SCRUMstudy
Targeting success

Una guía para el

CUERPO DE CONOCIMIENTO DE SCRUM (GUÍA SBOK™)

Tercera Edición

A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ Guide)

Incluye Escalamiento de Scrum en Grandes Proyectos y Escalamiento de Scrum para la Empresa

Una guía integral para la entrega de proyectos utilizando Scrum

SCRUMstudy

Targeting success

Una guía para el

CUERPO DE CONOCIMIENTO DE SCRUM

(Guía SBOK™)

3ra Edición

Una guía integral para la entrega de proyectos utilizando Scrum

© 2017 SCRUMstudy™, una marca de VMEdu, Inc. Todos los derechos reservados.

Información del catálogo de publicación de la Biblioteca del Congreso

Una guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™) – 3ra Edición
Título original en inglés*: A Guide to the Scrum Body Of Knowledge (SBOK™Guide)* – 3rd Edition

Incluye referencia bibliográfica e índice.
ISBN: 978-0-9899252-0-4

1. Scrum Framework. I. SCRUMstudy™. II. *SBOK™ Guide*

2013950625

ISBN: 978-0-9899252-0-4

Publicado por:
SCRUMstudy™, una marca de VMEdu, Inc.
12725 W. Indian School Road, Suite F-112
Avondale, Arizona 85392 USA
Correo Electrónico: sbok@scrumstudy.com
Sitio Web: www.scrumstudy.com

“SBOK”, el logotipo de SCRUMstudy, “SFC” “SDC”, “SMC”, “SAMC”, “SPOC” y “ESMC” son marcas registradas de SCRUMstudy™ (una marca de VMEdu, Inc.) Para obtener una lista completa de las marcas SCRUMstudy™, póngase en contacto con el Departamento Jurídico de SCRUMstudy™.

La G*uía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)* se ofrece para fines educativos. SCRUMstudy™ o VMEdu, Inc. no garantizan que sea adecuada para cualquier otro propósito; no se brinda una garantía expresa o implícita de ningún tipo, ni se asume responsabilidad por errores u omisiones. No se asume responsabilidad por daños incidentales o consecuentes en conexión que deriven de su uso o que surjan del uso de la información contenida en la guía.

SCRUMstudy™ acepta correcciones y comentarios sobre sus libros. Envíe por favor sus comentarios sobre errores tipográficos, de formato o de otro tipo. Puede fotocopiar la página indicada, señalar el error y enviarla al domicilio antes mencionado o por correo electrónico a: sbok@scrumstudy.com.

Se prohíbe la reproducción o transmisión parcial de cualquier tipo o por cualquier medio, ya sea electrónico, manual, de fotocopiado, grabación o cualquier sistema de almacenamiento y recuperación, sin la autorización previa por escrito de la editorial.

10 9 8 7 6 5 4 3 2

PREFACIO

La *Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)* proporciona directrices para la aplicación exitosa de Scrum: el desarrollo ágil de productos y el más popular método de entrega de proyectos. Scrum, tal como se define en la *Guía SBOK™,* es un framework que se aplica a portafolios, programas o proyectos de cualquier tamaño o complejidad; y se puede aplicar de manera efectiva en *cualquier* industria para crear un producto, servicio o cualquier otro resultado.

La *Guía SBOK™* está diseñada para utilizarse como referencia y guía de conocimiento tanto por profesionales con experiencia en Scrum y demás practicantes de desarrollo de productos o servicios, como por personas que no tengan experiencia previa o conocimiento de Scrum o cualquier otro método de entrega de proyectos. Esta nueva edición de la *Guía SBOK™* brinda información adicional sobre las mejores prácticas de Scrum, particularmente en las áreas para escalar en Scrum. Se han agregado dos capítulos a la *Guía SBOK™* para abordar en forma específica la forma de escalar en Scrum en grandes proyectos (Capítulo 13) y escalar en Scrum para la empresa (Capítulo 14). A medida que aumenta la popularidad y aplicación del framework de Scrum y evoluciona a nivel global, nuestra meta es compartir las lecciones aprendidas y las mejores prácticas como parte de la *Guía SBOK™*.

La *Guía SBOK™* se basa en el conocimiento y la visión combinada obtenida de miles de proyectos a través de una variedad de organizaciones e industrias. Esta tercera edición se suma a las aportaciones colectivas de expertos en Scrum y la entrega de proyectos. En forma particular, la retroalimentación de la comunidad global de Scrum jugó un papel importante en la identificación de mejoras y adiciones a la *Guía SBOK™*. Su desarrollo ha sido un verdadero esfuerzo de colaboración de un gran número de expertos y practicantes en una variedad de disciplinas.

La amplia adopción del framework de la *Guía SBOK™* estandariza la forma en la que Scrum se aplica a todo tipo de proyectos a través de las organizaciones a nivel mundial y ayuda de igual forma a mejorar considerablemente su retorno sobre la inversión. Promueve además una mayor reflexión y deliberación sobre la aplicación de Scrum para muchos tipos de proyectos, que a su vez contribuirán a ampliar y enriquecer el acervo de conocimientos y consecuentemente actualizaciones futuras de esta guía.

Aunque la *Guía SBOK™* es un framework y una guía integral para la entrega de proyectos que utilizan Scrum, su contenido está organizado para una fácil consulta, independientemente de los conocimientos previos que el lector tenga sobre el tema. Espero que cada lector pueda aprender y disfrutar tanto como los autores y revisores aprendieron y disfrutaron del proceso de compaginación de los conocimientos y la sabiduría colectiva que aquí se incluye.



Tridibesh Satpathy
Autor principal, *Guía SBOK™*

TABLA DE CONTENIDO

1. INTRODUCCIÓN 1
1.1 Información general de Scrum 2
1.1.1 Breve historia de Scrum 3
1.2 ¿Por qué utilizar Scrum? 4
1.2.1 Escalabilidad de Scrum 5
1.3 Propósito de la *Guía SBOK*™ 6
1.4 Framework de la *Guía SBOK*™ 7
1.4.1 ¿Cómo utilizar la *Guía SBOK*™? 8
1.4.2 Principios de Scrum 9
1.4.3 Aspectos de Scrum 11
1.4.4 Procesos de Scrum 16
1.5 Scrum vs. Gestión tradicional de proyectos 22
2. PRINCIPIOS 23
2.1 Introducción 23
2.2 Guía de roles 24
2.3 Control del proceso empírico 24
2.3.1 Transparencia 24
2.3.2 Inspección 26
2.3.3 Adaptación 26
2.4 Auto-organización 29
2.4.1 Beneficios de la auto-organización 29
2.5 Colaboración 31
2.5.1 Beneficios de la colaboración en los proyectos Scrum 31
2.5.2 La importancia de la co-ubicación en la colaboración 33
2.6 Priorización basada en valor (*Value-based Prioritization*) 33
2.7 Time-boxing 36
2.7.1 Time-boxes de Scrum 36
2.8 Desarrollo iterativo 39
2.9 Scrum vs. Gestión tradicional de proyectos 41
3. ORGANIZACIÓN 43
3.1 Introducción 43

3.2 Guía de los roles 44
3.3 Roles de un proyecto Scrum 44
3.3.1 Roles centrales 45
3.3.2 Roles no centrales 46
3.4 Product Owner 48
3.4.1 Voz del cliente (VOC) 49
3.4.2 Chief Product Owner 49
3.4.3 Program Product Owner 50
3.4.4 Portfolio Product Owner 50
3.5 Scrum Master 51
3.5.1 Chief Scrum Master 52
3.5.2 Program Scrum Master 54
3.5.3 Portfolio Scrum Master 54
3.6 Equipo Scrum 54
3.6.1 Selección de personal 56
3.6.2 Tamaño del Equipo Scrum 56
3.7 Scrum en proyectos, programas y portafolios 57
3.7.1 Definición de proyecto, programa y portafolios 57
3.7.2 Mantener la participación de los stakeholders 59
3.8 Resumen de responsabilidades 60
3.9 Scrum vs. Gestión tradicional de proyectos 61
3.10 Teorías populares de recursos humanos y su relevancia para Scrum 62
3.10.1 Modelo de dinámica de grupo de Tuckman (*Tuckman's Model of Group Dynamics*) 62
3.10.2 Gestión de conflictos (*Conflict Management*) 63
3.10.3 Técnicas de gestión de conflictos 63
3.10.4 Estilos de liderazgo 65
3.10.5 Teoría de jerarquía de necesidades de Maslow 67
3.10.6 Teoría X y Teoría Y 68
4. JUSTIFICACIÓN DEL NEGOCIO 69
4.1 Introducción 69
4.2 Guía de roles 70
4.3 Entrega basada en valor (*Value-driven Delivery*) 70
4.3.1 Responsabilidades del Product Owner en la justificación del negocio 72

4.3.2 Responsabilidades de otros roles de Scrum en la justificación del negocio 72
4.4 Importancia de la justificación del negocio 73
4.4.1 Factores que se utilizan para determinar la justificación del negocio 73
4.4.2 Justificación del negocio y el ciclo de vida del proyecto 74
4.5 Técnicas de justificación del negocio 76
4.5.1 Estimación del valor del proyecto 76
4.5.2 Planificar para el valor 78
4.5.3 Clasificación relativa de priorización (*Relative Prioritization Ranking*) 80
4.5.4 Mapeo de historias 80
4.6 Justificación continua de valor 80
4.6.1 Análisis del valor ganado (AVG) 81
4.6.2 Diagrama de flujo acumulativo (DFA) 84
4.7 Confirmar la realización de beneficios 85
4.7.1 Prototipos, simulaciones y demostraciones 85
4.8 Resumen de responsabilidades 86
4.9 Scrum vs. Gestión tradicional de proyectos 87
5. CALIDAD 89
5.1 Introducción 89
5.2 Guía de roles 90
5.3 Definición de calidad 90
5.3.1 Calidad y alcance 91
5.3.2 Calidad y valor del negocio 91
5.4 Criterios de aceptación y Backlog Priorizado del Producto 92
5.4.1 Redacción de criterios de aceptación 94
5.4.2 Definición de terminado 94
5.4.3 Criterios mínimos de terminado 95
5.4.4 Aceptación o rechazo de elementos del Backlog Priorizado del Producto 96
5.5 Gestión de calidad en Scrum 96
5.5.1 Planificación de calidad 97
5.5.2 Control de calidad y garantía de calidad 98
5.5.3 Ciclo de planificar, hacer, verificar y actuar (Ciclo PDCA, por sus siglas en inglés) 99
5.6 Resumen de responsabilidades 100
5.7 Scrum vs. Gestión tradicional de proyectos 101

6. CAMBIO 103
6.1 Introducción 103
6.2 Guía de los roles 104
6.3 Descripción 104
6.3.1 Solicitudes de cambio aprobadas y no aprobadas 105
6.4 Cambio en Scrum 106
6.4.1 Equilibrio entre flexibilidad y estabilidad 106
6.4.2 Lograr la flexibilidad 107
6.5 Integración del cambio 112
6.5.1 Cambios a un sprint 113
6.6 Cambio en portafolios y programas 118
6.6.1 En portafolios 118
6.6.2 En programas 118
6.7 Resumen de responsabilidades 120
6.8 Scrum vs. Gestión tradicional de proyectos 121
7. RIESGO 123
7.1 Introducción 123
7.2 Guía de roles 124
7.3 ¿Qué es un riesgo? 124
7.3.1 Diferencia entre riesgos y problemas 124
7.3.2 Actitud de riesgo 125
7.4 Procedimiento de gestión de riesgos 126
7.4.1 Identificación de riesgos 126
7.4.2 Evaluación de riesgos 127
7.4.3 Priorización de riesgos 131
7.4.4 Mitigación de riesgos 132
7.4.5 Comunicación de riesgos 133
7.5 Minimizar riesgos por medio de Scrum 135
7.6 Riesgos en portafolios y programas 136
7.6.1 En el portafolio 136
7.6.2 En los programas 136
7.7 Resumen de responsabilidades 138
7.8 Scrum vs Gestión tradicional de proyectos 139

8. INICIO 141
8.1 Crear la visión del proyecto 145
8.1.1 Entradas 147
8.1.2 Herramientas 149
8.1.3 Salidas 151
8.2 Identificar al Scrum Master y stakeholder(s) 153
8.2.1 Entradas 155
8.2.2 Herramientas 157
8.2.3 Salidas 158
8.3 Formar el Equipo Scrum 159
8.3.1 Entradas 161
8.3.2 Herramientas 162
8.3.3 Salidas 163
8.4 Desarrollar épica(s) 165
8.4.1 Entradas 167
8.4.2 Herramientas 170
8.4.3 Salidas 172
8.5 Crear el Backlog Priorizado del Producto 174
8.5.1 Entradas 176
8.5.2 Herramientas 177
8.5.3 Salidas 179
8.6 Realizar la planificación del lanzamiento 181
8.6.1 Entradas 183
8.6.2 Herramientas 184
8.6.3 Salidas 185
8.7 Diagrama de flujo de datos de fase 187
9. PLANIFICACIÓN Y ESTIMACIÓN 189
9.1 Crear historias de usuario 193
9.1.1 Entradas 195
9.1.2 Herramientas 196
9.1.3 Salidas 198
9.2 Estimar historias de usuarios 200
9.2.1 Entradas 202

9.2.2 Herramientas 202
9.2.3 Salidas 205
9.3 Comprometer historias de usuario 206
9.3.1 Entradas 208
9.3.2 Herramientas 208
9.3.3 Salidas 209
9.4 Identificar tareas 210
9.4.1 Entradas 212
9.4.2 Herramientas 212
9.4.3 Salidas 213
9.5 Estimar tareas 215
9.5.1 Entradas 217
9.5.2 Herramientas 218
9.5.3 Salidas 219
9.6 Crear el Sprint Backlog 220
9.6.1 Entradas 222
9.6.2 Herramientas 222
9.6.3 Salidas 223
9.7 Diagrama de flujo de datos de fase 225
10. IMPLEMENTACIÓN 227
10.1 Crear entregables 231
10.1.1 Entradas 233
10.1.2 Herramientas 235
10.1.3 Salidas 236
10.2 Realizar el Daily Standup 238
10.2.1 Entradas 240
10.2.2 Herramientas 241
10.2.3 Salidas 242
10.3 Refinar el Backlog Priorizado del Producto 244
10.3.1 Entradas 246
10.3.2 Herramientas 247
10.3.3 Salidas 248
10.4 Diagrama de flujo de datos de fase 250

11. REVISIÓN Y RETROSPECTIVA 251
11.1 Demostrar y validar el sprint 255
11.1.1 Entradas 257
11.1.2 Herramientas 258
11.1.3 Salidas 259
11.2 Retrospectiva del sprint 260
11.2.1 Entradas 262
11.2.2 Herramientas 262
11.2.3 Salidas 264
11.3 Diagrama de flujo de datos de fase 266
12. LANZAMIENTO 267
12.1 Enviar entregables 271
12.1.1 Entradas 273
12.1.2 Herramientas 274
12.1.3 Salidas 275
12.2 Retrospectiva del proyecto 276
12.2.1 Entradas 278
12.2.2 Herramientas 279
12.2.3 Salidas 280
12.3 Diagrama de flujo de datos de fase 281
13. ESCALAMIENTO DE SCRUM EN GRANDES PROYECTOS 283
13.1 Crear componentes de grandes proyectos 287
13.1.1 Entradas 289
13.1.2 Herramientas 291
13.1.3 Salidas 293
13.2 Realizar y coordinar sprints 297
13.2.1 Entradas 299
13.2.2 Herramientas 301
13.2.3 Salidas 302
13.3 Preparar el lanzamiento de grandes proyectos 303
13.3.1 Entradas 305
13.3.2 Herramientas 305
13.3.3 Salidas 306

13.4 Impacto de los grandes proyectos en los procesos fundamentales de Scrum 307
14. ESCALAMIENTO DE SCRUM PARA LA EMPRESA 311
14.1 Crear componentes de programa o portafolio 315
14.1.1 Entradas 316
14.1.2 Herramientas 317
14.1.3 Salidas 318
14.2 Revisar y actualizar el Scrum Guidance Body 319
14.2.1 Entradas 321
14.2.2 Herramientas 321
14.2.3 Salidas 322
14.3 Crear y refinar el backlog del programa o portafolio 324
14.3.1 Entradas 326
14.3.2 Herramientas 328
14.3.3 Salidas 330
14.4 Coordinar componentes del programa o portafolio 331
14.4.1 Entradas 333
14.4.2 Herramientas 335
14.4.3 Salidas 336
14.5 Retrospectiva de los lanzamientos del programa o portafolio 338
14.5.1 Entradas 339
14.5.2 Herramientas 340
14.5.3 Salidas 340
APÉNDICE A. DESCRIPCIÓN GENERAL DE ÁGIL 341
APÉNDICE B. AUTORES Y COLABORADORES DE LA *GUÍA SBOK*™ 351
APÉNDICE C. ACTUALIZACIONES A LA TERCERA EDICIÓN 355
REFERENCIAS 359
GLOSARIO 361
ÍNDICE TEMÁTICO 401

LISTA DE FIGURAS

Figura 1-1: Flujo de Scrum para un sprint....2
Figura 1-2: Framework de la *Guía SBOK™*....7
Figura 1-3: Principios de Scrum....9
Figura 1-4: Organización en Scrum....13
Figura 2-1: Transparencia en Scrum....25
Figura 2-2: Inspección de Scrum....26
Figura 2-3: Adaptación en Scrum....27
Figura 2-4: Retos en la gestión tradicional de proyectos....28
Figura 2-5: Objetivos de un equipo de auto-organizado....30
Figura 2-6: Beneficios de la colaboración en proyectos Scrum....32
Figura 2-7: Priorización basada en valor....35
Figura 2-8: Duración de los bloques de tiempo (Time-Box) para las reuniones de Scrum....38
Figura 2-9: Scrum vs Cascada tradicional....40
Figura 3-1: Roles de Scrum—Descripción General....46
Figura 3-2: Las preguntas formuladas durante una reunión de Scrum de Scrums....53
Figura 3-3: Características deseadas de los roles centrales de Scrum....56
Figura 3-6: Etapas de Tuckman de desarrollo de grupos....62
Figura 3-7: Teoría de jerarquía de necesidades de Maslow....67
Figura 4-1: Entrega de valor en Scrum vs. Proyectos tradicionales....71
Figura 4-2: Jerarquía de responsabilidades en la justificación del negocio....72
Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto....75
Figura 4-4: Análisis de Kano....79
Figura 4-5: Ejemplo de diagrama de flujo acumulativo (DFA)....84
Figura 5-1: Diagrama de flujo del incremento del proyecto....93
Figura 5-3: Ciclo PDCA en Scrum....100
Figura 6-1: Ejemplo del proceso de aprobación de cambios....105
Figura 6-2: Actualización del Backlog Priorizado del Producto con los cambios aprobados....106
Figura 6-3: Características de Scrum para lograr flexibilidad....107
Figura 6-4: Motivación de los stakeholders para la solicitud de cambios....108
Figura 6-5: Motivos del equipo principal de Scrum para la solicitud de cambios....109
Figura 6-6: Integración del cambio en Scrum....113
Figura 6-7: Impacto del cambio esperado en la duración del sprint....115
Figura 6-8: Incorporación de cambios en portafolios y programas....119
Figura 7-1: Ejemplo de árbol de probabilidad....128
Figura 7-2: Ejemplo de un diagrama de Pareto....129
Figura 7-3: Ejemplo de matriz de probabilidad e impacto....130
Figura 7-4: Proceso de priorización de riesgos....132
Figura 7-5: Ejemplo de un Risk Burndown Chart....134
Figura 8-1: Información general sobre el Inicio....143
Figura 8-2: Resumen sobre el inicio (esenciales)....144
Figura 8-3: Crear la visión del proyecto – Entradas, herramientas y salidas....145
Figura 8-4: Crear la visión del proyecto - Diagrama del flujo de datos....146

Figura 8-5: El proceso de Análisis de brecha....151
Figura 8-6: Identificar al Scrum Master y stakeholder(s)—Entradas, herramientas y salidas....153
Figura 8-7: Identificar al Scrum Master y stakeholder(s)—Diagrama de flujo de datos....154
Figura 8-8: Formar el Equipo Scrum—Entradas, herramientas y salidas....159
Figura 8-9: Formar el Equipo Scrum—Diagrama de flujo de datos....160
Figura 8-10: Desarrollar épica(s)—Entradas, herramientas y salidas....165
Figura 8-11: Desarrollar épica(s)—Diagrama de flujo....166
Figura 8-12: Crear el Backlog Priorizado del Producto—Entradas, herramientas y salidas....174
Figura 8-13: *Crear el Backlog Priorizado del Producto*—Diagrama de flujo de datos....175
Figura 8-14: Realizar la planificación del lanzamiento—Entrada, herramientas y salidas....181
Figura 8-15: Realizar el plan de lanzamiento—Diagrama de flujo de datos....182
Figura 8-16: Fase de inicio—Diagrama del flujo de datos....187
Figura 9-1: Resumen de la planificación y estimación....191
Figura 9-2: Resumen de la Planificación y Estimación (Esenciales)....192
Figura 9-3: Crear historias de usuario—Entradas, herramientas y salidas....193
Figura 9-4: Crear historias de usuario—Diagrama de flujo de datos....194
Figura 9-5: Estimar historias de usuario—Entradas, herramientas y salidas....200
Figura 9-6: Estimar historias de usuario— Diagrama de flujo de datos....201
Figura 9-7: Comprometer tareas—Entradas, herramientas y salidas....206
Figura 9-8: Comprometer historias de usuario—Diagrama de flujo de datos....207
Figura 9-12: Estimar tareas—Entradas, herramientas y salidas....215
Figura 9-13: Crear el Sprint Backlog—Entradas, herramientas y salidas....220
Figura 10-1: Descripción general de la Implementación....229
Figura 10-2: Resumen de la implementación (Esenciales)....230
Figura 10-3: Crear entregables—Entradas, herramientas y salidas....231
Figura 10-4: Crear entregables—Diagrama de flujo de datos....232
Figura 10-5: Scrumboard....233
Figura 10-6: Realizar el Daily Standup—Entradas, herramientas y salidas....238
Figura 10-7: Realizar el Daily Standup—Diagrama de flujo de datos....239
Figura 10-8: Refinar el Backlog Priorizado del Producto—Entradas, herramientas y salidas....244
Figura 10-9: Refinar el Backlog Priorizado del Producto—Diagrama de flujo de datos....245
Figura 10-10: Fase de implementación—Diagrama de flujo de datos....250
Figura 11-1: Descripción general de la Revisión y retrospectiva....253
Figura 11-2: Descripción general de la Revisión y retrospectiva (esenciales)....254
Figura 11-5: Demostrar y validar el sprint—Entradas, herramientas y salidas....255
Figura 11-6: Demostrar y validar el sprint—Diagrama de flujo de datos....256
Figura 11-7: Retrospectiva del sprint—Entradas, herramientas y salidas....260
Figura 11-9: Fase de revisión y retrospectiva—Diagrama de flujo de datos....266
Figura 12-1: Descripción general del lanzamiento....269
Figura 12-2: Resumen del lanzamiento (Esenciales)....270
Figura 12-3: Enviar entregables—Entradas, herramientas y salidas....271
Figura 12-4: Envío de entregables—Diagrama de flujo de datos....272
Figura 12-5: Retrospectiva del proyecto—Entrada, herramientas y salidas....276
Figura 12-6: Retrospectiva del proyecto—Diagrama de flujo de datos....277

Figura 12-7: Fase de lanzamiento—Diagrama de flujo de datos 281
Figura 13-1: Descripción general del Escalamiento de Scrum en grandes proyectos 285

LISTA DE TABLAS

Tabla 1-1: Resumen de los procesos fundamentales de Scrum....16
Tabla 1-2: Scrum vs. Gestión tradicional de proyectos....22
Tabla 3-1: Responsabilidades del Product Owner en los procesos de Scrum....49
Tabla 3-2: Responsabilidades del Scrum Master en los procesos de Scrum....52
Tabla 3-3: Responsabilidades del Equipo Scrum en los procesos de Scrum....55
Tabla 3-4: Resumen de las responsabilidades pertinentes a la organización....60
Tabla 4-1: Fórmulas del valor ganado....82
Tabla 4-2: Resumen de las responsabilidades pertinentes a la justificación del negocio....86
Tabla 5-1: Resumen de las responsabilidades pertinentes a la calidad....101
Tabla 6-1: Resumen de las responsabilidades pertinentes al cambio....120
Tabla 7-1: Resumen de las responsabilidades pertinentes a los riesgos....138

1. INTRODUCCIÓN

La *Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)* proporciona directrices para la aplicación con éxito de Scrum: el desarrollo ágil de productos y el método de entrega de proyectos más popular. Brinda un framework integral que incluye los principios, aspectos y procesos de Scrum.

Scrum, tal como se define en la *Guía SBOK™,* aplica a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria
- Productos, servicios o cualquier otro resultado que se les entregarán a los interesados
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la Guía SBOK™ puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse en forma efectiva a cualquier proyecto en cualquier industria, desde proyectos pequeños o equipos con tan sólo seis miembros, hasta proyectos grandes y complejos con varios cientos de integrantes.

Este primer capítulo describe la finalidad y el framework de la *Guía SBOK™* y proporciona una introducción a los conceptos claves de Scrum. Contiene un resumen de los principios de Scrum, al igual que los aspectos y los procesos sobre el tema. El capítulo 2 amplía la información sobre los seis principios de Scrum, que son la base del mismo. Los capítulos del 3 al 7 tratan en detalle los cinco aspectos de Scrum que se deben abordar en cualquier proyecto: organización, justificación del negocio, la calidad, el cambio y el riesgo. Los capítulos del 8 al 12 cubren los 19 procesos de Scrum que forman parte de la creación de un proyecto Scrum. Estos procesos forman parte de las cinco fases de Scrum: Inicio; Planificación y estimación; Implementación, Revisión y retrospectiva y Lanzamiento. Estas fases describen a detalle las entradas y salidas asociadas con cada proceso, así como las diferentes herramientas que pueden utilizarse en cada una. Algunas entradas, herramientas y salidas son obligatorias y estas se indican como tales; otras son opcionales dependiendo del proyecto específico, de los requisitos de la organización y/o lineamientos establecidos por el Scrum Guidance Body de la organización (SGB). Los capítulos 13 y 14 son nuevas adiciones a la *Guía SBOK™*, mismas que brindan las directrices para escalar en Scrum en grandes proyectos y escalar Scrum para la empresa.

Este capítulo está dividido en las siguientes secciones:

1.1 Información general de Scrum

1.2 ¿Por qué utilizar Scrum?

1.3 Propósito de la *Guía SBOK™*

1.4 Framework de la *Guía SBOK™*

1.5 Scrum vs. Gestión tradicional de proyectos

1.1 Información general de Scrum

Un proyecto Scrum consiste en un esfuerzo de colaboración para crear un nuevo producto, servicio u otro resultado tal como se define en la Declaración de la visión del proyecto (*Project Vision Statement*). Los proyectos se ven afectados por limitaciones de tiempo, costos, alcance, calidad, recursos, capacidades organizacionales y demás limitaciones que dificultan su planificación, ejecución, administración y, por último, su éxito. Sin embargo, la implementación exitosa de los resultados de un proyecto terminado le proporciona ventajas económicas considerables a una organización. Por lo tanto, es importante que las organizaciones seleccionen e implementen un método adecuado de gestión de proyectos.

Scrum es uno de los métodos ágiles más populares. Es un framework adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto. Scrum garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo. El framework de Scrum, tal como se define en la *Guía SBOK™,* está estructurado de tal manera que es compatible con el desarrollo de productos y servicios en todo tipo de industrias y en cualquier tipo de proyecto, independientemente de su complejidad.

Una fortaleza clave de Scrum radica en el uso de equipos interfuncionales (*cross-functional*), auto-organizados y empoderados que dividen su trabajo en ciclos de trabajo cortos y concentrados llamados *Sprints*. La figura 1-1 proporciona una visión general de flujo de un proyecto Scrum.

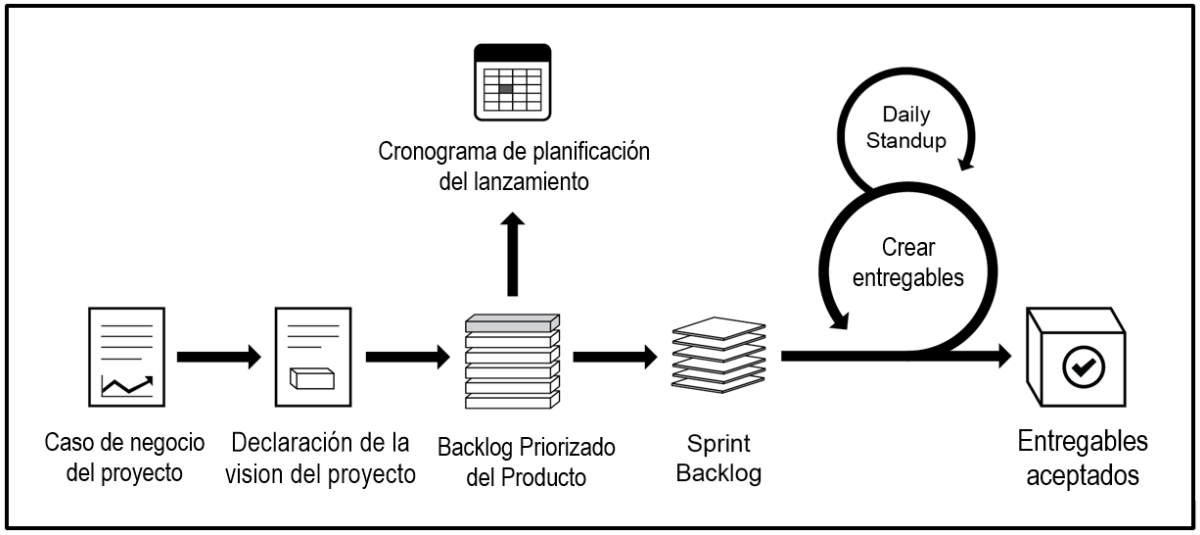


Figura 1-1: Flujo de Scrum para un sprint

El ciclo de Scrum empieza con una reunión de stakeholders, durante la cual se crea la visión del proyecto. Después, el Product Owner desarrolla una Backlog Priorizado del Producto (*Prioritized Product Backlog)* que contiene una lista requerimientos del negocio y del proyecto por orden de importancia en forma de una historia de usuario. Cada sprint empieza con una reunión de planificación del sprint (*Sprint Planning Meeting*) durante la cual se consideran las historias de usuario de alta prioridad para su inclusión en el sprint. Un sprint generalmente tiene una duración de una a seis semanas durante las cuales el Equipo

Scrum trabaja en la creación de entregables (del inglés *deliverables*) en incrementos del producto. Durante el sprint, se llevan cabo Daily Standups muy breves y concretos, donde los miembros del equipo discuten el progreso diario. Haca el final del sprint, se lleva a cabo una Reunión de Revisión del Sprint (*Sprint Review Meeting*) en la cual se proporciona una demostración de los entregables al Product Owner y a los stakeholders relevantes. El Product Owner acepta los entregables sólo si cumplen con los criterios de aceptación predefinidos. El ciclo del sprint termina con una Reunión de Retrospectiva del Sprint (*Retrospect Sprint Meeting*), donde el equipo analiza las formas de mejorar los procesos y el rendimiento a medida que avanzan al siguiente sprint.

1.1.1 Breve historia de Scrum

A mediados de la década de los 80s, Hirotaka Takeuchi y Ikujiro Nonaka definieron una estrategia de desarrollo de producto flexible e incluyente donde el equipo de desarrollo trabaja en unidad para alcanzar un objetivo común. Describieron un método innovador para el desarrollo de productos al que llamaron enfoque holístico o “rugby”, “donde un equipo intenta llegar hasta el final como una unidad, pasando el balón hacia atrás y adelante”. Basaron su enfoque en los estudios de casos de diversas industrias de fabricación. Takeuchi y Nonaka propusieron que el desarrollo de productos no debe ser como una carrera de relevos secuencial, sino que debería ser análogo al del juego de rugby, donde el equipo trabaja en conjunto, pasando el balón hacia atrás y hacia adelante a medida que se desplaza en unidad por el campo. El concepto de rugby de un “Scrum” (donde un grupo de jugadores se junta para reiniciar el juego) se introdujo en este artículo para describir la propuesta de los autores de que el desarrollo de productos debe implicar “mover al Scrum campo abajo”.

Ken Schwaber y Jeff Sutherland desarrollaron el concepto de Scrum y su aplicabilidad al desarrollo de software durante una presentación en la Conferencia internacional sobre programación, lenguajes y aplicaciones orientadas a objetos (*Object-Oriented Programming, Systems, Languages & Applications,* o OOPSLA) en 1995 en Austin, Texas. Desde entonces, varios practicantes, expertos y autores de Scrum han seguido perfeccionando la conceptualización y framework de Scrum. En los últimos años, Scrum ha aumentado en popularidad, y es hoy en día el método de desarrollo de proyectos predilecto de muchas organizaciones a nivel mundial.

1.2 ¿Por qué utilizar Scrum?

Algunas de las ventajas principales del uso de Scrum en cualquier proyecto son:

1. **Adaptabilidad—**El control del proceso empírico y el desarrollo iterativo hacen que los proyectos sean adaptables y abiertos a la incorporación del cambio.

2. **Transparencia—**Todos los radiadores de información tales como un Scrumboard y el Sprint Burndown Chart se comparten, lo cual conduce a un ambiente de trabajo abierto.

3. **Retroalimentación continua—**La retroalimentación continua se proporciona a través de los procesos de *Realizar Daily Standup* y *Demostrar y validar el sprint*.

4. **Mejora continua—**Los entregables se mejoran progresivamente sprint por sprint a través del proceso de *Refinar el Backlog Priorizado del Producto.*

5. **Entrega continúa de valor—**Los procesos iterativos permiten la entrega continua de valor tan frecuentemente como el cliente lo requiere a través del proceso de *Envío de entregables*.

6. **Ritmo sostenible—**Los procesos Scrum están diseñados de tal manera que las personas involucradas pueden trabajar a un ritmo sostenible que, en teoría, puede continuar indefinidamente.

7. **Entrega anticipada de alto valor—**El proceso de *Crear el Backlog Priorizado del Producto* asegura que los requisitos de mayor valor del cliente sean los primeros en cumplirse.

8. **Proceso de desarrollo eficiente—**El *Time-boxing* y la reducción al mínimo del trabajo que no es esencial conducen a mayores niveles de eficiencia.

9. **Motivación—**Los procesos de *Realizar Daily Standup* y *Retrospectiva del sprint* conducen a mayores niveles de motivación entre los empleados.

10. **Resolución de problemas de forma más rápida—**La colaboración y co-ubicación de equipos interfuncionales conducen a la resolución de problemas con mayor rapidez.

11. **Entregables efectivos—**El proceso de *Crear el Backlog Priorizado del Producto*, y las revisiones periódicas después de la creación de entregables aseguran entregas eficientes al cliente.

12. **Centrado en el cliente—**El poner énfasis en el valor del negocio y tener un enfoque de colaboración con los stakeholders asegura un framework orientado al cliente.

13. **Ambiente de alta confianza—**Los procesos de *Realizar Daily Standup* y la *Retrospectiva del Sprint* promueven la transparencia y colaboración, dando lugar a un ambiente de trabajo de alta confianza que garantiza una baja fricción entre los empleados.

14. **Responsabilidad colectiva—**El proceso de *Comprometer Historias de Usuarios* permite que los miembros del equipo hagan suyo el proyecto y su trabajo lleve a una mejor calidad.

15. **Alta velocidad—**Un framework de colaboración permite a los equipos interfuncionales altamente cualificados alcanzar su potencial y una alta velocidad.

16. **Ambiente innovador—**Los procesos de *Retrospectiva de Sprint* y *Retrospectiva del Proyecto* crean un ambiente de introspección, aprendizaje y capacidad de adaptación que conllevan a un ambiente de trabajo innovador y creativo.

1.2.1 Escalabilidad de Scrum

Para ser eficaces, los equipos Scrum idealmente deben tener de seis a diez miembros. Esta práctica pudiera ser la razón de la idea errónea de que el framework de Scrum sólo puede utilizarse para proyectos pequeños. Sin embargo, este framework puede aumentarse fácilmente para utilizarse de manera eficaz en grandes proyectos, programas y portafolios. En situaciones donde el tamaño del Equipo Scrum excede diez personas, se pueden formar diversos equipos Scrum para trabajar en el proyecto. El enfoque lógico de las directrices y los principios de este framework pueden utilizarse para gestionar proyectos de cualquier tamaño, que abarcan grandes geografías y organizaciones. Los proyectos grandes pueden tener múltiples equipos Scrum que trabajan forma paralela, por lo que es necesario sincronizarse y facilitar el flujo de información y mejorar la comunicación. Los proyectos grandes y complejos generalmente se implementan como parte de un programa o portafolio.

Los detalles sobre la escalabilidad en Scrum para grandes proyectos se proporcionan en el Capítulo 13 y lo relacionado a escalar Scrum para la empresa se cubre en el Capítulo 14.

1.3 Propósito de la *Guía SBOK*™

En los últimos años ha sido evidente que las organizaciones que utilizan Scrum como el framework de ejecución de proyectos preferido, consistentemente obtienen altos retornos sobre la inversión. El enfoque de Scrum en la entrega impulsada por el valor ayuda a que los equipos de Scrum rindan resultados durante el proyecto tan pronto como les sea posible.

La *Guía SBOK*™ ha sido desarrollada como medio para crear una guía necesaria para organizaciones y profesionales de gestión de proyectos que deseen implementar Scrum, así como para quienes ya lo hacen y deseen mejorar sus procesos. Se basa en la experiencia adquirida de miles de proyectos a través de una variedad de organizaciones e industrias. En su desarrollo se tomaron en cuenta las aportaciones de muchos expertos en Scrum y profesionales de gestión de proyectos.

La *Guía SBOK*™ es especialmente valiosa:

- para los miembros del equipo principal de Scrum, incluyendo los siguientes:
  - Product Owners que deseen comprender plenamente el framework de Scrum y particularmente las inquietudes del cliente o stakeholder relacionadas a la justificación del negocio, a la calidad, el cambio y los aspectos de riesgo asociados con los proyectos Scrum.
  - Scrum Masters que quieran aprender su rol específico en la supervisión de la aplicación del framework de Scrum en proyectos de este tipo.
  - Miembros del Equipo Scrum que deseen comprender mejor los procesos de Scrum y las herramientas asociadas que pueden utilizarse para crear el producto o servicio del proyecto.
- como una guía integral para todos los practicantes de Scrum que trabajan en proyectos Scrum en cualquier organización o industria.
- como fuente de referencia para cualquier persona que interactúe con el equipo principal de Scrum, incluyendo, pero sin limitarse, al Portfolio Product Owner, Portfolio Scrum Master, Program Product Owner, Program Scrum Master, Scrum Guidance Body y Stakeholders (patrocinador, cliente y usuarios).
- como un manual para cualquier persona que no tenga experiencia previa o conocimiento del framework de Scrum, pero quiera aprender más sobre el tema.

El contenido de la *Guía SBOK*™ también es útil para las personas que se preparan para tomar los siguientes exámenes de certificación de SCRUMstudy™:

- Scrum Developer Certified (SDC™)
- Scrum Master Certified (SMC™)
- Scaled Scrum Master Certified (SSMC™)
- SCRUMstudy Agile Master Certified (SAMC™)
- Scrum Product Owner Certified (SPOC™)
- Scaled Scrum Product Owner Certified (SSPOC™)
- Expert Scrum Master Certified (ESMC™)

1.4 Framework de la *Guía SBOK*™

La *Guía SBOK*™ se divide en las siguientes tres áreas:

1. Los **principios** que se contemplan en el capítulo 2 amplían la información sobre los seis principios que constituyen el fundamento sobre el que se basa Scrum.

2. Los **aspectos** que se cubren en los capítulos del 3 al 7 describen los cinco aspectos que se consideran importantes para todos los proyectos Scrum.

3. Los **procesos** que se cubren en los capítulos del 8 al 12 incluyen los diecinueve procesos fundamentales de Scrum y sus entradas, herramientas y salidas asociadas. Los capítulos 13 y 14 abordan procesos adicionales específicos sobre escalar Scrum en grandes proyectos y escalar Scrum para la empresa.

La figura 1-2 ilustra el framework de la *Guía SBOK*™, que muestra que los principios, aspectos y procesos interactúan entre sí y son de igual importancia al tratar de obtener una mejor comprensión del framework de Scrum.

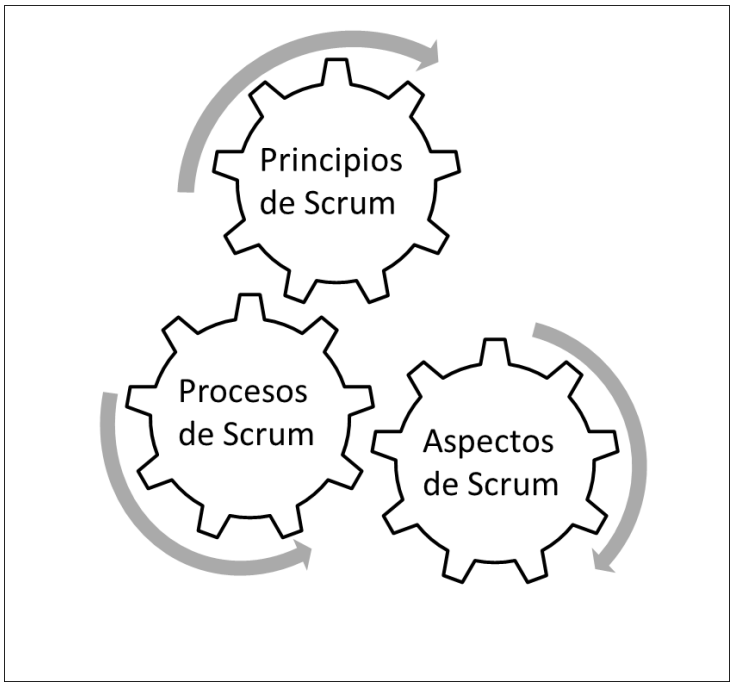


Figura 1-2: Framework de la *Guía SBOK*™

1.4.1 ¿Cómo utilizar la *Guía SBOK*™?

La *Guía SBOK*™ puede utilizarse como una referencia y guía de conocimiento tanto por practicantes de Scrum con experiencia y demás profesionales de desarrollo de productos y servicios, como por personas sin experiencia previa o conocimiento de Scrum o de métodos de gestión de proyectos. Los contenidos se organizan para facilitar la consulta de los tres roles principales del Equipo Scrum: Scrum Master, Product Owner y Equipo Scrum.

Los capítulos que abarcan los seis principios de Scrum (capítulo 2) y los cinco aspectos de Scrum (capítulos del 3 al 7), incluyen una guía de roles. Esta guía brinda información sobre los roles del equipo principal de Scrum (Scrum Core Team).

A fin de facilitar la mejor aplicación del framework de Scrum, la *Guía SBOK*™ ha diferenciado claramente entre las entradas, las herramientas y las salidas obligatorias de las opcionales. Las entradas, herramientas y salidas que se indican con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales. Se recomienda que las personas que empiezan a aprender sobre Scrum se enfoquen principalmente en las entradas, las herramientas y las salidas obligatorias, mientras que los profesionales con más experiencia deben leer todos los capítulos del proceso a fin de beneficiarse de las entradas, herramientas y salidas sugeridas como mejores prácticas opcionales.

Scrum es un framework y no pretende ser prescriptivo, lo cual significa que hay espacio para la flexibilidad en su aplicación. Todos los procesos fundamentales de Scrum detallados en el SBOK (capítulos del 8 al 12) son obligatorios para cada proyecto Scrum, pero se aplicarían con base en las necesidades específicas de la organización, del proyecto, del producto o el equipo. Los procesos adicionales aplicarían solamente cuando se escale Scrum en grandes proyectos (capítulo 13) o cuando se escale Scrum para las empresas (capítulo 14).

1.4.2 Principios de Scrum

Los principios de Scrum son las pautas básicas para aplicar el framework de Scrum y deben implementarse en forma obligatoria en todos los proyectos Scrum. Los seis principios de Scrum que se presentan en el capítulo 2 son los siguientes:

1. Control del proceso empírico (*Empirical Process Control*)
2. Auto-organización (*Self-organization*)
3. Colaboración (*Collaboration*)
4. Priorización basada en valor (*Value-based Prioritization*)
5. Time-boxing
6. Desarrollo iterativo (*Iterative Development*)

La figura 1-3 ilustra los seis principios de Scrum.

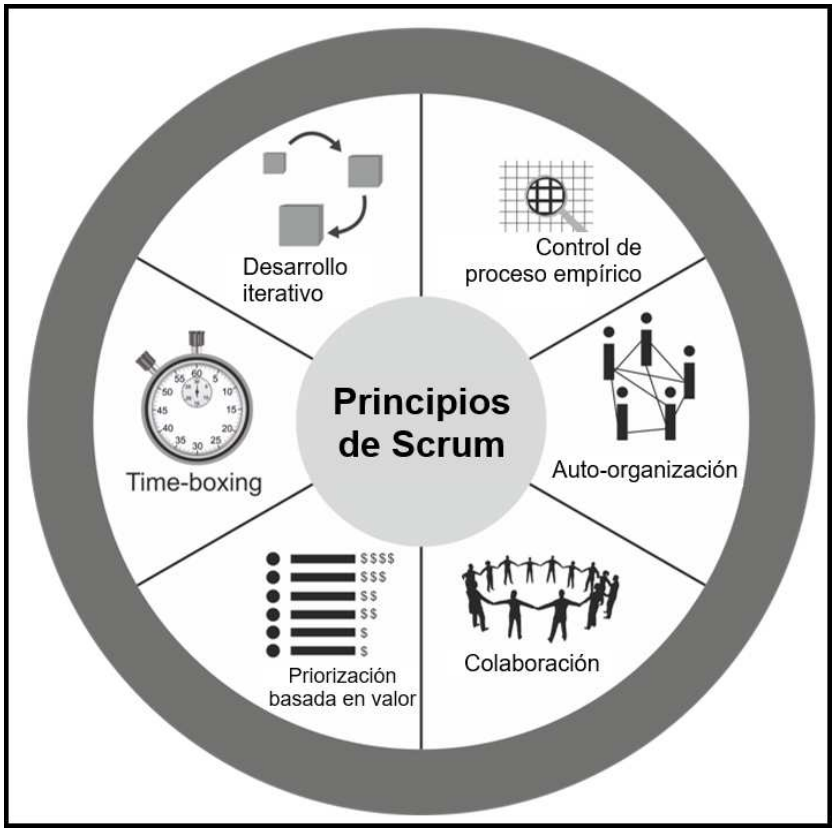


Figura 1-3: Principios de Scrum

Los principios de Scrum se pueden aplicar a cualquier tipo de proyecto en cualquier organización y deben cumplirse a fin de garantizar la aplicación efectiva del framework de Scrum. Los principios de Scrum no están abiertos a la discusión ni pueden modificarse, y deben aplicarse tal como se especifica en la *Guía SBOK™.* El mantener los principios intactos y usarlos apropiadamente infunde confianza en el framework de Scrum respecto al cumplimiento de los objetivos del proyecto. Los aspectos y procesos de Scrum, sin embargo, pueden modificarse para cumplir con los requisitos del proyecto o la organización.

1. **Control del proceso empírico—**Este principio enfatiza la filosofía central de Scrum con base a las tres ideas principales de transparencia, inspección y adaptación.

2. **Auto-organización—**Este principio se enfoca en los trabajadores de hoy en día, que entregan un valor considerablemente mayor cuando se auto-organizan, lo cual resulta en equipos que poseen un gran sentido de compromiso y responsabilidad; a su vez, esto produce un ambiente innovador y creativo que es más propicio para el crecimiento.

3. **Colaboración—**Este principio se centra en las tres dimensiones básicas relacionadas con el trabajo colaborativo: conocimiento, articulación y apropiación. También fomenta la gestión de proyectos como un proceso de creación de valor compartido con equipos que trabajan e interactúan conjuntamente para ofrecer el mayor valor.

4. **Priorización basada en valor**—Este principio pone de relieve el enfoque de Scrum para ofrecer el máximo valor de negocio, desde el principio del proyecto hasta su conclusión.

5. **Time-boxing—**Este principio describe cómo el tiempo se considera una restricción limitante en Scrum, y cómo este se utiliza para ayudar a manejar eficazmente la planificación y ejecución del proyecto. Los elementos del time boxing en Scrum incluyen sprints, Daily Standups, reuniones de planificación del sprint y reuniones de revisión del sprint.

6. **Desarrollo iterativo—**Este principio define el desarrollo iterativo y hace énfasis en cómo gestionar mejor los cambios y crear productos que satisfagan las necesidades del cliente. También delinea las responsabilidades del Product Owner y las de la organización relacionadas con el desarrollo iterativo.

1.4.3 Aspectos de Scrum

Los aspectos de Scrum deben abordarse y gestionarse durante todo un proyecto Scrum. Los cinco aspectos de Scrum que se presentan en los capítulos del 3 al 7 son los siguientes:

1.4.3.1 Organización

Entender los roles y responsabilidades definidos en un proyecto Scrum es muy importante a fin de asegurar la implementación exitosa de Scrum.

Los roles de Scrum se dividen en dos amplias categorías:

1. **Roles centrales**—Los roles centrales son aquellos que se requieren obligadamente para crear el producto o servicio del proyecto. Las personas a quienes se les asignan los roles centrales están plenamente comprometidas con el proyecto y son las responsables del éxito de cada iteración del mismo, así como del proyecto en su totalidad.

   Estos roles incluyen:

   - El **Product Owner** es la persona responsable de lograr el máximo valor empresarial para el proyecto. Este rol también es responsable de la articulación de requisitos del cliente y de mantener la justificación del negocio para el proyecto. El Product Owner representa la voz del cliente.

   - El **Scrum Master** es un facilitador que asegura que el Equipo Scrum cuente con un ambiente propicio para completar el proyecto con éxito. El Scrum Master guía, facilita y enseña las prácticas de Scrum a todos los involucrados en el proyecto; elimina los impedimentos que pueda tener el equipo y se asegura de que se estén siguiendo los procesos de Scrum.

   - El **Equipo Scrum** es el grupo o equipo de personas responsables de entender los requisitos especificados por el Product Owner y de crear los entregables del proyecto.

2. **Roles no centrales**—Los roles no centrales son los que no son necesariamente obligatorios para el proyecto Scrum, y estos pueden incluir a miembros de los equipos que estén interesados en el proyecto. No tienen ningún rol formal en el equipo del proyecto, y pueden interactuar con el equipo, pero pueden no ser responsables del éxito del proyecto. Los roles no centrales deben tenerse en cuenta en cualquier proyecto de Scrum.

   Los roles no centrales incluyen los siguientes:

   - **Stakeholder(s)** es un término colectivo que incluye a clientes, usuarios y patrocinadores, que con frecuencia interactúan con el equipo principal de Scrum, e influyen en el proyecto a lo largo de su desarrollo. Lo más importante es que el proyecto produzca beneficios colaborativos para los stakeholders.

   - El **Scrum Guidance Body** (SGB) es un rol opcional, que generalmente consiste en un conjunto de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de los objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros parámetros claves de la organización. El SGB guía el trabajo llevado a cabo por el Product Owner, el Scrum Master y el Equipo Scrum.

   - Los **vendedores**, incluyendo a individuos u organizaciones externas, ofrecen productos y/o servicios que no están dentro de las competencias centrales de la organización del proyecto.

La figura 1-4 ilustra la estructura de la organización Scrum.

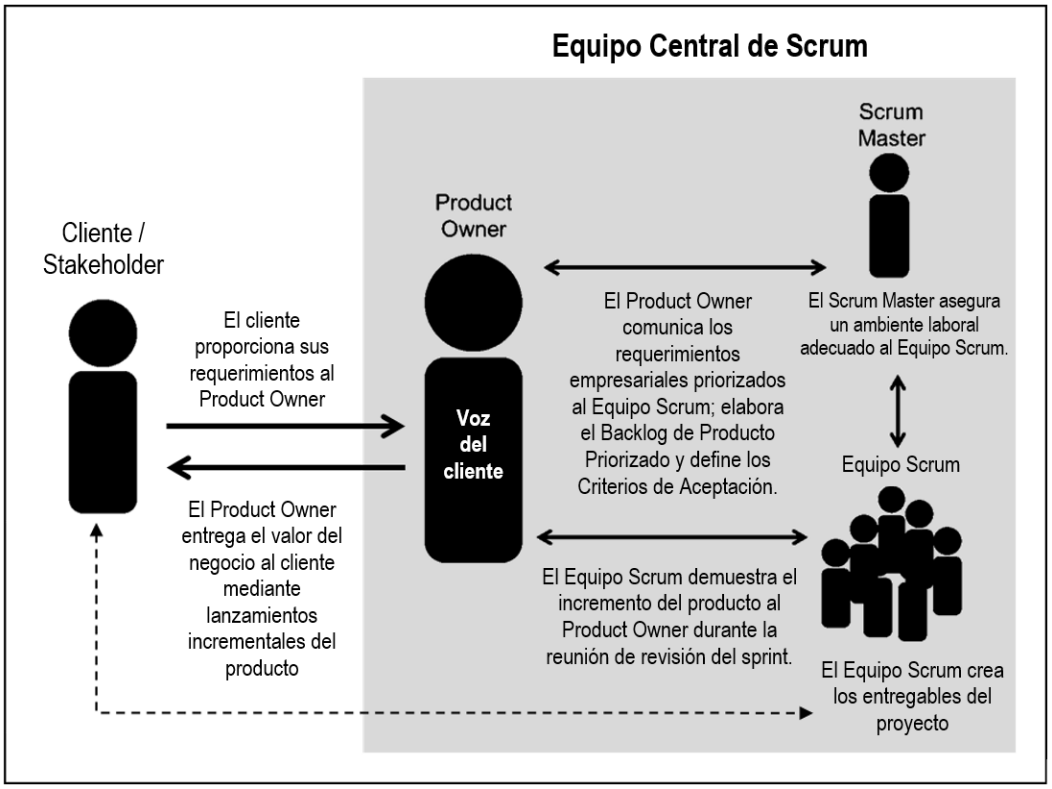


Figura 1-4: Organización en Scrum

El aspecto de organización de Scrum aborda también los requisitos de estructura del equipo para implementar Scrum en grandes proyectos, programas y portafolios.

1.4.3.2 Justificación del negocio

Es importante que una organización lleve a cabo una evaluación adecuada del negocio antes de iniciar cualquier proyecto. Esto ayuda a los tomadores de decisiones clave a entender la necesidad de cambio en la empresa o de un nuevo producto o servicio, la justificación para seguir adelante con un proyecto y su viabilidad.

En Scrum, la justificación del negocio se basa en el concepto de entrega impulsada por el valor (*Value-driven Delivery*). Una de las características claves de cualquier proyecto es la incertidumbre sobre los resultados. Es imposible garantizar el éxito de un proyecto, independientemente del tamaño o la complejidad del mismo. Considerando esta inseguridad de alcanzar el éxito, Scrum busca iniciar la entrega de resultados lo antes posible en el proyecto. Esta entrega temprana de resultados, y por lo tanto de valor, proporciona una oportunidad para la reinversión y demuestra el valor del proyecto a los stakeholders interesados.

La adaptabilidad de Scrum permite que los objetivos y procesos del proyecto cambien si cambia su justificación del negocio. Es importante señalar que, si bien el Product Owner es el responsable principal de la justificación del negocio, otros miembros del equipo también contribuyen considerablemente.

1.4.3.3 Calidad

En Scrum, la calidad se define como la capacidad con la que cuenta el producto o los entregables para cumplir con los criterios de aceptación y de alcanzar el valor de negocio que el cliente espera.

Para garantizar que un proyecto cumpla con los requisitos de calidad, Scrum adopta un enfoque de mejora continua mediante el cual el equipo aprende de sus experiencias y de la participación de los stakeholders para mantener constantemente actualizado el Backlog Priorizado del Producto con cualquier cambio en los requisitos. El Backlog Priorizado del Producto nunca se completa sino hasta el cierre o conclusión del proyecto. Cualquier cambio en los requisitos debe reflejar los cambios en el entorno empresarial, ya sean internos o externos, y permitirle al equipo trabajar continuamente y adaptarse para lograr dichos requerimientos.

Debido a que Scrum requiere que el trabajo se realice en incrementos durante los sprints, esto hace que los errores o defectos se noten con más facilidad mediante pruebas de calidad repetitivas y no simplemente cuando el producto final o servicio esté casi terminado. Por otra parte, las tareas relacionadas a la calidad (por ejemplo, desarrollo, pruebas y documentación) se completan como parte del mismo sprint por el mismo equipo. Esto asegura que la calidad sea inherente a cualquier entregable que se crea como parte de un sprint. A tales entregables de proyectos Scrum, que son potencialmente enviables, se les conoce como “terminado”.

Por lo tanto, la mejora continua con pruebas repetitivas optimiza la probabilidad de alcanzar los niveles esperados de calidad en un proyecto Scrum. Las discusiones constantes entre el equipo principal de Scrum y los stakeholders (incluyendo los clientes y los usuarios), junto con incrementos reales del producto que se entregan al final de cada sprint, aseguran que la brecha entre las expectativas de los clientes del proyecto y los verdaderos entregables se reduzca constantemente.

El Scrum Guidance Body también puede proporcionar directrices sobre la calidad que pueden ser de interés para todos los proyectos Scrum en la organización.

1.4.3.4 Cambio

Cada proyecto, independientemente del método o framework que se utilice, está expuesto a cambios. Es importante que los miembros del equipo del proyecto entiendan que los procesos de desarrollo de Scrum están diseñados para aceptar el cambio. Las organizaciones deben tratar de maximizar los beneficios que se deriven de los cambios y minimizar cualquier impacto negativo a través de procesos de gestión de cambio diligentes, según los principios de Scrum.

Un principio fundamental de Scrum es su reconocimiento de que **a)** los stakeholders (clientes, usuarios y patrocinadores) cambian de opinión acerca de lo que quieren y lo que necesitan durante un proyecto (a esto se le conoce en ocasiones como: "requisitos volátiles" o *requirements churn*); y, **b)** que es muy difícil, si no es que imposible, que los stakeholders definan todos los requisitos al inicio del proyecto.

Los proyectos Scrum aceptan los cambios mediante el uso de sprints breves e iterativos que incorporan la retroalimentación del cliente en cada entregable del sprint. Esto permite que el cliente interactúe regularmente con los miembros del Equipo Scrum, que vea los entregables a medida que estén listos y que cambie los requisitos si es necesario antes del siguiente sprint.

Asimismo, los equipos de gestión de programa o portafolio pueden responder a las solicitudes de cambio pertenecientes a los proyectos Scrum aplicables a su nivel.

1.4.3.5 Riesgo

El riesgo se define como un evento incierto o serie de eventos que pueden afectar los objetivos de un proyecto y pueden contribuir a su éxito o fracaso. A los riegos que pueden tener un impacto positivo en el proyecto se les conoce como oportunidades, mientras que las amenazas son riesgos que pudieran afectar negativamente al proyecto. La gestión de riesgos debe hacerse de forma preventiva, y es un proceso iterativo que debe comenzar al inicio del proyecto y continuar a lo largo del ciclo de vida del mismo. El proceso de gestión de riesgos debe seguir algunos pasos estandarizados para asegurar que estos se identifiquen y evalúen, y que se determine un curso adecuado de acción y se proceda en consecuencia.

Los riesgos deben ser identificados, evaluados y atendidos con base a dos factores: la probabilidad de ocurrencia de cada riesgo y el posible impacto en el caso de tal ocurrencia. Los riesgos con una alta probabilidad y valor de impacto (que se calcula multiplicando ambos factores) deben ser atendidos primero que aquellos con un valor relativamente bajo. En general, una vez que se detecta un riesgo, es importante entender el mismo en relación con las causas probables y los posibles efectos.

1.4.4 Procesos de Scrum

Los procesos de Scrum abordan las actividades específicas y el flujo de un proyecto de Scrum. En total hay diecinueve procesos fundamentales de Scrum que aplican a todos los proyectos. Estos procesos se agrupan en cinco fases y se presentan en los capítulos del 8 al 12 de la *Guía SBOK*™ tal como se muestra en la Tabla 1-1.

Capítulo	Fase	Procesos fundamentales de Scrum
8	Inicio	1. Crear la visión del proyecto 2. Identificar al Scrum Master y Stakeholder(s) 3. Formar Equipos Scrum 4. Desarrollar épica(s) 5. Crear el Backlog Priorizado del Producto 6. Realizar la planificación de lanzamiento
9	Planificación y estimación	7. Crear historias de usuario 8. Estimar historias de usuario 9. Comprometer historias de usuario 10. Identificar tareas 11. Estimar tareas 12. Crear el Sprint Backlog
10	Implementación	13. Crear entregables 14. Realizar Daily Standup 15. Refinar el Backlog Priorizado del Producto
11	Revisión y retrospectiva	16. Demostrar y validar el sprint 17. Retrospectiva del sprint
12	Lanzamiento	18. Enviar entregables 19. Retrospectiva del proyecto

Tabla 1-1: Resumen de los procesos fundamentales de Scrum

Estas fases describen a detalle cada proceso, incluyendo sus entradas, herramientas y salidas asociadas. En cada proceso, algunas entradas, herramientas y salidas son obligatorias (las que tienen un asterisco [*]), mientras que otras son opcionales. La inclusión de las entradas, herramientas y/o salidas opcionales dependerá del proyecto en particular, de la organización o la industria. Las entradas, herramientas y salidas señaladas con un asterisco son consideradas obligatorias o importantes para la implementación exitosa de Scrum en cualquier organización.

Para proyectos Scrum a grande escala que requieren de una coordinación entre múltiples equipos, existen tres procesos adicionales de Scrum que se definen en el capítulo 13: Escalar Scrum en grandes proyectos (*Scaling Scrum for Large Projects*). Existen también procesos específicos definidos cuando se implementa Scrum al nivel empresarial, lo cual se aborda en el capítulo 14: Escalar Scrum para la empresa (*Scaling Scrum for the Enterprise*). Estos procesos adicionales de Scrum se resumen en la tabla 1-2.

Capítulo	Aplicabilidad	Procesos adicionales de Scrum
13	Scrum para grandes proyectos	1. Crear componentes de grandes proyectos 2. Realizar y coordinar sprints 3. Preparar el lanzamiento de grandes proyectos
14	Scrum para la empresa	1. Crear componentes de programa o portafolio 2. Revisar y actualizar el Scrum Guidance Body 3. Crear y refinar el backlog del programa o portafolio 4. Coordinar los componentes del programa o portafolio 5. Retrospectiva de los lanzamientos del programa o portafolio

1.4.4.1 Inicio

1. *Crear la visión del proyecto*—En este proceso se revisa el caso de negocio del proyecto (*Project Business Case*) a fin de crear una Declaración de la visión del proyecto, que servirá de inspiración y proporcionará un enfoque para todo el proyecto. En este proceso se identifica al Product Owner.

2. *Identificar al Scrum Master y Stakeholder(s)*—En este proceso se identifica al Scrum Master y stakeholders utilizando criterios de selección específicos.

3. *Formar Equipos Scrum*—En este proceso se identifican a los miembros del Equipo Scrum. Normalmente, el Product Owner es el responsable principal de la selección de los miembros del equipo, pero con frecuencia lo hace en colaboración con el Scrum Master.

4. *Desarrollar épica(s)*—En este proceso la Declaración de visión del proyecto sirve como base para el desarrollo de épicas. Se pueden llevar a cabo reuniones de grupos de usuarios para hablar sobre las épicas adecuadas.

5. *Crear el Backlog Priorizado del Producto*—En este proceso se refinan y se crean las épicas, y después se priorizan para crear un Backlog Priorizado del Producto para el proyecto. A este punto también se establecen los criterios de terminado.

6. *Realizar la planificación del lanzamiento*—En este proceso el equipo principal de Scrum revisa las historias de usuario en el Backlog Priorizado del Producto para desarrollar un cronograma de planificación del lanzamiento, que es esencialmente un programa de implementación por fases que se puede compartir con los stakeholders del proyecto. En este proceso también se determina la duración del sprint.

1.4.4.2 Planificación y estimación

7. *Crear historias de usuario*—En este proceso se crean las historias de usuario y los criterios de aceptación de las historias de usuario. Las historias de usuario generalmente las escribe el Product Owner y están diseñadas para asegurar que los requisitos del cliente estén claramente representados y puedan ser plenamente comprendidos por todos los stakeholders. Se pueden llevar a cabo ejercicios de redacción de historias de usuario, lo cual incluyan a los miembros del Equipo Scrum, resultando en la creación de dichas historias. Estas se incorporan al Backlog Priorizado del Producto.

8. *Estimar historias de usuario*—En este proceso, el Product Owner aclara las historias de usuario para que el Scrum Master y el Equipo Scrum puedan estimar el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario.

9. *Comprometer historias de usuario*—En este proceso, el Equipo Scrum se compromete a entregar al Product Owner las historias de usuario aprobadas para un sprint. El resultado de este proceso serían las historias de usuario comprometidas.

10. *Identificar tareas*—En este proceso, las historias de usuario comprometidas se desglosan en tareas específicas y se compilan en una lista de tareas.

11. *Estimar tareas*—En este proceso, el equipo principal de Scrum estima el esfuerzo necesario para cumplir con cada tarea en la lista de tareas. El resultado de este proceso es una: Effort Estimated Task List.

12. *Crear el Sprint Backlog*—En este proceso, el equipo principal de Scrum elabora un Sprint Backlog que contiene todas las tareas a ser completadas en un sprint como parte de la Reunión de Planificación del Sprint.

1.4.4.3 Implementación

13. *Crear entregables*—En este proceso, el Equipo Scrum trabaja en las tareas en el Sprint Backlog para crear los entregables del sprint. Generalmente se utiliza un Scrumboard para dar seguimiento a las actividades que se llevan a cabo. Las asuntos o problemas que enfrenta el equipo Scrum pudieran actualizar se en un Impediment Log (o registro de impedimentos).

14. *Realizar Daily Standup*—En este proceso, se lleva a cabo diariamente una reunión altamente focalizada con un time-box, conocida como Daily Standup. Es aquí donde los miembros del Equipo Scrum se actualizan el uno al otro referente a sus progresos y sobre los impedimentos que pudieran enfrentar.

15. *Refinamiento del Backlog Priorizado del Producto*—En este proceso, el Backlog Priorizado del Producto se actualiza y se refina continuamente. Se puede considerar realizar una reunión de revisión del Backlog Priorizado del Producto, en la que se analiza cualquier cambio o actualización al backlog y se incorpora a dicho backlog según sea necesario.

1.4.4.4 Revisión y retrospectiva

16. *Demostrar y validar el sprint*—En este proceso, el Equipo Scrum muestra los entregables del sprint al Product Owner y a los stakeholders relevantes en una Reunión de Revisión del Sprint. El propósito de esta reunión es asegurar que se obtenga la aprobación y aceptación del Product Owner respecto a los entregables elaborados en el sprint.

17. *Retrospectiva del sprint*—En este proceso, el Scrum Master y el Equipo Scrum se reúnen para analizar las lecciones aprendidas durante todo el Sprint. Esta información se documenta en forma lecciones aprendidas que pueden aplicarse a futuros sprints. Frecuentemente, como resultado de esta discusión, puede haber mejoras aceptadas (*Agreed Actionable Improvements*) o recomendaciones actualizadas por parte del Scrum Guidance Body

1.4.4.5 Lanzamiento

18. *Enviar entregables*—En este proceso, los entregables aceptados se entregan o se envían a los stakeholders relevantes. Un documento denominado *Working Deliverables Agreement* (Acuerdo de entregables funcionales) documenta la conclusión satisfactoria del sprint.

19. *Retrospectiva del proyecto*—En este proceso, mismo que concluye el proyecto, los stakeholders y miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones aprendidas. A menudo, estas lecciones llevan a la documentación de Agreed Actionable Improvements, que se implementarán en futuros proyectos.

1.4.4.6 Scrum para grandes proyectos

Crear componentes de grandes proyectos—Este proceso define la forma en la que los Product Owners trabajan en conjunto y de cómo varios equipos de Scrum trabajan juntos. También se identifican componentes comunes, así como recursos comunes y especializados.

Realizar y coordinar sprints—Este proceso generalmente solo es relevante en grandes proyectos y aborda aspectos específicos que deben ser considerados durante cada sprint. De ser necesario, se pueden llevar a cabo reuniones de Scrum de Scrums a fin de coordinar los esfuerzos entre los distintos equipos de Scrum.

Preparar el lanzamiento de grandes proyectos—En algunos proyectos grandes, pudiera tener sentido empresarial llevar a cabo un sprint especial con anticipación a un lanzamiento a fin de preparar el lanzamiento del producto (que será decidido por el equipo del proyecto con base en las necesidades del negocio). Este proceso aborda dicho sprint preparatorio.

1.4.4.7 Scrum para la empresa

Crear componentes de programa o portafolio—En este proceso, el Program Product Owner o el Portfolio Product Owner, así como los stakeholders clave identifican componentes comunes y

recursos necesarios para el programa o portafolio. Los criterios mínimos de terminado se definen y se identifican a todos los stakeholders.

Revisar y actualizar el Scrum Guidance Body—En este proceso, las recomendaciones del Scrum Guidance Body se revisan constantemente por parte de sus miembros y se actualizan cuando sea necesario. En este proceso, también se atienden los cambios en los integrantes del Scrum Guidance Body.

Crear y refinar el backlog de programa o portafolio—En este proceso, se elabora, se actualiza y se refina el backlog del programa o portafolio. Se pueden hacer recomendaciones de mejoramiento por parte del Scrum Guidance Body y se pueden modificar los plazos con base en los cambios en los requerimientos y/o procesos del proyecto en el programa o portafolio.

Coordinar los componentes del programa o portafolio—En este proceso se coordinan los componentes del programa o portafolio. Se atienden las dependencias entre proyectos; se discuten los impedimentos comunes y se comparten las mejores prácticas. En ocasiones, se hacen recomendaciones de mejoramiento por parte del Scrum Guidance Body.

Retrospectiva de lanzamientos del programa o portafolio—En este proceso, el Program Product Owner o el Portfolio Product Owner y los stakeholders clave se reúnen para hacer una retrospectiva sobre el lanzamiento de un programa o portafolio e internalizar las lecciones aprendidas. Por lo general dichas lecciones aprendidas llevan a mejoras aceptadas (*Agreed Actionable Improvements*) para ser implementadas a futuro.

1.5 Scrum vs. Gestión tradicional de proyectos

La tabla 1-2 resume muchas de las diferencias entre los modelos tradicionales de gestión de proyectos.

	Scrum	Gestión tradicional de proyectos
El énfasis está en	Las personas	Los procesos
Documentación	Sólo mínima; según se requiera	Integral
Estilo de procesos	Iterativo	Lineal
Planificación por adelantado	Baja	Alta
Priorización de requerimientos	Según el valor del negocio y regularmente actualizada	Fijo en el plan de proyecto
Garantía de calidad	Centrada en el cliente	Centrada en el proceso
Organización	Auto-organizada	Gestionada
Estilo de gestión	Descentralizado	Centralizado
Cambio	Actualizaciones al Backlog Priorizado del Producto	Sistema formal de gestión del cambio
Liderazgo	Liderazgo colaborativo y servicial	Mando y control
Medición del rendimiento	El valor del negocio	Conformidad con el plan
Retorno sobre la inversión (RSI)	Al comienzo y a lo largo del proyecto	Al final del proyecto
Participación del cliente	Alta durante todo el proyecto	Varía dependiendo del ciclo de vida del proyecto

Tabla 1-2: Scrum vs. Gestión tradicional de proyectos

2. PRINCIPIOS

2.1 Introducción

Los principios de Scrum son el fundamento sobre el que se basa su framework. Estos principios pueden aplicarse a cualquier tipo de proyecto u organización y deben respetarse a fin de garantizar la aplicación adecuada de Scrum. Los aspectos y procesos de Scrum pueden modificase para cumplir con los requerimientos del proyecto, o la organización que lo usa, pero sus principios no están abiertos a discusión ni pueden modificarse, y deben aplicarse como se describe en el framework presentado en *Una guía para el conocimiento de Scrum (Guía SBOK™).* El mantener los principios intactos y usarlos apropiadamente infunde confianza en el usuario del framework de Scrum respecto al cumplimiento de los objetivos del proyecto. Los principios se consideran los lineamientos básicos para la aplicación del framework de Scrum.

Los *principios,* tal como se definen en la *Guía SBOK™,* son aplicables a lo siguiente:

- Portafolios, programas, y/o proyectos de *cualquier* industria
- Productos, servicios, o cualquier otro resultado que se entregue a los stakeholders
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la *Guía SBOK™* puede ser un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos pequeños o equipos con tan sólo seis miembros, hasta proyectos grandes y complejos con hasta varios cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

2.2 Guía de los roles—Aquí se describe qué sección o subsección es más relevante para cada uno de los roles centrales de Scrum tales como el Product Owner, Scrum Master y Equipo Scrum.

2.3 Control de proceso empírico—Esta sección describe el primer principio de Scrum y las tres ideas principales de transparencia, inspección y adaptación.

2.4 Auto-organización—Esta sección destaca el segundo principio de Scrum que se enfoca en los trabajadores de hoy en día, quienes entregan un valor considerablemente mayor cuando se auto-organizan. Esta auto-organización se traduce en un mejor sentido de compromiso del equipo y responsabilidad compartida; esto a su vez proporciona un ambiente de trabajo innovador y creativo que es más propicio para el crecimiento.

2.5 Colaboración—Esta sección hace énfasis en el tercer principio de Scrum donde el desarrollo de los productos es un proceso de creación de valor compartido que necesita que todos los stakeholders trabajen e interactúen en conjunto para ofrecer el mayor valor. También se centra en las dimensiones básicas de trabajo colaborativo: conocimiento, articulación y apropiación.

2.6 Priorización basada en el valor—Esta sección presenta el cuarto principio de Scrum, que pone de relieve la unidad del framework de Scrum para entregar el máximo valor empresarial en un período.

2.7 Time boxing—Esta sección explica el quinto principio de Scrum que aborda el tiempo como un factor limitante. También aborda el sprint, el Daily Standup y otros sprints relacionados con las reuniones, tales como la Reunión de Planificación del Sprint y la Reunión de Revisión del Sprint, las cuales están bajo un time-box asignado.

2.8 Desarrollo iterativo—En esta sección se aborda el sexto principio de Scrum, que hace énfasis en el desarrollo iterativo, mismo que ayuda a gestionar mejor los cambios y crear productos que satisfagan las necesidades del cliente.

2.9 Scrum vs. Gestión tradicional de proyectos—Esta sección destaca las principales diferencias entre los principios de Scrum y los principios de gestión tradicional de proyectos (modelo en cascada) y explica cómo Scrum funciona mejor en el mundo tan cambiante de hoy en día.

2.2 Guía de roles

Todas las secciones de este capítulo son importantes para todos los roles del equipo principal de Scrum: Product Owner, Scrum Master y Equipo Scrum. Es esencial que exista una comprensión clara de los principios de Scrum por parte de todos stakeholders para que el framework de Scrum sea exitoso en cualquier organización.

2.3 Control del proceso empírico

En Scrum, las decisiones se basan en la observación y la experimentación en vez de la planificación inicial detallada. El control del proceso empírico se basa en las tres ideas principales de la transparencia, inspección y adaptación.

2.3.1 Transparencia

La transparencia permite que todas las facetas de cualquier proceso de Scrum sean observadas por cualquiera. Esto promueve un flujo de información fácil y transparente en toda la organización y crea una cultura de trabajo abierta. En Scrum, la transparencia se representa mediante lo siguiente:

- Una declaración de la visión del proyecto (*Project Vision Statement*) que pueden ver todos los stakeholders y el Equipo Scrum.
- Un Backlog Priorizado del Producto abierto con historias de usuario priorizadas que todos pueden ver tanto dentro como fuera del Equipo Scrum.

- Un cronograma de planificación del lanzamiento (*Release Planning Schedule*) que se puede coordinar a través de múltiples equipos Scrum
- Una clara visibilidad sobre el progreso del equipo a través del uso de Scrumboard, Burndown Chart y otros radiadores de información
- Daily Standups que se llevan a cabo durante el proceso de *Realizar Dialy Standup* en las que todos los miembros del equipo informan sobre lo que hicieron el día anterior, lo que van a hacer hoy y cualquier problema que les impida completar sus tareas en el sprint actual
- Las reuniones de revisión del sprint se llevan a cabo durante el proceso de *Demostrar y validar el sprint*, donde el Equipo Scrum muestra los entregables del sprint que potencialmente se pueden enviar a los Product Owners y a los stakeholders.

La figura 2-1 resume el concepto de transparencia en Scrum.

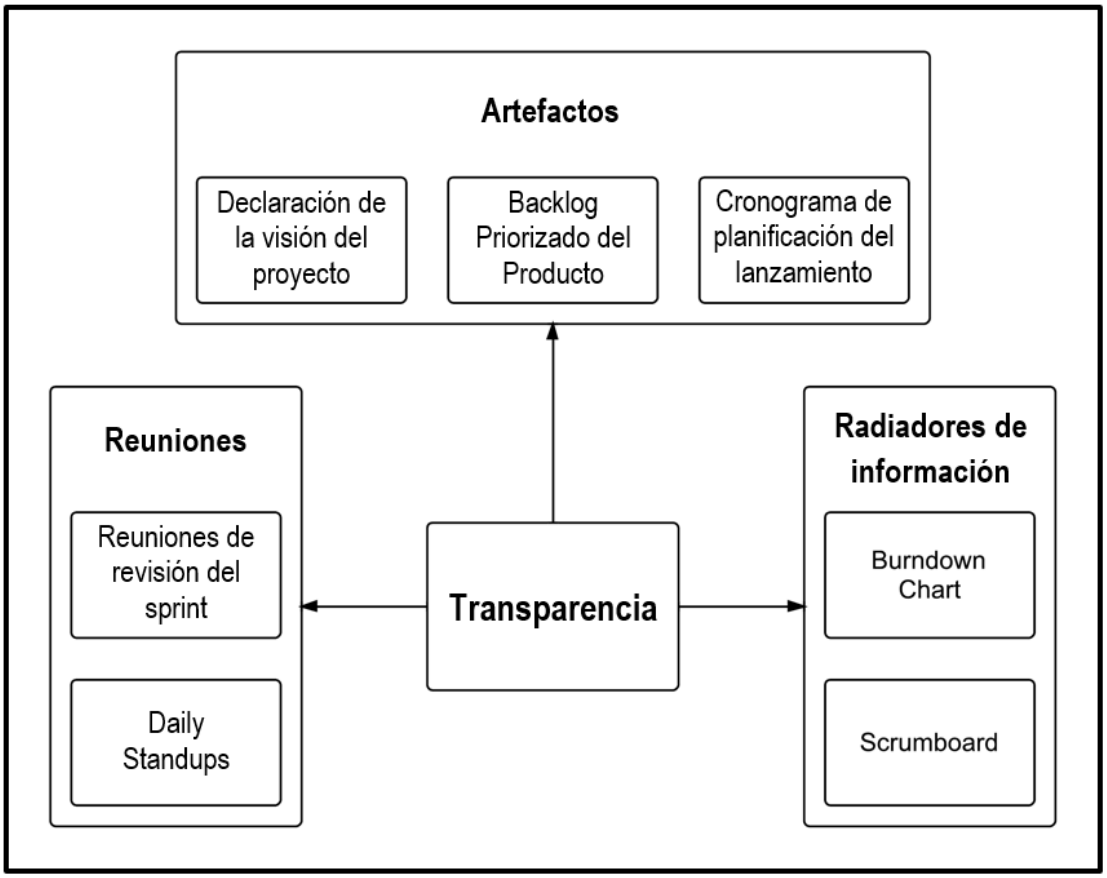


Figura 2-1: Transparencia en Scrum

2.3.2 Inspección

La inspección en Scrum se representa mediante lo siguiente:

- Uso de un Scrumboard común y otros radiadores de información que muestran el progreso del Equipo Scrum en completar las tareas del sprint actual.
- Recopilación de la retroalimentación del cliente y otros stakeholders durante los procesos de *Desarrollar de épica(s)*, *Crear Backlog Priorizado del Producto* y *Realizar planificación del lanzamiento.*
- La inspección y aprobación de los entregables por parte del Product Owner y el cliente en el proceso de *Demostrar y validar el sprint.*

La figura 2-2 resume el concepto de inspección en Scrum.

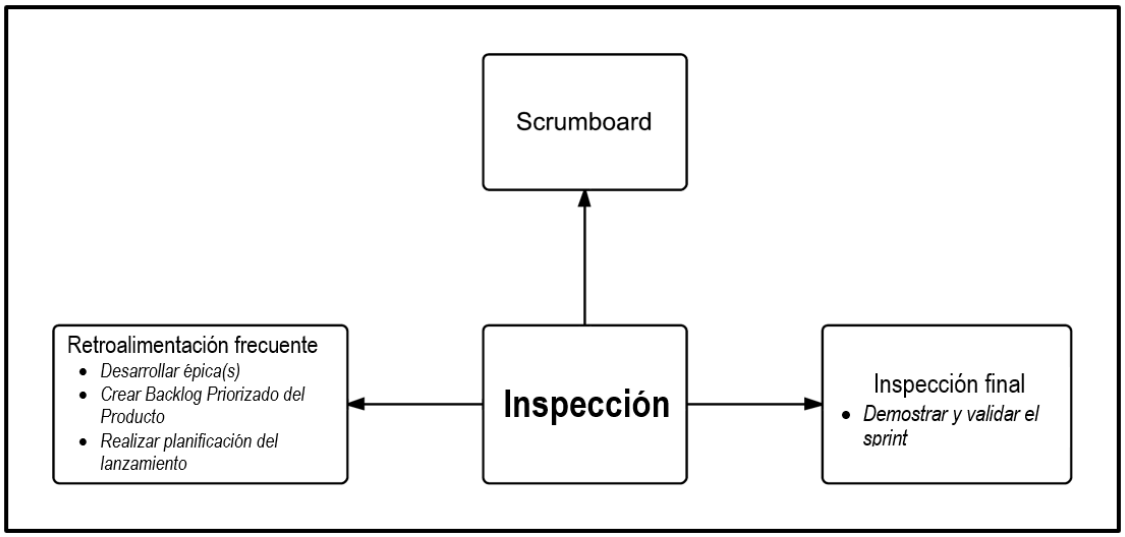


Figura 2-2: Inspección de Scrum

2.3.3 Adaptación

La adaptación se da cuando el equipo principal de Scrum y los stakeholders aprenden mediante la transparencia y la inspección, y después se adaptan al hacer mejoras en el trabajo que llevan a cabo. Algunos ejemplos de adaptación incluyen:

- En los Daily Standups, los miembros del Equipo Scrum hablan abiertamente sobre los impedimentos para completar sus tareas y buscan la ayuda de otros miembros del equipo. Los miembros con más experiencia en el Equipo Scrum también ayudan de aquellos quienes tienen relativamente menos experiencia y conocimiento del proyecto o de la tecnología.
- Se lleva a cabo la identificación del riesgo y se repite a lo largo del proyecto. Los riegos que se identifican se convierten en entradas para varios procesos de Scrum, incluyendo el de *Crear*

Backlog Priorizado del Producto, de *Refinamiento del Backlog Priorizado del Producto* y de *Demostrar y validar el sprint*.

- Las mejoras pueden resultar en solicitudes de cambios que se discuten y aprueban durante los procesos de *Desarrollar* épica(s), *Crear Backlog Priorizado del Producto* y *Refinamiento del Backlog Priorizado del Producto.*
- El Scrum Guidance Body interactúa con los miembros del Equipo Scrum durante los procesos de *Crear historias de usuario*, *Estimar tareas*, *Crear entregables* y *Refinamiento del Backlog Priorizado del Producto* para ofrecer orientación y también proporcionar conocimientos según sea necesario.
- En el proceso de *Retrospectiva del sprint* se determinan las mejoras aceptadas con base en las salidas del proceso de *Demostrar y validar el sprint*.
- En la reunión de retrospectiva del proyecto, los participantes documentan las lecciones aprendidas y realizan revisiones en busca de oportunidades para mejorar los procesos y atender ineficiencias.

La figura 2-3 resume el concepto de adaptación en Scrum.

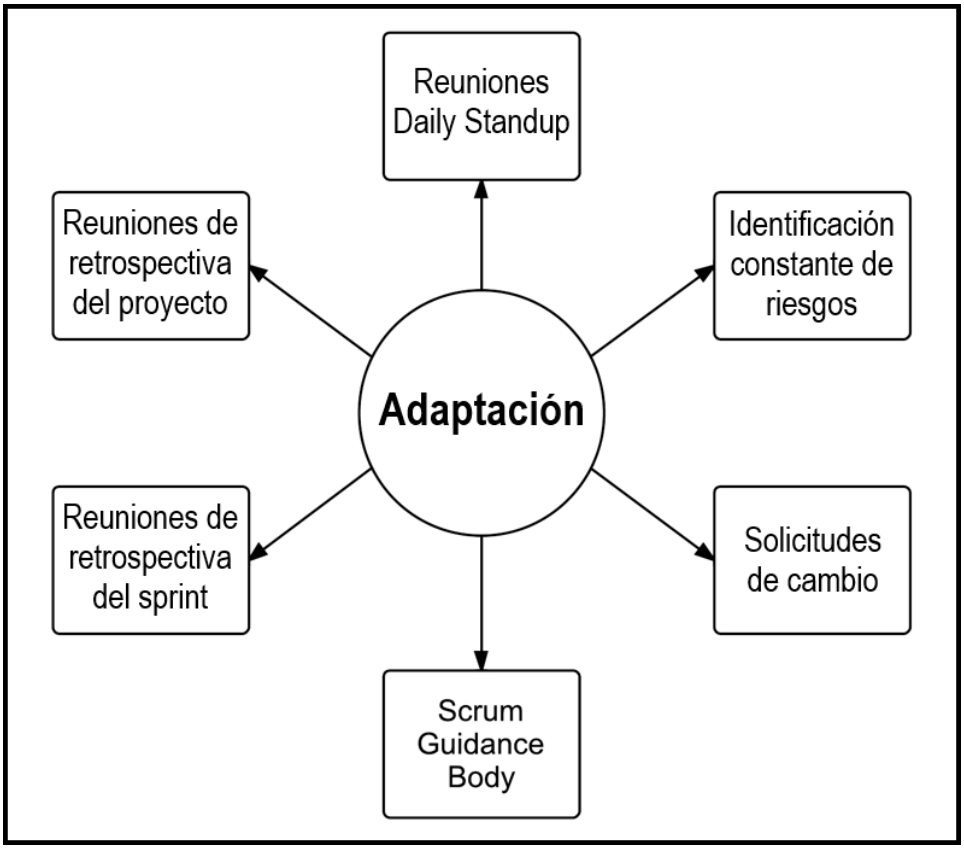


Figura 2-3: Adaptación en Scrum

Con otros métodos, como el modelo tradicional en Cascada, se requiere de una planificación considerable que debe hacerse por adelantado, y el cliente generalmente no revisa los componentes del producto hasta casi el final de una fase o del proyecto. Con frecuencia este método presenta enormes riesgos al éxito del proyecto, ya que tiene más potencial para impactar considerablemente la entrega de proyectos y la aceptación del cliente. La interpretación y comprensión del cliente sobre el producto final puede ser muy

diferente de lo que realmente se entendió originalmente y fue producido por el equipo, algo que no se sabría hasta muy tarde en el desarrollo del proyecto.

La figura 2-4 muestra un ejemplo de estos desafíos.

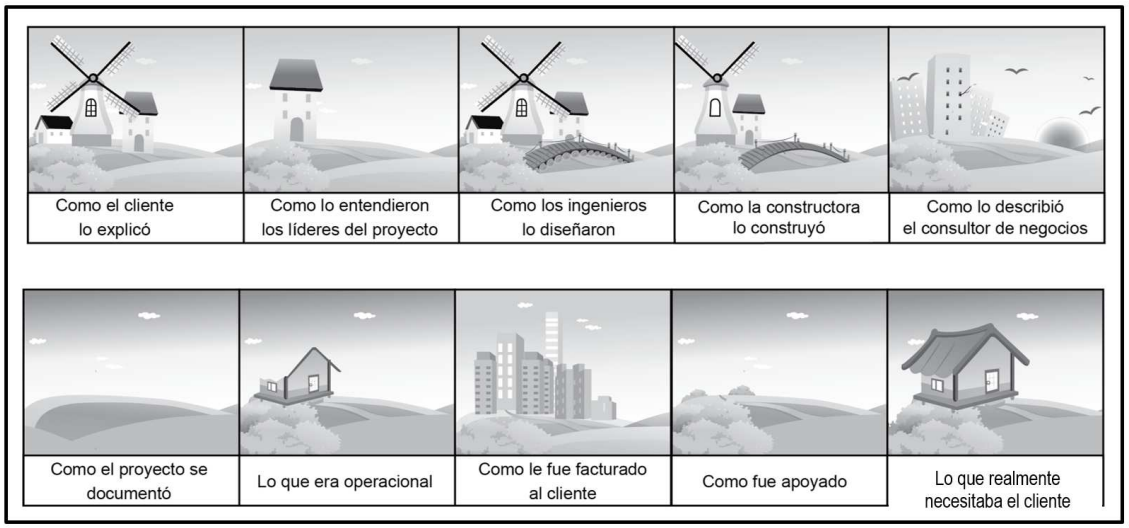


Figura 2-4: Retos en la gestión tradicional de proyectos

2.4 Auto-organización

Scrum sostiene que los empleados cuentan con motivación propia y que buscan aceptar mayores responsabilidades. Por tanto, ofrecen mucho más valor cuando se organizan por cuenta propia.

El estilo de liderazgo preferido en Scrum es el de "liderazgo servicial", el cual enfatiza el logro de resultados, centrándose en las necesidades del Equipo Scrum. Consulte la sección 3.10.3, en donde se describen varios estilos de liderazgo y de gestión.

2.4.1 Beneficios de la auto-organización

La auto-organización, como un principio esencial en Scrum, conduce a:

- Un sentido de compromiso del equipo y de responsabilidad compartida
- Motivación, lo cual conduce a un mejor nivel de rendimiento del equipo
- Un ambiente de trabajo innovador y creativo que conduzca al crecimiento

La auto-organización no significa que a los miembros del equipo se les permita actuar como quieran. Sólo significa que una vez que la visión del producto se define en el proceso de *Crear la visión del proyecto*, se identifica al Product Owner, al Scrum Master y al Equipo Scrum. Asimismo, el equipo principal de Scrum trabaja estrechamente con los stakeholders para perfeccionar los requisitos a medida que avanzan a través de los procesos de *Desarrollar épica(s)* y *Crear historias de usuario*. La experiencia del equipo se utiliza para evaluar las entradas necesarias para realizar la obra prevista del proyecto. Este juicio y experiencia se aplican a todos los aspectos técnicos y de gestión de proyectos durante el proceso de la *Crear entregables*.

Aunque la priorización la hace principalmente el Product Owner, quien representa la voz del cliente, el Equipo Scrum auto-organizado participa en la distribución y estimación de tareas durante los procesos de *Identificación de tareas* y *Estimación de tareas*. Durante estos procesos, cada miembro del equipo tiene la responsabilidad de determinar qué tipo de trabajo hará. Durante la ejecución de un sprint, los miembros del equipo tal vez necesiten ayuda para completar sus tareas. Scrum aborda esto mediante de la interacción constante obligatoria en las reuniones Daily Standup. El propio Equipo Scrum interactúa con otros equipos mediante las reuniones de Scrum de Scrums y, de ser necesario, puede buscar orientación adicional del Scrum Guidance Body.

Por último, el Equipo Scrum y el Scrum Master trabajan de cerca para demostrar el incremento del producto creado durante el sprint en el proceso de *Demostrar y validar el sprint*, donde se aceptan los entregables debidamente completados. Dado que los entregables son potencialmente enviables (y el Backlog Priorizado del Producto se prioriza por historias de usuario en el orden del valor creado por ellas), el Product Owner y el cliente pueden visualizar y articular claramente el valor creado después de cada sprint; y el Equipo Scrum a su vez tiene la satisfacción de obtener la aceptación de su por parte del cliente y los otros stakeholders.

Los principales objetivos de los equipos auto-organizados son los siguientes:

- Entender la visión del proyecto y por qué el proyecto aporta valor a la organización
- Estimar historias de usuario durante el proceso de *Estimar historias de usuario* y asignarse tareas durante el proceso de *Crear el Sprint Backlog*.
- Identificar tareas en forma independiente durante el proceso de *Identificación de tareas*
- Aplicar y aprovechar la experiencia de ser un equipo interfuncional al trabajar en las tareas durante el proceso de *Crear entregables*
- Entregar resultados tangibles que sean aceptados por el cliente y otros stakeholders durante el proceso de *Demostrar y validar el sprint*
- Resolver problemas individuales analizándolos durante los Daily Standups
- Aclarar cualquier discrepancia o duda y tener la disposición de aprender cosas nuevas
- Actualizar los conocimientos y habilidades de manera continua a través de constantes interacciones dentro del equipo
- Mantener la estabilidad de los miembros del equipo durante la duración del proyecto al no cambiar los miembros, a menos que sea inevitable

La figura 2-5 ilustra los objetivos de un equipo auto-organizado.

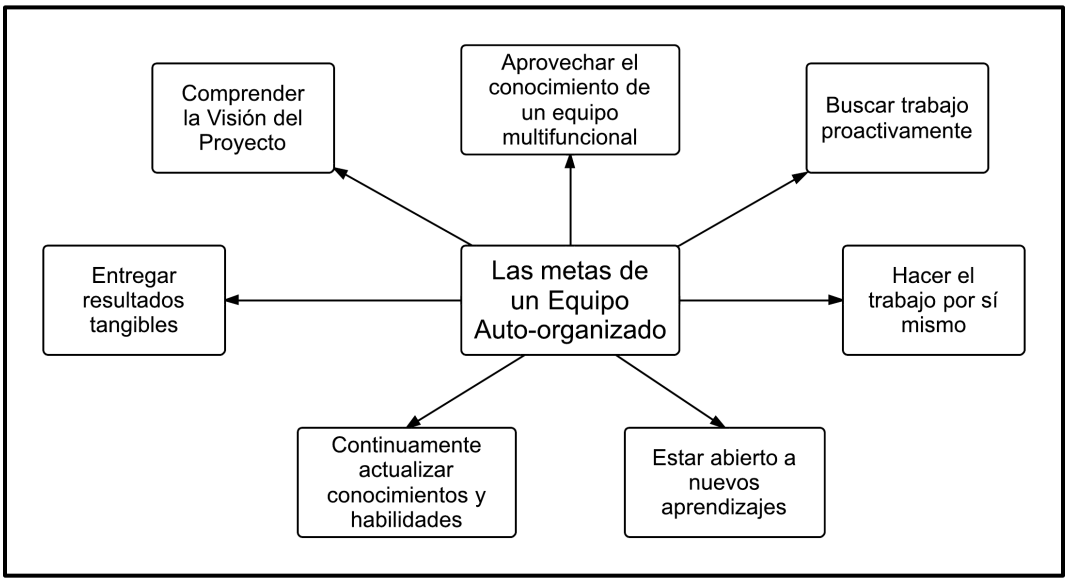


Figura 2-5: Objetivos de un equipo de auto-organizado

2.5 Colaboración

La colaboración en Scrum se refiere a que el equipo principal de Scrum trabaja e interactúa con los stakeholders para crear y validar los resultados del proyecto a fin de cumplir con los objetivos que se plantean en la visión del proyecto. Es importante tener en cuenta la diferencia entre cooperación y colaboración. La cooperación se da cuando el trabajo que se produce consiste en la suma de los esfuerzos del trabajo de varias personas en un equipo. La colaboración, en cambio, se produce cuando un equipo trabaja en conjunto para contraponer los aportes del otro a fin de producir algo más grande.

Las dimensiones básicas de trabajo en la colaboración son las siguientes:

- ***C****onocimiento*—Las personas que trabajan juntas deben estar al tanto del trabajo de los demás.
- ***A****rticulación*—Los colaboradores deben distribuir el trabajo en unidades; dividir las unidades entre los miembros del equipo y después reintegrarlo cuando el trabajo esté hecho.
- ***A****propiación*—Adaptar la tecnología a la situación individual; la tecnología se puede utilizar de forma completamente distinta a lo esperado por los diseñadores.

2.5.1 Beneficios de la colaboración en los proyectos Scrum

El Manifiesto Ágil (Fowler y Highsmith, 2001) hace énfasis en la "colaboración con el cliente sobre negociación contractual". Por lo tanto, el framework de Scrum adopta un enfoque donde los miembros del equipo principal de Scrum (el Product Owner, el Scrum Master y el Equipo Scrum) colaboran entre sí y con los stakeholders para crear los entregables que proporcionan el mayor valor posible para el cliente. Esta colaboración se produce durante todo el proyecto.

La colaboración asegura que los siguientes beneficios del proyecto se realicen:

1. La necesidad de cambios debido a requisitos mal clarificados se reduce al mínimo. Por ejemplo, durante los procesos de *Crear la visión del proyecto*, *Desarrollar épica(s)* y *Crear Backlog Priorizado del Producto*, el Product Owner colabora con los stakeholders para crear la visión del proyecto, la épica(s) y el Backlog Priorizado del Producto, respectivamente. Esto asegurará que haya claridad entre los miembros principales del Equipo Scrum sobre el trabajo que se requiere para completar el proyecto. El Equipo Scrum colabora continuamente con el Product Owner y los stakeholders a través de un Backlog Priorizado del Producto transparente para crear los entregables del proyecto. Los procesos de *Realizar Dialy Standup*, *Refinamiento del Backlog Priorizado del Producto* y la *Retrospectiva del sprint* dan margen a los miembros del equipo principal de Scrum para discutir lo que se ha hecho y colaborar en lo que hay que hacer. De esta manera se minimiza el número de solicitudes de cambio por parte del cliente.

2. Los riesgos se identifican y se atienden con eficiencia. Por ejemplo, los riesgos del proyecto se identifican y evalúan en los procesos del *Desarrollar épica(s)*, *Crear entregables* y *Realizar Daily*

Standup por parte de los miembros del equipo principal de Scrum. Las herramientas de la reunión de revisión de Scrum, tales como el Daily Standup, la Reunión de Planificación del Sprint, la Reunión de Revisión del Backlog Priorizado del Producto, entre otras, brindan oportunidades para el equipo no solo identifique y evalúe los riesgos, sino que implemente también respuestas para atender riesgos de alta prioridad.

3. Se logra el verdadero potencial del equipo. Por ejemplo, el proceso de *Realizar Daily Standup* ofrece un margen para que el Equipo Scrum colabore y entienda las fortalezas y debilidades de sus miembros. Si un miembro del equipo excedió el plazo para completar una tarea, los miembros del Equipo Scrum se alinean en colaboración para completarla y cumplir con los objetivos acordados para completar el sprint.

4. Se garantiza la mejora continua a través de las lecciones aprendidas. Por ejemplo, el Equipo Scrum utiliza el proceso de *Retrospectiva del sprint* para identificar lo que salió bien y lo que no salió bien en el sprint anterior. Esto proporciona una oportunidad para que el Scrum Master trabaje con el equipo y esté más preparado para el próximo sprint. Esto también garantiza que la colaboración sea aún más eficaz en el próximo sprint.

La figura 2-6 ilustra los beneficios de la colaboración en los proyectos Scrum.

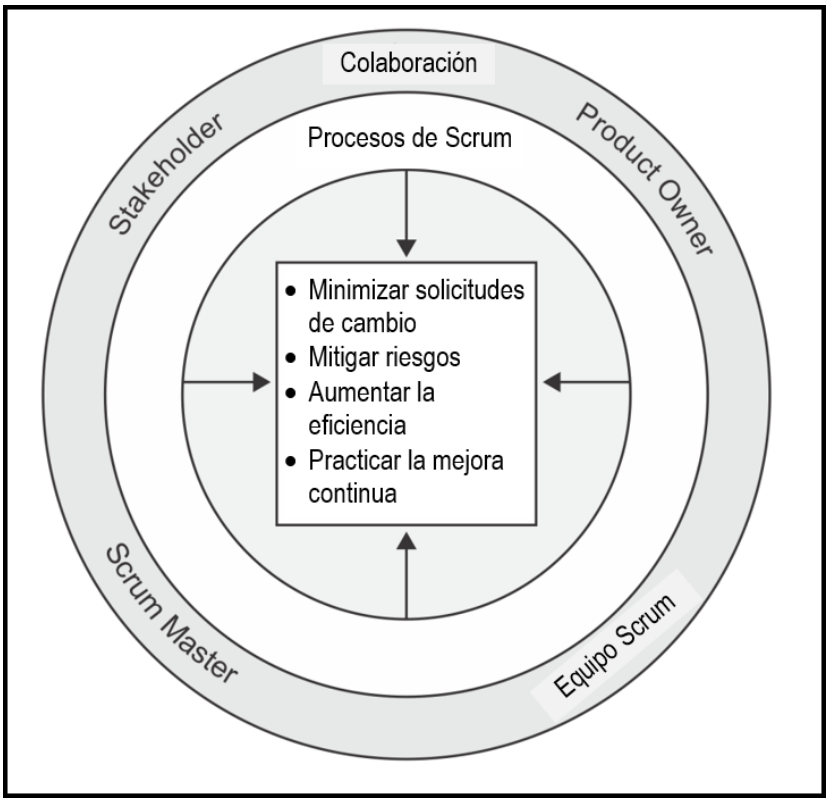


Figura 2-6: Beneficios de la colaboración en proyectos Scrum

2.5.2 La importancia de la co-ubicación en la colaboración

Para muchas de las prácticas de Scrum se requiere una comunicación de high-bandwidth. Por tanto, es preferible que los miembros del equipo estén co-ubicados. La co-ubicación (del inglés *colocation*—es decir, cuando todos los miembros del equipo del proyecto se encuentran trabajando en el mismo lugar), permite la interacción formal e informal entre los miembros del equipo. Esto brinda la ventaja de contar con miembros del equipo siempre a la mano para facilitar la coordinación, la resolución de problemas y el aprendizaje. Algunos de los beneficios de la co-ubicación son los siguientes:

- Las preguntas se contestan rápidamente.
- Los problemas se solucionan en ese momento.
- Existe menor fricción entre las interacciones.
- La confianza se gana con mucha más rapidez.

Las herramientas de colaboración que pueden utilizarse para los equipos co-ubicados o distribuidos geográficamente son las siguientes:

1. **Equipos co-ubicados** (equipos que trabajan en la misma oficina)—En Scrum, es preferible contar con equipos co-ubicados. Si los equipos están co-ubicados, los métodos de comunicación preferidos incluyen interacciones de cara a cara, salas de decisión o War Rooms, Scrumboards, información visual colocada en la pared, mesas compartidas, etcétera.

2. **Equipos distribuidos geográficamente** (es decir, equipos que trabajan en diferentes ubicaciones físicas)—Aunque es preferible contar con equipos co-ubicados, en ocasiones el Equipo Scrum puede estar disperso debido a la subcontratación, offshoring, diferentes ubicaciones físicas, las opciones de trabajo desde casa, etc. Algunas herramientas que podrían utilizarse para tener una colaboración eficaz entre los equipos distribuidos geográficamente incluyen videoconferencias, mensajes instantáneos, chats, redes sociales, pantallas compartidas y herramientas de software que simulan la funcionalidad de los Scrumboards, la información visual colocada en la pared, etc.

2.6 Priorización basada en valor (*Value-based Prioritization*)

El framework de Scrum se guía por la finalidad de ofrecer el máximo valor empresarial en un mínimo período de tiempo. Una de las herramientas más eficaces para entregar el mayor valor en el menor tiempo posible es la priorización.

La priorización se puede definir como la determinación del orden y la separación de lo que debe hacerse ahora, de lo que debe hacerse después. El concepto de priorización no es nuevo para la gestión de proyectos. El modelo tradicional de gestión de proyectos llamado Cascada o *Waterfall* propone el uso de múltiples herramientas de priorización. Desde el punto de vista del Project Manager, la priorización es integral debido a que ciertas tareas deben llevarse a cabo primero a fin de acelerar el proceso de desarrollo

y el cumplimiento de los objetivos del proyecto. Algunas de las técnicas tradicionales de la priorización de tareas incluyen el establecimiento de plazos para las tareas delegadas y el uso de matrices de priorización.

Sin embargo, Scrum utiliza la priorización basada en valor (*Value-based Prioritization*) como uno de los principios básicos que impulsa la estructura y funcionalidad de todo el framework de Scrum—ayuda a que los proyectos se beneficien mediante la capacidad de adaptación y el desarrollo iterativo del producto o servicio. Y lo que es más importante, Scrum tiene como finalidad entregar un producto o servicio valioso para el cliente en forma oportuna y continua.

La priorización se lleva a cabo por el Product Owner cuando prioriza las historias de usuario en el Backlog Priorizado del Producto. Este contiene una lista de todos los requisitos necesarios para llevar el proyecto a buen término.

Una vez que el Product Owner ha recibido los requerimientos del negocio del cliente y los ha escrito en forma de historias de usuario viables, este trabaja con el cliente y el patrocinador para entender los requerimientos del negocio que proporcionan el máximo valor empresarial. El Product Owner debe entender lo que quiere el cliente y valora a fin de organizar los elementos del Backlog Priorizado del Producto (historias de usuario), según su importancia relativa. En ocasiones, un cliente puede ordenar que todas las historias de usuario sean de alta prioridad. Aunque este pudiera ser el caso, incluso una lista de alta prioridad de historias de usuario también debe ser priorizada dentro de la lista misma. Darle prioridad a un backlog implica determinar la importancia de cada historia de usuario. Los requisitos de alto valor se identifican y trasladan a la parte superior del Backlog Priorizado del Producto. Los procesos por los cuales el principio de la priorización basada en valor se pone en práctica son la *Crear Backlog Priorizado del Producto* y *Refinamiento del Backlog Priorizado del Producto.*

Al mismo tiempo, el Product Owner debe trabajar con el Equipo Scrum para entender los riesgos y la incertidumbre del proyecto, ya que estos pueden tener consecuencias negativas. Esto riesgos se deben de tener en cuenta al priorizar las historias de usuario con enfoque basado en el valor (véase el capítulo de Riesgos para leer más información al respecto). El Equipo Scrum también alerta al Product Owner sobre las dependencias que surgen de la implementación. Estas dependencias deben tenerse en cuenta durante la priorización. Esta puede basarse en una estimación subjetiva del valor proyectado del negocio o la rentabilidad, o puede basarse en los resultados y análisis del mercado utilizando herramientas, incluyendo, pero sin limitarse a, entrevistas del cliente, encuestas y modelos financieros y técnicas analíticas.

El Product Owner debe interpretar las entradas y las necesidades de los proyectos de los stakeholders para crear el Backlog Priorizado del Producto. Por lo tanto, mientras se priorizan las historias de usuario en el Backlog Priorizado del Producto, se consideran los siguientes tres factores (véase la figura 2-7):

1. Valor
2. Riesgo o incertidumbre
3. Dependencias

De esta forma, la priorización resulta en entregables que satisfacen los requisitos del cliente con el objetivo de ofrecer el máximo valor de negocio en el menor tiempo posible.

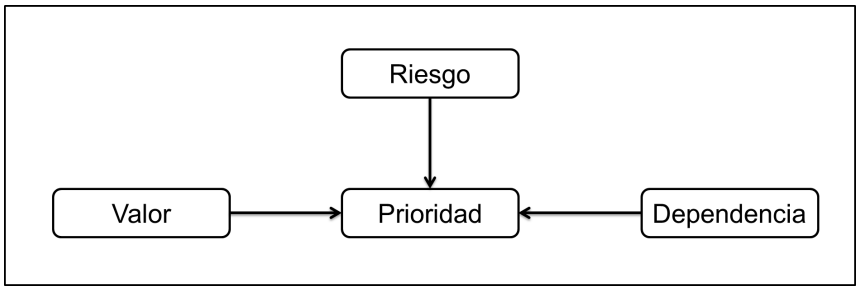


Figura 2-7: Priorización basada en valor

2.7 Time-boxing

Scrum trata al tiempo como uno de los limitantes más importantes en la gestión de un proyecto. Para hacer frente a la restricción del tiempo, Scrum introduce un concepto de Time-boxing (o asignación de un bloque de tiempo), que propone la fijación de una cierta cantidad de tiempo para cada proceso y actividad en un proyecto Scrum. Esto garantiza que los miembros del Equipo Scrum no ocupen demasiado o muy poco tiempo para un trabajo determinado, y que no desperdicien su tiempo y energía en un trabajo para el cual tienen poca claridad.

Algunas de las ventajas del Time-boxing son las siguientes:

- Proceso de desarrollo eficiente
- Menos gastos generales
- Alta velocidad para los equipos

El Time-boxing puede utilizarse en muchos procesos de Scrum, por ejemplo, en el proceso de *Realizar Daily Standup*, la duración de dicha reunión tiene un time-box asignado. A veces, el Time-boxing puede utilizarse para evitar la mejora excesiva de un elemento (por ejemplo, *gold-plating*, un término en inglés que indica la incorporación de características o refinamientos costosos e innecesarios en un producto o servicio).

El Time-boxing es una práctica muy importante en Scrum y debe aplicarse con cuidado. Una time-box arbitrario puede llevar a la desmotivación del equipo y tener como consecuencia la creación de un ambiente tenso, por lo que se debe utilizar de manera apropiada.

2.7.1 Time-boxes de Scrum

- **Sprint—**Un sprint es una iteración con un time-box de una a seis semanas de duración durante el cual el Scrum Master guía, facilita y protege al Equipo Scrum de impedimentos tanto internos como externos durante el proceso de *Crear entregables*. Esto ayuda a evitar una expansión de la visión más allá de su objetivo original, lo que podría afectar la meta del sprint. Durante este tiempo, el equipo trabaja para convertir las necesidades del Backlog Priorizado del Producto en funcionalidades de productos fáciles de enviar. Para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint dentro de un Time-box de cuatro semanas, a menos que existan proyectos con requisitos muy estables, en los que los sprints pueden extenderse hasta seis semanas.

- **Daily Standup—**El Daily Standup es una breve reunión diaria con un time-box de 15 minutos. Los miembros del equipo se reúnen para informar sobre cómo avanza el proyecto, respondiendo a las siguientes tres preguntas:

1. ¿Qué he hecho desde la última reunión?
2. ¿Qué tengo planeado hacer antes de la siguiente reunión?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Esta reunión la lleva a cabo por el equipo como parte del proceso de *Realizar Daily Standup.*

- **Reunión de planificación del sprint**—Esta reunión se lleva a cabo antes del sprint, como parte de los procesos de *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y *Crear el Sprint Backlog*. Se asigna a un time-box de ocho horas durante un sprint de un mes de duración. La reunión de planificación del sprint se divide en dos partes:

    1. Definición del objetivo—Durante la primera parte de la reunión, el Product Owner explica las historias de usuario de más alta prioridad o los requerimientos en el Backlog Priorizado del Producto al Equipo Scrum. Después, el Equipo Scrum en colaboración con el Product Owner se compromete con las historias de usuario, las cuales definen la meta del sprint.

    2. Identificación y estimación de tareas—El Equipo Scrum decide "cómo" completar los elementos seleccionados en el Backlog Priorizado del Producto seleccionados para cumplir con la meta del sprint. Las historias de usuario comprometidas y las Effort Estimated Tasks se incluyen en el Backlog Priorizado del Producto al que se le dará seguimiento.

- **Reunión de revisión del sprint**—La reunión de revisión del sprint tiene un time-box de cuatro horas en un sprint de un mes. Durante la reunión de revisión del sprint que se lleva a cabo en el proceso de *Demostrar y validar el sprint*, el Equipo Scrum presenta los entregables del sprint actual al Product Owner. Este revisa el producto (o incremento del producto) para compararlo con los criterios de aceptación acordados y acepta o rechaza las historias de usuario completadas.

- **Reunión de retrospectiva del sprint**—Esta reunión tiene un time-box de cuatro horas en un sprint de un mes, y se lleva a cabo como parte del proceso *Retrospectiva del sprint*. Durante esta reunión, el Equipo Scrum se reúne para revisar y reflexionar sobre el sprint anterior en relación a los procesos que se siguieron, las herramientas empleadas, la colaboración y los mecanismos de comunicación, así como otros aspectos de interés para el proyecto. El equipo discute lo que salió bien durante el sprint anterior y lo que no salió bien, con el objetivo de aprender y mejorar sprints futuros. Algunas oportunidades de mejora o las mejores prácticas de esta reunión también podrían actualizarse como parte de los documentos del Scrum Guidance Body.

La figura 2-8 ilustra las duraciones de los time-box en las reuniones relacionadas con Scrum.

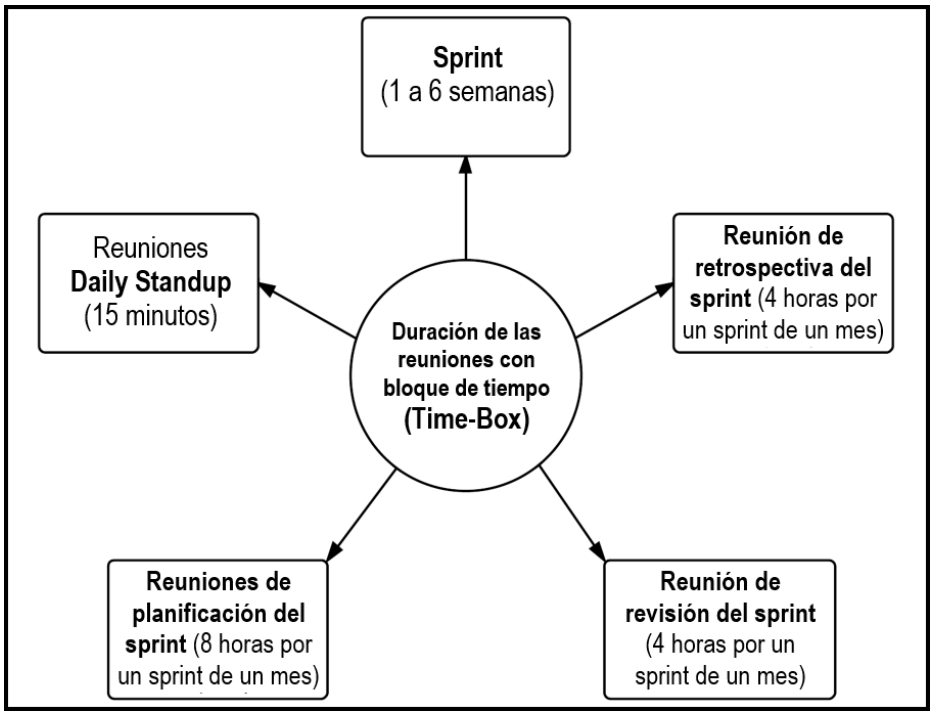


Figura 2-8: Duración de los bloques de tiempo (Time-Box) para las reuniones de Scrum

2.8 Desarrollo iterativo

El Scrum framework está guiado por el objetivo de ofrecer el máximo valor empresarial en un mínimo período de tiempo. Para lograr esto en forma práctica, Scrum cree en el desarrollo iterativo de entregables.

En la mayoría de los proyectos complejos, el cliente puede tal vez no pueda definir requisitos muy concretos o pudiera no estar seguro sobre cómo debería de ser el producto final. El modelo iterativo es más flexible para asegurar que cualquier cambio que solicite el cliente se pueda incluir como parte del proyecto. Las historias de usuario tal vez tengan que ser escritas constantemente durante la duración del proyecto. En las etapas iniciales de redacción, la mayoría de las historias son las funcionalidades de alto nivel. Estas historias de usuario se conocen como épica(s). Las épicas generalmente son muy grandes como para que los equipos las completen en un sólo sprint, y por lo tanto se dividen en pequeñas historias de usuario.

Cada aspecto complejo del proyecto se divide mediante la elaboración progresiva durante el proceso *Refinar el Backlog Priorizado del Producto*. Los procesos de *Crear historias de usuario* y de *Estimar, Aprobar y Comprometer historias de usuario* se utilizan para agregar nuevos requisitos al Backlog Priorizado del Producto. La tarea del Product Owner es asegurar un mayor retorno sobre la inversión (RSI), centrándose en el valor y en la entrega continua con cada sprint. El Product Owner debe entender bien la justificación del negocio y el valor que el proyecto debe entregar al redactar el Backlog Priorizado del Producto, y por lo tanto decidir qué entregables contractuales y valores se han de entregar en cada sprint. Posteriormente, los procesos de *Identificar tareas*, *Estimar tareas* y *Crear el Sprint Backlog* producen el Sprint Backlog, que utiliza el equipo para crear los entregables.

En cada sprint, el proceso de *Crear entregables* se utiliza para desarrollar las salidas del sprint. El Scrum Master tiene que garantizar que se sigan los procesos de Scrum y facilitar al equipo el trabajo de la manera más productiva. El Equipo Scrum se auto-organiza, teniendo como objetivo el crear entregables del sprint a partir de las historias de usuario que están en el Sprint Backlog. En grandes proyectos, varios equipos interfuncionales trabajan en paralelo a través de los sprints, proporcionando soluciones potencialmente entregables al final de cada sprint. Después de completar cada sprint, el Product Owner acepta o rechaza los entregables con base a los criterios de aceptación del proceso de *Demostrar y validar el sprint*.

Como se ilustra en la figura 2-9, los proyectos Scrum se completan de manera iterativa, entregando valor a lo largo del ciclo de vida del proyecto.

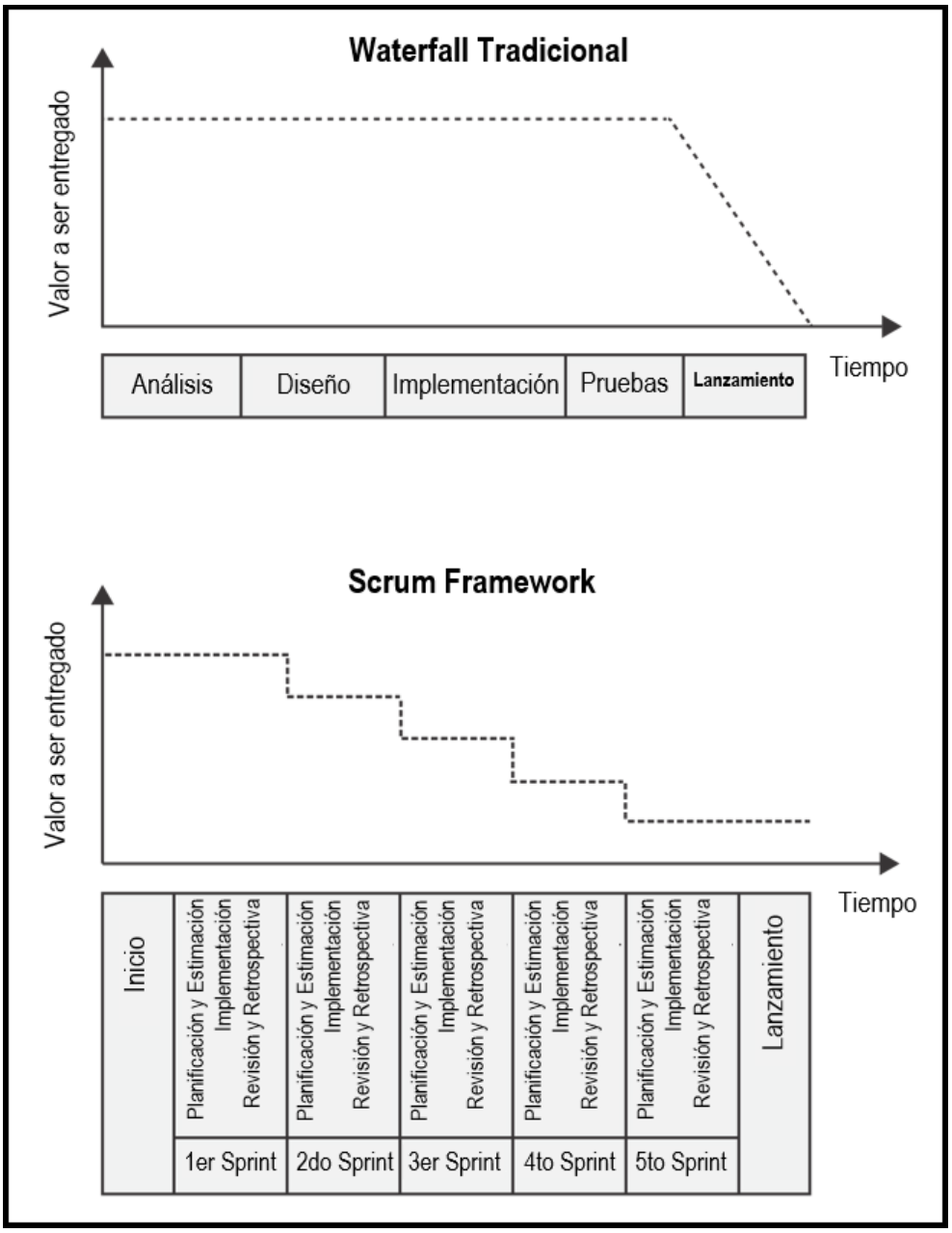


Figura 2-9: Scrum vs Cascada tradicional

El beneficio del desarrollo iterativo permite la corrección a medida que todas las personas involucradas obtengan una mejor comprensión de lo que se debe entregar como parte del proyecto, e incorporar lo aprendido de manera iterativa. Así, el tiempo y el esfuerzo requerido para alcanzar el punto final definitivo, se reduce considerablemente y el equipo produce entregables que se adaptan mejor al entorno empresarial.

2.9 Scrum vs. Gestión tradicional de proyectos

El énfasis en la gestión tradicional de proyectos es llevar a cabo la planificación detallada del proyecto por adelantado con el énfasis en gestionar y solucionar el alcance, costo, horarios y gestionar esos parámetros. En ocasiones, la gestión tradicional de proyectos puede llevar a una situación en la que, aunque el plan haya tenido éxito, el cliente no está satisfecho.

El framework de Scrum se basa en la creencia de que el conocimiento de los trabajadores de hoy en día puede ofrecer mucho más que solo su experiencia técnica, y en que tratar de asignar y planear en un ambiente de constante cambio no es eficiente. Por lo tanto, Scrum alienta a la toma de decisiones iterativa basada en datos. En Scrum, el enfoque principal es la entrega de productos que satisfagan los requisitos del cliente en pequeños incrementos iterativos que sean entregables.

Para entregar la mayor cantidad de valor en el menor tiempo posible, Scrum promueve la priorización y el Time-boxing en vez de la fijación del alcance, del costo y del cronograma de un proyecto. Una característica importante de Scrum es la auto-organización, lo cual permite a las personas que hacen el trabajo estimar y asumir la propiedad de las tareas.

3. ORGANIZACIÓN

3.1 Introducción

En esta sección veremos las diversas facetas de la organización de un proyecto Scrum, así como los roles centrales y los roles no centrales, y cómo formar equipos Scrum de alto rendimiento.

La organización, tal como se define en *Una guía para el conocimiento de Scrum (Guía SBOK™)*, se aplica a lo siguiente:

- Portafolios, programas y/o proyectos de cualquier sector
- Productos, servicios o cualquier otro resultado que se entregue a los stakeholders
- Proyectos de cualquier tamaño y complejidad

En la *Guía SBOK™*, el término "producto"" puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde pequeños proyectos o equipos con tan sólo seis miembros por equipo, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

3.2 Guía de los roles—Esta sección identifica qué sección o subsección es importante para un Product Owner, un Scrum Master y un Equipo Scrum.

3.3 Roles de un proyecto Scrum—Esta sección aborda los roles centrales y no centrales claves con un proyecto Scrum.

3.4 Product Owner—En esta sección se destacan las principales responsabilidades del Product Owner en relación con un proyecto Scrum.

3.5 Scrum Master—Esta sección se centra en las principales responsabilidades del Scrum Master en el contexto de un proyecto Scrum, programa o portafolio.

3.6 Equipo Scrum—Esta sección hace hincapié en las responsabilidades claves del Equipo Scrum en el contexto de un proyecto Scrum, programa o portafolio.

3.7 Scrum en proyectos, programas y portafolios—Esta sección se centra en cómo un framework de Scrum se puede adaptar y utilizar en los diferentes contextos de los programas y los portafolios. También se destacan las responsabilidades específicas de los miembros del Equipo Scrum en relación con la comunicación, la integración y el trabajo con los equipos empresariales y de gestión de programas.

3.8 Responsabilidades—En esta sección se describen las responsabilidades pertinentes al tema de la organización, para todos quienes trabajan en un proyecto, en función de sus roles.

3.9 Scrum vs. Gestión tradicional de proyectos—Esta sección explica las principales diferencias y ventajas del modelo Scrum en relación con el modelo tradicional en Cascada de gestión de proyectos.

3.10 Teorías populares de recursos humanos y su relevancia en Scrum—Esta sección contiene algunas de las teorías de recursos humanos más populares de utilidad para todos los miembros del equipo principal de Scrum.

3.2 Guía de los roles

1. Product Owner—Es importante que el Product Owner lea todo el capítulo.
2. Scrum Master—El Scrum Master también debe familiarizarse con todo este capítulo con enfoque principal en las secciones 3.3, 3.5, 3.6, 3.8 y 3.10.4.
3. Equipo Scrum—El Equipo Scrum debe centrarse principalmente en las secciones 3.3, 3.6 y 3.8.

3.3 Roles de un proyecto Scrum

El entendimiento de los roles y las responsabilidades definidas es muy importante para garantizar la implementación exitosa de los proyectos Scrum.

Los roles de Scrum se dividen en dos categorías:

1. **Roles centrales**—Los roles centrales son aquellos que se requieren obligadamente para crear el producto del proyecto, están comprometidos con el proyecto, y por último son los responsables del éxito de cada sprint del proyecto y del proyecto en su totalidad.

2. **Roles no centrales**—Los roles centrales son aquellos que no son necesariamente obligatorios para el proyecto Scrum, y pueden incluir miembros de los equipos que tengan interés en el proyecto, pero que no tienen ninguna función formal en el equipo del proyecto. Pueden interactuar con el equipo, pero no son responsables del éxito del proyecto. Los roles no centrales también deben tenerse en cuenta en cualquier proyecto de Scrum.

3.3.1 Roles centrales

Hay tres roles centrales en Scrum que en última instancia tienen la responsabilidad de cumplir con los objetivos del proyecto. Los roles centrales son el Product Owner, el Scrum Master y el Equipo Scrum. En conjunto se les conoce como el equipo principal de Scrum. Es importante tener en cuenta que, de estos tres roles, ningún rol tiene autoridad sobre los otros.

1. **Product Owner**

   El Product Owner es la persona responsable de maximizar el valor del negocio para el proyecto. Este rol es responsable de articular los requisitos del cliente y de mantener la justificación del negocio del proyecto. El Product Owner representa la voz del cliente.

   De manera similar al rol del Product Owner en un proyecto, pudiera haber un Program Product Owner o un Portfolio Product Owner, para un programa y un portafolio, respectivamente.

2. **Scrum Master**

   El Scrum Master es un facilitador que asegura que el Equipo Scrum esté dotado de un ambiente propicio para completar con éxito el desarrollo del producto. El Scrum Master guía, facilita e enseña las prácticas de Scrum a todos los participantes en el proyecto, elimina los impedimentos que enfrenta el equipo y se asegura de que se estén siguiendo los procesos de Scrum.

   Debe tenerse en cuenta que el rol de Scrum Master es muy diferente a la función que desempeña el Project Manager en un modelo tradicional de cascada en la gestión de proyectos, en el que el Project Manager trabaja como gerente o líder del mismo. El Scrum Master sólo trabaja como un facilitador y está en el mismo nivel jerárquico que cualquier otra persona en el Equipo Scrum—cualquier persona del Equipo Scrum que aprenda a facilitar proyectos Scrum puede convertirse en el Scrum Master de un proyecto o sprint.

   De manera similar al rol de Scrum Master en un proyecto, también pudiera haber un Program Scrum Master o un Portfolio Scrum Master, para un programa y un portafolio, respectivamente.

3. **Equipo Scrum**

   El Equipo Scrum es un grupo o equipo de personas responsables de entender los requerimientos del negocio especificados por el Product Owner, de estimar las historias de usuarios y de la creación final de los entregables del proyecto.

La figura 3-1 presenta una descripción general de los roles centrales del Equipo Scrum.

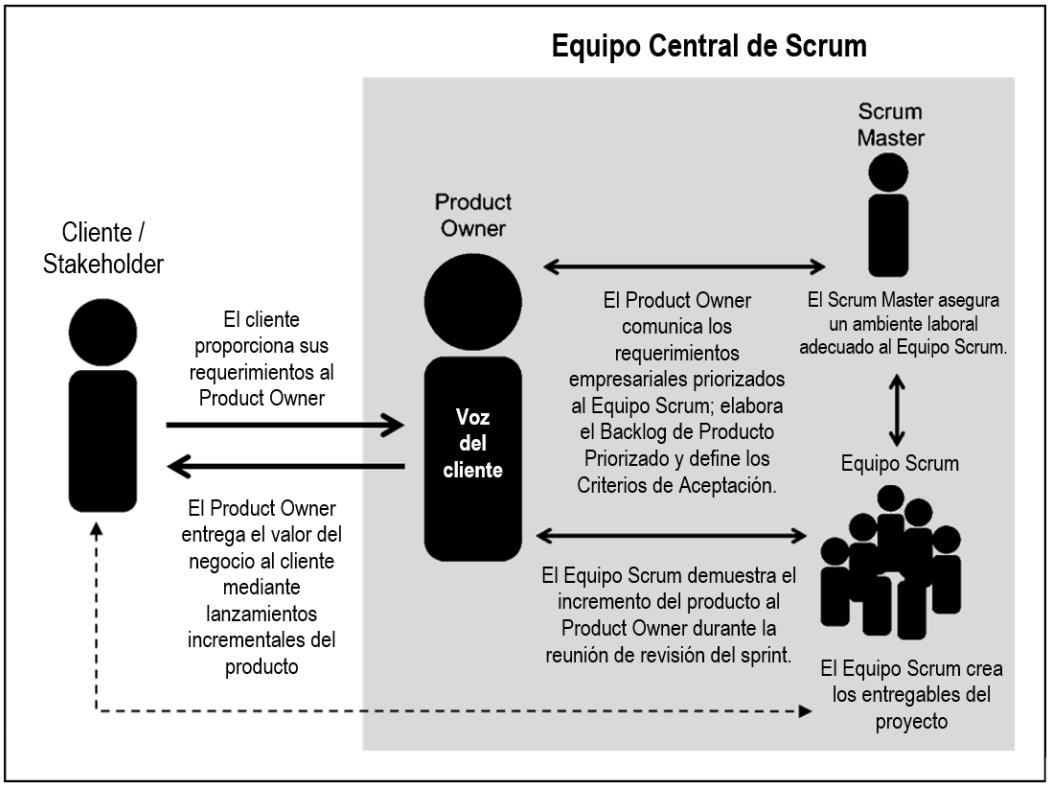


Figura 3-1: Roles de Scrum—Descripción General

3.3.2 Roles no centrales

Los roles no centrales son aquellos roles que no son obligatoriamente necesarios para el proyecto Scrum y pueden no participar en el proceso de Scrum. Sin embargo, es importante tener conocimiento sobre estos roles no centrales, ya que podrían desempeñar un rol importante en algunos proyectos de Scrum.

Los roles no centrales pueden incluir los siguientes:

1. **Stakeholder(s)**

   Stakeholder(s) es un término colectivo que incluye a clientes, usuarios y patrocinadores, que generalmente interactúan con el Product Owner, el Scrum Master y el Equipo Scrum para proporcionarles las entradas y facilitar la creación del producto del proyecto, servicio, o cualquier otro resultado. Los stakeholders influyen en el proyecto a lo largo del desarrollo del mismo. Los stakeholders también pueden desempeñar un rol en los procesos importantes de Scrum tales como *Desarrollar épica(s)*, *Crear Backlog Priorizado del Producto*, *Realizar la planificación del lanzamiento* y *Retrospectiva del sprint*.

- **Cliente**

  El cliente es la persona o la organización que adquiere el producto, servicio o cualquier otro resultado del proyecto. Para cualquier organización, dependiendo del proyecto, puede haber clientes internos (dentro de la misma organización) como clientes externos (fuera de la organización).

- **Usuarios**

  El usuario es el individuo o la organización que utiliza directamente el producto, servicio o cualquier otro resultado del proyecto. Al igual que los clientes, para cualquier organización, puede haber usuarios internos y externos. En algunas industrias los clientes y los usuarios pueden ser los mismos.

- **Patrocinador (*Sponsor*)**

  El patrocinador es la persona o la organización que provee recursos y apoyo para el proyecto. El patrocinador es también el stakeholder a quien todos le deben rendir cuentas al final.

En ocasiones, la misma persona u organización puede desempeñar múltiples roles de stakeholders–por ejemplo, el patrocinador y el cliente pueden ser el mismo.

2. Vendedores

Los vendedores incluyen a individuos u organizaciones externas que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.

3. Scrum Guidance Body

El Scrum Guidance Body (SGB) es un rol opcional, aunque altamente recomendado para formalizar las prácticas organizacionales relacionadas a Scrum. Por lo general, se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en definir los objetivos relacionados a la calidad, regulaciones gubernamentales, seguridad y otros parámetros clave de la organización. Estos objetivos guían el trabajo que lleva a cabo el Product Owner, el Scrum Master y el Equipo Scrum. El Scrum Guidance Body también ayuda a captar las mejores prácticas que deben utilizarse en todos los proyectos de Scrum en la organización.

El Scrum Guidance Body no toma decisiones relacionadas al proyecto. En cambio, actúa como una estructura de consultoría u orientación para todos los niveles de la jerarquía en el proyecto de organización del portafolio, programa y proyecto. Los equipos Scrum tienen la opción de solicitar ayuda al Scrum Guidance Body sobre cualquier recomendación que requieran.

3.4 Product Owner

El Product Owner representa los intereses de la comunidad de stakeholders para el Equipo Scrum. El Product Owner es responsable de asegurar una comunicación clara sobre el producto y los requisitos de funcionalidad del servicio con el Equipo Scrum, definir los criterios de aceptación y asegurar que se cumplan dichos criterios. En otras palabras, el Product Owner es responsable de asegurar que el Equipo Scrum entregue valor. Este rol central siempre debe mantener una visión dual. Debe entender y apoyar las necesidades e intereses de todos los stakeholders, al tiempo que comprende las necesidades y el funcionamiento del Equipo Scrum. Puesto que el Product Owner debe entender las necesidades y prioridades de los stakeholders, incluyendo los clientes y los usuarios, a este rol se le conoce comúnmente como la voz del cliente.

La tabla 3-1 resume las responsabilidades del Product Owner en los diferentes procesos de Scrum.

Proceso	Responsabilidades del Product Owner
8.1 Crear la visión del proyecto	• Define la visión del proyecto • Ayuda a crear el acta constitutiva del proyecto y su presupuesto
8.2 Identificar al Scrum Master y Stakeholder(s)	• Ayuda a finalizar la elección del Scrum Master para el proyecto • Identifica al(los) stakeholder(s)
8.3 Formar el Equipo Scrum	• Ayuda a determinar quiénes serán los miembros del Equipo Scrum • Ayuda a desarrollar un plan de colaboración • Ayuda a desarrollar el plan del equipo con el(los) Scrum Master(s)
8.4 Desarrollar épica(s)	• Crea épica(s) y prototipos (*Personas*)
8.5 Crear Backlog Priorizado del Producto	• Prioriza los elementos en el Backlog Priorizado del Producto • Define los criterios de terminado
8.6 Realizar la planificación del lanzamiento	• Elabora el cronograma de planificación del lanzamiento • Ayuda a determinar la duración del sprint
9.1 Crear historias de usuario	• Ayuda a crear historias de usuario • Define los criterios de aceptación para cada historia de usuario
9.2 Estimar historias de usuario	• Aclara las historias de usuario
9.3 Comprometer historias de usuario	• Facilita el Equipo Scrum y compromete historias de usuario
9.4 Identificar tareas	• Explica las historias de usuario al Equipo Scrum mientras elabora la lista de tareas
9.5 Estimar tareas	• Brinda orientación y aclaraciones al Equipo Scrum en la estimación de los esfuerzos para las tareas
9.6 Crear el Sprint Backlog	• Aclara los requerimientos al Equipo Scrum mientras elabora el Sprint Backlog
10.1 Crear entregables	• Aclara los requerimientos empresariales al Equipo Scrum.
10.3 Refinar Backlog Priorizado del	• Refina (*Grooms*) el Backlog Priorizado del Producto.

Producto	
11.1 Demostrar y validar el sprint	• Acepta/rechaza los entregables • Proporciona la retroalimentación necesaria al Scrum Master y a los equipos Scrum • Actualiza el plan de lanzamiento y el Backlog Priorizado del Producto
12.1 Enviar entregables	• Ayuda a enviar los lanzamientos del producto y se coordina con el cliente
12.2 Retrospectiva del proyecto	• Participa en las reuniones de retrospectiva del sprint

Tabla 3-1: Responsabilidades del Product Owner en los procesos de Scrum

Otras responsabilidades de un Product Owner incluyen:

- Determinar los requisitos generales iniciales del proyecto y dar inicio a las actividades del mismo; esto puede implicar una interacción con el Program Product Owner y el Portfolio Product Owner, a fin de asegurar que el proyecto se alinee con la dirección que proporciona la alta gerencia.
- Representar al(los) usuario(s) del producto o servicio con un conocimiento integral de la comunidad de usuarios.
- Asegurar los recursos financieros del proyecto al inicio y durante su transcurso
- Enfocarse en la creación de valor y el retorno sobre la inversión en general.
- Evaluar la viabilidad y garantizar la entrega del producto o servicio.

3.4.1 Voz del cliente (VOC)

Como representante del cliente, se dice que el Product Owner es la voz del cliente, ya que es quien se asegura de que las necesidades explícitas e implícitas del cliente se reflejen en las historias de usuario en el Backlog Priorizado del Producto y que más adelante se utilicen para crear los entregables del proyecto para el cliente.

3.4.2 Chief Product Owner

El Chief Product Owner también interactúa con el Program Product Owner para asegurar la alineación del proyecto grande con las metas y objetivos del programa.

En el caso de grandes proyectos con varios equipos Scrum y múltiples Product Owners, sigue siendo necesario contar con una sola persona que tome las decisiones diarias del negocio. Esta persona/rol es el Chief Product Owner. Este rol es responsable de coordinar el trabajo de múltiples Product Owners. Con la ayuda de los Product Owners, el Chief Product Owner prepara y mantiene el Backlog Priorizado del Producto en general para el proyecto grande utilizándolo para coordinar el trabajo a través de los Product

Owners de los equipos Scrum. El Chief Product Owner será el responsable de entregable final del proyecto, mientras que los Product Owners de los equipos individuales serán responsables solo de aquellos componentes y características que desarrollan sus respectivos equipos Scrum.

Un proyecto grande, el Chief Product Owner tendrá la tarea de priorizar los requerimientos competentes que presenten los Product Owners con base en su interacción los stakeholders. La complejidad de esta tarea aumenta en gran medida con cada aumento en la cantidad de equipos Scrum y el número de Product Owners. Una parte importante de la complejidad de esta tarea es asegurarse de que los varios componentes estén adecuadamente integrados en los momentos apropiados. Por lo tanto, es importante desarrollar una lista de componentes y recursos necesarios en común para todos los equipos durante el proyecto. Aunque el Chief Product Owner toma las decisiones finales del negocio, él/ella colabora con el Chief Scrum Master, con otros Product Owners y Scrum Masters para desarrollar dicha lista.

Los Chief Product Owners deben consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Product Owner, así como el capítulo 13 que describe lo relacionado a escalar Scrum para grandes proyectos.

3.4.3 Program Product Owner

Define los objetivos estratégicos y las prioridades del programa. El Program Product Owner es la persona responsable de maximizar el valor de negocio en un programa. Es responsable de articular los requerimientos del cliente y de mantener la justificación del negocio para el programa y puede brindar valiosos aportes sobre cómo deben concebirse los proyectos en un programa. El Program Product Owner también gestiona el Backlog del Producto del Programa.

El Program Product Owner interactúa con el Portfolio Product Owner para asegurar la alineación del programa con las metas y objetivos del portafolio. Participa también el nombramiento de Product Owners en proyectos individuales y se asegura de que la visión, los objetivos, los resultados y los lanzamientos de proyectos individuales en el programa estén alineados con los del programa.

Este rol es similar al del Product Owner, a excepción de que cumple con las necesidades el programa o unidad empresarial en vez de un solo Equipo Scrum.

Los Program Product Owners deben consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Product Owner, así como el capítulo 14 que describe lo relacionado a escalar Scrum para la empresa.

3.4.4 Portfolio Product Owner

El Portfolio Product Owner toma las decisiones a nivel de portafolio. Tendrá la mejor perspectiva para decidir cómo organizar la empresa para cumplir con la visión. El rol del Portfolio Product Owner es similar al

del Program Product Owner en un programa. Es responsable de la creación y refinación del Backlog del Producto del Portafolio.

Este rol es similar al del Product Owner, a excepción de que cumple con las necesidades del portafolio o unidad empresarial en vez de un solo Equipo Scrum.

Los Portfolio Product Owners deben consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Product Owner, así como el capítulo 14 que describe lo relacionado a escalar Scrum para la empresa.

3.5 Scrum Master

El Scrum Master es el "líder servicial" del Equipo Scrum y es quien modera y facilita las interacciones del equipo como coach y motivador del mismo. Este rol es responsable de asegurarse que el equipo tenga un ambiente de trabajo productivo protegiéndolo de influencias externas, eliminando todos los obstáculos y haciendo que se cumplan los principios, aspectos y procesos de Scrum.

La tabla 3-2 resume las responsabilidades del Scrum Master en los diferentes procesos de Scrum.

Procesos	Responsabilidades del Scrum Master
8.2 Identificar al Scrum Master y Stakeholder(s)	• Ayuda a identificar al(los) stakeholder(s) para el proyecto
8.3 Formar del Equipo Scrum	• Facilita la selección del Equipo Scrum • Facilita la creación del plan de colaboración y el plan de desarrollo del equipo • Garantizar que los recursos de respaldo estén disponibles para el funcionamiento del proyecto sin problemas
8.4 Desarrollar de épica(s)	• Facilitar la creación de épica(s) y prototipos (*Personas*)
8.5 Crear Backlog Priorizado del Producto	• Ayuda al Product Owner en la creación del Backlog Priorizado del Producto y en la definición de los criterios de terminado
8.6 Realizar la planificación del lanzamiento	• Coordina la creación del cronograma de planificación del lanzamiento • Determina de la duración del sprint
9.1 Crear historias de usuario	• Ayuda al Equipo Scrum en la creación de historias de usuario y sus criterios de aceptación
9.2 Estimar historias de usuario	• Organiza las reuniones del Equipo Scrum para estimar historias de usuario
9.3 Comprometer historias de usuario	• Organiza reuniones del Equipo Scrum para comprometer historias de usuario
9.4 Identificar tareas	• Ayuda al Equipo Scrum a crear la lista de tareas para el siguiente sprint

9.5 Estimar tareas	• Ayuda al Equipo Scrum a estimar el esfuerzo necesario para completar las tareas acordadas para el sprint
9.6 Crear el Sprint Backlog	• Ayuda al Equipo Scrum a desarrollar el Sprint Backlog y el Sprint Burndown Chart
10.1 Crear entregables	• Ayuda al Equipo Scrum a crear los entregables acordados para el sprint • Ayuda a actualizar el Scrumboard y el Impediment Log
10.2 Realizar Daily Standup	• Se asegura de que el Scrumboard y el Impediment Log permanezcan actualizados
10.3 Refinar el Backlog Priorizado del Producto	• Organiza las reuniones de revisión del Backlog Priorizado del Producto
11.1 Demostrar y validar sprints	• Facilita la presentación de los entregables completados por el Equipo Scrum para la aprobación del Product Owner
11.2 Retrospectiva del sprint	• Garantiza que exista un ambiente ideal para el Equipo Scrum del proyecto en los sucesivos sprints
12.2 Retrospectiva del proyecto	• Representa al equipo principal de Scrum para proporcionar lecciones del proyecto actual en caso de ser necesario

Tabla 3-2: Responsabilidades del Scrum Master en los procesos de Scrum

3.5.1 Chief Scrum Master

Los grandes proyectos requieren que varios equipos Scrum trabajen en paralelo. La información obtenida de un equipo pudiera ser comunicada apropiadamente a otros equipos. El Chief Scrum Master es responsable de dicha actividad.

El rol de un Chief Scrum Master es necesario para garantizar una colaboración apropiada entre los equipos Scrum. La coordinación entre los varios equipos Scrum que trabajan en un proyecto generalmente se da mediante la reunión de Scrum de Scrums (véase la sección 13.2.2.1). No existe una jerarquía entre los Scrum Masters: todos son compañeros. El Chief Scrum Master solo trabaja en un nivel de equipos múltiples, mientras que los Scrum Masters trabajan al nivel de un solo equipo.

La figura 3-2 proporciona las preguntas que se hacen durante una reunión de Scrum de Scrums.

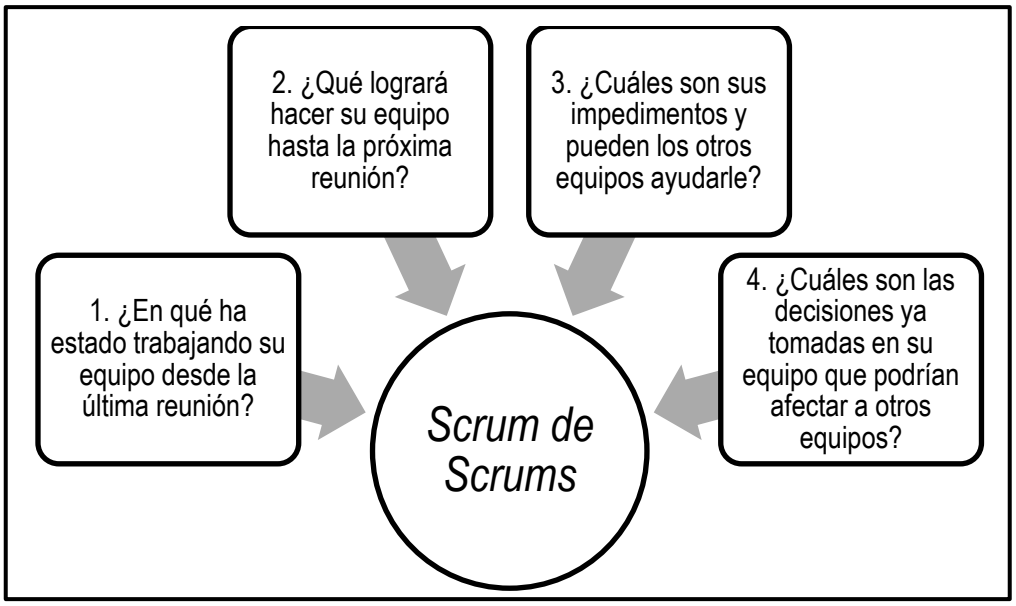


Figura 3-2: Las preguntas formuladas durante una reunión de Scrum de Scrums

Por lo general, cualquier problema dentro de los equipos lo atienden las partes interesadas en una sesión celebrada inmediatamente después de la reunión de Scrum de Scrum. El Chief Scrum Master organiza esta sesión.

El Chief Scrum Master puede ser seleccionado de entre los Scrum Masters de un proyecto grande o puede ser alguien más. Para proyectos muy grandes, se recomienda contar con un Chief Scrum Master que tampoco sea un Scrum Master, ya que el esfuerzo requerido para el rol de Chief Scrum Master evitará que este pueda dedicar suficiente tiempo al trabajo con su Equipo Scrum. En cualquiera de los casos, el Chief Scrum Master debe contar con la suficiente experiencia en Scrum para poder fomentar la colaboración y ayudar a entrenar a los demás en la implementación de Scrum para lograr una entrega sin contratiempos de los productos del proyecto.

Además de eliminar impedimentos y asegurar un entorno conductivo en el proyecto para los equipos Scrum, el Chief Scrum Master también colaboración con el Chief Product Owner, con otros Scrum Masters y Product Owners en actividades tales como el desarrollo de lista de componentes y recursos necesarios y comunes entre todos los equipos durante todo el proyecto. Él/ella facilita todo lo necesario que está por encima del alcance de un solo Equipo Scrum.

El Chief Scrum Master también interactúa con el Program Scrum Master a fin de garantizar la alineación de un proyecto grande con las metas y objetivos del programa.

Los Chief Scrum Masters consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Scrum Master, así como el capítulo 13 que describe lo relacionado a escalar Scrum para grandes proyectos.

3.5.2 Program Scrum Master

El Program Scrum Master es un facilitador que se asegura de que todos los equipos del proyecto cuentan con un ambiente conductivo para concluir con éxito sus proyectos. El Program Scrum Master guía, organiza y enseña las prácticas de Scrum a todos los involucrados en el programa; brinda orientación a los Scrum Masters de proyectos individuales; elimina impedimentos que enfrenten los distintos equipos del proyecto; se coordina con el Scrum Guidance Body para definir los objetivos relacionados a la calidad, regulaciones gubernamentales, seguridad y demás parámetros organizacionales clave; y se asegura de que los procesos de Scrum se sigan eficazmente durante el programa.

El Program Scrum Master interactúa con el Portfolio Scrum Master para asegurar la alineación del programa con las metas y objetivos del portafolio. También participa en el nombramiento de Scrum Masters en proyectos individuales y se asegura de que la visión, los objetivos, los resultados y los lanzamientos de proyectos individuales en el programa estén alineados con los del programa.

Este rol es similar al del Scrum Master a excepción de que cumple con las necesidades del programa o unidad empresarial en vez de un solo Equipo Scrum.

Los Program Scrum Masters deben consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Scrum Master, así como el capítulo 14 que describe lo relacionado a escalar Scrum para la empresa.

3.5.3 Portfolio Scrum Master

Este rol es similar al del Scrum Master a excepción de que cumple con las necesidades del portafolio o unidad empresarial en vez de un solo Equipo Scrum.

Los Portfolio Scrum Masters deben consular las secciones de la *Guía SBOK*™ definidas en la Guía de Roles para el Scrum Master, así como el capítulo 14 que describe lo relacionado a escalar Scrum para la empresa.

3.6 Equipo Scrum

Al Equipo Scrum en ocasiones se le conoce como equipo de desarrollo, ya que este es responsable del desarrollo del producto, servicio o de cualquier otro resultado. Consiste en un grupo de personas que trabajan en las historias de usuario en el Sprint Backlog para crear los entregables del proyecto.

La tabla 3-3 resume las responsabilidades del Equipo Scrum en los diversos procesos de Scrum.

Procesos	Responsabilidades del Equipo Scrum
8.3 Formar el Equipo Scrum	• Proporciona aportes para la creación del plan de colaboración y del plan de desarrollo del equipo
8.4 Desarrollar épica(s)	• Asegura una comprensión clara de la épica(s) y prototipos (*Personas*)
8.5 Crear el Backlog Priorizado del Producto	• Entiende las historias de usuario en el Backlog Priorizado del Producto
8.6 Realizar la planificación del lanzamiento	• Está de acuerdo con los demás miembros del equipo principal de Scrum sobre la duración del sprint • Busca clarificación sobre los nuevos productos o cambios, si los hay, en los productos existentes en el Backlog Priorizado del Producto.
9.1 Crear historias de usuario	• Proporciona aportes al Product Owner en la creación de historias de usuario
9.2 Estimar historias de usuario	• Estima las historias de los usuarios aprobadas por el Product Owner
9.3 Comprometer historias de usuario	• Compromete las historias de usuario a realizarse en un sprint
9.4 Identificar tareas	• Desarrolla una lista de tareas con base en las historias de usuario y dependencias acordadas
9.5 Estimar tareas	• Estima el esfuerzo de las tareas identificadas y, de ser necesario, actualiza la lista de tareas.
9.6 Crear el Sprint Backlog	• Desarrolla el Sprint Backlog y el Sprint Burndown Chart
10.1 Crear entregables	• Elabora los entregables • Identifica riesgos y ejecuta acciones de mitigación de riesgos, si los hay. • Actualiza el Impediment Log y las dependencias
10.2 Realizar el Daily Standup	• Actualiza el Burndown Chart, el Scrumboard y el Impediment Log • Discute los problemas que enfrenta cada miembro y busca soluciones para motivar al equipo • Identifica riesgos, si lo hay. • Presenta solicitudes de cambio, si se requieren.
10.3 Refinar el Backlog Priorizado del Producto	• Participa en las reuniones de revisión del Backlog Priorizado del Producto
11.1 Demostrar y validar sprints	• Muestra los entregables completados al Product Owner para su aprobación
11.2 Retrospectiva del sprint	• Identifica oportunidades de mejora, si las hay, del Sprint actual y decide si está de acuerdo sobre las posibles mejoras viables para el próximo sprint
12.2 Retrospectiva del proyecto	• Participa en la reunión de retrospectiva del proyecto

Tabla 3-3: Responsabilidades del Equipo Scrum en los procesos de Scrum

3.6.1 Selección de personal

La figura 3-3 enumera las características deseables para los roles centrales de Scrum.

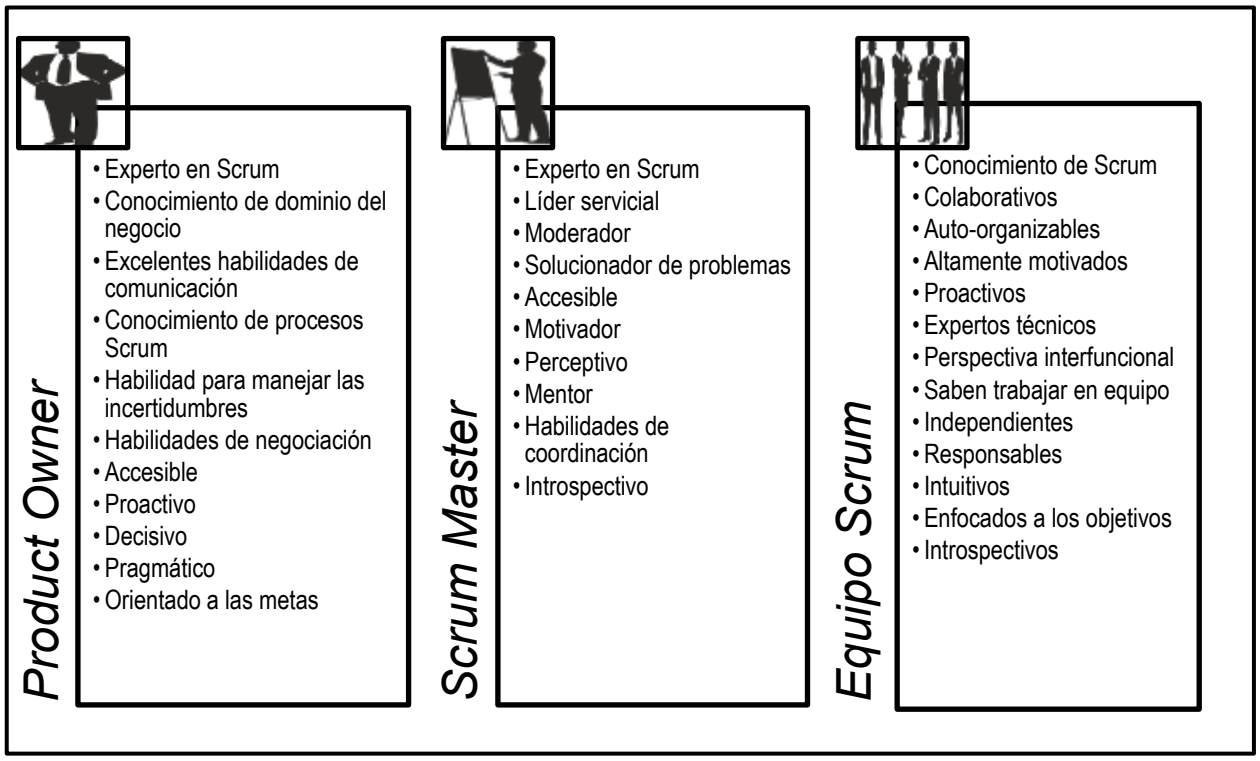


Figura 3-3: Características deseadas de los roles centrales de Scrum

3.6.2 Tamaño del Equipo Scrum

Es importante que el Equipo Scrum cuente con todas las habilidades esenciales necesarias para llevar a cabo el trabajo del proyecto. También es necesario contar con un alto nivel de colaboración para maximizar la productividad, de modo que se requiera una mínima coordinación para llevar a cabo el trabajo.

El tamaño óptimo de un Equipo Scrum es de seis a diez miembros, lo suficientemente grande para asegurar habilidades adecuadas, pero lo suficientemente pequeño como para facilitar la colaboración. Un beneficio clave de un equipo de seis a diez miembros es que la comunicación y la gestión suelen ser simples y requieren un esfuerzo mínimo. Sin embargo, también puede haber inconvenientes. Una desventaja importante es que los equipos más pequeños se ven considerablemente más afectados por la pérdida de un miembro del equipo, en comparación a los equipos más grandes, aunque sea por un corto tiempo. Este problema se puede solucionar si los miembros del equipo tienen conocimientos especializados y habilidades fuera de su rol específico. Sin embargo, esto puede ser difícil y depende del tipo de proyecto, la industria y el tamaño de la organización. También se recomienda tener suplentes para reemplazar a cualquier persona que tenga que abandonar el Equipo Scrum.

3.7 Scrum en proyectos, programas y portafolios

3.7.1 Definición de proyecto, programa y portafolios

- **Proyecto**—Un proyecto es un emprendimiento colaborativo para crear nuevos productos o servicios, o para obtener resultados como los que se definen en la declaración de la visión del proyecto (*Project Vision Statement*). Los proyectos por lo general se ven afectados por limitaciones de tiempo, costo, alcance, calidad, personal y la capacidad de la organización. El objetivo del equipo de proyecto es crear entregables tal como se define en el Backlog Priorizado del Producto.

- **Programa**—Un programa es un grupo de proyectos relacionados con la finalidad de entregar resultados de negocio definidos en la declaración de la visión del programa (*Program Vision Statement*). El Backlog Priorizado del Programa incorpora al Backlog Priorizado del Producto de todos los proyectos del programa.

- **Portafolio**—Un portafolio es un grupo de programas relacionados con la finalidad de entregar resultados de negocio como se define en la declaración de la visión del portafolio (*Portfolio Vision Statement*). El Backlog Priorizado del Portafolio incorpora el Backlog Priorizado del Producto de todos los programas en el portafolio

Los siguientes son ejemplos de proyectos, programas y portafolios de diferentes industrias y sectores:

Ejemplo 1: Empresa constructora

- Proyecto—Construcción de una casa
- Programa—Construcción de un complejo de viviendas
- Portafolio—Todos los proyectos de vivienda de la empresa

Ejemplo 2: Organización aeroespacial

- Proyecto—Construcción del vehículo de lanzamiento
- Programa—Lanzamiento exitoso de un satélite
- Portafolio —Todos los programas de satélites activos

Ejemplo 3: Empresa de tecnología de la información (IT)

- Proyecto—Desarrollo del módulo del carrito de compras
- Programa—Desarrollo de un sitio web de comercio electrónico (*e-commerce*) completamente funcional
- Portafolio—Todos los sitios web desarrollados por la empresa hasta ahora

La figura 3-5 ilustra la forma en la que Scrum se puede utilizar para portafolios, programas o proyectos en la organización.

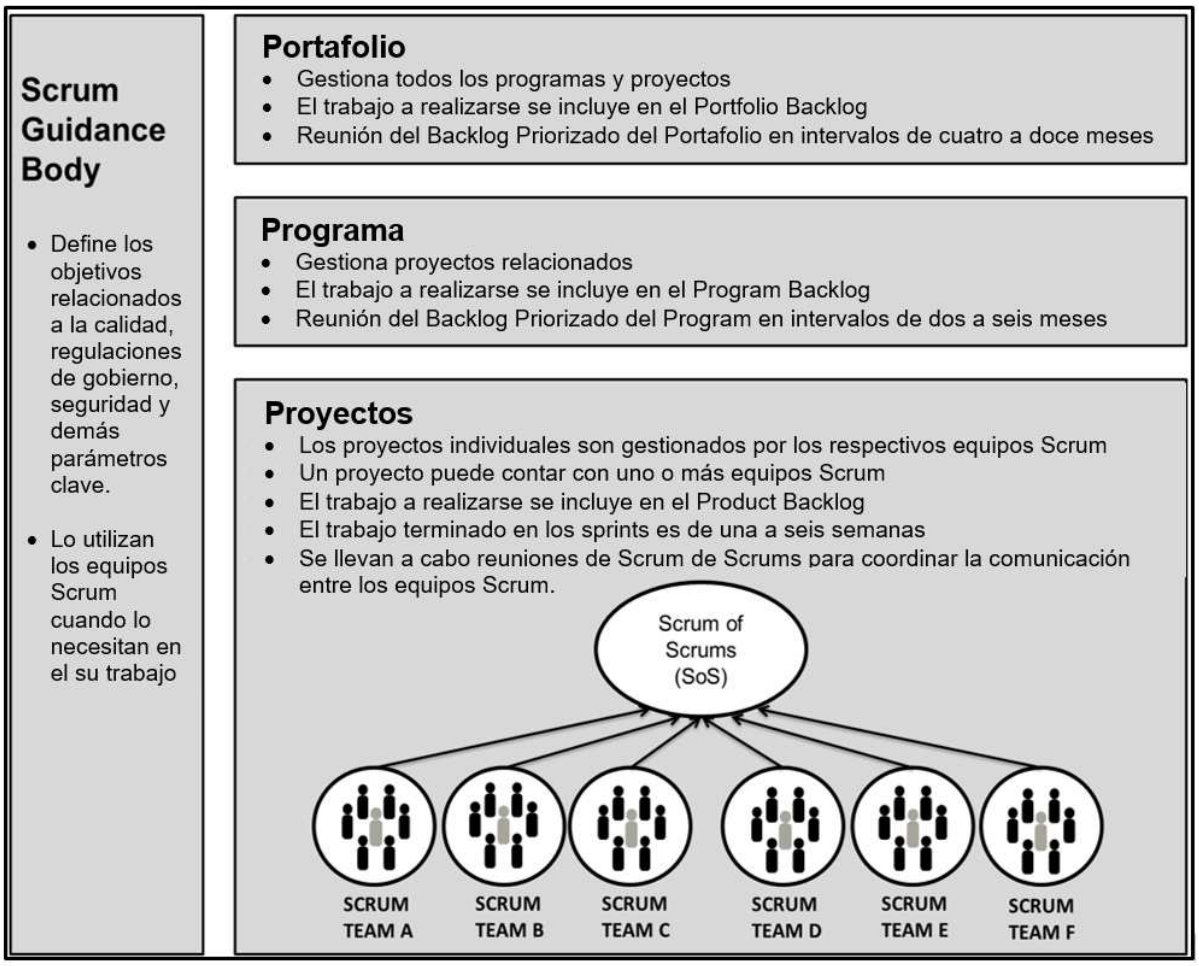


Figura 3-5: Scrum para proyectos, programas y portafolios en la organización

3.7.1.1 Trabajar con equipos de portafolios y programas

Al aplicar Scrum para gestionar proyectos en el contexto de un programa o un portafolio, se recomienda enfáticamente el apego a los principios generales de Scrum que se presentan en esta publicación. Sin embargo, se entiende que, a fin de adaptar el programa en su totalidad o actividades relacionadas con el portafolio y las interdependencias, pueden ser necesarios pequeños ajustes en el conjunto de herramientas, así como a la estructura organizacional. Si existe Scrum Guidance Body, éste puede ser responsable de examinar la organización en diferentes niveles a fin de entender y definir la aplicación adecuada de Scrum, y actuar como facilitador de consulta para todos los que trabajan en un proyecto, programa o portafolio.

Los portafolios y los programas cuentan con equipos separados y con diferentes series de objetivos. Los equipos de gestión de programas tienen por objetivo ofrecer capacidades y llevar a cabo ciertas metas que contribuyan a objetivos específicos del programa. Por el contrario, el equipo del portafolio tiene que

equilibrar los objetivos de los distintos programas para alcanzar los objetivos estratégicos de la organización en su totalidad. El capítulo 14 aborda a detalle lo relacionado a escalar Scrum para la empresa.

3.7.1.2 Gestionar la comunicación con equipos de portafolios y programas

Los problemas y los asuntos que se enfrentan al utilizar Scrum dentro de un programa o portafolio implican principalmente la coordinación entre los numerosos equipos. Esto puede conducir al fracaso si no se maneja con cuidado. Las herramientas que se utilizan para la comunicación deben ampliarse para que coincidan con los requisitos de los varios equipos que participan en un programa o portafolio. Cada Equipo Scrum debe atender no sólo la comunicación interna, sino también la comunicación externa con otros equipos y los stakeholders relevantes del programa o portafolio.

3.7.2 Mantener la participación de los stakeholders

Scrum requiere del apoyo total de los stakeholders de los proyectos. La responsabilidad de mantener la participación de los stakeholders depende del Product Owner. Las siguientes son acciones que se recomiendan para conservar de la participación y el apoyo de los stakeholders:

- Asegurar la colaboración efectiva y la participación de los stakeholders en el proyecto
- Evaluar continuamente el impacto en el negocio
- Mantener una comunicación constante con los stakeholders
- Administrar las expectativas de los stakeholders

Un stakeholder clave es el patrocinador (*sponsor*): la persona que provee los fondos y demás recursos para un proyecto. Los patrocinadores buscan entender los resultados financieros relacionados con un producto o servicio, y están, por lo general, más interesados en los resultados finales, que en las tareas individuales.

Es importante que los patrocinadores que financian el proyecto tengan en claro los siguientes asuntos:

- Beneficios de la implementación de Scrum
- Plazos del objetivo y costos estimados de los proyectos Scrum
- Los riesgos en general involucrados en proyectos Scrum y las medidas para mitigarlos
- Fechas de lanzamiento esperadas y entregables finales

3.8 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	• Asume la responsabilidad colectiva y garantiza que los entregables del proyecto sean elaborados según los requerimientos • Asegura al Product Owner y el Scrum Master que el trabajo asignado se está llevando a cabo según el plan
Product Owner/ Chief Product Owner	• Elabora los requerimientos generales iniciales del proyecto y arranca el proyecto • Nombra a las personas apropiadas en los roles de Scrum Master y Equipo Scrum • Ofrece los recursos económicos iníciales y constantes para el proyecto • Determina la visión del producto • Evalúa la viabilidad y asegura la entrega del producto o servicio • Asegura la transparencia y claridad de los elementos del Backlog Priorizado del Producto • Decide el contenido comercial mínimo que se puede presentar • Proporciona los criterios de aceptación de las historias de usuario a ser desarrolladas en un sprint • Inspecciona los entregables • Decide la duración del sprint
Scrum Master/ Chief Scrum Master	• Se asegura de que todos los miembros del equipo, incluyendo el Product Owner estén cumpliendo correctamente los procesos de Scrum • Se asegura de que el desarrollo del producto o servicio avance sin contratiempos y que los integrantes del Equipo Scrum cuenten con todas las herramientas necesarias para realizar el trabajo • Supervisa la reunión de planificación del lanzamiento y programa otras reuniones
Program Product Owner	• Define los objetivos estratégicos y las prioridades de los programas
Program Scrum Master	• Resuelve problemas y coordina reuniones para los programas
Portfolio Product Owner	• Define los objetivos estratégicos y las prioridades de los portafolios
Portfolio Scrum Master	• Resuelve problemas y coordina reuniones para los portafolios
Stakeholder(s)	• Término colectivo que incluye a clientes, usuarios y patrocinadores (*sponsors*). • Interactúa frecuentemente con el Product Owner, el Scrum Master y el Equipo Scrum para brindarles aportes y facilitar la creación de los entregables del proyecto
Scrum Guidance Body	• Establece las directrices generales y los parámetros para desarrollar la descripción de roles para los integrantes del Equipo Scrum • Funge como consultor de proyectos en toda la organización en distintos niveles • Entiende y define los niveles apropiados de agrupación, roles y reuniones para el proyecto Scrum

Tabla 3-4: Resumen de las responsabilidades pertinentes a la organización

3.9 Scrum vs. Gestión tradicional de proyectos

La estructura organizativa y la definición de roles y responsabilidades correspondientes son algunas de las áreas en las que Scrum difiere de manera importante de los métodos tradicionales de gestión de proyectos.

En los métodos tradicionales de gestión de proyectos, la estructura de la organización es jerárquica y la autoridad para todos los aspectos del proyecto se delega desde el nivel superior al inferior (por ejemplo, el patrocinador, o *sponsor* del proyecto delega autoridad al Project Manager, y este, a su vez, la delega a los miembros del equipo). Los métodos tradicionales de gestión de proyectos hacen énfasis en la responsabilidad individual respecto a las responsabilidades del proyecto, en vez de la responsabilidad propiedad colectiva. Cualquier desviación de la autoridad delegada se considera como una señal de problemas y puede ser llevada a un nivel más alto en la jerarquía de la organización. Por lo general, es el Project Manager, quien es responsable de la finalización con éxito del proyecto, así como de la toma de decisiones sobre diversos aspectos del proyecto, incluyendo el inicio, planificación, estimación, ejecución, seguimiento y control y cierre.

El énfasis en Scrum está en la auto-organización y la auto-motivación, donde el equipo asume una mayor responsabilidad en lograr el éxito de un proyecto. Esto también garantiza la existencia de un sentido de compromiso del equipo y responsabilidad compartida. Esto, a su vez, da lugar a la motivación del equipo que lleva a una optimización de su eficiencia. El Product Owner, el Scrum Master y el Equipo Scrum trabajan de cerca con el(los) stakeholder(s) relevante(s) para ajustar los requisitos a medida que avanzan a través de los procesos de *Desarrollar de épica(s)*, *Crear el Backlog Priorizado del Producto* y *Crear historias de usuario*. Esto asegura que no haya margen para la planificación aislada en Scrum. El conocimiento y las experiencias del equipo sobre el desarrollo de productos se utilizan para evaluar las entradas necesarias para planificar, calcular y ejecutar el trabajo del proyecto. La colaboración entre los miembros del equipo principal de Scrum asegura que el proyecto se lleve a cabo en un ambiente innovador y creativo propicio para el crecimiento y la armonía del equipo.

3.10 Teorías populares de recursos humanos y su relevancia para Scrum

3.10.1 Modelo de dinámica de grupo de Tuckman (*Tuckman's Model of Group Dynamics*)

El enfoque y método de Scrum pueden parecer al principio muy diferentes y difíciles para un nuevo Equipo Scrum. Al igual que cualquier otro equipo nuevo, un nuevo Equipo Scrum se desarrolla, por lo general, a través de un proceso de cuatro etapas durante su primer proyecto de Scrum. Este proceso se conoce como modelo de dinámica de grupo de Tuckman (Tuckman, 1965), conocido en inglés como: *Tuckman's Model of Group Dynamics*. La idea principal es que las cuatro etapas: Formación, Enfrentamiento, Normalización y Desempeño, son imprescindibles para que un equipo se desarrolle y mitigue los problemas y desafíos, encuentre soluciones, planifique el trabajo y entregue resultados.

Las cuatro etapas del modelo son las siguientes:

1. **Formación (*Forming*)**—Esto a menudo se experimenta como un escenario ameno, ya que todo es nuevo y el equipo aún no ha encontrado ninguna dificultad con el proyecto.
2. **Enfrentamiento (*Storming*)**—Durante esta etapa, el equipo trata de cumplir con el trabajo; sin embargo, puede encontrar conflictos de poder y, con frecuencia, existe un caos o confusión entre los miembros del equipo.
3. **Normalización (*Norming*)**—Esto es cuando el equipo empieza a madurar, resolver sus diferencias internas, y encontrar soluciones para así trabajar juntos. Se considera un período de ajuste.
4. **Desempeño (*Performing*)**—Durante esta etapa, el equipo está unido y opera en su nivel más alto en términos de rendimiento. Los miembros se han convertido en un equipo eficiente de profesionales que son consistentemente productivos.

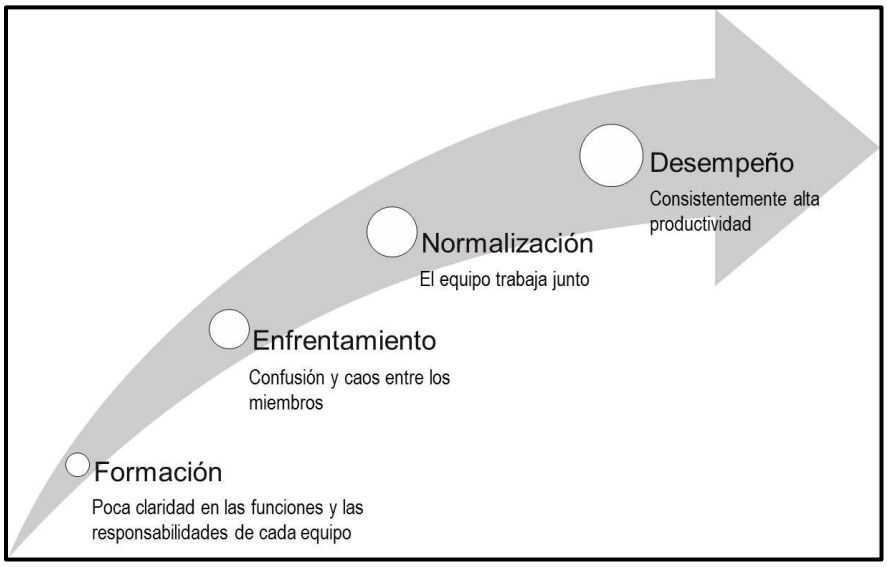


Figura 3-4: Etapas de Tuckman de desarrollo de grupos

3.10.2 Gestión de conflictos (*Conflict Management*)

Las organizaciones que aplican el framework de Scrum fomentan un ambiente abierto y de diálogo entre los empleados. Los conflictos entre los integrantes del Equipo Scrum generalmente se resuelven en forma independiente, con poca o ninguna participación de la gerencia o de otros fuera del equipo.

El conflicto puede ser sano cuando promueve las discusiones en equipo y fomenta al debate, ya que por lo general esto se traduce en beneficios para el proyecto y para los respectivos miembros de los equipos. Por lo tanto, es importante que se fomente la resolución de los conflictos, promoviendo un entorno abierto donde los miembros del equipo se sientan alentados a expresar sus inquietudes y opiniones con los demás sobre el proyecto, y por último que estén de acuerdo en lo que habrá de entregarse y cómo se realizará el trabajo en cada sprint.

Los miembros del equipo utilizan técnicas de gestión de conflictos para atender los conflictos que surjan durante un proyecto Scrum. Las fuentes de conflictos evolucionan principalmente debido a los horarios, prioridades, recursos, informes de jerarquía, cuestiones técnicas, procedimientos, personalidad y costos.

3.10.3 Técnicas de gestión de conflictos

Por lo general, hay cuatro enfoques para la gestión de conflictos en una organización que aplica procesos de Scrum:

1. Ganar-Ganar
2. Perder-Ganar
3. Perder-Perder
4. Ganar-Perder

3.10.3.1 Ganar-Ganar

Por lo general es mejor que los miembros del equipo enfrenten los problemas directamente con una actitud de cooperación y abiertos al diálogo a fin de abordar cualquier desacuerdo y lograr un consenso. Este enfoque se denomina ganar-ganar (*Win-Win*). Las organizaciones que implementan Scrum deben promover un ambiente donde los empleados se sientan cómodos para dialogar y enfrentarse a los problemas y atenderlos para lograr resultados de ganar-ganar.

3.10.3.2 Perder-Ganar

Algunos miembros del equipo a veces pueden sentir que sus aportaciones no son reconocidas ni valoradas por los demás, o que no se les trata igual. Esto puede conducir a que no contribuyan de manera efectiva al proyecto y a que estén de acuerdo con todo lo que se les pida, incluso si están en desacuerdo. Este enfoque se llama perder-ganar (*Lose-Win*). Esta situación puede darse si hay miembros en el equipo (inclusive los administradores) que utilizan un estilo autoritario o directivo para dar órdenes y/o no tratan a todos los miembros del equipo con igualdad. Este enfoque no es una técnica preferencial para la gestión de conflictos en los proyectos Scrum, ya que el aporte colectivo de todos los miembros del equipo es obligatorio para completar con éxito cada uno de los sprints. El Scrum Master debe fomentar la participación de los miembros del equipo que busquen alejarse de las situaciones de conflicto. Por ejemplo, es importante que todos los miembros del equipo hablen y contribuyan en cada Daily Standup para dar a conocer y manejar eficazmente cualquier problema o impedimento.

3.10.3.3 Perder-Perder

En situaciones de conflicto, los miembros del equipo pueden intentar negociar o buscar soluciones que aporten sólo un grado parcial o medida temporal de satisfacción a las partes en una disputa. Esta situación podría presentarse en equipos Scrum si los integrantes intentan negociar soluciones sub-óptimas para resolver un problema. Este enfoque suele implicar un poco de “dar para recibir” a fin de satisfacer a cada miembro del equipo, en vez de intentar resolver el verdadero problema. En general, esto se traduce en un resultado de perder-perder (*Lose-Lose*) para los individuos involucrados y, en consecuencia, para el proyecto. El Equipo Scrum debe tener cuidado de que sus integrantes no lleguen a una mentalidad de perder-perder. Los Daily Standups y otras reuniones de Scrum se llevan a cabo para asegurar que los problemas reales se resuelvan mediante de discusiones mutuas.

3.10.3.4 Ganar-Perder

En ocasiones, un Scrum Master u otro miembro influyente del equipo, pueden creer ser un líder *de facto* o manager y tratar de imponer su punto de vista sin tomar en cuenta los puntos de vista de los demás. Esta técnica de gestión de conflictos a menudo se caracteriza por la competitividad y por lo general resulta en ganar-perder (*Win-Lose*). Este enfoque no es recomendable cuando se trabaja en proyectos Scrum, ya que los equipos Scrum son por naturaleza auto-organizados y empoderados, donde nadie tiene una verdadera autoridad sobre otro integrante del equipo. Aunque el Equipo Scrum puede incluir personas con diferentes niveles de experiencia y conocimientos, cada miembro se trata por igual y nadie tiene la autoridad de ser el principal tomador de decisiones.

3.10.4 Estilos de liderazgo

Los estilos de liderazgo varían dependiendo de la organización, la situación, e incluso en los individuos y los objetivos específicos del proyecto Scrum. Algunos estilos de liderazgo comunes son los siguientes:

- **Líder servicial**—El liderazgo servicial (*Servant Leadership*) implica escuchar cuidadosamente, tener empatía, comprometerse al servicio, tener visión y compartir el poder y la autoridad con los miembros del equipo. Este estilo de liderazgo logra resultados centrándose en las necesidades del equipo. Asimismo, es la personificación del rol del Scrum Master.
- **Líder delegador—**Los líderes delegadores participan en la mayoría de la toma de decisiones; sin embargo, delegan parte de las responsabilidades de planificación a los miembros del equipo, particularmente si son competentes para manejar las tareas asignadas. Este estilo de liderazgo es apropiado en situaciones en las que el líder está en sintonía con los detalles específicos del proyecto y cuando el tiempo es limitado.
- **Autocrático—**Los líderes autocráticos toman decisiones por su cuenta, dándoles poco o nada de tiempo a los miembros del equipo antes de tomar una decisión. Este estilo de liderazgo debe utilizarse solamente en raras ocasiones.
- **Líder directivo—**Los líderes directivos instruyen a los miembros del equipo sobre las tareas que se requieren y sobre cómo y cuándo deben llevarse a cabo.
- **Líder *Laissez Faire*—**Es un estilo de liderazgo en el que el equipo se queda sin supervisión y donde el líder no interfiere con las actividades laborales diarias. Esto a menudo conduce a un estado de anarquía.
- **Líder de apoyo y entrenamiento—**Los líderes de apoyo y entrenamiento (*Coaching and Supportive*) dan instrucciones; después apoyan y monitorean a los miembros del equipo al escuchar, ayudar, alentar y presentar una perspectiva positiva en tiempos de incertidumbre.
- **Líder enfocado en las tareas—**Los líderes que se enfocan en las tareas (*Task-Oriented*) se encargan de que se hagan las tareas con apego a los plazos.
- **Líder asertivo—**Los líderes asertivos confrontan los problemas y demuestran confianza para establecer autoridad con respeto.

3.10.4.1 Liderazgo servicial (*Servant Leadership*)

El estilo de liderazgo preferido para los proyectos Scrum es el liderazgo servicial. Este término fue descrito por primera vez por Robert K. Greenleaf, en un ensayo titulado *The Servant as Leader*. A continuación, se proporciona un extracto en el que se explica este concepto:

> El líder servicial es primero que nada un servidor... Empieza con el sentimiento natural de que uno quiere servir, servir *primero*. Luego la selección consciente lleva a uno a aspirar a liderar. Esa persona es completamente diferente a aquel que es *líder primero*, tal vez debido a la necesidad de

> mando inusual o de adquirir posesiones materiales... El líder primero y el líder servicial son dos tipos opuestos. Entre ellos hay matices y mezclas que forman parte de la infinita variedad de la naturaleza humana…
>
> La diferencia se manifiesta en el cuidado del que es siervo primero para asegurarse de que las necesidades más prioritarias de la gente estén siendo atendidas. La mejor prueba, que es difícil de administrar, es: ¿A los que se les sirve, crecen como personas? ¿Mientras se les sirve, se vuelven más sanos, más sabios, más libres, más autónomos, más propensos a ser servidores ellos mismos? Y, ¿cuál es el efecto sobre los más desfavorecidos en la sociedad? ¿Van a ser beneficiados o al menos ya no más desfavorecidos? (Greenleaf 1970, 6)

Abundando sobre los escritos de Greenleaf, Larry Spears identifica diez rasgos que cada líder servicial eficaz debe poseer:

1. **Escuchar**—Se busca que los líderes serviciales escuchen con atención y sean receptivos a lo que se dice o no se dice. Estos son capaces de ponerse en contacto con su voz interior para comprender y reflexionar sobre sus propios sentimientos.

2. **Empatía**—Los líderes serviciales buenos aceptan y reconocen a los individuos por sus destrezas únicas y habilidades especiales. Asumen que los trabajadores tienen buenas intenciones y los aceptan como individuos, incluso cuando existen problemas de comportamiento o rendimiento.

3. **Recuperación**—La motivación y la capacidad de recuperarse a sí mismo y la relación con los demás es un fuerte rasgo de los líderes serviciales. Reconocen y se dan la oportunidad de ayudar a sus colegas que están pasando por dolor emocional.

4. **Toma de conciencia (*Awareness*)**—Ser consciente, especialmente ser auto-consciente, es un rasgo de los líderes serviciales. Esto les permite entender mejor e integrar los problemas, tales como los relacionados con la ética, el poder y los valores.

5. **Persuasión**—Los líderes serviciales usan la persuasión, en vez de su posición de autoridad para obtener el consenso colectivo y tomar decisiones. En vez de forzar el cumplimiento y la coerción como es costumbre en algunos estilos autoritarios de gestión, los líderes serviciales practican la persuasión.

6. **Conceptualización**—Una habilidad especial de los líderes serviciales es ver y analizar los problemas (en una organización) desde una perspectiva conceptual y visionaria más amplia, en vez de centrarse en los objetivos inmediatos a corto plazo.

7. **Prospectiva**—Su mente intuitiva les permite a los líderes serviciales utilizar y aplicar las lecciones del pasado y la realidad actual para prever el resultado de situaciones y decisiones actuales.

8. **Administración (*Stewardship*)**—La administración (*Stewardship*) exige un compromiso de servir a los demás. Los líderes serviciales prefieren la persuasión por encima del control para obtener la confianza de los demás en la organización.

9. **Compromiso con el crecimiento de los demás**—Los líderes serviciales tienen un profundo compromiso con el crecimiento de las personas dentro de su organización. Asumen la responsabilidad de nutrir el crecimiento personal, profesional y espiritual de los demás (por ejemplo, facilitando el acceso a lrecursos para el desarrollo personal y profesional, alentando a los trabajadores a participar en la toma de decisiones).

10. **Desarrollo de una comunidad**—Los líderes serviciales están interesados en el desarrollo de comunidades dentro de un ambiente de trabajo. Esto es de gran importancia, en especial dado al cambio en muchas sociedades que dejan de ser comunidades pequeñas para convertirse en grandes instituciones que dan forma y que controlan las vidas humanas.

Scrum cree que todos los líderes de proyectos Scrum (incluyendo al Scrum Master y el Product Owner) deben ser líderes serviciales que tengan las características mencionadas anteriormente.

3.10.5 Teoría de jerarquía de necesidades de Maslow

Maslow (1943) presenta una jerarquía de necesidades la cual reconoce que diferentes personas se encuentran en diferentes niveles en sus necesidades. Por lo general, la gente se preocupa de las necesidades fisiológicas y después se desplaza progresivamente hacia las necesidades jerárquicas.

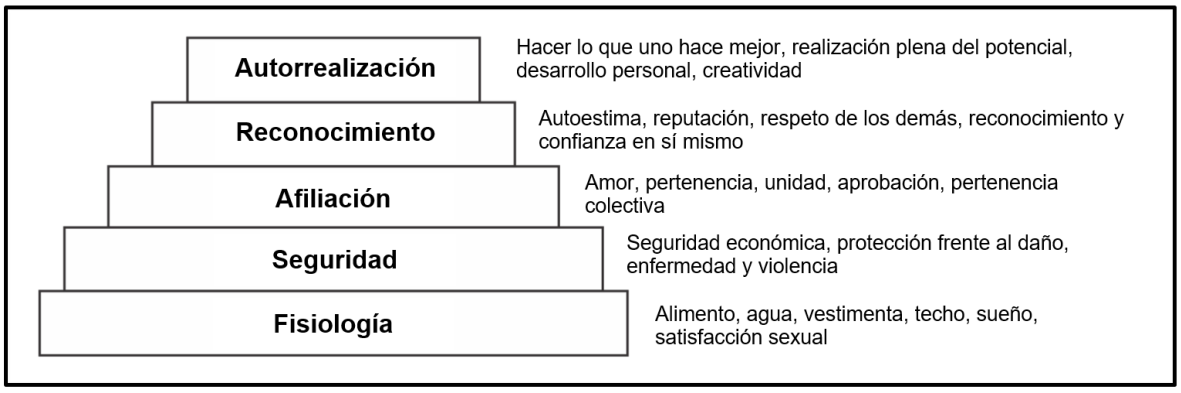


Figura 3-5: Teoría de jerarquía de necesidades de Maslow

Para lograr el éxito, un Equipo Scrum necesita miembros del equipo centrales como los no centrales que hayan alcanzado los niveles de reconocimiento y autorrealización. El concepto de equipos auto-organizados, lo cual es un principio clave en Scrum, requiere que los miembros del equipo sean auto-motivados, que participen y contribuyan plenamente en el cumplimiento de los objetivos del proyecto.

Como líder, el Scrum Master debe entender dónde se encuentra cada persona del equipo está en relación la pirámide. Este entendimiento le ayudará a determinar el mejor enfoque para motivar a cada individuo.

Asimismo, todas las personas fluctúan de arriba a abajo en los niveles de jerarquía de necesidades a lo largo de la vida. Esto se debe a su propia motivación y esfuerzos para avanzar en la jerarquía o, a veces debido a factores fuera de su control. El objetivo del Scrum Master es trabajar con personas en el equipo para mejorar sus habilidades y conocimientos y ayudarle a ascender en la jerarquía de necesidades. Este apoyo resulta en un equipo que está formado por individuos motivados y que son fuertes colaboradores al proyecto y a la organización.

3.10.6 Teoría X y Teoría Y

Douglas McGregor (1960) propuso dos teorías de gestión:

- **Teoría X**—En la Teoría X, los líderes asumen que los empleados están inherentemente desmotivados y evitarán el trabajo si es posible, lo cual justifica un estilo autoritario de gestión.

- **Teoría Y**—En la Teoría Y, los líderes asumen que los empleados son auto motivados y buscan aceptar una mayor responsabilidad. La Teoría Y implica un estilo más participativo de gestión.

Es improbable que los proyectos Scrum tengan éxito si el Scrum Master o el Product Owner son líderes de la Teoría X. Todos los líderes de proyectos Scrum deben apegarse a la Teoría Y, mediante la cual vean a los empleados por sus cualidades importantes, a la vez que ayudan a empoderar y desarrollar las habilidades de los miembros de su equipo. Es de suma importancia también expresar a los miembros del equipo aprecio por el trabajo que han hecho para lograr los objetivos del proyecto.

4. JUSTIFICACIÓN DEL NEGOCIO

4.1 Introducción

El propósito de este capítulo es entender el concepto y el propósito de la justificación del negocio en su relación a proyectos Scrum. Es importante que una organización lleve a cabo una adecuada justificación del negocio y elabore una declaración de la visión del proyecto (*Project Vision Statement*) viable antes de dar inicio a cualquier proyecto. Esto ayudará a las personas claves que toman decisiones a entender la necesidad empresarial de cambio o de un nuevo producto o servicio, así como la justificación para seguir adelante con un proyecto. Esto también ayuda a que el Product Owner elabore un Backlog Priorizado del Producto tomando en cuenta las expectativas empresariales de la alta gerencia y de (los) stakeholder(s).

La *justificación del negocio*, según se define en la *Guía para el conocimiento de Scrum* (*Guía SBOK™*), aplica a lo siguiente:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se le entregue a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

4.2 Guía de roles—Esta sección ofrece una orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: Product Owner, Scrum Master y Equipo Scrum.

4.3 Entrega basada en el valor—Esta sección describe el concepto del valor del negocio y su importancia en cualquier proyecto. Brinda también información sobre las responsabilidades de los distintos individuos, incluyendo el Product Owner, quien participa en el logro del valor para el negocio.

4.4 Importancia de la justificación del negocio—En esta sección se detalla la importancia de la justificación del negocio, los factores que la determinan y cómo se mantiene y se verifica durante todo el proyecto.

4.5 Técnicas de justificación del negocio—Esta sección describe a detalle cómo se evalúa y se verifica la justificación del negocio utilizando diversas herramientas.

4.6 Justificación continua de valor—En esta sección se detalla la importancia de la justificación continua de valor y abunda sobre la forma en la que esta se logra.

4.7 Confirmar la realización de beneficios—Esta sección describe la forma en la que se logran los beneficios durante el proyecto.

4.8 Resumen de responsabilidades—Esta sección define las responsabilidades pertinentes a la justificación del negocio para los miembros del equipo del proyecto en función de sus roles.

4.9 Scrum vs. Gestión tradicional de proyectos—Esta sección destaca los beneficios empresariales del método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

4.2 Guía de roles

1. Product Owner—La justificación del negocio la realiza principalmente el Product Owner; por lo tanto, todo este capítulo aplica más a este rol.

2. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 4.3, 4.4, 4.6, 4.7 y 4.8.

3. Equipo Scrum—El Equipo Scrum debe enfocarse principalmente en las secciones 4.3, 4.7 y 4.8.

4.3 Entrega basada en valor (*Value-driven Delivery*)

Un proyecto es un emprendimiento colaborativo para crear nuevos productos o servicios, o para obtener resultados como los que se definen en la declaración de la visión del proyecto (*Project Vision Statement*). Los proyectos por lo general se ven afectados por limitaciones de tiempo, costo, alcance, calidad, personal y la capacidad de la organización. Por lo general, se busca que los resultados que generen los proyectos resulten en algún tipo de valor de negocio o servicio.

Dado a que el valor es una razón principal de cualquier organización para seguir adelante con un proyecto, la entrega basada en valor (del inglés: *Value-Driven Delivery)* debe ser el principal enfoque. El ofrecer valor es algo que está arraigado en el framework de Scrum. Scrum facilita la entrega anticipada de valor en el proyecto y lo sigue haciendo a lo largo del ciclo de vida del mismo.

Una de las características claves de cualquier proyecto es la incertidumbre de los resultados. Es imposible garantizar el éxito del proyecto, independientemente de su tamaño o complejidad. Por lo tanto, tomando en cuenta esta incertidumbre de alcanzar el éxito, es importante empezar a entregar resultados durante el proyecto tan pronto como sea posible. Esta entrega temprana de buenos resultados, y por lo tanto de valor, brinda una oportunidad para la reinversión, demostrando el valor del proyecto a los stakeholders interesados.

A fin de ofrecer una entrega basada en valor, es importante:

1. Entender lo que agrega valor a los clientes y usuarios y dar prioridad a los requerimientos de alto valor que encabezan el Backlog Priorizado del Producto.
2. Disminuir la incertidumbre y atender constantemente de los riesgos que potencialmente pudieran reducir el valor en caso de materializarse. También es importante trabajar en estrecha colaboración con los stakeholders del proyecto mostrándoles incrementos del producto al final de cada sprint, lo cual permite una gestión efectiva de cambios.
3. Crear entregables basados en las prioridades determinadas por la producción de incrementos del producto potencialmente entregables durante cada sprint. De esta forma, los clientes empiezan a ver el valor desde el principio del proyecto.

En Scrum, el concepto de la entrega basada en valor, hace que el framework de Scrum sea muy atractivo para los stakeholders del negocio y para la alta gerencia. Este concepto es muy diferente en comparación a los modelos tradicionales de gestión de proyectos, donde:

1. A los requerimientos no se les da prioridad con base al valor del negocio;
2. Los cambios en los requerimientos después del inicio del proyecto resultan difíciles y solo se pueden hacer mediante un tardado proceso de gestión;
3. El valor se logra sólo al final del proyecto, al entregarse el producto o servicio final.

La figura 4-1 contrasta la entrega basada en valor en Scrum en comparación a los proyectos tradicionales.

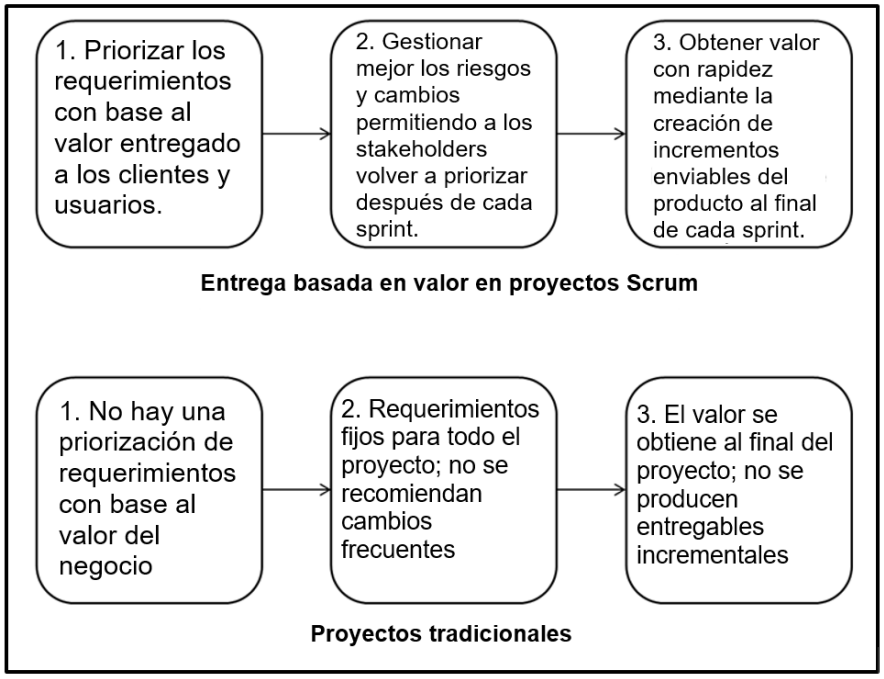


Figura 4-1: Entrega de valor en Scrum vs. Proyectos tradicionales

4.3.1 Responsabilidades del Product Owner en la justificación del negocio

La responsabilidad de priorizar y entregar valor de negocio para los proyectos en una organización le corresponde principalmente al Product Owner. Para los programas y portafolios, la responsabilidad recae en el Program Product Owner y en el Portfolio Product Owner, respectivamente. Su función es la de actuar como representantes efectivos del cliente y/o del patrocinador. Las directrices para la evaluación y medición del valor de negocio típicamente pueden ser establecidas por un Scrum Guidance Body.

La figura 4-2 ilustra las responsabilidades de la justificación del negocio en orden jerárquico.

Portfolio Product Owner	• Entrega el valor para los portafolios • Crea la justificación del negocio para los portafolios • Proporciona una guía de valor para los programas • Aprueba la justificación del negocio para los programas
Program Product Owner	• Entrega el valor de los programas • Crea la justificación del negocio para los programas • Proporciona una guía de valor para los proyectos • Aprueba la justificación del negocio para los proyectos
Product Owner	• Entrega el valor de los proyectos • Crea la justificación del negocio para los proyectos • Le confirma el logro de beneficios a los stakeholders

Figura 4-2: Jerarquía de responsabilidades en la justificación del negocio

4.3.2 Responsabilidades de otros roles de Scrum en la justificación del negocio

Es importante señalar que, aunque el Product Owner es el responsable principal de la justificación del negocio, hay otras personas en proyectos Scrum que también contribuyen considerablemente de la siguiente forma:

1. El **patrocinador** o *sponsor*, proporciona los fondos para el proyecto y supervisa constantemente el proyecto para confirmar el logro de los beneficios.

2. Los **clientes** y **usuarios** participan en la definición de la lista priorizada de los requisitos y de las historias de usuario en Backlog Priorizado del Producto; en la revisión de los entregables después de cada sprint o lanzamiento y en la confirmación de los beneficios.

3. El **Scrum Guidance Body** puede proporcionar directrices y recomendaciones relacionadas a las técnicas de justificación del negocio y confirmar la realización de beneficios, etc. Tales directrices y recomendaciones pueden después remitirse los equipos centrales de Scrum y a los stakeholders.

4. El **Scrum Master** facilita la creación de entregables del proyecto; está a cargo de la gestión riesgos, de cambios e impedimentos durante los procesos de *Realizar Daily Standup* y de la *Retrospectiva del sprint*, entre otros. El Scrum Master se coordina con el Equipo Scrum para crear los entregables y con el Product Owner y otros stakeholders para asegurar que los beneficios del proyecto se materialicen.

5. El **Equipo Scrum** trabaja en la creación de entregables del proyecto y contribuye a la creación de valor del negocio para todos los stakeholders y para el proyecto. El Equipo Scrum también participa en los procesos de *Desarrollar de épica(s)*, *Crear el Backlog Priorizado del Producto*, *Crear historias de usuario*, *Estimar historias de usuario*, *Comprometer historias de usuario,* así como en procesos asociados donde se definen y se priorizan los requerimientos del negocio. El Equipo Scrum también ayuda en la identificación de riesgos y presenta solicitudes de cambio para las mejoras durante las reuniones de retrospectiva del sprint y otras reuniones.

4.4 Importancia de la justificación del negocio

La justificación del negocio demuestra las razones para emprender un proyecto. Responde a la pregunta: "¿Por qué es necesario este proyecto?" La justificación del negocio es lo que impulsa todas las decisiones relacionadas aun proyecto. Por lo tanto, es importante evaluar la viabilidad de un proyecto, no solo antes de comprometerse a gastos o inversiones considerables en las etapas iniciales, sino también a lo largo del ciclo de vida del proyecto. Un proyecto debe cancelarse si se considera que no es viable; la decisión debe ser escalada a los stakeholders pertinentes y a la alta gerencia. La justificación del negocio de un proyecto debe ser evaluada al inicio de este, en intervalos predefinidos durante todo el proyecto y en cualquier momento cuando surgen grandes problemas o riesgos que amenacen su viabilidad.

4.4.1 Factores que se utilizan para determinar la justificación del negocio

Existen numerosos factores que un Product Owner debe tomar en cuenta para determinar la justificación del negocio de un proyecto. Los siguientes son algunos de los factores más importantes:

1. **Razonamiento del proyecto**

   El razonamiento del proyecto incluye todos los factores que este requiere, ya sean positivos, negativos, elegidos o no (por ejemplo: capacidad inadecuada para cumplir con la demanda actual y

la demanda prevista, la disminución en la satisfacción del cliente, baja utilidad, requerimientos legales, etc.).

2. **Necesidades del negocio**

   Las necesidades del negocio son aquellos resultados del negocio que se espera que cumpla el proyecto, tal como se documenta en la declaración de visión del proyecto.

3. **Beneficios del proyecto**

   Los beneficios del proyecto incluyen todas las mejoras cuantificables de un producto, servicio o resultado que se pudieran obtener durante la conclusión satisfactoria de un proyecto.

4. **Costo de oportunidad**

   El costo de oportunidad es el valor de la siguiente mejor opción de negocio o proyecto que fue descartado en favor del proyecto seleccionado.

5. **Riesgos mayores**

   Los riesgos incluyen eventos inciertos o no planeados que pudieran afectar la viabilidad y el posible éxito del proyecto.

6. **Escalas de tiempo del proyecto (*Project Timescales*)**

   Las escalas de tiempo reflejan la duración de un proyecto y el tiempo durante el cual se obtendrán los beneficios del mismo.

7. **Costos del proyecto**

   Los costos del proyecto son las inversiones y demás costos de desarrollo en un proyecto.

4.4.2 Justificación del negocio y el ciclo de vida del proyecto

Antes de iniciar un proyecto, primero se evalúa la justificación del negocio y se verifica constantemente durante todo el ciclo de vida del mismo. Los siguientes pasos captan la forma en la que se determina la justificación del negocio:

1. **Evaluar y presentar un caso de negocio**
   La justificación del negocio para un proyecto normalmente la analiza y la confirma el Product Owner. Se documenta y se presenta en forma de caso de negocio del proyecto antes de la fase de inicio y consiste en tomar en cuenta los diversos factores especificados en la sección 4.4.1. Una

vez documentado, el Product Owner debe crear una declaración de la visión del proyecto y obtener la aprobación de aquellos que toman las decisiones claves en la organización. Generalmente, son los ejecutivos y/o algún consejo administrativo del proyecto o del programa.

2. Justificación continua de valor

Una vez que los tomadores de decisiones aprueban la declaración de la visión del proyecto, esta se utiliza como base de referencia y forma la justificación del negocio. La justificación del negocio se valida durante toda la ejecución del proyecto, por lo general en intervalos predefinidos, como en reuniones del portafolio, del programa o del Backlog Priorizado del Producto y cuando se identifican los principales problemas y riesgos que amenazan la viabilidad del proyecto. Esto puede darse en varios procesos de Scrum, incluyendo el proceso de *Realizar Daily Standup* y en el *Refinar el Backlog Priorizado del Producto*. A lo largo del proyecto, el Product Owner debe mantener actualizada la justificación del negocio en la declaración de la visión del proyecto con información relevante del proyecto para que los que toman decisiones importantes continúen tomando decisiones informadas.

2. Confirmar la realización de beneficios

El Product Owner confirma el logro de los beneficios organizacionales durante el proyecto y al completar las historias de usuario en el Backlog Priorizado del Producto. Los beneficios de los proyectos Scrum se materializan durante los procesos de *Demostrar y validar el sprint*, *Retrospectiva del sprint*, *Enviar entregables* y *Retrospectiva del proyecto*.

La figura 4-3 resume los pasos para determinar justificación del negocio.

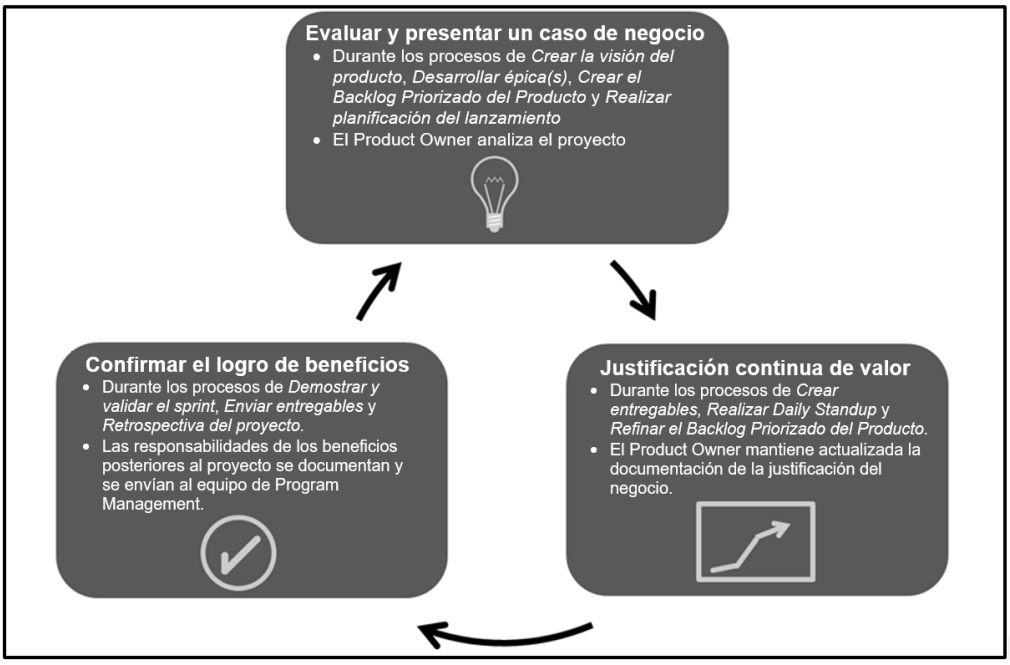


Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto

4.5 Técnicas de justificación del negocio

Las siguientes secciones abordan algunas de las herramientas que se utilizan para valorar y evaluar la justificación del negocio, así como otros aspectos relacionados a la justificación y selección del proyecto. No es necesario, ni se recomienda utilizar todas las técnicas disponibles para cada proyecto. Algunas técnicas no son apropiadas dependiendo del proyecto específico y también se pueden utilizar para evaluar los proyectos en forma individual o para comparar el valor esperado de múltiples proyectos.

El Scrum Guidance Body (SGB), —que puede consistir en un grupo de expertos o en un conjunto de documentos sobre normas y procedimientos de la organización—, define los lineamientos y parámetros que se utilizarán para evaluar el valor del negocio. Sin embargo, cada Product Owner es responsable de llevar a cabo las actividades para verificar y dar seguimiento al valor de negocio para sus respectivos proyectos, programas o portafolios.

4.5.1 Estimación del valor del proyecto

El valor que habrán de brindar los proyectos empresariales puede calcularse utilizando diversos métodos, tales como el retorno sobre la inversión (RSI), valor presente neto (VPN) y la tasa interna de retorno (TIR).

1. **Retorno sobre la inversión (RSI)**

   El retorno sobre la inversión (RSI), al utilizarse para la justificación de un proyecto, evalúa los ingresos netos esperados que se buscan obtener a partir de un proyecto. Se calcula deduciendo los costos esperados o la inversión en un proyecto de su ingreso previsto; después se divide (la utilidad neta) por los costos previstos a fin de obtener la tasa de retorno. Otros factores, tales como la inflación y las tasas de interés sobre el dinero prestado pudieran incluirse en los cálculos del RSI.

   Fórmula del RSI:

   RSI = (Ingresos del proyecto – Costo del proyecto) / Costo del proyecto

   Ejemplo: El RSI para un proyecto que tendrá un costo de $125,000 en desarrollarse y con beneficios económicos estimados en $300,000, se calcula de la siguiente forma:

   RSI = ($300,000 – $125,000) / $125,000 = 1.4

   Por lo tanto, el RSI es 1.4 veces la inversión (o 140 %).

   Los incrementos frecuentes del servicio o producto son una base fundamental de Scrum que permiten la verificación temprana del RSI. Esto ayuda en la evaluación de la justificación continua de valor.

2. Valor presente neto (VPN)

El valor presente neto (VPN) es un método que se utiliza para determinar el valor neto actual de un futuro beneficio económico, dada una inflación o tasa de interés prevista. En otras palabras, el VPN es el ingreso total esperado o los ingresos de un proyecto, restando el costo total previsto del proyecto y tomando en cuenta el valor actual de la divisa.

Ejemplo: ¿Cuál de los siguientes dos proyectos es la mejor opción si se utiliza el VPN como criterio de selección?

- El proyecto A tiene un VPN de $1,500 y se completará en 5 años.
- El proyecto B tiene un VPN de $1,000 y se completará en 1 año.

Solución: El proyecto A, ya que su VPN es más elevado. Aquí no se toma en cuenta el hecho de que el proyecto B tiene una duración más corta que el proyecto A, pues el tiempo ya está representado en los cálculos del VPN (debido a que es el valor actual y no el valor futuro que se considera en el cálculo).

3. Tasa interna de retorno (TIR)

La tasa interna de retorno (TIR), es una tasa de descuento sobre una inversión en la cual el valor actual de los flujos de efectivo se iguala al valor actual del flujo de salidas de efectivo a fin de evaluar la tasa de retorno del proyecto. Al hacer un comparativo de los proyectos, generalmente resulta mejor aquel que cuenta con una tasa interna de retorno más elevada.

Aunque no se utiliza para justificar proyectos con tanta frecuencia como con algunas otras técnicas, tales como el VPN, la TIR es un concepto importante que hay que conocer.

Ejemplo: Basado en el TIR, ¿cuál proyecto es más conveniente?

- Proyecto A, que tiene una TIR del 15 % y se completará en 5 años.
- Proyecto B, que tiene una TIR del 10 % y se completará en 1 año.

Solución: El proyecto A, ya que su TIR es mayor. Aquí no se toma en cuenta el hecho de que el proyecto B tiene una duración más corta que el proyecto A, pues el tiempo ya está representado en los cálculos del TIR (tal como en el VPN, es el valor actual y no el valor futuro el que se utiliza para determinar la TIR).

4.5.2 Planificar para el valor

Después de justificar y confirmar el valor de un proyecto, el Product Owner debe considerar las políticas de la organización, los procedimientos, las plantillas y las normas generales dictadas por el Scrum Guidance Body (o el puesto similar o una junta organizacional del proyecto) en la planificación de un proyecto; y a la vez, maximizar la entrega basada en valor. La planificación para el valor es la justificación y confirmación del valor del proyecto. La responsabilidad de determinar cómo se crea valor recae en los stakeholders (patrocinadores, clientes y/o usuarios), mientras que el Equipo Scrum se concentra en lo que se habrá de desarrollar. Algunas de las herramientas comunes recomendadas por un Scrum Guidance Body pudieran ser las siguientes:

1. **Mapa de flujo de valor (*Value Stream Mapping*)**

   El mapa de flujo de valor utiliza diagramas de flujo para ilustrar el flujo de información necesaria para completar un proceso. Esta técnica pudiera utilizarse para racionalizar un proceso ayudando a determinar los elementos que no aportan valor.

2. **Priorización basada en el valor para el cliente (*Customer Value-based Prioritization*)**

   La priorización basada en el valor para el cliente le da importancia primordial al cliente y se esfuerza primero en implementar las historias de usuario con más alto valor. Dichas historias de usuario de alto valor se identifican y se colocan en la parte superior del Backlog Priorizado del Producto.

   Un equipo puede utilizar una variedad de esquemas de priorización para determinar las características de alto valor.

   a. **Esquemas simples**

      Los esquemas simples implican etiquetar elementos como prioridad “1”, “2”, “3” o “alta”, “media” y “baja”, y así sucesivamente. Aunque se trata de un método sencillo y directo, puede llegar a ser problemático, ya que a menudo hay una tendencia en etiquetar todo como prioridad “1” o “alta”. Incluso los métodos de priorización tales como “alta”, “media” y “baja” pueden encontrarse con dificultades similares.

   b. **Priorización MoSCoW**

      El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: “Debe tener” (*Must have*), “Debería tener” (*Should have*), “Podría tener” (*Could have*) y “No tendrá” (*Won’t have*). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de “debería tener”, siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de “gustaría que tuviera” siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.

c. Dinero de Monopoly

El dinero de Monopoly es una técnica que consiste en darle al cliente dinero del juego Monopoly, o "dinero falso", equivalente a la cantidad del presupuesto del proyecto, solicitando que lo distribuyan entre las historias de usuario que están a consideración. De esta forma, el cliente prioriza con base en lo que está dispuesto a pagar por cada historia de usuario.

d. Método de los 100 puntos

El método de los 100 puntos fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes.

e. Análisis de Kano

El análisis de Kano fue desarrollado en 1984 por Noriaki Kano y consiste en clasificar las características o requisitos en cuatro categorías con base en las preferencias del cliente:

1. *Calidad atractiva (*Exciters/Delighters*):* Características que son nuevas o de gran valor para el cliente
2. *Calidad unidimensional* (Satisfiers): Características que le ofrecen valor al cliente
3. *Calidad requerida (*Dissatisfiers*):* Características que, si no están presentes, pudieran causar la insatisfacción del cliente respecto al producto, pero que no afectan el nivel de satisfacción si se cuenta con ellas.
4. *Calidad indiferente (*Indifferent*):* Características que no afectarán al consumidor de ninguna manera y deben ser eliminadas.

La figura 4-4 muestra una ilustración del Análisis de Kano:

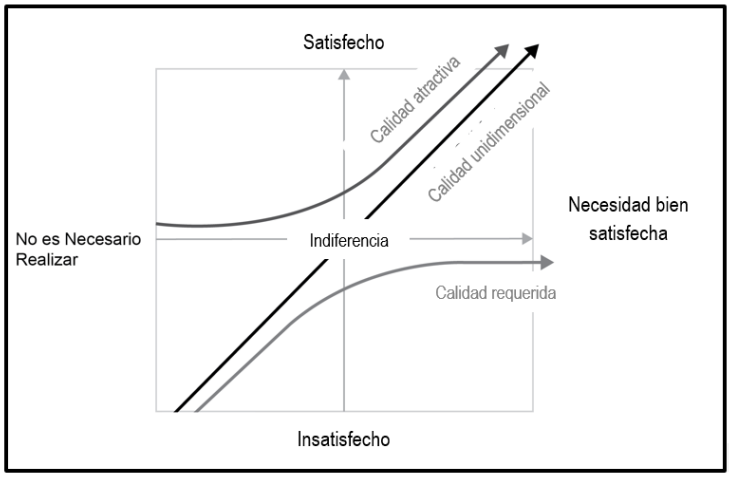


Figura 4-4: Análisis de Kano

Curiosamente, con el tiempo, las características por lo general se desplazan hacia abajo en la lista de clasificación; los clientes esperan ciertas características (por ejemplo, las cámaras en los teléfonos) y dichas funciones dejaron de ser una calidad atractiva para convertirse, eventualmente, en calidad requerida.

4.5.3 Clasificación relativa de priorización (*Relative Prioritization Ranking*)

La clasificación relativa de priorización, conocida en inglés como: *Relative Prioritization Ranking*, es una simple enumeración de historias de usuario por orden de prioridad. Es un método eficaz para determinar las historias deseadas para cada iteración o lanzamiento del producto o servicio. El objetivo es crear una lista simple, con el único objetivo de dar prioridad a las características, en vez de distraerse con múltiples esquemas de priorización.

Esta sencilla lista también brinda una base para incorporar los cambios y los riesgos identificados cuando sea necesario. Cada cambio o riesgo identificado se puede incluir en la lista con base en su prioridad relativa a las demás historias de usuario en la lista. Típicamente, los nuevos cambios se incluirán a expensas de las características que se han asignado como con prioridad más baja.

Definir las Características Mínimas de Mercado (*Minimum Marketable Features - MMF*) es de suma importancia durante este proceso, de modo que la primera versión o iteración ocurra tan pronto como sea posible, lo cual lleva a un aumento en el retorno sobre la inversión. Normalmente, estas historias de usuario se clasifican como alta prioridad.

4.5.4 Mapeo de historias

El mapeo de historias, conocido en inglés como *Story Mapping*, es una técnica para proporcionar un esquema visual del producto y sus componentes clave. El mapeo de historias, formulado por primera vez por Jeff Patton (2005), se utiliza comúnmente para ilustrar la ruta del producto.

Los mapas de historia representan la secuencia de las iteraciones de desarrollo del producto y trazan las características que serán incluidas en el primer, segundo, tercero y subsecuentes lanzamientos.

4.6 Justificación continua de valor

La justificación continua de valor describe la evaluación con regularidad del valor del negocio para determinar si aún existe la justificación o viabilidad en la ejecución del proyecto. La evaluación frecuente de la inversión en el proyecto en relación con el valor del negocio que se está creando, califica la viabilidad de un proyecto. Los requisitos esperados del proyecto pueden cambiar con frecuencia, lo que puede afectar tanto la inversión del proyecto, como la creación de valor. Un aspecto clave de Scrum es su capacidad para adaptarse rápidamente al caos creado por un modelo de negocio que cambia rápidamente. En los

proyectos donde los requerimientos de usuario son ambiguos y los cambios son frecuentes, Scrum ofrece considerables ventajas sobre otros modelos de desarrollo.

Monitorear la tasa de entrega de valor es un requisito importante para los proyectos Scrum. El seguimiento constante y los reportes sobre la creación de valor ayudan a evaluar el estado del proyecto y ofrecen información importante para el cliente y otros stakeholders.

4.6.1 Análisis del valor ganado (AVG)

Aunque se utilizan comúnmente, las herramientas tales como las gráficas de barras y diagramas de Gantt tienen sus limitaciones en el seguimiento y presentación de informes de progreso cuando se trata de rendimiento del proyecto. El análisis del valor ganado (AVG) se utiliza para tales efectos.

El análisis del valor ganado analiza el verdadero rendimiento del proyecto en comparación al rendimiento planeado en un punto previsto. Para que las técnicas de rastreo sean eficaces, el plan de base del proyecto debe ser precio. El AVG generalmente utiliza gráficas y demás elementos visuales (Curva S) como forma de representar la información sobre el estado del proyecto.

El análisis del valor ganado mide las variaciones actuales en la agenda del proyecto, así como el costo de funcionamiento y prevé del costo final con base en el rendimiento actual determinado. El AVG generalmente se lleva a cabo al final de cada sprint una vez que se han concluido las historias de usuario y el Sprint Backlog.

La tabla 4-1 resume las fórmulas que se utilizan en el análisis del valor ganado.

Definición de término	Siglas (Inglés)	Fórmula
Valor planeado	PV	
Valor ganado	EV	
Costo actual	AC	
Presupuesto al finalizar	BAC	
Varianza del cronograma	SV	EV – PV
Varianza del costo	CV	EV – AC
Índice de desempeño del programa	SPI	EV / PV
Índice de desempeño de costo	CPI	EV / AC
Porcentaje completado	% Completado	(EV / BAC) x 100
Estimado al completar 1. Las suposiciones de estimación no válidas 2. Las diferencias actuales son atípicas	EAC	 1. AC + ETC 2. AC + BAC – EV

3. Las variaciones actuales son típicas		3. BAC / CPI
Estimado para completar	ETC	EAC – AC
Variancia al completar	VAC	BAC – EAC

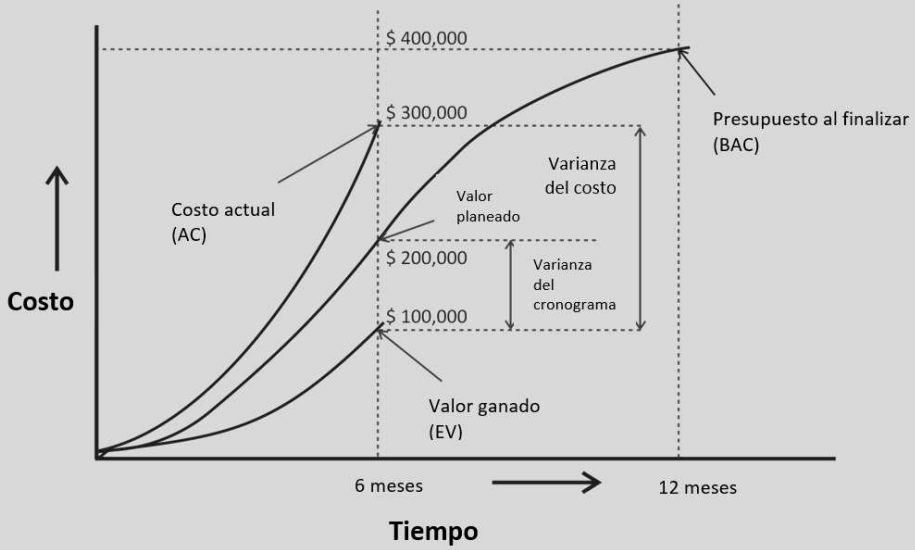
Tabla 4-1: Fórmulas del valor ganado

Ejemplo: Un sitio web con 4,000 páginas necesita ser desarrollado: asumimos que cada página web lleva el mismo tiempo en completar, y que cada página web es una historia de usuario única de igual prioridad en el Backlog Priorizado del Producto. El costo estimado para completar el proyecto es de $400,000 y el límite de tiempo para el proyecto es de 12 meses. Después de 6 meses, se han gastado $300,000 y el trabajo realizado es 1,000 páginas web.

¿Qué información se nos ha proporcionado?

- Presupuesto al finalizar (BAC) = $400,000 (Costo base del proyecto)
- Valor planeado (PV) = $200,000 (ya que habíamos planeado completar 2,000 páginas web)
- Valor ganado (EV) = $100,000 (el valor de 1,000 páginas web que se han completado)
- Costo actual (AC) = $300,000 (lo que se ha gastado hasta el momento)

Curva S para los datos:



Fórmulas:

- Varianza del cronograma (SV) = EV – PV = $100,000 – $200,000 = –$100,000
- Varianza del costo (CV) = EV – AC = $100,000 – $300,000 = –$200,000
  - Las variaciones negativas en nuestro proyecto indican que estamos atrasados en el programa y por encima del presupuesto.
- Índice de desempeño del programa (SPI) = EV / PV = $100,000 / $200,000 = 0.5
  - El SPI < 1 indica que el trabajo realizado hasta el momento es solo el 50 % de lo que habíamos planeado terminar en 6 meses.
- Índice de desempeño de costo (CPI) = EV / AC = $100,000 / $300,000 = 0.33
  - El CPI < 1 indica que solo termina el 33 % del trabajo por la cantidad de dinero que se gasta.
- Porcentaje completado = EV / BAC x 100 = $100,000 / $400,000 x 100 = 25 %
  - Por lo tanto, hasta el momento se ha completado el 25 % del trabajo en el proyecto.

4.6.2 Diagrama de flujo acumulativo (DFA)

El diagrama de flujo acumulativo (DFA) es una herramienta útil para la elaboración de informes y para el seguimiento de los resultados del proyecto. Proporciona una representación sencilla y visual del avance del proyecto en un punto de tiempo determinado. Se utiliza generalmente para brindar un estado de mayor nivel de la totalidad del proyecto y no para actualizaciones individuales diarias de sprints.

La figura 4-5 muestra un ejemplo de un DFA para un proyecto grande. Muestra cuántas historias de usuario están por crearse, las que están en proceso de creación y las que ya se han creado. A medida que cambian los requisitos de los clientes, se produce un cambio en las historias de usuario acumuladas que han de ser entregadas. Los puntos de cambio 1 y 2 son donde el Product Owner ha eliminado las historias de usuario existentes en el Backlog Priorizado del Producto del riesgo ajustado, mientras que los puntos de cambio 3 y 4 son donde el Product Owner agregó nuevas historias de usuario en dicha lista.

Este tipo de diagrama puede ser una gran herramienta para la identificación de obstáculos y embotellamiento en los procesos. Por ejemplo, si el diagrama muestra una banda cada vez más estrecha, mientras la banda anterior con el tiempo está cada vez más amplia, pudiera haber un embotellamiento, por lo que pudieran ser necesarios cambios para aumentar la eficiencia y/o mejorar el desempeño del proyecto.

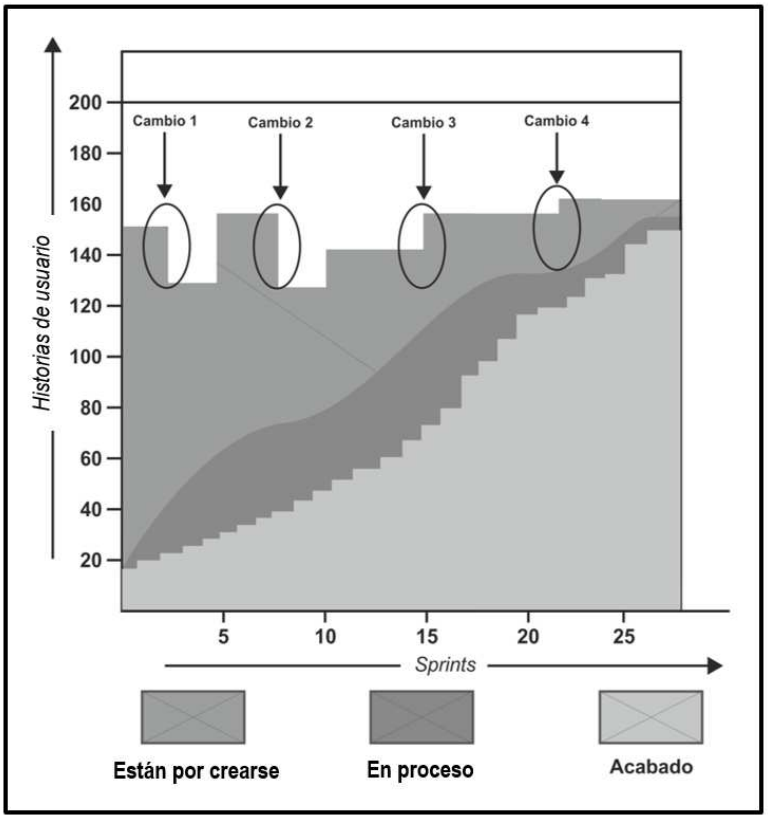


Figura 4-5: Ejemplo de diagrama de flujo acumulativo (DFA)

4.7 Confirmar la realización de beneficios

Durante todo un proyecto, es importante verificar si se están logrando los beneficios. Ya sea si los productos de un proyecto Scrum son tangibles o intangibles, se requieren técnicas adecuadas de verificación para confirmar que el equipo esté creando los entregables que lograrán los beneficios y el valor definido al inicio del proyecto.

4.7.1 Prototipos, simulaciones y demostraciones

La demostración de los prototipos (no confundirse con los “prototipos de usuario”, conocidos en inglés como *personas*) a los clientes y simular su funcionalidad, son técnicas comúnmente utilizadas para confirmar el valor.

Generalmente, después de usar las características o después de haberlas visto, los clientes pueden determinar con mayor claridad si las características son aptas y adecuadas para sus necesidades. Pueden darse cuenta de la necesidad de características adicionales, o pueden decidir modificar los requisitos de características previamente definidos. En el desarrollo de productos, esta experiencia del cliente ha llegado a ser conocida como IKIWISI, por sus siglas en inglés (*I’ll know it when I see it*) o “Lo sabré cuando lo vea”.

Por medio de las demostraciones o del acceso a las iteraciones anticipadas, los clientes pueden también evaluar a qué grado el equipo ha sabido interpretar sus necesidades y cumplir con sus expectativas.

4.8 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	• Se asegura de que los entregables del proyecto se completen de acuerdo con los criterios de aceptación acordados • Lleva a cabo la justificación continua de valor para los proyectos
Product Owner/ Chief Product Owner	• Asegura la entrega de valor en los proyectos • Mantiene la justificación del negocio para los proyectos • Confirma y comunica los beneficios del proyecto a los stakeholders
Scrum Master/ Chief Scrum Master	• Garantiza que los resultados esperados del proyecto sean comunicados y comprendidos por el Equipo Scrum • Lleva a cabo la justificación continua de valor para los proyectos
Program Product Owner	• Garantiza la entrega de valor para los programas • Crea la justificación del negocio para los programas • Proporciona una guía de valor para los programas en los programas • Aprueba la justificación del negocio de los programas en un programa
Program Scrum Master	• Garantiza que los resultados esperados del programa se comuniquen y se entiendan • Lleva a cabo la justificación continua de valor para los programas
Portfolio Product Owner	• Garantiza la entrega de valor para los portafolios • Crea la justificación del negocio para los portafolios • Proporciona una guía de valor para los programas en los portafolios • Aprueba la justificación del negocio de los programas en un portafolio
Portfolio Scrum Master	• Garantiza que se logren los resultados deseados del portafolio • Lleva a cabo la justificación continua de valor para los portafolios
Stakeholder(s)	• Ayuda a priorizar las historias de usuario y los requerimientos del Backlog Priorizado del Producto. • Se comunica con el Equipo Scrum y confirma la realización del valor al final de cada sprint, de cada lanzamiento y del proyecto.
Scrum Guidance Body	• Establece las directrices generales y los parámetros para evaluar el valor • Actúa en forma consultiva y brinda asesoría en proyectos, programas y portafolios según sea necesario

Tabla 4-2: Resumen de las responsabilidades pertinentes a la justificación del negocio

4.9 Scrum vs. Gestión tradicional de proyectos

Los proyectos tradicionales hacen énfasis en una amplia planificación y en el apego al plan de proyecto creado por el Project Manager. Por lo general, los cambios se administran mediante un sistema formal de gestión de cambio y el valor se crea al final del proyecto cuando se entrega el producto final.

En los proyectos Scrum, no se realiza una extensa planificación de largo plazo antes de la ejecución del proyecto. La planificación se realiza de manera iterativa antes de cada sprint. Esto permite una respuesta rápida y eficaz a los cambios, lo cual se traduce en menores costos y en última instancia, aumenta la rentabilidad y el retorno sobre la inversión (RSI). Asimismo, la entrega basada en valor (sección 4.3) es un beneficio clave del framework de Scrum y brinda una mejor priorización considerablemente mejor, así como una más rápido del valor del negocio. Debido a la naturaleza iterativa del desarrollo Scrum, siempre hay al menos una versión disponible del producto con las características mínimas de mercado (MMF, por sus siglas en inglés). Incluso si un proyecto se cancela antes de completarse, por lo general hay algunos beneficios o valor generado antes de su interrupción.

5. CALIDAD

5.1 Introducción

El propósito de este capítulo es definir la calidad en lo que respecta a los proyectos y presentar el enfoque de Scrum para alcanzar los niveles de calidad requeridos.

La *calidad*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™),* es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

5.2 Guía de roles—Esta sección ofrece orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: Product Owner, Scrum Master y el Equipo Scrum.

5.3 Definición de calidad—Esta sección brinda la definición de Scrum sobre calidad, con una clara distinción de alcance y describe la relación entre la calidad y el valor del negocio.

5.4 Criterios de aceptación y la Backlog Priorizado del Producto—Esta sección hace énfasis en la importancia de los criterios de aceptación, el Backlog Priorizado del Producto y la relación entre estos ambos. También se explica la definición de *terminado* en el contexto de Scrum.

5.5 Gestión de calidad en Scrum—Esta sección proporciona detalles sobre planificación de calidad, control de calidad y garantía de calidad en el contexto de Scrum.

5.6 Resumen de responsabilidades—Esta sección describe las responsabilidades pertinentes a la calidad para cada persona o rol en un proyecto.

5.7 Scrum vs. Gestión tradicional de proyectos—Esta sección destaca los beneficios de la gestión de calidad en el método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

5.2 Guía de roles

1. Product Owner—Es importante que cualquier persona que asuma el rol de Product Owner en proyectos Scrum, lea completamente este capítulo.
2. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 5.3, 5.4, 5.5.3 y 5.6.
3. Equipo Scrum—El Equipo Scrum debe enfocarse principalmente en las secciones 5.3, 5.4 y 5.6.

5.3 Definición de calidad

Hay muchas maneras de definir la calidad.

En Scrum, la calidad se define como la capacidad con la que cuenta un producto terminado o los entregables para cumplir con los criterios de aceptación y lograr el valor del negocio que espera el cliente.

Para asegurar que un proyecto cumpla con los requisitos de calidad, Scrum adopta un enfoque de mejora continua donde el equipo aprende de sus experiencias y de la participación de los stakeholders para mantener constantemente actualizado el Backlog Priorizado del Producto con cualquier cambio en los requerimientos. Dicha lista nunca está completa hasta el cierre o conclusión del proyecto. Cualquier cambio en los requisitos refleja los cambios en el entorno empresarial interno y externo y permite que el equipo funcione continuamente y se adapte para alcanzar esos requisitos. Scrum requiere que el trabajo se realice en incrementos durante los sprints, lo cual significa que los errores o defectos se detectan durante las pruebas de calidad repetitivas y no cuando el producto final o servicio está casi terminado. Asimismo, las tareas importantes relacionadas a la calidad (por ejemplo: desarrollo, pruebas y documentación) se completan como parte del mismo sprint por el mismo equipo; esto asegura que la calidad sea inherente a cualquier entregable terminado creado como parte de un sprint. Por lo tanto, la mejora continua con sus repetitivas pruebas optimiza la probabilidad de alcanzar los niveles esperados de calidad en un proyecto Scrum. Las discusiones constantes entre el equipo principal de Scrum y lo stakeholders (incluyendo los clientes y usuarios) con incrementos reales del producto que se entregan al final de cada sprint, aseguran que se reduzca constantemente la brecha entre las expectativas de los clientes sobre el proyecto y los entregables producidos.

5.3.1 Calidad y alcance

Los requerimientos de alcance y calidad para un proyecto se determinan al tomarse en cuenta varios factores tales como los siguientes:

- La necesidad del negocio que habrá de cumplir el proyecto
- La capacidad y la disposición de la organización para cumplir con las necesidades del negocio
- Las necesidades futuras y actuales del público meta

El alcance (*scope*) de un proyecto es la suma total de todos los incrementos del producto, así como el trabajo necesario para el desarrollo del producto final. La calidad es la capacidad que tienen los entregables para cumplir con los requisitos de calidad del producto y satisfacer las necesidades del cliente. En Scrum, el alcance y la calidad del proyecto se capturan en Backlog Priorizado del Producto, mientras que el alcance de cada sprint se determina por la refinación de los amplios elementos del Backlog Priorizado del Producto (PBIs, por sus siglas en inglés) en un conjunto de pequeñas, pero detalladas historias de usuario que pueden ser planeadas, desarrolladas y verificadas en un sprint.

El Product Owner refina constante el Backlog Priorizado del Producto. El Product Owner se asegura de que cualquier historia de usuario que se espera que realice el Equipo Scrum en un sprint sea refinada antes del inicio del sprint. En general, los requerimientos más importantes en la resolución de los problemas del cliente o en el cumplimiento de sus necesidades, se consideran de alta prioridad, mientras que al resto se les da menor prioridad. Las historias de usuario de menor importancia se desarrollan en posteriores sprints o se pueden dejar de lado por completo según los requerimientos del cliente. Durante la ejecución del sprint, el Product Owner, el cliente y el Equipo Scrum pueden analizar la lista de características del producto para cumplir con las necesidades cambiantes de los clientes.

5.3.2 Calidad y valor del negocio

La calidad y el valor del negocio están estrechamente vinculados. Por lo tanto, es fundamental entender la calidad y el alcance de un proyecto a fin de trazar correctamente los resultados y beneficios que debe lograr tanto el proyecto como su producto para ofrecer valor empresarial. Para determinar el valor empresarial de un producto, es importante entender la necesidad del negocio que impulsa los requisitos del mismo. Por lo tanto, la necesidad del negocio determina cuál es el producto requerido, y este, a su vez, proporciona el valor empresarial esperado.

La calidad es una variable compleja. Aumentar el alcance sin incrementar el tiempo o los recursos, tiende a reducir la calidad. De igual forma, una reducción de tiempo o de recursos sin disminuir el alcance generalmente resulta en una disminución de la calidad. En Scrum se cree en el trabajo a “ritmo sostenible”, lo cual ayuda a mejorar la calidad durante cierto periodo.

El Scrum Guidance Body puede definir los requisitos mínimos de calidad y estándares que se deben cumplir en todos los proyectos de la organización. Todos los equipos de Scrum en la empresa deben apegarse a dichos estándares.

5.4 Criterios de aceptación y Backlog Priorizado del Producto

El Backlog Priorizado del Producto es un solo documento de requisitos que define el alcance del proyecto, proporcionando una lista de prioridades de las características del producto o servicio a ser entregado por el proyecto. Las características necesarias se describen en forma de historias de usuario. Dichas historias son requisitos específicos señalados por varios stakeholders que se relacionan con el producto o servicio propuesto. Cada historia de usuario contará con sus criterios de aceptación de historia de usuario (también conocidos como "criterios de aceptación"), que son los componentes objetivos mediante los cuales se juzga la funcionalidad de una historia de usuario. Los criterios de aceptación los desarrolla el Product Owner según su experiencia en los requerimientos del cliente. El Product Owner después comunica las historias de usuario que están en el Backlog Priorizado del Producto a los miembros del Equipo Scrum, buscando un común acuerdo. Los criterios de aceptación deben delinear explícitamente las condiciones que deben satisfacer las historias de usuario. Los criterios de aceptación claramente definidos son de suma importancia para la entrega eficaz y oportuna de la funcionalidad definida en las historias de usuario, lo cual, en última instancia, determinan el éxito del proyecto.

Al final de cada sprint, el Product Owner utiliza estos criterios para verificar los entregables completados y puede aceptar o rechazar entregables individuales, así como sus respectivas historias de usuario. Si los entregables son aceptados por el Product Owner, la historia de usuario se considera entontes como terminada. Es importante contar con una clara definición de "terminado", ya que ayuda a poner en claro los requerimientos y permite que el equipo se apegue a las normas de calidad. Ayuda también a que el equipo piense desde la perspectiva del usuario al momento de trabajar con historias de usuario.

Las historias de usuario que corresponden a los entregables rechazados se agregan de nuevo al Backlog Priorizado del Producto durante el proceso de dar mantenimiento a dicha lista para que puedan ser completados en futuros sprints. El rechazo de unos cuantos entregables individuales y sus correspondientes historias de usuario no significa el rechazo del producto final o del incremento del producto. El producto o el incremento del producto pudiera ser potencialmente enviable incluso si se rechazan algunas historias de usuario.

La figura 5-1 ilustra el concepto de los criterios de aceptación y el flujo de incremento del producto.

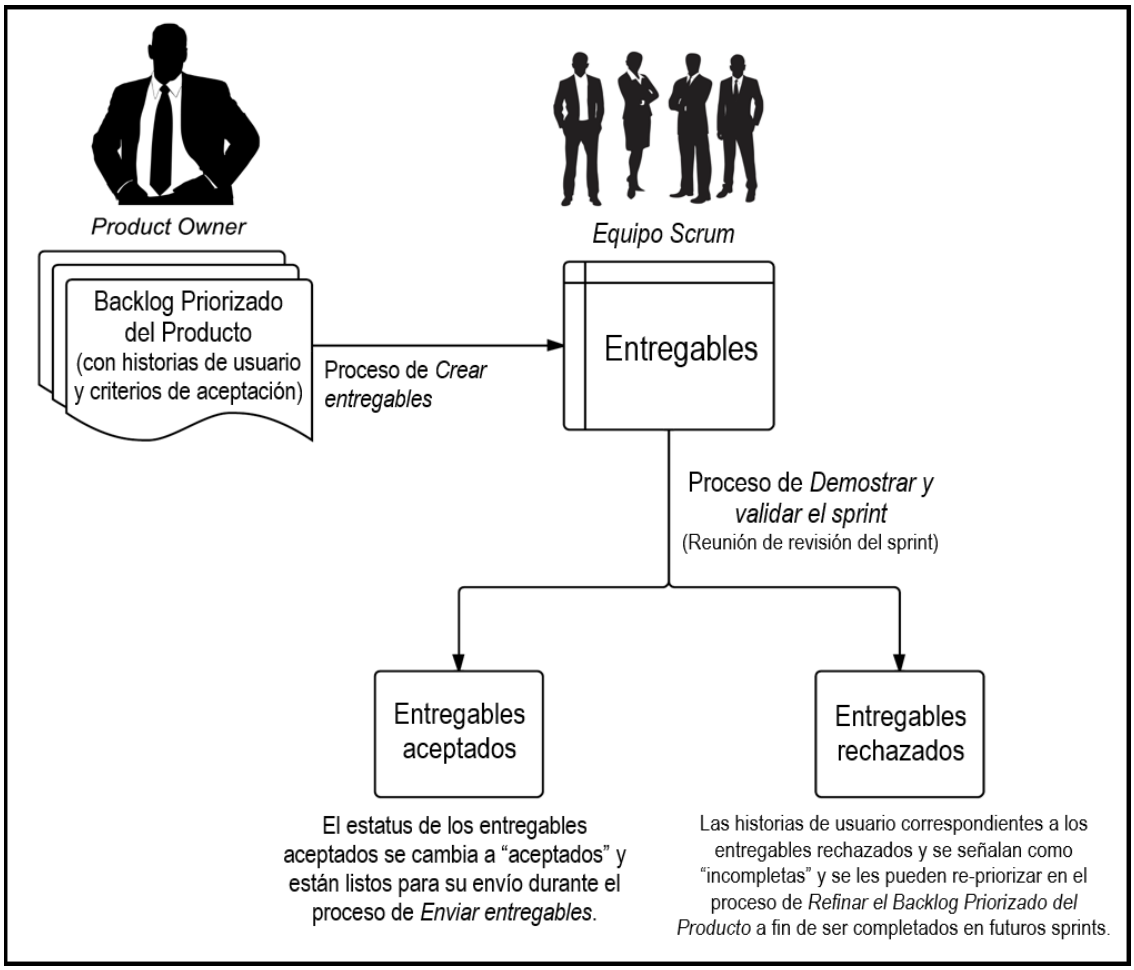


Figura 5-1: Diagrama de flujo del incremento del proyecto

5.4.1 Redacción de criterios de aceptación

Los criterios de aceptación son únicos en cada historia de usuario y no son un substituto de la lista de requerimientos.

Ejemplo:

Prototipo del cliente (*persona*): Janine es una profesionista de 36 años; está casada y tiene tres hijos. Es una mujer ocupada y exitosa que equilibra su vida profesional y personal. Se siente cómoda con la tecnología y le gusta adoptar productos y servicios innovadores. Siempre está conectada al Internet a través de múltiples dispositivos y regularmente hace compras en portales de comercio electrónico.

Historia de usuario: "Como compradora en línea, debo contar con la posibilidad de ahorrar y ver mi orden desde cualquiera de mis dispositivos para poder completar el proceso de orden cuando mejor me convenga".

Criterios de aceptación:

- Todas las órdenes en progreso deben guardarse como pendiente (*draft*) cada 5 segundos en la cuenta del usuario que haya iniciado una sesión.
- Las nuevas órdenes pendientes deben mostraste como notificaciones en cualquier dispositivo de donde el usuario inicie sesión.

Es importante que el Product Owner tome nota que las historias de usuario que cumplan con la mayoría, aunque no con todos los criterios de aceptación, no pueden aceptarse como terminadas. Los proyectos Scrum operan en sprints con un time-box asignado, con Sprint Backlog asignado para cada sprint. Generalmente, la última parte del trabajo pudiera ser la parte más complicada de la historia de usuario y pudiera tardar más de lo esperado. Si se les dio crédito a historias de usuario incompletas como si estuvieran terminadas —y si se llevaron al siguiente sprint—, el progreso de posteriores sprints pudiera verse interrumpido. Por lo tanto, el estado de "terminado" es a blanco y negro. Una historia de usuario puede ser, ya sea "terminada" o "no terminada".

5.4.2 Definición de terminado

Hay una diferencia clave entre los "criterios de terminado" y los "criterios de aceptación". Mientras que los criterios de aceptación son únicos para en las historias de usuario individuales, los criterios de terminado son una serie de reglas aplicables a todas las historias de usuario en un determinado sprint. Los criterios de terminado pueden incluir cualquiera de los siguientes:

- Fueron revisados por otros miembros del equipo;
- Completaron la prueba de unidad de la historia de usuario;
- Conclusión de las pruebas de garantía de calidad;

- Conclusión de toda la documentación relacionada a la historia de usuario;
- Se corrigieron todos los problemas;
- Demostración satisfactoria a los stakeholders y/o representantes empresariales.

Al igual que con los criterios de aceptación, se deben cumplir todas las condiciones de los criterios de terminado para que la historia de usuario se considere terminada.

El Equipo Scrum debe utilizar una *checklist* (o lista de verificación) de los criterios de terminado generales para garantizar que una tarea está terminada y de que el resultado cumpla con la con la definición de terminado (DoD, por sus siglas en inglés). Es importante contar con una clara definición de terminado, ya que ayuda a eliminar la ambigüedad y permite al equipo apegarse a las normas de calidad requeridas. La definición de terminado típicamente la determina y la documenta el Scrum Guidance Body.

Los registros y datos necesarios para cumplir con los requisitos de documentación del proyecto se pueden generar a medida que el equipo procede a través de sprints y lanzamientos.

La inclusión de actividades tales como reuniones de revisión y la redacción de documentos de diseño pueden ayudar a asegurar el cumplimiento de las normas de calidad internas y externas. Los principios básicos de Scrum, tales como iteraciones cortas, la construcción gradual, la participación del cliente, la adaptación a las nuevas necesidades y el constante ajuste de tiempo y costo en el proyecto seguirán siendo aplicables.

5.4.3 Criterios mínimos de terminado

Una unidad empresarial de más alto nivel puede anunciar un criterio de aceptación mínimo y obligatorio, lo cual después forma parte de los criterios de aceptación para cualquier historia de usuario en esta unidad empresarial. Cualquier funcionalidad definida por la unidad empresarial debe satisfacer dichos criterios mínimos de aceptación si es que se busca la aceptación del respectivo Product Owner. La introducción a estos criterios de aceptación puede llevar a una serie en cascada de criterios de aceptación para el portafolio, el programa y el proyecto (véase la figura 5-2). De tal forma, los criterios de aceptación de una historia de usuario en un proyecto habrán de incluir implícitamente todos los criterios mínimos de aceptación de los niveles más elevados, según corresponda.

Chief Product Owner/ Product Owner	• Establece los criterios mínimos de terminado del proyecto, mismos que incluyen los criterios de terminado del programa. • Revisa los entregables del proyecto
Program Product Owner	• Establece los criterios mínimos de terminado para todo el programa, mismos que incluyen los criterios de terminado del portafolio. • Revisa los entregables del programa
Portfolio Product Owner	• Establece los criterios mínimos de terminado de todo el portafolio. • Revisa los entregables del portafolio.

Figura 5-2: Criterios de terminado en cascada

Una vez que los criterios mínimos de terminado están definidos, se pueden incluir en los documentos del Scrum Guidance Body y presentarse a los equipos Scrum cuando según sea necesario.

5.4.4 Aceptación o rechazo de elementos del Backlog Priorizado del Producto

Hacia el final de cualquier iteración, los stakeholders y la unidad empresarial correspondiente participan en una reunión de revisión del sprint en la cual se demuestran los incrementos del producto al Product Owner, al patrocinador, al cliente y al usuario. Aunque se recoge la opinión de todos los usuarios, únicamente el Product Owner tiene la autoridad para aceptar o rechazar como terminada una historia de usuario en particular, según los criterios de aceptación que fueron acordados. Por lo tanto, el rol de los criterios de aceptación en conservar la calidad es de suma importante y debe entenderse claramente por el equipo. Es responsabilidad del Scrum Master asegurase de que los criterios de aceptación de una historia de usuario no sean modificados por el Product Owner a mitad de un sprint. Las historias de usuario parcialmente completadas son rechazadas como “no terminadas” y se devuelven al Backlog Priorizado del Producto.

5.5 Gestión de calidad en Scrum

El cliente es el stakeholder más importante en cualquier proyecto; por lo tanto, es importante entender sus necesidades y requerimientos. La voz del cliente puede describirse como los requerimientos explícitos e implícitos del cliente, los cuales deben entenderes antes del diseño de un producto o servicio. Generalmente, en un entorno de Scrum, el Product Owner se enfoca en los requerimientos y objetivos del negocio, los cuales, juntos, representan la voz del cliente. El Product Owner se puede beneficiar considerablemente de la guía disponible del Scrum Guidance Body (ya sea a través de documentos o estándares de calidad o de expertos en calidad). Estos especialistas deben trabajar con el Product Owner y con el cliente a fin de garantizar el nivel apropiado de detalle e información en las historias de usuario, ya que estas son la base del éxito de cualquier proyecto Scrum.

Cabe señalar que los stakeholders externos no participan directamente al nivel del Equipo Scrum y, en cambio, interactúan principalmente con el Product Owner. Para cualquier proyecto Scrum, el cliente puede ser cualquiera de los siguientes:

- Interno (dentro de la misma organización)
- Externo (fuera de la organización)

En Scrum, el control de calidad les permite a los clientes conocer cualquier problema en el proyecto en forma anticipada, ayudándoles a reconocer si el proyecto habrá o no de funcionarles. En Scrum, la calidad gira en torno a la satisfacción del cliente y de un producto funcional y no necesariamente en cumplir con parámetros arbitrarios. Dicha distinción resulta muy importante desde el punto de vista del cliente, ya que son quienes invierten tiempo y dinero en el proyecto.

En Scrum, la gestión de calidad se facilita mediante tres actividades interrelacionadas:

1. Planificación de calidad
2. Control de calidad
3. Garantía de calidad

5.5.1 Planificación de calidad

Uno de los principios rectores de Scrum es desarrollar primero la funcionalidad de más alta prioridad para el cliente. Las características de menor importancia se desarrollan en posteriores sprints o pueden dejarse de lado completamente según los requerimientos del cliente. Este enfoque le brinda al Equipo Scrum el tiempo para centrarse en la calidad de la funcionalidad esencial. Un beneficio clave de la planificación de calidad es la reducción de la deuda técnica. La deuda técnica, conocida también como deuda de diseño o deuda de código, es el trabajo al que los equipos dan menor prioridad; el trabajo que omiten o que no terminan a medida que trabajan en la creación de los principales entregables asociados al producto del proyecto. La deuda técnica se acumula y se debe saldar a futuro.

Algunas de las causas de la deuda técnica pueden ser:

- Rápida solución y elaboración de entregables que no cumplan con los estándares de calidad, seguridad, metas de arquitectura a largo plazo, etc.
- Evaluación inadecuada o incompleta;
- Documentación inadecuada o incompleta;
- Falta de coordinación entre los distintos miembros del equipo, o si hay diferentes equipos de Scrum que empiezan a trabajar en forma aislada con menor enfoque en la integración final de los componentes requeridos para realizar un proyecto o programa exitoso.
- Intercambio deficiente del conocimiento empresarial y de procesos entre los stakeholders y equipos del proyecto.
- Demasiado énfasis en los objetivos del proyecto a corto plazo en vez de objetivos a largo plazo en la empresa. Esta supervisión puede resultar en una baja calidad de los entregables funcionales que pudiera incurrir en considerables costos de mantenimiento y actualización.

En los proyectos Scrum, cualquier deuda técnica no debe llevarse más allá de un sprint, ya que debe de haber criterios de aceptación y de terminado debidamente definidos. La funcionalidad debe satisfacer dichos criterios para considerarlos terminados. Conforme se refina el Backlog Priorizado del Producto y se da prioridad a las historias de usuario, el equipo elabora regularmente entregables funcionales, previniendo con ello la acumulación de una deuda técnica considerable. El Scrum Guidance Body pudiera también incluir documentación y definición de los procesos que ayudan a disminuir dicha deuda.

Para conservar una cantidad mínima de deuda técnica, es importante definir el producto requerido en un sprint y del proyecto, así como los criterios de terminado, cualquier método de desarrollo a seguir y las responsabilidades clave de los miembros del Equipo Scrum respecto a la calidad. Definir los criterios de

calidad es una parte importante de la planificación de calidad y permite que se lleve a cabo un control eficaz de la misma durante el proyecto.

La deuda técnica es un gran reto con algunas técnicas de gestión tradicional de proyectos donde el desarrollo, la evaluación, la documentación, etc., se realizan en forma subsecuente y por lo general por distintas personas, donde ninguna es responsable de cualquier entregable funcional en particular. Como resultado, la deuda técnica se acumula, llevando a un mantenimiento considerablemente más elevado, integración y costos de lanzamiento del producto en las etapas finales su lanzamiento. Asimismo, el costo de los cambios es muy elevado en tales circunstancias, ya que los problemas surgen en etapas posteriores del proyecto. El framework de Scrum evita los problemas relacionadas a la deuda técnica, garantizando que se definan los entregables terminados con criterios de aceptación como parte del Sprint Backlog y que las tareas clave, incluyendo el desarrollo, la evaluación y la documentación se lleven a cabo como parte del mismo sprint y por el mismo Equipo Scrum.

5.5.1.1 Integración continua y ritmo sostenible

Mantener un ritmo sostenible es uno de los principios más importantes de Scrum. El ritmo sostenible se traduce en una mayor satisfacción del empleado, en estabilidad y una mayor precisión en la estimación; todo ello conlleva, en última instancia, a un aumento en la satisfacción del cliente. Para desarrollar un producto verdaderamente de alta calidad y conservar un sano ambiente laboral, es importante realizar periódicamente actividades de integración, en vez de retrasar el trabajo de integración hasta el final en tales circunstancias. Para brindar valor en intervalos frecuentes, el equipo debe continuamente desarrollar, evaluar e integrar las funcionalidades de cada elemento en el Backlog Priorizado del Producto en cada sprint con el uso de técnicas, tales como la integración continua y la evaluación automática del producto. También es importante, desde la perspectiva del equipo, garantizar que el esfuerzo realizado en el actual sprint sea similar al esfuerzo realizado en el sprint anterior a fin de sostener un ritmo constante durante los sprints en el proyecto. Esto ayuda al equipo a evitar fases de intensos periodos de trabajo, garantizando que puedan siempre presentar el esfuerzo requerido para logar el trabajo que debe realizarse.

5.5.2 Control de calidad y garantía de calidad

El control de calidad es la ejecución de las actividades de calidad planeadas por el Equipo Scrum en el proceso de creación de entregables que la potencialidad de enviarse. Incluye también el aprendizaje de cada serie de actividades realizadas a fin de lograr una mejora continua. Dentro del equipo inter-funcional, es importante contar con las habilidades necesarias para llevar a cabo actividades de control de calidad. Durante la reunión de retrospectiva del sprint, los miembros del equipo analizan las lecciones aprendidas. Dichas lecciones sirven de entradas en la mejora continua y contribuyen al mejoramiento del constante control de calidad.

Se requiere de la calidad no solo en los productos, sino también en los procesos. La garantía de calidad es el proceso de evaluación y los estándares que rigen la gestión de calidad en un proyecto a fin de garantizar que continúen siendo relevantes. Las actividades relacionadas a la garantía de calidad se llevan a cabo como parte del trabajo. De hecho, la garantía de calidad es un factor considerable en la definición de terminado. El entregable no se considera completo si no se ha realizado una garantía adecuada de calidad. Generalmente, la garantía de calidad se demuestra durante la reunión de revisión del sprint.

El Product Owner de los proyectos respectivos, programas y portafolios, puede monitorear y evaluar las actividades de garantía de calidad para asegurarse de que cada equipo sigua de acuerdo y cumpla con los estándares de calidad que se han establecido. La garantía de calidad de un extremo al otro puede abordarse durante la evaluación final del producto, de un lanzamiento de un sprint. Se puede realizar una comparación de la cantidad de problemas que se encontraron en relación a la cantidad de historias de usuario completadas. Los componentes del producto que tienen defectos se pueden incorporar como elementos del Backlog Priorizado del Producto, mismos que se pueden abordar ya sea por el equipo o por una persona durante ciertos periodos durante el sprint, dependiendo del número de defectos.

En ocasiones, el Scrum Guidance Body puede definir los procesos y documentos que pueden remitirse a los equipos de Scrum al momento de hacer sus proyectos a fin de asegurase de que se les dé seguimiento a las normas uniformes de calidad en todos los proyectos al interior de la empresa.

5.5.3 Ciclo de planificar, hacer, verificar y actuar (Ciclo PDCA, por sus siglas en inglés)

El ciclo de planificar, hacer, verificar y actuar (PDCA, por sus siglas en inglés), conocido también como Ciclo Deming o Shewhart, fue desarrollado por el doctor W. Edwards Deming, considerado como el padre del control de calidad moderno, y por el doctor Walter A. Shewhart. Estas son algunos de los puntos más importantes de la filosofía de Deming:

- Las directrices de la gerencia son las que definen la calidad. Cuando la gerencia puede ofrecer un ambiente conductivo y motivar a sus empleados para que mejore la calidad en forma continua, cada empleado podrá contribuir hacia una calidad superior en el producto. La “Teoría del conocimiento profundo” de Deming aboga por lo que debe hacer la gerencia a fin de crear un ambiente en el cual cada empleado puede realizar contribuciones considerables para mejorar la calidad.

Deming modificó después el nombre de “planificar, hacer, verificar y actuar” a “planificar, hacer, estudiar y actuar” (PDSA, por sus siglas en inglés), ya que consideraba que el término “estudiar” hacía énfasis en el análisis en vez de simplemente la inspección, como lo implica el término “verificar”.

Tanto Scrum como el Ciclo Deming/Shewhart/PDCA son métodos iterativos enfocados en la mejora continua. La figura 5-3 ilustra las etapas del ciclo PDCA y su correlación con los diversos procesos de Scrum.

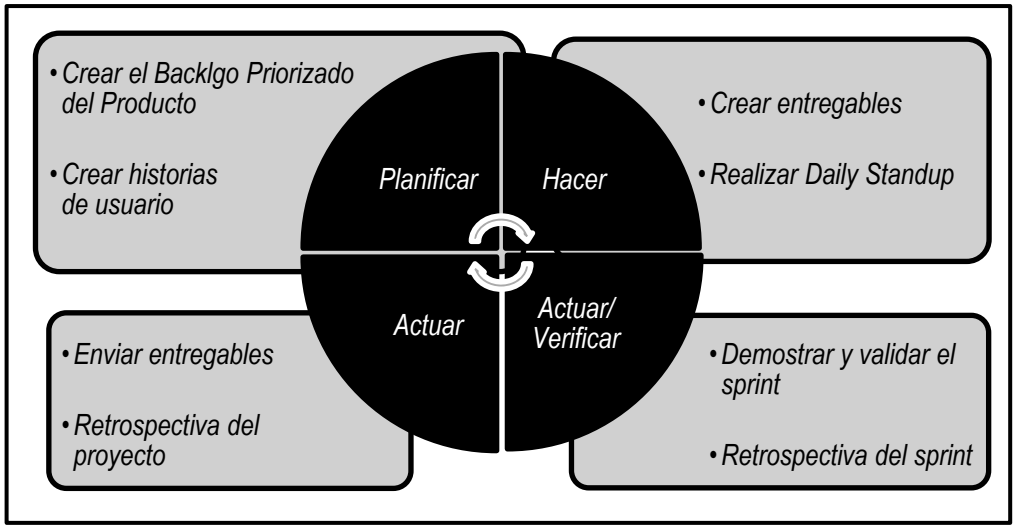


Figura 5-2: Ciclo PDCA en Scrum

5.6 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	• Desarrolla y mantiene todos los entregables durante los sprints hasta que pasen a manos de los usuarios finales. • Pone en práctica y fomenta la buena comunicación a fin de que los requerimientos queden clarificados y completamente entendidos. • Comparte conocimiento para garantizar que los miembros del equipo se familiaricen con la serie de características y con ello se beneficien de la experiencia de otros. • Hace cambios adecuados con rapidez a los entregables.
Product Owner/ Chief Product Owner	• Presenta los requerimientos del negocio para el producto y define con claridad los requerimientos en el Backlog Priorizado del Producto. • Evalúa la viabilidad y se asegura de que los entregables cumplan con los requisitos de calidad. • Establece los criterios mínimos de terminado para todo el proyecto, incluyendo los criterios de aceptación del programa respectivo. • Facilita la creación de los criterios de aceptación para las historias de usuario. • Revisa y valida los entregables durante el proceso de *Demostrar y validar el sprint*.

Scrum Master/ Chief Scrum Master	• Facilita la mentalidad de "primero el equipo" cuando se traja de calidad. • Elimina obstáculos ambientales que pudieran afectar la calidad de los entregables y procesos. • Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea calidad de las características en vez de estrictamente en la velocidad. • Se asegura de que todos los integrantes del equipo, incluyendo el Product Owner, sigan correctamente los procesos de Scrum.
Program Product Owner	• Establece criterios mínimos de aceptación para todo el programa. • Revisa los entregables del programa.
Program Scrum Master	• Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.
Portfolio Product Owner	• Establece criterios mínimos de aceptación para todo el portafolio. • Revisa los entregables del portafolio.
Portfolio Scrum Master	• Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.
Stakeholder(s)	• Revisa y acepta los entregables y el producto final.
Scrum Guidance Body	• Proporciona la definición de terminado. • Proporciona un framework y una guía para desarrollar los criterios de aceptación. • Define la gama de herramientas que puede utilizar el Equipo Scrum para desarrollar y verificar el producto.

Tabla 5-1: Resumen de las responsabilidades pertinentes a la calidad

5.7 Scrum vs. Gestión tradicional de proyectos

Aunque hay similitudes entre Scrum y los métodos tradicionales de gestión de proyectos en relación a la definición de "calidad" (la capacidad con la que cuenta el producto para cumplir con los criterios de aceptación acordados y lograr el valor empresarial que espera el cliente), existen diferencias en términos de cómo los métodos abordan la implementación y logro de los niveles exigidos de calidad.

En los métodos tradicionales de gestión de proyectos, los usuarios aclaran sus expectativas; el Project Manager define dichas expectativas en términos cuantificarles y obtiene el acuerdo de los usuarios. Después de una planificación detallada, el equipo del proyecto desarrolla el producto durante un período acordado. Si hay necesidad de cambiar alguno de los criterios acordados, los cambios se pueden realizar sólo a través de un sistema formal de gestión de cambio, en el cual se estima el impacto de los cambios y el Project Manager consigue la aprobación de todos los stakeholders relevantes.

En Scrum, sin embargo, el Product Owner colabora con el Equipo Scrum y define los criterios de aceptación para las historias de usuario en relación con el producto que se debe entregar. El Equipo Scrum, después

desarrolla el producto a partir de una serie de iteraciones cortas denominadas sprints. El Product Owner puede realizar cambios en los requisitos para mantenerse al ritmo de las necesidades del usuario y estos cambios pueden ser abordados por el Equipo Scrum, ya sea al concluir el actual sprint, o al incluir los requisitos ajustados en el próximo sprint, ya que cada uno es de muy corta duración (de una a seis semanas).

Una de las principales ventajas de Scrum es el énfasis en la creación de entregables potencialmente listos para ser enviados al final de cada ciclo de sprint, en vez de esperar al final de todo el proyecto. Así, el Product Owner y los clientes constantemente inspeccionan, aprueban y aceptan entregables después de cada sprint. Incluso, si un proyecto Scrum se cancela antes de tiempo, siempre existe algo de valor que fue creado antes de la terminación como resultado de los entregables creados en sprints individuales.

6. CAMBIO

6.1 Introducción

Cada proyecto, independientemente de su método o framework, está expuesto al cambio. Es importante que los miembros del equipo del proyecto entiendan que los procesos de desarrollo de Scrum están diseñados para aceptar el cambio. Las organizaciones deben tratar de maximizar los beneficios que se derivan de los cambios y disminuir los impactos negativos a través de los procesos de gestión de cambio diligente según los principios de Scrum.

El *cambio,* tal como se define en *Una guía para el conocimiento de Scrum (Guía SBOK™),* es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la Guía SBOK™ puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

6.2 Guía de los roles—En esta sección se proporciona orientación sobre qué secciones son relevantes para cada uno de los roles principales de Scrum: Product Owner, Scrum Master y Equipo Scrum.

6.3 Descripción—Esta sección define el concepto de cambio, específicamente dentro del contexto de los procesos de Scrum. También ilustra la forma en la que las solicitudes de cambios se manejan en los procesos de Scrum.

6.4 Cambio en Scrum—En esta sección se detalla la importancia de gestionar con eficacia el cambio en un proyecto Scrum. También se habla de cómo la flexibilidad y la estabilidad se pueden lograr a través del manejo adecuado de las solicitudes de cambio que surgen a lo largo de un proyecto.

6.5 Integración del cambio—Esta sección detalla cómo las solicitudes de cambio se evalúan y aprueban (o se rechazan) en la aplicación del framework de Scrum.

6.6 Cambio a programas y portafolios—En esta sección se describe el impacto de los cambios en los programas y portafolios.

6.7 Resumen de las responsabilidades—Esta sección define las responsabilidades de los miembros de un equipo de proyecto al gestionar cambios.

6.8 Scrum vs. Gestión tradicional de proyecto—Esta sección analiza los beneficios de gestionar cambios utilizando Scrum en comparación a los modelos tradicionales de gestión de proyectos.

6.2 Guía de los roles

4. Product Owner—La responsabilidad de iniciar el cambio en un proyecto recae principalmente en el Product Owner, por lo tanto, todo este capítulo es aplicable a este rol.

5. Scrum Master—El Scrum Master también debe familiarizarse con todo este capítulo, con un enfoque principal en las secciones 6.3, 6.4, 6.5 y 6.7.

6. Equipo Scrum—El Equipo Scrum debe centrarse principalmente en las secciones 6.3, 6.4.2, 6.5 y 6.7.

6.3 Descripción

El cambio es inevitable en todos los proyectos. En el mundo excesivamente competitivo de hoy en día, donde la tecnología, las condiciones del mercado y los patrones de negocio están cambiando de forma continua, el cambio es la única constante.

Un principio fundamental de Scrum es su reconocimiento de que **a)** los stakeholders (clientes, usuarios y patrocinadores) cambian de opinión acerca de lo que quieren y lo que necesitan durante un proyecto (a esto se le conoce en ocasiones como: "requisitos volátiles" o *requirements churn*); y, **b)** que es muy difícil, si no es que imposible, que los stakeholders definan todos los requisitos al inicio del proyecto.

Los proyectos Scrum aceptan los cambios mediante el uso de sprints breves e iterativos que incorporan la retroalimentación del cliente sobre los entregables del proyecto después de cada sprint. Esto permite que el cliente interactúe regularmente con los miembros del Equipo Scrum, que vea los entregables a medida que estén listos y que cambie los requisitos tempranamente en el ciclo de desarrollo. Asimismo, los equipos de administración del portafolio o el programa pueden responder a las solicitudes de cambio relacionadas a proyectos de Scrum que apliquen en su nivel.

Scrum personifica un principio primordial del Manifiesto Ágil (Fowler y Highsmith, 2001): "Responder al cambio, en vez de seguir un plan". La práctica de Scrum se basa en la aceptación del cambio y de convertirlo en una ventaja competitiva. Por lo tanto, es más importante ser flexible que seguir un plan estricto y predefinido. Esto significa que es esencial abordar la gestión de proyectos de una manera adaptativa que permita el cambio durante el desarrollo del producto o de los ciclos rápidos de servicio.

Ser adaptable al cambio es una ventaja clave del framework de Scrum. Aunque Scrum funciona bien en todo tipo de proyectos e industrias, puede ser muy eficaz cuando el producto o los requisitos del proyecto no se identificaron o no están bien definidos desde el principio. Scrum también es eficaz cuando el mercado del producto es volátil y/o cuando la atención se centra en hacer que el equipo sea lo suficientemente flexible para incorporar los requisitos cambiantes. Scrum es especialmente útil para proyectos complejos con mucha incertidumbre. La planificación y previsión a largo plazo suele ser ineficaz para este tipo de proyectos e implican grandes cantidades de riesgo. Scrum lleva al equipo a obtener resultados valiosos de negocio utilizando la *transparencia*, la *inspección* y la *adaptación*.

6.3.1 Solicitudes de cambio aprobadas y no aprobadas

Las solicitudes para hacer cambios se presentan por lo general como solicitudes de cambio o *Change Requests*. Las solicitudes de cambio permanecen como no aprobadas hasta que se obtiene una aprobación formal. El Scrum Guidance Body por lo general define el proceso de decisión y gestión de cambios en la organización. En ausencia de un proceso formal, se recomienda que los pequeños cambios que no tienen un impacto significativo en el proyecto se aprueben directamente por el Product Owner. La tolerancia de estos pequeños cambios podría definirse a nivel organizacional o por el patrocinador de un proyecto en particular. En la mayoría de los proyectos, el 90 % de las solicitudes de cambio podrían clasificarse como pequeños cambios que deben aprobarse por el Product Owner. Por lo tanto, el Product Owner juega un papel muy importante en la gestión de los cambios en un proyecto Scrum.

Los cambios que están más allá del nivel de tolerancia del Product Owner pueden necesitar de la aprobación de los stakeholders que trabajan con él.

En ocasiones, si un cambio que se ha solicitado puede tener un impacto sustancial en el proyecto u organización, la autorización de la alta gerencia (por ejemplo, el patrocinador ejecutivo, el Portfolio Product Owner, el Program Product Owner o el Chief Product Owner) puede ser necesaria.

Las solicitudes de cambio para el proyecto se discuten y aprueban durante los siguientes procesos: *Desarrollar épica(s)*, *Crear el Backlog Priorizado del Producto* y *Refinar el Backlog Priorizado del Producto*. Posteriormente, las solicitudes de cambio aprobadas se priorizan junto con otros requisitos del producto y sus respectivas historias de usuario y se incorporan después en el Backlog Priorizado del Producto.

La figura 6-1 resume el proceso de aprobación de cambio y la figura 6-2 explica cómo se actualiza el Backlog Priorizado del Producto con los cambios aprobados.

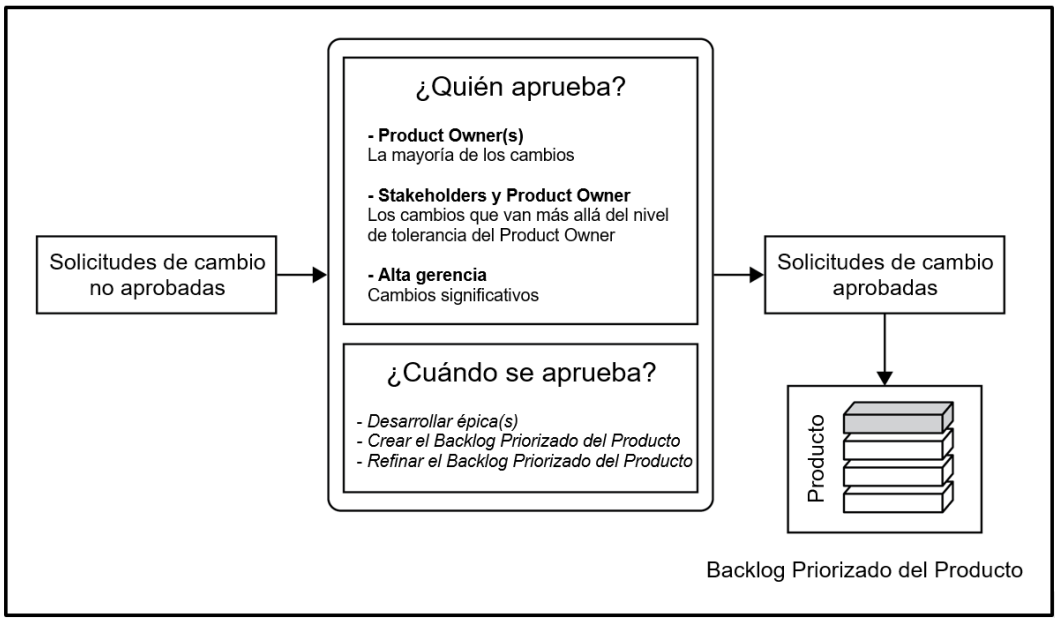


Figura 6-1: Ejemplo del proceso de aprobación de cambios

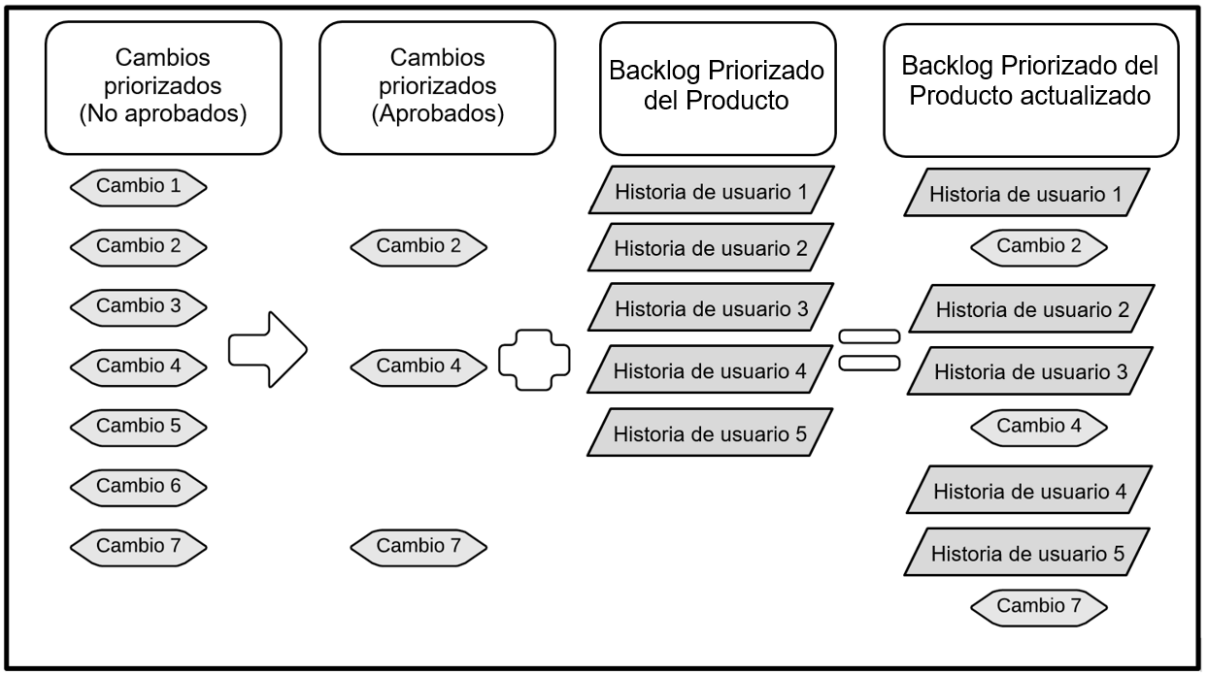


Figura 6-2: Actualización del Backlog Priorizado del Producto con los cambios aprobados

6.4 Cambio en Scrum

6.4.1 Equilibrio entre flexibilidad y estabilidad

Scrum ayuda a las organizaciones a ser más flexibles y receptivas al cambio. Sin embargo, es importante entender que, aunque el framework de Scrum hace énfasis en la flexibilidad, también es importante mantener la estabilidad durante todo el proceso de cambio. De la misma manera que la rigidez extrema es ineficaz, la flexibilidad extrema también es improductiva. La clave es encontrar el equilibrio adecuado entre la flexibilidad y la estabilidad, ya que la estabilidad se necesita a fin de realizar el trabajo. Por lo tanto, Scrum utiliza el desarrollo iterativo y sus demás características y principios para lograr dicho equilibrio. Scrum mantiene la flexibilidad de que las solicitudes de cambio se pueden crear y aprobar en cualquier momento durante el proyecto; sin embargo, estas se priorizan cuando se crea o se actualiza el Backlog Priorizado del Producto. Al mismo tiempo, Scrum asegura que la estabilidad permanezca al refinar el Sprint Backlog y al no permitir la interferencia con el Equipo Scrum durante un sprint.

En Scrum, todos los requisitos relacionados con el sprint en curso se suspenden durante el sprint. Ningún cambio se introduce hasta que se termina el sprint, a menos que un cambio se considere lo suficientemente importante como para detener el sprint. En el caso de un cambio urgente, el sprint se termina y el equipo se reúne para planificar uno nuevo. Es así como Scrum acepta los cambios sin crear problemas de cambio en las fechas de lanzamiento o inestabilidad.

6.4.2 Lograr la flexibilidad

Scrum facilita la flexibilidad a través de la *transparencia*, la *inspección* y la *adaptación* para lograr los resultados de negocio más valiosos. Scrum proporciona un mecanismo de adaptación para la gestión de proyectos en el que un cambio en los requisitos se puede acomodar sin afectar significativamente el progreso general del proyecto. Es necesario adaptarse a las realidades de los negocios emergentes como parte del ciclo de desarrollo. La flexibilidad en Scrum se logra a través de cinco características claves (véase la figura 6-3): el desarrollo de productos iterativos, la asignación de un time-box, los equipos interfuncionales, la priorización basada en el cliente y la integración continua.

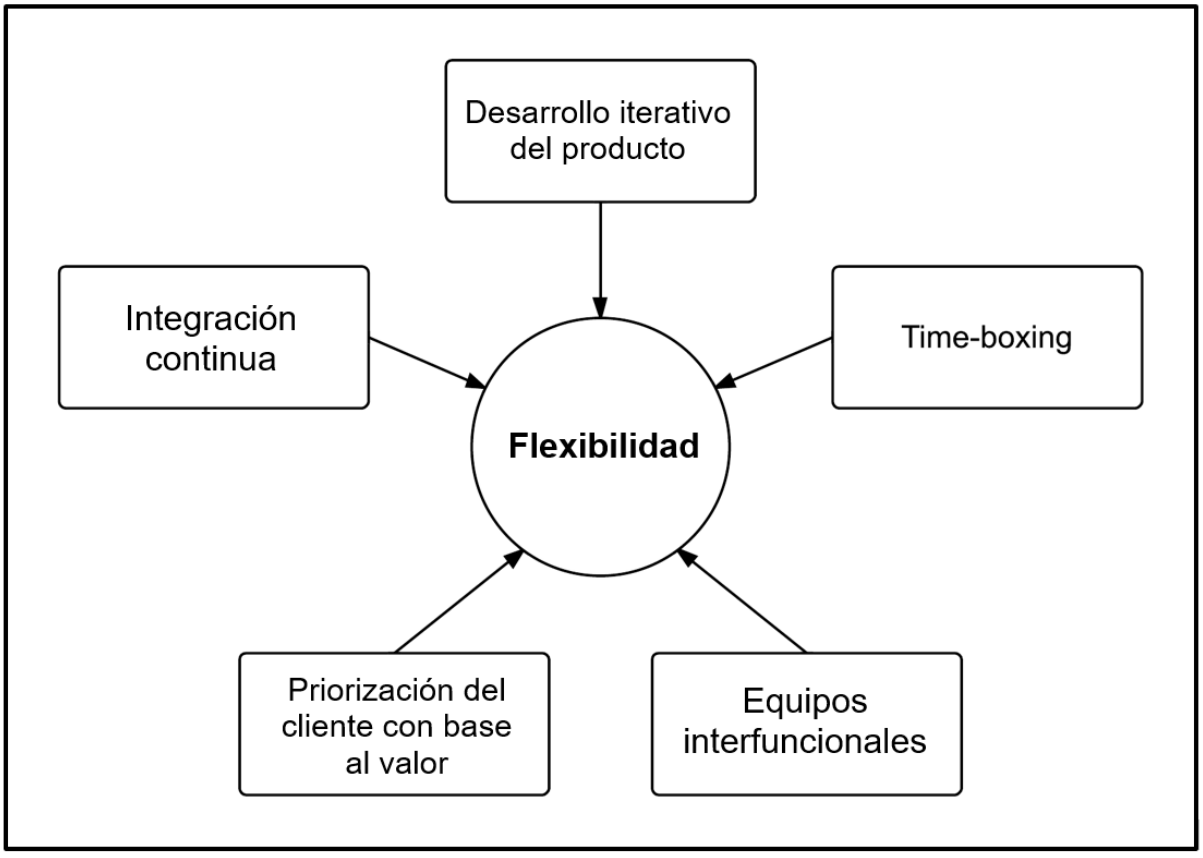


Figura 6-3: Características de Scrum para lograr flexibilidad

6.4.2.1 La flexibilidad mediante el desarrollo iterativo del producto

Scrum sigue un enfoque iterativo e incremental de desarrollo de productos y servicios, por lo que es posible la incorporación de cambios en cualquier paso en el proceso de desarrollo. A medida que se desarrolla el producto, una solicitud de cambio para el proyecto puede provenir de múltiples fuentes de la siguiente manera:

1. **Stakeholders**

   Los stakeholders del proyecto —en particular los patrocinadores, clientes y los usuarios— pueden presentar solicitudes de cambio en cualquier momento durante todo el proyecto. Las solicitudes podrían deberse a cambios en las condiciones del mercado, la dirección de la organización, asuntos legales o reglamentarios, o a varias otras razones. Por otra parte, los stakeholders pueden presentar dichas solicitudes a medida que van revisando los entregables durante los procesos de *Demostrar y validar el sprint*, *Retrospectiva del sprint* o *Retrospectiva del proyecto*. Todas las solicitudes de cambio se añaden Backlog Priorizado del Producto del proyecto (también denominado Backlog Priorizado del Producto o Product Backlog), una vez que se aprueba. La figura 6-4 muestra algunas de las razones por las que los stakeholders inician el proceso de solicitud de cambio.

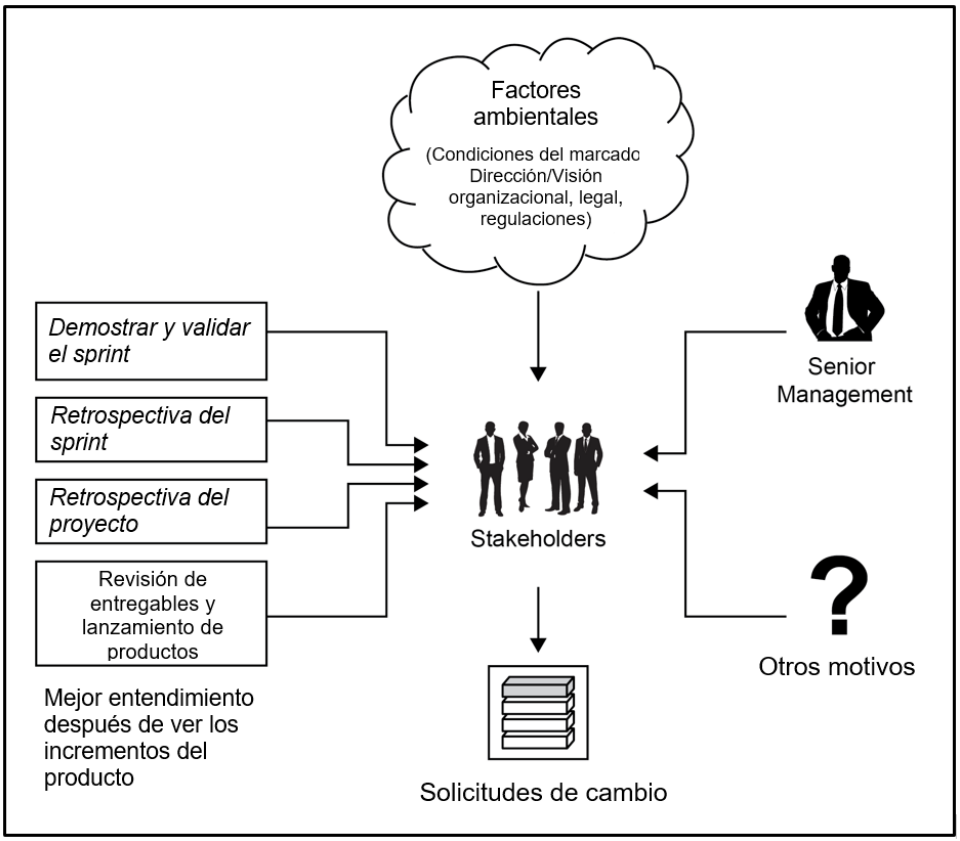


Figura 6-4: Motivación de los stakeholders para la solicitud de cambios

2. El equipo principal de Scrum (*Scrum Core Team*)

El equipo principal de Scrum (Product Owner, Scrum Master y Equipo Scrum) participa en la creación de los entregables del producto. La interacción continua entre los miembros principales de un Equipo Scrum y otros (como otros equipos Scrum del proyecto, y los stakeholders internos y externos del proyecto) puede motivarlos a sugerir cambios o mejoras en el producto, servicio, o cualquier otro aspecto del proyecto. Normalmente, estos cambios —al igual que otros— se incluyen en las solicitudes de cambio, y el Product Owner toma una decisión final acerca de qué sugerencias de cambio por parte del Equipo Scrum o Scrum Master se deben considerar como solicitudes de cambio formales.

En ocasiones pueden existir retos con la creación de ciertos entregables, lo que puede resultar en solicitudes de cambio. Por ejemplo, el equipo puede añadir o modificar una característica para mejorar el rendimiento del producto. En la mayoría de los proyectos Scrum, las recomendaciones de cambios por parte del equipo principal de Scrum suceden durante el proceso de *Crear entregables*, o cuando participa en las Daily Standups o las reuniones de retrospectiva del sprint. La figura 6-5 muestra algunas de las razones por las que el equipo principal de Scrum puede iniciar solicitudes de cambio.

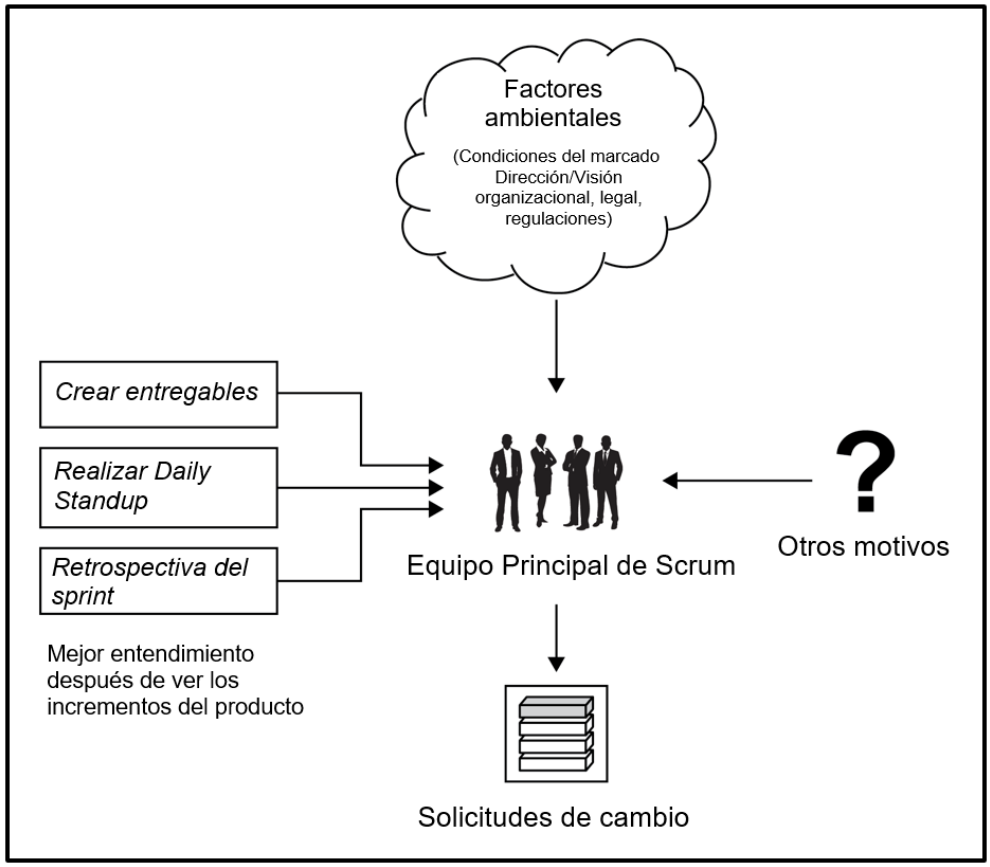


Figura 6-5: Motivos del equipo principal de Scrum para la solicitud de cambios

3. **La alta gerencia (*Senior Management*)**

   La alta gerencia —incluyendo el Portfolio Management y Program Management— puede recomendar cambios que afecten el proyecto. Esto puede deberse a cambios estratégicos en la dirección de la empresa, a un entorno competitivo, a problemas financieros, etcétera. Es importante tener en cuenta que estos cambios se añaden al Backlog Priorizado del Producto y tienen que pasar por el proceso normal de gestión del cambio. Si algunos de estos cambios son urgentes, es posible que cualquier sprint afectado necesite terminarse (véase la sección 6.6 para más detalles).

4. **Scrum Guidance Body**

   El Scrum Guidance Body puede presentar solicitudes de cambio que afecten a todos los proyectos debido a alguno de los siguientes ejemplos:

   - Cambios en las regulaciones gubernamentales (por ejemplo, la privacidad, las normas de seguridad, o nuevas leyes)
   - Directivas corporativas de calidad, seguridad u otras iniciativas de la organización que deban ponerse en práctica en toda la compañía
   - Puntos de referencia o mejores prácticas para alcanzar cierto nivel
   - Lecciones aprendidas de proyectos anteriores que se puedan implementar por otros equipos Scrum

El sello distintivo de Scrum es su tolerancia y adaptación al cambio. Scrum no promueve el determinar y establecer planes con mucha firmeza y anticipación, ya que opera en la premisa de que el desarrollo del proyecto es muy propenso al cambio y al riesgo. El resultado es un alto grado de flexibilidad y tolerancia al cambio. El proyecto se lleva a cabo y se gestiona de forma incremental, por lo que generalmente es fácil incorporar cambios a lo largo del proyecto.

6.4.2.2 La flexibilidad mediante el Time-boxing

El Time-boxing es la fijación de breves periodos de periodos para hacer el trabajo. Si el trabajo realizado queda incompleto al final del time-box, este se asigna a un nuevo bloque. Algunos ejemplos de Time-boxing incluyen el limitar los Daily Standups a 15 minutos y el establecer la duración del sprint de una a seis semanas. El asignar time-boxes proporciona la estructura necesaria para los proyectos Scrum, los cuales tienen un elemento de incertidumbre, son dinámicos por naturaleza y propensos a cambios frecuentes. Estos bloques de tiempo ayudan a medir el progreso del proyecto y permiten que el equipo identifique fácilmente cuándo se necesitará modificar un proceso o método.

Los sprints que se asignan a un time-box contribuyen en gran medida al cumplimiento de los plazos, así como al logro de altos niveles de productividad. Los sprints promueven el orden y la consistencia en un ambiente de trabajo volátil y proporcionan una plataforma para medir resultados, así como para obtener información en un corto espacio de tiempo. Los sprints también permiten una evaluación frecuente de los

progresos y los métodos que se utilizan para gestionar el proyecto, incluyendo la gestión del cambio eficaz. Los errores o problemas pueden identificarse en una etapa temprana y rectificarse rápidamente.

Mediante el Time-boxing en los sprints, el equipo repasa con frecuencia el proceso de estimación del trabajo que se va a realizar, por lo que la proyección de tiempo y esfuerzo que se requieren es más precisa con cada sprint subsecuente a medida que el proyecto avanza. Estos ciclos iterativos también motivan a los miembros del equipo a lograr los objetivos previstos y las metas incrementales, a fin de alcanzar el objetivo mayor.

6.4.2.3 La flexibilidad a través de equipos interfuncionales y auto-organizados

Las estructuras interfuncionales y auto-organizadas del Equipo Scrum les permiten a los miembros del equipo enfocarse en los resultados deseados del sprint. El equipo tiene un conjunto definido de objetivos durante cada sprint, así como la flexibilidad para dar cuenta de un cambio en los objetivos antes de comenzar el siguiente sprint.

El uso de equipos interfuncionales garantiza también de que todas las habilidades y conocimientos necesarios para llevar a cabo el trabajo del proyecto existan dentro del equipo. Esto proporciona un modelo de trabajo eficiente que da lugar a la creación de entregables listos para mostrarlos al Product Owner y/u otros stakeholders.

La auto-organización garantiza que los miembros del Equipo Scrum decidan por sí mismos la forma de hacer el trabajo del proyecto sin la microgestión de las tareas por un alto directivo.

Tener equipos interfuncionales y auto-organizados permite al grupo adaptarse y administrar los trabajos en curso y algunos problemas menores o cambios sin tener que obtener el apoyo o la experiencia de miembros fuera del equipo, y en el proceso, crear entregables que estén listos para enviarse en caso de ser necesario.

6.4.2.4 Flexibilidad a través de la priorización basada en el valor para el cliente

La priorización de las necesidades y del trabajo en un proyecto Scrum siempre se determina con base al valor proporcionado al cliente. Primero, al inicio de un proyecto, los requisitos iniciales se priorizan en función del valor que cada requisito proveerá; esto está documentado en Backlog Priorizado del Producto. Cuando se realiza una solicitud para un requisito nuevo o un cambio a uno ya existente, esto se evalúa durante el proceso de *Refinar el Backlog Priorizado del Producto*. Si se considera que el cambio proporcionará más valor que otros requisitos existentes, se añadirá y priorizará de acuerdo a la versión actualizada del Backlog Priorizado del Producto. Así, dicho backlog ofrece la posibilidad de incorporar cambios y agregar nuevos requisitos cuando sea necesario.

Es importante tener en cuenta que los nuevos requisitos y cambios añadidos al Backlog Priorizado del Producto puede disminuir la prioridad de otras historias de usuarios existentes en el backlog: por tanto, estas historias de usuario de menos prioridad se pueden implementar más adelante dependiendo de su nueva priorización. Debido a que los clientes están íntimamente involucrados en la priorización de las necesidades y de sus historias de usuario correspondientes en el Backlog Priorizado del Producto, esta práctica asegura que los requisitos que los clientes consideran de "alto valor" se completen lo antes posible, y que el proyecto inicie la entrega de valor con mayor anticipación.

6.4.2.5 Flexibilidad a través de la integración continua

Utilizando técnicas de integración continua, los miembros del Equipo Scrum pueden incorporar características nuevas y modificadas en las entregas siempre que sea posible. Esto reduce el riesgo de que varios miembros del equipo hagan cambios en componentes redundantes (por ejemplo, código obsoleto en los productos de software, o diseños antiguos para la fabricación de piezas). Esto asegura que se esté trabajando solamente en la última función de la versión y en evitar problemas de compatibilidad.

6.5 Integración del cambio

Dependiendo de la industria y del tipo de proyecto, la prioridad de las características y los requisitos en un proyecto pueden permanecer fijos durante periodos considerables de tiempo, o bien, pueden cambiar con frecuencia. Si los requisitos del proyecto son generalmente estables, normalmente hay pequeños cambios realizados en el Backlog Priorizado del Producto en todo el desarrollo, y los equipos Scrum pueden trabajar secuencialmente en completar los requisitos que le proporcionan el valor máximo al cliente como se priorizó en el Backlog Priorizado del Producto. En entornos estables, la duración del sprint generalmente es más larga, de 4 a 6 semanas.

Si los requisitos del proyecto cambian durante la duración del mismo debido a, por ejemplo, las modificaciones de los requerimientos de negocio, el mismo método sigue siendo eficaz. Antes de comenzar un sprint —durante los procesos de *Crear el Backlog Priorizado del Producto* o *Refinamiento del Backlog Priorizado del Producto*—, los requisitos de mayor prioridad en el Backlog Priorizado del Producto se seleccionan normalmente para completarse en ese sprint. Dado a que los cambios se han tenido en cuenta en el Backlog Priorizado del Producto, el equipo sólo tiene que determinar el número de tareas que se pueden realizar en el sprint, basado en el tiempo y los recursos proporcionados. La gestión del cambio se lleva a cabo en los procesos de priorización actuales, y se le agregan tareas al Backlog Priorizado del Producto

6.5.1 Cambios a un sprint

Si hay una solicitud de cambio que puede tener un impacto considerable sobre un sprint en curso, el Product Owner, después de consultar con stakeholders relevantes, decide si el cambio puede esperar hasta el próximo sprint o si representa una situación urgente que pueda requerir finalizar el sprint actual y comenzar uno nuevo.

El framework de Scrum especifica claramente que el alcance de un sprint no puede cambiarse una vez que este comience. Si el cambio requerido es tan importante que los resultados del sprint no tendrían ningún valor sin él, entonces el sprint debe terminarse. De lo contrario, entonces el cambio se incorpora más adelante en un sprint (como se muestra en la figura 6-6).

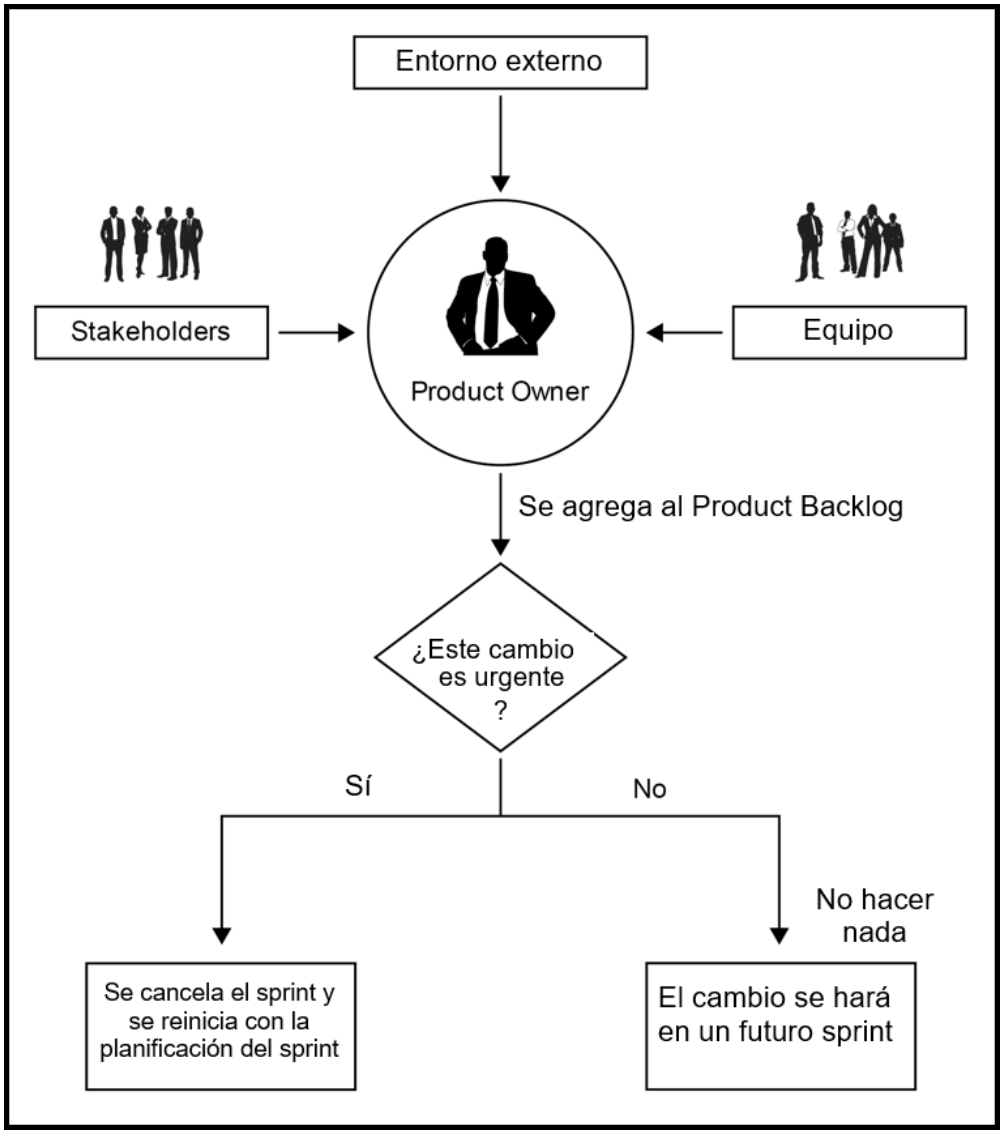


Figura 6-6: Integración del cambio en Scrum

Sólo hay una excepción a la regla de no modificar el alcance de un sprint una vez que ha comenzado. Si el Equipo Scrum determina que se ha sobrestimado en gran medida el esfuerzo durante el sprint y no tiene capacidad para poner en práctica historias de usuario adicionales, el equipo puede preguntarle al Product Owner cuáles historias de usuario deben incorporarse al sprint actual.

Al bloquear el alcance de cada sprint, el equipo es capaz de optimizar y administrar con eficiencia su trabajo y esfuerzo. Un beneficio adicional es que el equipo no tiene que preocuparse por la gestión de los cambios una vez que empieza a trabajar en un sprint. Esta es una gran ventaja del framework de Scrum en comparación a la gestión tradicional de proyectos.

En la gestión tradicional de proyectos, los cambios pueden solicitarse y aprobarse en cualquier momento durante el ciclo de vida del proyecto. Esto a menudo causa confusión entre los miembros del equipo del proyecto, disminuye la motivación del equipo debido a la discontinuidad, da lugar a una falta de concentración y el equipo tiene la sensación de que "nunca se termina nada". En cambio, en los proyectos Scrum, los cambios no se permiten una vez que se inicia un sprint. Esto garantiza que en cada sprint el equipo complete entregables y que las tareas se lleven a cabo. Por otra parte, el negocio reconoce los beneficios tangibles de los entregables que están potencialmente listos para la entrega al final de cada sprint.

Además, como el Product Owner y los stakeholders están conscientes de que los cambios no se permiten una vez que inicia el sprint, y que un sprint dura entre 1 y 6 semanas; definen y priorizan las necesidades durante los procesos adecuados de *Crear épica(s)*, *Crear el Backlog Priorizado del Producto* y el *Refinar el Backlog Priorizado del Producto*.

6.5.1.1 Impacto del cambio esperado en la duración del sprint

Dado que los cambios no están permitidos durante un sprint, el impacto y la frecuencia de los cambios previstos pueden tener un impacto en la decisión relacionada a la duración del sprint cuando esta se determina durante el proceso de *Realizar la planificación del lanzamiento*.

Si los requisitos del proyecto son generalmente estables y no se esperan grandes cambios en un futuro próximo, la duración de un sprint se puede ajustar para que sea más larga, de 4 a 6 semanas. Esto les proporciona estabilidad a los miembros del Equipo Scrum para trabajar en los requisitos del Backlog Priorizado del Producto durante largos periodos de tiempo sin tener que pasar por los procesos de *Crear historias de usuario*, *Estimar historias de usuario* y *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y otros procesos relacionados que se llevan a cabo para cada sprint.

Sin embargo, si los requisitos del proyecto no están muy bien definidos, o si se esperan cambios considerables en un futuro inmediato, la duración del sprint puede ser relativamente corta, de 1 a 3 semanas. Esto les brinda estabilidad a los miembros del Equipo Scrum para trabajar en sprints más cortos y entregar resultados, los que se pueden evaluar por el Product Owner y los stakeholders al final del sprint. Esto también proporciona la flexibilidad suficiente para que puedan aclarar los requisitos y realizar cambios en el Backlog Priorizado del Producto al final de cada sprint.

Para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint bajo un time-box de cuatro semanas, a menos que existan proyectos con requisitos muy estables, donde los sprints pueden extenderse hasta seis semanas.

La figura 6-7 muestra el impacto del cambio esperado en la duración del sprint.

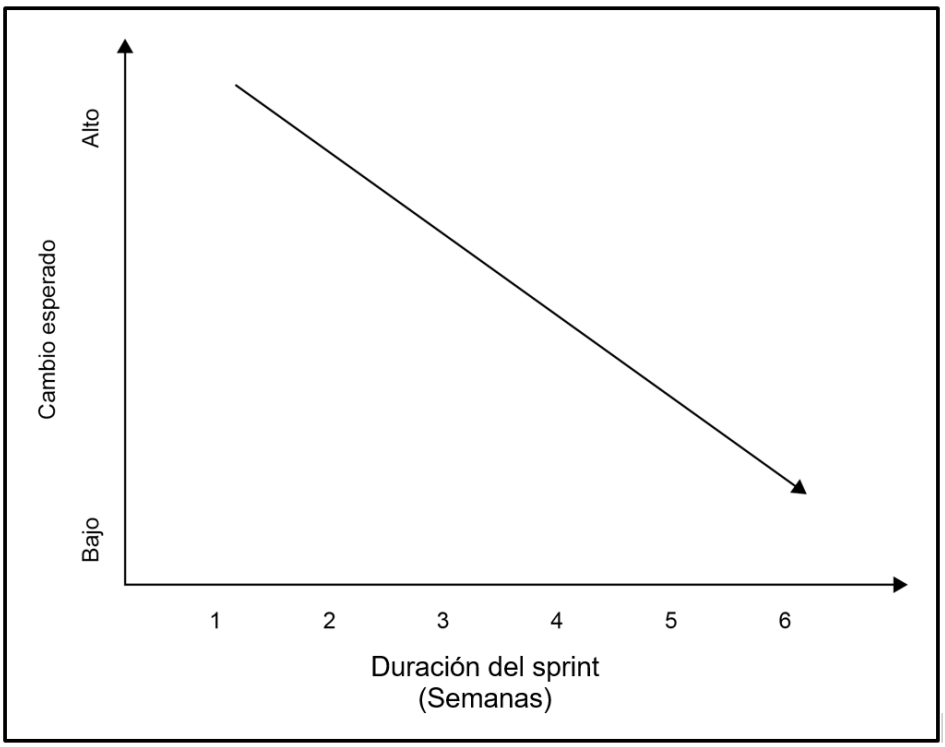


Figura 6-7: Impacto del cambio esperado en la duración del sprint

Sin embargo, es importante tener en cuenta que el cambio esperado no es el único factor que se utiliza para determinar la duración del sprint. Otros factores que también deben tomarse en cuenta son:

- El tiempo real para realizar su trabajo (si el proyecto o entorno corporativo necesita un tiempo específico para realizar tareas de forma, eso podría determinar la duración del sprint)
- La fecha prevista para su lanzamiento (la duración del sprint debe tener en cuenta las fechas de lanzamiento para el producto o el servicio en general)
- Cualquier otro factor que determine el Product Owner o el Scrum Master que deben tenerse en cuenta al determinar la duración del sprint

Es importante tener en cuenta que el cambio en la duración del sprint no debe decidirse a la ligera o de manera periódica (por ejemplo, no es recomendable tener un sprint de tres semanas, luego uno de dos semanas y el siguiente de cuatro semanas, etc.) De preferencia, la duración del sprint debe ser consistente. Uno de los mayores impactos del cambio de la duración del sprint es que causa un restablecimiento en todo el seguimiento a nivel de proyecto. Las velocidades anteriores pueden llegar a ser inútiles para la previsión y la planificación de los futuros sprints. Sin una velocidad precisa (que es una medida primaria en cualquier

proyecto Scrum), el Equipo Scrum no puede medir la eficacia o elegir adecuadamente el número de historias para asumir la planificación del próximo sprint. Por lo tanto, una vez que la duración del sprint se decide, de preferencia debe permanecer constante durante toda la duración del proyecto o a través de múltiples ciclos de sprint.

6.5.1.2 Gestión de cambios mediante el refinamiento del Backlog Priorizado del Producto

Un típico Backlog Priorizado del Producto incluirá todas las historias de usuarios, sus estimaciones de tiempo (incluyendo las estimaciones revisadas) y el estado de las necesidades de mayor prioridad. También se incorporan historias de usuario nuevas o revisadas que resultaron de cambios en los requerimientos de negocio, pedidos de los clientes, condiciones externas del mercado y/o lecciones aprendidas en sprints anteriores.

Una de las responsabilidades principales de los Product Owners es preparar el Backlog Priorizado del Producto para garantizar que los requisitos priorizados en dicho backlog se incluyan en los próximos dos o tres sprints, y se refinen en acuerdo con las historias de usuario. Se recomienda que el Product Owner dedique una cantidad considerable de tiempo en cada sprint para refinar el backlog. El Product Owner es responsable de añadir y modificar elementos a dicho backlog en respuesta a los cambios, así como de proporcionar historias de usuario más detalladas que se utilizarán en el próximo sprint.

Este refinamiento o *grooming* ayuda a asegurar que la refinación de los requisitos y sus historias de usuario se hagan mucho antes de la reunión de planificación del sprint, a fin de que el equipo tenga un conjunto de historias muy bien analizadas y claramente definidas que puedan dividirse fácilmente en tareas y, posteriormente estimadas. Con base en las lecciones aprendidas del sprint actual, puede haber cambios en los requisitos, o bien una priorización nueva que pueda incorporarse fácilmente en sprints posteriores. Este refinamiento apoya y mejora la flexibilidad del modelo Scrum mediante la incorporación de los últimos avances técnicos y de negocio en futuros sprints.

La reunión de revisión del Backlog Priorizado del Producto es una reunión formal durante el proceso de refinamiento de dicho backlog, que ayuda al Equipo Scrum a repasar y lograr el consenso sobre el *grooming* del Backlog Priorizado del Producto. Sin embargo, además de la reunión de revisión del Backlog Priorizado del Producto, su refinamiento debe darse durante todo el proyecto y puede incluir situaciones en las que el Product Owner escriba nuevas historias de usuarios o vuelva a priorizar las historias de usuario en el Backlog Priorizado del Producto vigente, y los miembros del Equipo Scrum o stakeholders ofrezcan sugerencias sobre las nuevas historias al Product Owner, y así sucesivamente.

Es importante tener en cuenta que cualquier elemento del Backlog Priorizado del Producto está siempre abierto para la re-estimación hasta que el Sprint Backlog sea finalizado en el proceso de *Crear el Sprint Backlog*. Después de ello, los cambios se podrán seguir haciendo inclusive hasta momentos antes de la reunión de planificación del sprint, si es necesario.

6.5.1.2.1 Reunión eficaz de revisión del Backlog Priorizado del Producto (o Sesión de refinamiento del Backlog Priorizado del Producto)

El Product Owner es quien está encargado de que se lleve a cabo una reunión de revisión del Backlog Priorizado del Producto durante el proceso de *Refinar el Backlog Priorizado del Producto*. Es importante que el Product Owner establezca los objetivos y que de preferencia desarrolle un orden del día antes de iniciar la reunión. Sin esto, la sesión no tendría estructura y podría resultar improductiva. También es importante limitar el número de stakeholders que participan en la reunión. El tener demasiados participantes tiende a disminuir la eficiencia general de la reunión. El Product Owner debe invitar sólo a los stakeholders cuyas votaciones se requieren para la sesión de la preparación. Se deben incluir todos los miembros del Equipo Scrum debido a que su opinión es valiosa para el trabajo que se realiza y los problemas que se encontraron. Si los resultados de la sesión de preparación o cuidado resultan en nueva priorización o cambio en el Backlog Priorizado del Producto, es importante que el equipo esté de acuerdo con esos cambios.

Una eficaz sesión de *grooming* debe resultar en los elementos claramente definidos en el Backlog Priorizado del Producto para que el Equipo Scrum entienda los requisitos del cliente. Esto también ayuda a que el equipo se familiarice con todas las historias de usuario en caso de que una o más de ellas sean incluidas en un sprint a corto plazo. También pueden tratarse los criterios de aceptación y de terminado durante las sesiones de preparación.

En Scrum los ejercicios de refinación no tienen a un time-box. El refinamiento del Backlog Priorizado del Producto es una actividad continua del Product Owner.

6.5.1.3 Gestión de cambios durante la demostración y validación del sprint

Aunque el Product Owner tiene la última palabra sobre los elementos del Backlog Priorizado del Producto y decide si se aceptan o rechazan las historias de usuario (correspondientes a las solicitudes de cambio aprobadas) presentadas durante el proceso de *Demostrar y validar el sprint*, es la responsabilidad del Scrum Master garantizar que los requisitos y criterios de aceptación no se modifiquen durante la reunión de revisión del sprint de las historias de usuario completadas en el sprint actual. Esto evita el rechazo de futuras historias de usuario basado en el hecho de que no cumplen los requisitos recién cambiados. Si los requisitos se debieran cambiar, cualquier elemento del Backlog Priorizado del Producto correspondiente debe revisarse para adaptarse a los requisitos modificados en un sprint futuro.

6.6 Cambio en portafolios y programas

Cualquier cambio que surja, ya sea en los programas o portafolios, puede tener un efecto en cascada en todos los proyectos dependientes y sprints. Por lo tanto, se recomienda minimizar los cambios en estos niveles más altos. Si se requiere un cambio, y todos los stakeholders están de acuerdo en hacerlo a estos niveles, se debe tomar en cuenta lo siguiente:

6.6.1 En portafolios

1. No se recomienda hacer cambios entre dos reuniones de Portfolio Backlog.
2. Si el cambio es menor, el Portfolio Product Owner debe contar con la aprobación de los stakeholders correspondientes (por ejemplo, el patrocinador, el cliente y el usuario meta) y después añadir los requisitos al Portfolio Backlog. Los Product Owners y del programa y del proyecto analizarán los requisitos para su inclusión en futuros sprints.
3. Si el cambio es importante, las actividades del portafolio, junto con los programas asociados, los proyectos y los sprints tienen que detenerse y se debe llevar a cabo una reunión del Portfolio Backlog para determinar cuáles serán los siguientes pasos.
4. Las reuniones del Backlog Priorizado del Producto del Portfolio (también conocidas como reuniones del Program Backlog), deben llevarse a cabo en intervalos de 4 a 12 meses. La frecuencia y el impacto de los cambios en un portafolio determinan en gran medida la duración de tiempo entre dos reuniones del Portfolio Backlog. Si son varios los cambios esperados en el portafolio, es preferible llevar a cabo este tipo de reuniones en intervalos más frecuentes (por ejemplo, de cuatro a seis meses); pero si hay menos cambios esperados, y si los requisitos son estables, la duración entre dos reuniones podría aumentarse (por ejemplo, de 9 a 12 meses).

6.6.2 En programas

1. No se recomienda hacer cambios entre dos reuniones del Program Backlog.
2. Si el cambio es menor, el Program Product Owner debe contar con la aprobación de los stakeholders correspondientes (por ejemplo, el patrocinador, el cliente y el usuario meta) y después añadir los requisitos al Program Backlog. Los Product Owners y del programa y del proyecto analizarán los requisitos para su inclusión en futuros sprints.
3. Si el cambio es importante, las actividades del programa, así como los proyectos y sprints relacionados deben detenerse y se debe llevar a cabo una reunión del Product Backlog para determinar cuáles serán los siguientes pasos.
4. Las reuniones del Backlog Priorizado del Producto del programa (también conocidas como reuniones del Program Backlog), deben llevarse a cabo, de preferencia, en intervalos de dos a seis meses. La frecuencia y el impacto de los cambios en un programa determinan en gran medida la duración de tiempo entre dos reuniones del Program Backlog. Si hay varios cambios previstos en el programa, es preferible llevar a cabo este tipo de reuniones en intervalos más regulares (por

ejemplo, de dos a tres meses); pero si hay menos cambios esperados y si los requisitos son estables, la duración entre dos reuniones podría aumentarse (por ejemplo, de cinco a seis meses).

La figura 6-8 muestra cómo se pueden administrar los cambios dentro del flujo de Scrum tanto para los portafolios como para los programas.

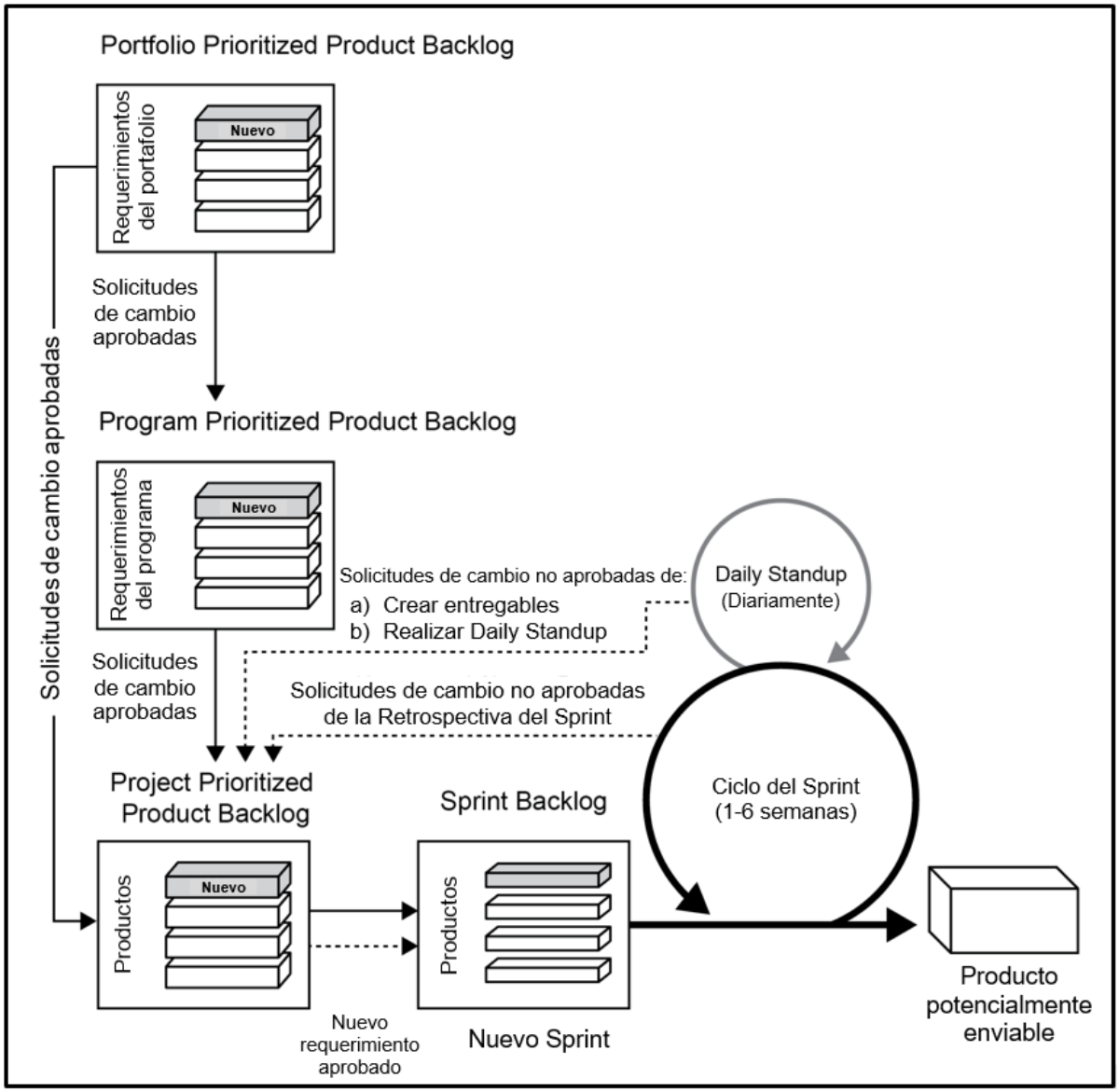


Figura 6-8: Incorporación de cambios en portafolios y programas

6.7 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	• Sugiere mejoras o cambios durante los procesos de *Crear entregables* y *Realizar Daily Standup*.
Product Owner/ Chief Product Owner	• Proporciona solicitudes de cambio de un proyecto • Evalúa el impacto de las solicitudes de cambio planteadas por el portafolio, el programa o el proyecto • Prioriza las historias de usuario en el Backlog Priorizado del Producto del proyecto • Evalúa el impacto de los problemas sobre los objetivos del proyecto identificados por el Equipo Scrum • Proporciona una comunicación clara a los stakeholders sobre los elementos del Product Backlog que se han vuelto a priorizar
Scrum Master/ Chief Scrum Master	• Facilita la identificación y evaluación de los problemas y solicitudes de cambio por el Equipo Scrum
Program Product Owner	• Proporciona solicitudes de cambio para los programas • Aprueba los productos que son modificados, eliminados o agregados de acuerdo con los requisitos del programa
Program Scrum Master	• Facilita la identificación, evaluación y gestión de las solicitudes de cambio para los programas
Portfolio Product Owner	• Proporciona solicitudes de cambio para los portafolios • Aprueba los productos que son modificados, eliminados o agregados de acuerdo con los requisitos del portafolio
Portfolio Scrum Master	• Facilita la identificación, evaluación y gestión de las solicitudes de cambio para los portafolios
Stakeholder(s)	• Proporciona solicitudes de cambios • Participa en la aprobación y priorización de las solicitudes de cambio
Scrum Guidance Body	• Proporcionar una guía general para los procedimientos de gestión de cambios que deben seguirse durante todo el proyecto

Tabla 6-1: Resumen de las responsabilidades pertinentes al cambio

6.8 Scrum vs. Gestión tradicional de proyectos

La gestión de cambios en los proyectos que se gestionan en forma tradicional está estrechamente relacionada con la gestión de la configuración. Todos los cambios se consideran con base en su magnitud de variación desde un valor base. Al Project Manager se le permite gestionar las actividades y decisiones diarias del proyecto. Cuando una solicitud de cambio supera las tolerancias definidas, el Project Manager debe llevar la propuesta de cambio a niveles superiores de gestión y esperar la decisión antes de hacerla efectiva. El Project Manager registra primero la petición de cambio en un registro de problemas o cambios (*Issue Log* o *Change Log*) y después presenta el cambio a autoridades superiores. Estas podrían incluir al patrocinador del proyecto (*Project Sponsor*), así como a otros stakeholders relevantes y a aquellos que toman decisiones sobre el caso. En algún momento se deberá llevar a cabo una evaluación del impacto. Con base al impacto estimado del cambio, se tomará una decisión respecto a si el cambio debe aplicarse o no. El Project Manager también podrá proponer posibles soluciones a los problemas planteados por el cambio. Si las autoridades superiores deciden proceder con el cambio, el Project Manager es responsable de asegurar que el cambio se implemente correctamente.

El cambio en Scrum funciona de manera muy diferente en comparación con la gestión tradicional de proyectos. El framework de Scrum está altamente enfocado en la gestión de cambios de manera eficaz y eficiente. Cada vez que el Product Owner o el Equipo Scrum reconoce un problema o defecto o identifica un elemento del Backlog Priorizado del Producto que deba modificarse, sustituirse o añadirse, el cambio se realiza en el Backlog Priorizado del Producto. Del mismo modo, la alta gerencia, el Product Owner o el(los) stakeholder(s) puede(n) añadir solicitudes de cambio a dicho backlog. El Product Owner y el(los) stakeholders aprueba(n) las solicitudes de cambio y las nuevas prioridades del portafolio según corresponda. Siempre que hay un problema o una nueva exigencia que se deba atender, la cual resulta en cambios inmediatos que afectan el sprint actual, el Product Owner debe cancelar el sprint con la aprobación de los stakeholders relevantes. Una vez terminado, el sprint se vuelve a planificar y a reiniciar para incorporar los nuevos requisitos.

Sin embargo, si el problema o cambio no es importante y no garantiza un cambio dentro del sprint actual, el cambio se añadirá al Backlog Priorizado del Producto y se incorporará en la planificación para un futuro sprint. Esto les da a los stakeholders la capacidad para responder a los cambios en el ambiente externo, mientras se mantiene un cierto nivel de control sobre las actividades en curso dentro del proyecto. Además, al final de cada sprint el Equipo Scrum muestra los entregables clasificados como terminados. Estos entregables potencialmente enviables pueden revisarse por el Product Owner y otros stakeholders.

7. RIESGO

7.1 Introducción

El propósito de este capítulo es definir los riesgos, analizar la gestión de riesgos en un entorno de Scrum y considerar las herramientas que facilitan la gestión de los riesgos. Para garantizar la viabilidad del negocio, reducir la probabilidad de fracaso de los proyectos y tomar decisiones de negocio más informadas, es importante que los riesgos se gestionen con eficacia mediante un enfoque metódico y bien organizado.

En un entorno de Scrum, los riesgos generalmente se minimizan, en gran parte debido al trabajo que se realiza en los sprints, donde se produce una serie continua de entregables en ciclos muy cortos. Los entregables se comparan con las expectativas y el Product Owner participa activamente en el proyecto. Sin embargo, hasta en el más simple de los proyectos, las cosas pueden salir mal, por lo que es importante contar con una estrategia para identificar y atender los riesgos.

El *riesgo*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™),* es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

7.2 Guía de roles—Esta sección ofrece orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: Product Owner, Scrum Master y Equipo Scrum.

7.3 ¿Qué es un riesgo?—En esta sección se define el riesgo y se explica cómo puede afectar los objetivos de un proyecto y contribuir al éxito o al fracaso del mismo.

7.4 Procedimiento de gestión de riesgos—Esta sección presenta las técnicas claves de la gestión de riesgos y profundiza en el desarrollo de estrategias para identificar, evaluar y gestionar riesgos.

7.5 Minimizar el riesgo mediante el uso de Scrum—Esta sección explica los aspectos claves de Scrum que lo convierten en un framework de gestión ideal para manejar con eficacia los riesgos en varios niveles: portafolio, programa y proyecto.

7.6 Resumen de responsabilidades—Esta sección describe las responsabilidades pertinentes a la calidad para cada persona o rol en un proyecto.

7.7 Scrum vs. Gestión tradicional de proyectos—Esta sección destaca los beneficios de la gestión de calidad en el método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

7.2 Guía de roles

4. Product Owner—La principal responsabilidad en la gestión de riesgos en un proyecto recae sobre el Product Owner, por ello, todo este capítulo aplica principalmente a este rol.
5. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 7.3, 7.4 y 7.7.
6. Equipo Scrum—El Equipo Scrum debe enfocarse principalmente en las secciones 7.3 y 7.7.

7.3 ¿Qué es un riesgo?

El riesgo se define como un evento incierto o serie eventos que pueden afectar los objetivos de un proyecto y pudieran contribuir a su éxito o fracaso. Los riesgos con un potencial de impacto positivo en el proyecto se denominan “oportunidades”, mientras que las amenazas son riesgos que pudieran afectar negativamente a un proyecto. La gestión de riesgos debe hacerse proactivamente y es un proceso iterativo que debería empezar al inicio del proyecto y continuar durante toda la vida del mismo. El proceso de gestión de riesgos debe seguir algunos pasos estandarizados para garantizar que los riesgos sean identificados, evaluados y se determine un curso de acción para actuar en consecuencia.

Es necesario identificar, evaluar y responder a los riesgos basándose principalmente en dos factores: la probabilidad de ocurrencia y el impacto probable en caso de que ocurra. Los riesgos de alta probabilidad y con un alto índice de impacto deben ser abordados antes de aquellos con una calificación más baja. En general, una vez que se detecte un riesgo, es importante comprender los aspectos básicos del riesgo respecto a las posibles causas, el área de la incertidumbre y los efectos potenciales si se produce el riesgo.

7.3.1 Diferencia entre riesgos y problemas

Los riesgos son las incertidumbres relacionadas a un proyecto que podrían alterar considerablemente el resultado del proyecto de manera positiva o negativa. Debido a que los riesgos son incertidumbres a futuro, no tienen ningún impacto actual en el proyecto, pero podrían tener un impacto potencial en el futuro. Los siguientes son algunos ejemplos de riesgos:

- Incluso después de una amplia capacitación, es posible que los representantes de servicio al cliente no estén listos para tomar pedidos el día oficial del lanzamiento.
- Es posible que una cuadrilla de pintores se retrase debido a las fuertes lluvias, lo cual pudiera influir negativamente en el cronograma del proyecto.

Los problemas generalmente son certezas que se están suscitando en el proyecto, por lo que no hay necesidad de realizar una evaluación de la probabilidad como lo haríamos con un riesgo. Los problemas deben atenderse. Los siguientes son algunos ejemplos de problemas:

- No se autoriza el financiamiento.
- Los requisitos no están claros.

Si no se atienden a tiempo los riesgos, estos podrían convertirse en problemas. El objetivo de la gestión de riesgos es estar preparados con planes para poder abordar cualquier riesgo que pudiera presentarse.

7.3.2 Actitud de riesgo

Entre los stakeholders se incluyen a todas aquellas personas u organizaciones afectadas por el proyecto, así como aquellas que cuentan con la capacidad para afectarlo. Es importante entender la actitud de riesgo de los stakeholders. La actitud de riesgo (conocida en inglés como *Risk Attitude*) se ve influenciada por los siguientes tres factores:

1. **Apetito de riesgo:** Es la cantidad de incertidumbre que está dispuesta a asumir el stakeholder o la organización.
2. **Tolerancia al riesgo:** Indica el grado, cantidad o volumen de riesgo que resistirán los stakeholders
3. **Umbral de riesgo:** Es el nivel al cual es riesgo aceptable para la organización del stakeholder. Un riesgo caerá por encima o por debajo del umbral de riesgo. Si está por debajo, es más probable que el stakeholder o la organización acepte el riesgo.

En esencia, la actitud de riesgo de los stakeholders determina cuánto riesgo considera aceptable el stakeholder(s). Es un factor determinante cuando deciden tomar acciones para mitigar posibles riesgos adversos. Por lo tanto, es importante entender los niveles de tolerancia de los stakeholders en relación a diversos factores como el costo, la calidad, el alcance y los plazos.

La función de utilidad (del inglés: *Utility Function*) es un modelo utilizado para medir la preferencia del stakeholder por el riesgo o su actitud hacia el riesgo. Esto define el nivel del stakeholder para aceptar riesgos. Las tres categorías de la función de utilidad son las siguientes:

1. **Aversión al riesgo:** Describe al stakeholder que no está dispuesto a aceptar un riesgo sin importar el beneficio o la oportunidad anticipada.
2. **Neutral al riesgo:** Describe al stakeholder que ni tiene aversión al riesgo, ni busca riesgos; cualquier decisión que no se ve afectada por el nivel de incertidumbre de los resultados. Cuando

dos posibles escenarios llevan el mismo nivel de beneficio, el stakeholder neutral al riesgo no se preocupará si uno de dichos casos es más riesgoso que el otro.

3. **Búsqueda de riesgos:** Se refiere a que un stakeholder está dispuesto a aceptar el riesgo, incluso si ofrece un aumento marginal de retorno o beneficio al proyecto.

7.4 Procedimiento de gestión de riesgos

La gestión de riesgos se compone de los siguientes cinco pasos, mismos que deben llevarse a cabo en forma iterativa durante el proyecto:

1. Identificación de riesgos: Utilizar diversas técnicas para identificar todos los riesgos potenciales.
2. Evaluación de riesgos: Evaluar y estimar los riesgos identificados.
3. Priorización de riesgos: Dar prioridad al riesgo que habrá de incluirse en el Backlog Priorizado del Producto.
4. Mitigación de riesgos: Desarrollar de una estrategia adecuada para hacer frente a un riesgo.
5. Comunicación de riesgos: Comunicar a los stakeholders apropiados los resultados de los primeros cuatros pasos de la gestión de riesgos y determinar su percepción respecto a eventos inciertos.

7.4.1 Identificación de riesgos

Los miembros del Equipo Scrum deben hacer un intento por identificar todos los riesgos que pudieran afectar el proyecto. Con tal solo observar el proyecto desde una perspectiva diferente —y con el uso de una variedad de técnicas—, pueden lograrlo a fondo. La identificación de riesgos se lleva a cabo a lo largo del proyecto y los riesgos identificados se convierten en entradas en varios procesos de Scrum, incluyendo: *Crear el Backlog Priorizado del Producto*, *Refinar el Backlog Priorizado del Producto* y *Demostrar y validar el sprint*.

Las siguientes técnicas se utilizan comúnmente para identificar riesgos:

7.4.1.1 Técnicas de identificación de riesgos

1. **Revisar las lecciones aprendidas de los procesos de Retrospectiva del sprint o Retrospectiva del proyecto**

   Aprender de proyectos similares y de sprints anteriores en el mismo proyecto, al igual que explorar las incertidumbres que afectan a dichos proyectos y sprints, puede ser una forma útil de identificar riesgos.

2. **Listas de verificación de riesgos (*Risk Checklists*)**

   Las listas de verificación de riesgos (conocidas en inglés como: *Risk Checklists*) pueden incluir puntos clave a considerarse cuando se identifican los riesgos, riesgos comunes encontrados en un proyecto Scrum, o incluso categorías de riesgo que el equipo debe atender. Las listas de verificación son una valiosa herramienta que ayuda a garantizar una identificación integral del riesgo.

3. **Lista corta de riesgos (*Risk Prompt List*)**

   Conocidas en inglés como *Risk Prompt List*, estas listas se utilizan para estimular el pensamiento respecto a la fuente de donde se pudieran originar los riesgos. Dichas listas para distintas industrias y tipos de proyectos están disponibles al público.

4. **Lluvia de ideas (*Brainstorming*)**

   Son sesiones donde los stakeholders y los miembros del equipo principal de Scrum comparten abiertamente ideas por medio de discusiones y sesiones de intercambio de conocimientos, generalmente dirigidas por un facilitador.

5. **Estructura de distribución de riesgos (*Risk Breakdown Structure*)**

   Una de las herramientas clave que se utiliza para identificar riesgos es la estructura de distribución de riesgos, conocida en inglés como: *Risk Breakdown Structure*. En esta estructura se agrupan los riesgos con base en sus categorías o modalidades. Por ejemplo, los riesgos se pueden clasificar como financieros, técnicos o en materia de seguridad.

7.4.2 Evaluación de riesgos

La evaluación de riesgos ayuda a entender el impacto potencial de un riesgo; a ver qué posibilidades hay de que suceda y cuándo pudiera materializarse. Se debe estimar el efecto generalizado en el valor del negocio. Si dicho impacto es lo suficientemente considerable como para superar la justificación del negocio, se debe de tomar la decisión si se le da continuidad al proyecto.

La evaluación de riesgo se lleva a cabo en relación a la probabilidad, proximidad e impacto. La probabilidad de riesgo es la probabilidad de su ocurrencia, mientras que la proximidad se refiere a cuándo pudiera suscitarse un riesgo. El impacto es el efecto probable del riesgo sobre un proyecto u organización.

Para estimar la probabilidad de un riesgo, se pueden utilizar varias técnicas, incluyendo árboles de probabilidad, análisis de Pareto y matriz de impacto.

Además de la probabilidad, en la evaluación de riesgos se evalúa también el efecto potencial neto de los riesgos sobre el proyecto o la organización. Dichos efectos se pueden estimar utilizando técnicas tales como modelos de riesgo y valor monetario esperado.

7.4.2.1 Técnicas de evaluación de riesgos

1. Reunión de riesgos

Los riesgos de pueden priorizar con facilidad por el Product Owner convocando a una reunión del equipo principal de Scrum con la opción de invitar a los stakeholders relevantes a dicha reunión. El equipo pudiera reunirse y dar prioridad a distintos riesgos con base a su evaluación subjetiva del impacto del riesgo sobre los objetivos del proyecto.

2. Árboles de probabilidad

Los eventos potenciales se representan en un diagrama con una rama para cada resultado posible de los acontecimientos. La probabilidad de cada resultado se indica en la rama apropiada, y estos valores se pueden utilizar para calcular el impacto general de la ocurrencia de riesgos en un proyecto. Los valores resultantes se suman para calcular el impacto general esperado de un riesgo en un proyecto (véase la figura 7-1).

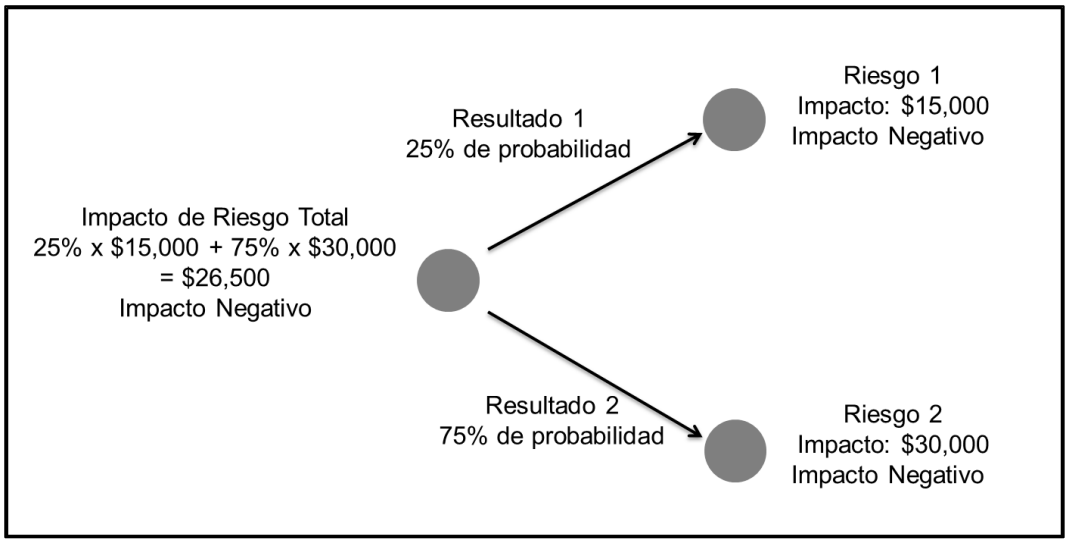


Figura 7-1: Ejemplo de árbol de probabilidad

3. Análisis de Pareto

Esta técnica de evaluación de riesgos implica la clasificación de riesgos por magnitud. Ayuda al Equipo Scrum a atender los riesgos por orden de impacto probable en un proyecto. Por ejemplo, en la figura 7-2, el Riesgo 1 tiene el mayor impacto y de preferencia debe atenderse primero.

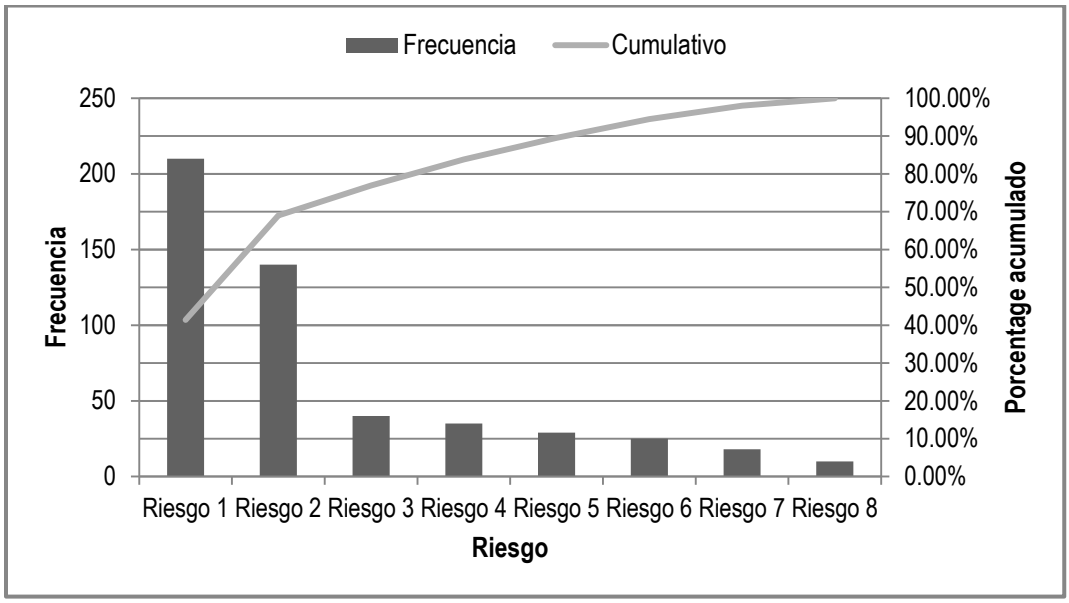


Figura 7-2: Ejemplo de un diagrama de Pareto

4. Cuadrícula de probabilidad e impacto (*Probability Impact Grid*)

Es una cuadrícula donde se evalúan los riesgos para establecer la probabilidad de ocurrencia y del impacto potencia en los objetivos del proyecto. Generalmente, se asigna una clasificación numérica tanto para la probabilidad, como para el impacto en forma independiente. Ambos valores se multiplican para obtener una puntuación de la gravedad del riesgo, lo cual se puede utilizar para priorizar los riesgos.

Por ejemplo, la puntuación de la gravedad del riesgo para un riesgo con una probabilidad del 50 % y un índice de impacto de 0.6 se calcularía de la siguiente forma:

0.5 (Probabilidad) x 0.6 (Impacto) = 0.3

Los esquemas de calificación implementados se determinan dentro de la organización o para el proyecto. Por lo general, se utiliza una escala decimal, del cero al uno, donde un índice de probabilidad de 0.5 indicaría un 50 % de posibilidad. Otras opciones incluyen una escala del uno al diez, o alta (3), mediana (2) y baja (1).

La figura 7-3 muestra el uso de la escala decimal. Cada riesgo se califica con base a su probabilidad de ocurrencia e impacto en una escala objetiva.

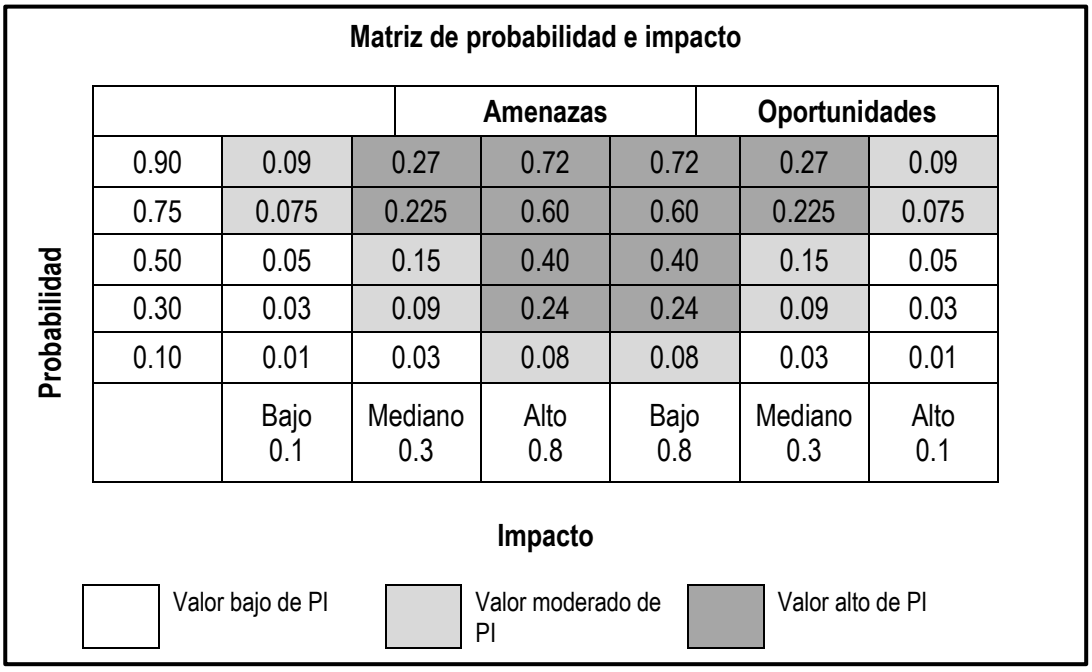


Figura 7-3: Ejemplo de matriz de probabilidad e impacto

El método de asignar valores de probabilidad e impacto a los riesgos varía dependiendo del proyecto y de la cantidad de riesgos que se evalúen, así como los procesos y procedimientos organizacionales existentes. Sin embargo, si se aplica la sencilla fórmula: P x I, la gravedad del riesgo se puede calcular en una escala numérica o categórica.

5. Valor monetario esperado

El valor monetario del riesgo está basado en su valor monetario esperado (EMV, por sus siglas en inglés). Dicho valor se calcula multiplicando el impacto monetario por la probabilidad de riesgo, según la aproximación del cliente.

Valor monetario esperado = Impacto del riesgo (en dólares) x Probabilidad del riesgo (porcentaje)

Por ejemplo, un riesgo con un pacto negativo estimado de 1,000 dólares y un 50 % de probabilidad de ocurrencia resultaría en el siguiente valor monetario esperado:

Valor monetario esperado = $1,000 x 0.50 = $500

7.4.3 Priorización de riesgos

Scrum permite una rápida identificación y evaluación de riesgos. Los riesgos identificados se toman en cuenta en el proceso de *Crear el Backlog Priorizado del Producto*, o bien, cuando se actualiza dicho backlog en el proceso de su refinamiento; de tal forma que un Backlog Priorizado del Producto pudiera también conocerse como: Backlog Priorizado del Producto del riesgo ajustado.

Los riesgos se pueden identificar y evaluar con base en cualquier técnica de identificación y evaluación de riesgos que se mencionan anteriormente.

En los procesos de *Crear el Backlog Priorizado del Producto* y *Refinar el Backlog Priorizado del Producto*, las historias de usuario priorizadas del Backlog Priorizado del Producto existente y la lista priorizada de riesgos se combinan para crear un backlog actualizado que incluya los riesgos identificados:

Pasos para actualizar el Backlog Priorizado del Producto con riesgos identificados:

1. Crear una lista de riesgos priorizados (por ejemplo: los riesgos se pueden priorizar por valor utilizando la técnica de valor monetario esperado).

2. Seleccionar aquellos riesgos identificados que pudieran mitigarse; y para loa cuáles el equipo decide tomar acción específica de riesgos durante el sprint a fin de mitigar tales riesgos.

3. Crear una lista de historias de usuarios el Backlog Priorizado del Producto, mismas que se priorizan por valor (por ejemplo: el valor de cada historia de usuario se puede evaluar con base en su retorno sobre la inversión esperado).

4. Combinar las listas de los pasos 2 y 3 y priorizarlos por valor para llegar al Backlog Priorizado del Producto.

La figura 7-4 muestra el proceso de priorización de riesgos:

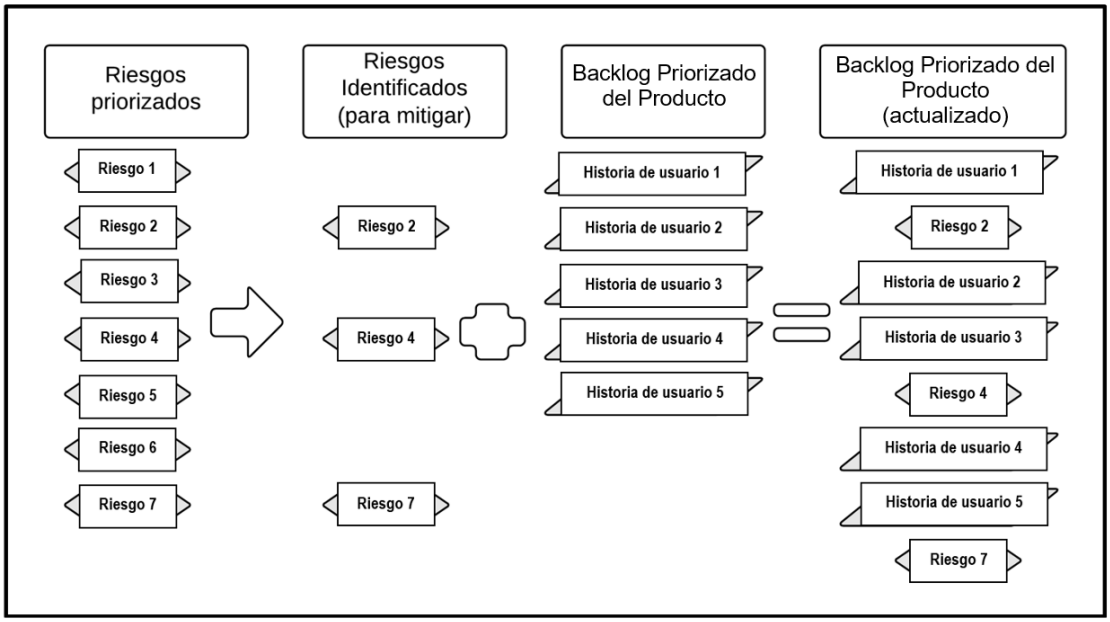


Figura 7-4: Proceso de priorización de riesgos

7.4.4 Mitigación de riesgos

La respuesta a cada riesgo dependerá de la probabilidad y el impacto del mismo. Sin embargo, la naturaleza iterativa de Scrum —con sus ciclos rápidos de respuesta y retroalimentación—, permite que las fallas se detecten de forma temprana; por lo tanto, hablando en términos prácticos, tiene una función de mitigación natural construida dentro del sistema.

Un riesgo puede ser mitigado mediante la implementación de una serie de respuestas. En la mayoría de los casos, las respuestas son proactivas o reactivas. En el caso de un riesgo, se puede formular un plan B, que se puede utilizar como una alternativa en caso de que el riesgo se materialice; en este caso, el plan B es una respuesta reactiva. En ocasiones, los riesgos se aceptan y son un ejemplo de una respuesta al riesgo que no es ni preventiva ni reactiva. Los riesgos se aceptan debido a varias razones, como en una situación en la que la probabilidad o el impacto de riesgo son muy bajos para una respuesta. La aceptación también puede ser el caso en una situación en la que la aprehensión de riesgos secundarios puede disuadir al Product Owner de tomar cualquier acción. El esfuerzo realizado por el Product Owner para reducir la probabilidad del riesgo o del impacto (o ambos), es un ejemplo de una respuesta proactiva a la mitigación de riesgos.

Una vez que los riesgos identificados se incluyen como parte del Backlog Priorizado del Producto (véase la figura 7-4), varios riesgos se mitigan durante el proceso de *Crear entregables* cuando se completan las tareas relacionadas a las historias de usuario definidas en el proceso de *Crear el Backlog Priorizado del Producto*.

En Scrum, la propiedad del riesgo es claramente del Product Owner para la gestión de los riesgos relacionados a los aspectos del negocio; la responsabilidad es también del Equipo Scrum para la implementación de respuestas al riesgo durante el curso de un sprint. El Scrum Guidance Body puede consultarse para pedir orientación sobre la forma de implementar la respuesta a los riesgos y ver si las acciones se alinean con las directrices de la organización en su conjunto. El Scrum Master mantiene una estrecha vigilancia sobre los riesgos potenciales que pudieran afectar el proyecto y mantiene informado al Product Owner y al Equipo Scrum.

7.4.4.1 Despunte basado en riesgo (*Risk-Based Spike*)

Los despuntes basados en riesgo son básicamente experimentos que implican una investigación o hacer un prototipo para entender mejor los riesgos potenciales. En un *spike*, se lleva a cabo un intenso ejercicio de dos a tres días (de preferencia al inicio de un proyecto, antes del proceso de *Desarrollar épica(s)* o de *Crear el Backlog Priorizado del Producto*) para ayudar al equipo a determinar las incertidumbres que pudieran afectar al proyecto. Los despuntes basados en riesgos son útiles cuando el Equipo Scrum trabaja con el cliente trabaja y se acostumbra a las nuevas tecnologías y herramientas, o bien, cuando las historias de usuario son muy extensas. Ayuda también a estimar con más precisión el tiempo y el esfuerzo.

7.4.5 Comunicación de riesgos

Debido a que los stakeholders tienen un interés en el proyecto, es importante comunicarles los asuntos relacionados a los riesgos. La información proporcionada a los stakeholders relacionada al riesgo debe incluir el impacto potencial, así como los planes para hacerle frente a cada riesgo. Esta comunicación siempre está en curso y debe ocurrir a la par de los cuatro pasos secuenciales discutidos hasta ahora: identificación, evaluación, priorización y mitigación de riesgos. El Equipo Scrum pudiera también discutir con el Scrum Master los riesgos específicos relacionados a sus tareas durante los Daily Standups. El Product Owner es responsable de la priorización de riesgos y de la comunicación del Backlog Priorizado del Producto al Equipo Scrum.

Una herramienta importante que se puede utilizar para comunicar la información relacionada a los riesgos es el Risk Burndown Chart.

7.4.5.1 Risk Burndown Chart

La gestión de riesgos es esencial para garantizar la creación de valor; por lo tanto, las actividades de gestión de riesgos se llevan a cabo durante todo el ciclo de vida del proyecto y no sólo durante el inicio del mismo.

Cada riesgo se pude evaluar usando diferentes herramientas de evaluación de riesgos. Sin embargo, la herramienta preferida para evaluación riesgos y crear una gráfica denominada *Risk Burndown Chart* es el Valor Monetario Esperado (VME), tal como se describe en la sección 7.4.2.1.

La información obtenida durante la evaluación de riesgos se puede utilizar para crear un Risk Burndown Chart. Esta representa la severidad del riesgo del proyecto acumulativo con en el tiempo. Las probabilidades de los diversos riesgos se trazan una sobre otra para mostrar el riesgo acumulativo en el eje vertical. La identificación y evaluación inicial de los riesgos en el proyecto, así como la creación de el Risk Burndown Chart se realizan al inicio. Después, en intervalos de tiempo predeterminados, los nuevos riesgos pueden ser identificados y evaluados mientras que los riesgos restantes deben ser reevaluados en la gráfica según sea el caso. Un momento apropiado para hacerlo es durante la reunión de planificación del sprint. El seguimiento de los riesgos de esta forma le permite al equipo reconocer tendencias en exposición de riesgo y tomar acciones adecuadas conforme sea necesario.

La figura 7-5 muestra un ejemplo de un Risk Burndown Chart.

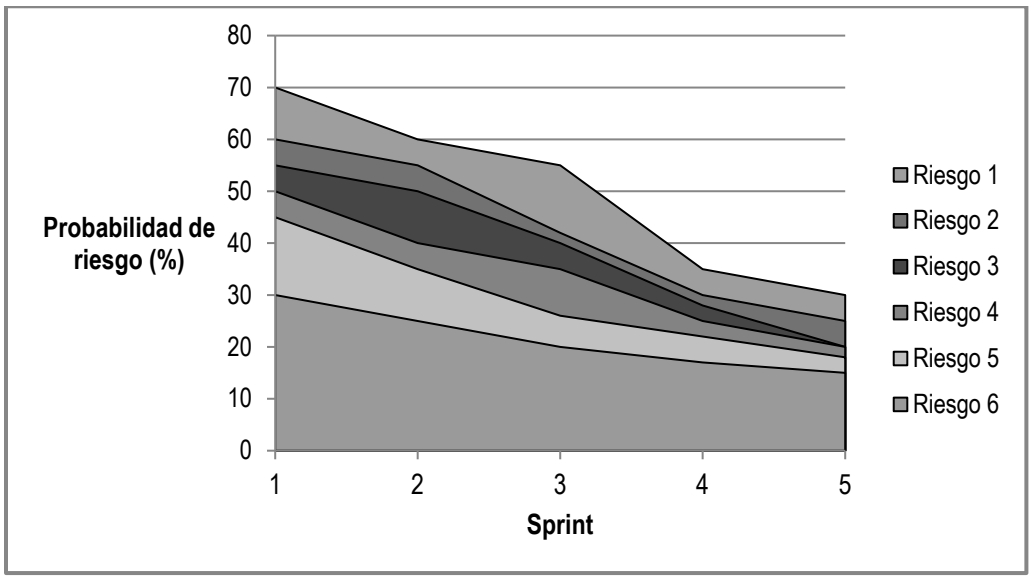


Figura 7-5: Ejemplo de un Risk Burndown Chart

7.5 Minimizar riesgos por medio de Scrum

Al ser un proceso ágil e iterativo, el framework de Scrum minimiza inherentemente el riesgo. Las siguientes prácticas de Scrum facilitan la gestión efectiva del riesgo:

1. **La flexibilidad reduce el riesgo relacionado al entorno empresarial**
   El riesgo se reduce en gran medida en Scrum debido a la flexibilidad en la adición o modificación de los requisitos en cualquier momento del ciclo de vida del proyecto. Esto le permite a la organización responder a las amenazas u oportunidades en el entorno empresarial, así como a las necesidades imprevistas cada vez que surjan, por lo general con un bajo costo de la gestión de tales riesgos.

2. **La retroalimentación constante reduce el riesgo relacionado a las expectativas**
   Al ser iterativo, el framework de Scrum proporciona amplias oportunidades para obtener información y establecer expectativas en todo el ciclo de vida del proyecto. Esto asegura que los stakeholders del proyecto, así como el equipo, no sean tomados por sorpresa debido a requisitos mal comunicados.

3. **La propiedad del equipo reduce la estimación de riesgo**
   El Equipo Scrum hace estimaciones y se hace responsable de los elementos del Sprint Backlog, lo cual conduce a una estimación más precisa y a la entrega oportuna de los incrementos del producto.

4. **La transparencia reduce el riesgo de no detección**
   El principio de la transparencia en Scrum, en torno al cual se construye el framework, asegura que los riesgos se detecten y se comuniquen oportunamente, lo cual conduce a un mejor manejo y mitigación de riesgos. Por otra parte, al llevar a cabo reuniones de Scrum de Scrums, los impedimentos que un equipo enfrenta en la actualidad pueden considerarse como riesgos para otros equipos Scrum a futuro. Esto debe reconocerse en el Impediments Log actualizado.

5. **La entrega iterativa reduce el riesgo de inversión**
   La entrega continua de valor a lo largo del ciclo de vida del proyecto Scrum, como entregables potencialmente enviables, se crean después de cada sprint, reduciendo así el riesgo de la inversión para el cliente.

7.6 Riesgos en portafolios y programas

Mientras que algunos riesgos están específicamente relacionados con proyectos individuales, otros pueden tener su origen en los programas y portafolios, y generalmente serán administrados allí mismo. Sin embargo, los riesgos relacionados a un portafolio o programa también tendrán un impacto en los proyectos que forman parte del respectivo portafolio o programa. Durante la evaluación de riesgos en portafolios y programas —si se determina que el riesgo puede afectar un proyecto individual—, la información relevante sobre el riesgo debe ser comunicada al Product Owner y al Equipo Scrum.

Dependiendo de la gravedad o de la prioridad, cuando el equipo del programa o el portafolio comunica el riesgo que habrá de tener un impacto en un proyecto individual, el Equipo Scrum tal vez tenga que detenerse y volver a planificar el actual sprint para atender el riesgo. Para los riesgos de menor urgencia, el equipo puede continuar con el actual sprint y atender el riesgo en un sprint a futuro.

7.6.1 En el portafolio

1. Cuando se identifican los riesgos en el portafolio, el Portfolio Product Owner tenderá que capturarlos y evaluar la proximidad, probabilidad e impacto de cada riesgo identificado a fin de priorizarlo y determinar la respuesta adecuada para el portafolio.

2. El Portfolio Product Owner tendrá también que comunicar los riesgos a los stakeholders relevantes, a los equipos del programa y del proyecto. En algunos casos, el equipo del portafolio pudiera tener que asumir la responsabilidad de riesgos específicos.

7.6.2 En los programas

1. Cuando se identifican los riesgos en el programa, el Product Owner del programa debe ingresarlo en el Backlog Priorizado del Producto del riesgo ajustado, evaluar la proximidad, probabilidad e impacto de cada riesgo identificado a fin de priorizarlo y determinar la respuesta adecuada para los programas.

2. El Program Product Owner tendrá también que comunicar los riesgos a los stakeholders relevantes y a los equipos del proyecto. En algunos casos, el equipo del portafolio pudiera tener que asumir la responsabilidad de riesgos específicos.

La figura 7-6 muestra cómo los riesgos se pueden administrar dentro del flujo de Scrum tanto en los portafolios como en los programas.

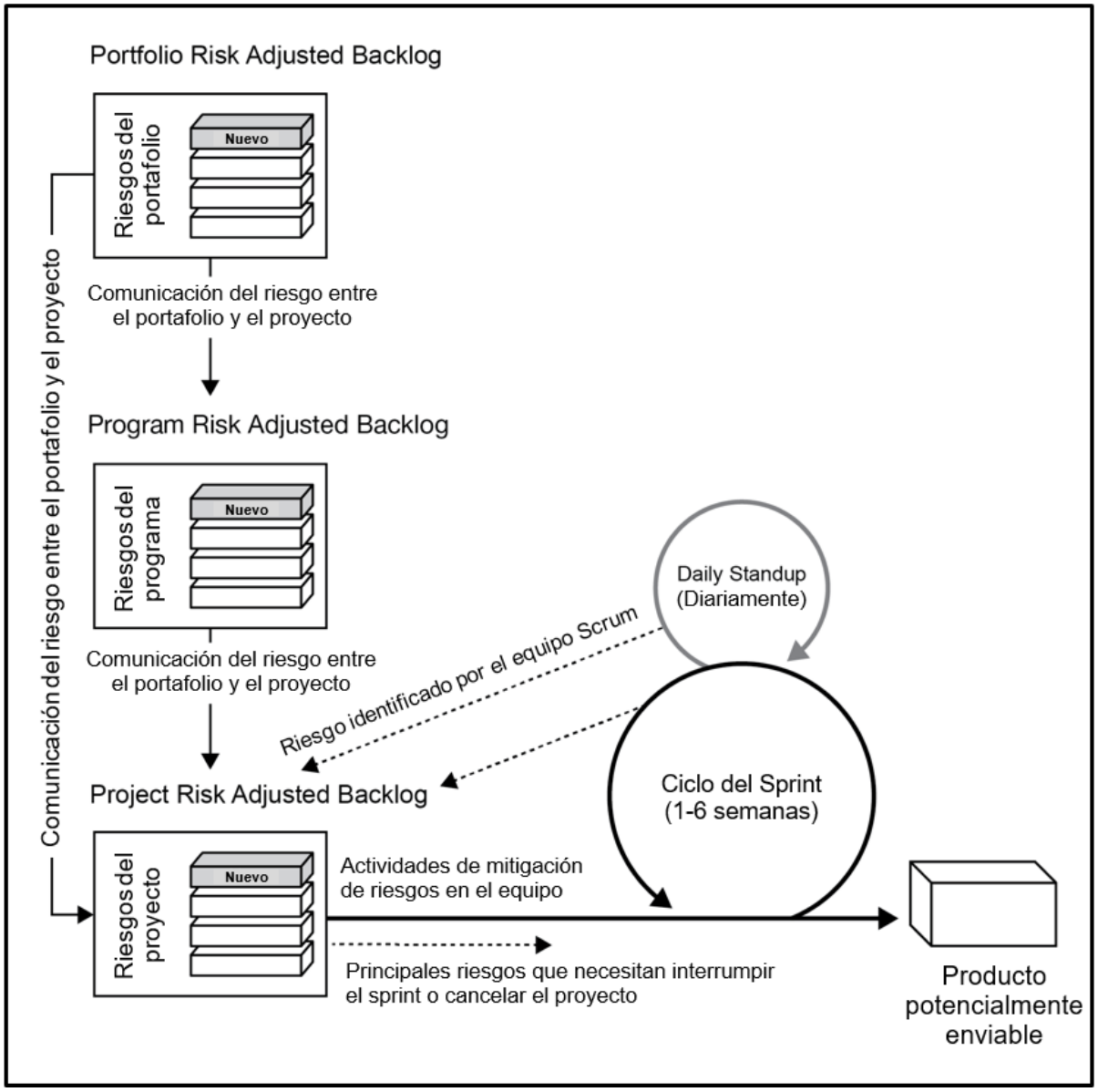


Figura 7-6: Manejo de riesgos en portafolios y programas

7.7 Resumen de responsabilidades

En Scrum, las actividades de gestión de riesgos se dividen entre varios roles donde algunas de las responsabilidades recaen sobre todos dentro del Equipo Scrum y donde el Scrum Master facilita el proceso.

Rol	Responsabilidades
Equipo Scrum	• Identifica los riesgos durante el desarrollo del producto durante el proceso de *Crear entregables*. • Implementa actividades de gestión de riesgo tal como lo recomiende el Product Owner.
Product Owner/ Chief Product Owner	• Captura y evalúa riesgos del proyecto • Prioriza y comunica los riesgos a los stakeholders relevantes y equipos del portafolio. • Se asegura de que los niveles de riesgo del proyecto estén dentro de los límites aceptables.
Scrum Master/ Chief Scrum Master	• Facilita al Equipo Scrum la identificación y escalamiento de riesgos.
Program Product Owner	• Captura y evalúa los riesgos de los programas • Priorizar y comunica los riesgos a los stakeholders relevantes y equipos del proyecto.
Program Scrum Master	• Facilita la identificación, evaluación y comunicación de los riesgos de los programas
Portfolio Product Owner	• Captura y evalúa los riesgos del portafolio • Prioriza y comunica los riesgos a los stakeholders relevantes, así como a los equipos del programa y del proyecto.
Portfolio Scrum Master	• Facilita la identificación, evaluación y comunicación de los riesgos del portafolio
Stakeholder(s)	• Interactúa con el equipo principal de Scrum para brindarle información sobre la gestión de riesgos que afectan el logro de los resultados esperados, así como los beneficios del proyecto.
Scrum Guidance Body	• Brinda una guía general para el procedimiento de gestión de riesgos para que se siga durante todo el proyecto.

Tabla 7-1: Resumen de las responsabilidades pertinentes a los riesgos

7.8 Scrum vs Gestión tradicional de proyectos

Scrum y la mayoría de los métodos tradicionales de gestión de proyectos definen los “riesgos” como “evento(s) inciertos que pudieran afectar de forma positiva o negativa los logros de los objetivos del proyecto”. Asimismo, los riesgos se identifican, se atienden, se planifican y se comunican continuamente.

En los modelos tradicionales de gestión de proyectos, se hace énfasis en la planificación inicial detallada para identificar, evaluar y determinar las respuestas de riesgo para todos los riesgos del proyecto. Durante la ejecución del proyecto, cualquier miembro del equipo del proyecto puede identificar los riesgos y el Project Manager o la dependencia encargada de gestión de proyectos o el personal de apoyo a los proyectos puede actualizarlos en la lista de riesgos o en el registro de riesgos. El Project Manager monitorea y controla con frecuencia todos los riesgos y por lo general identifica a individuos específicos en el equipo para que se hagan cargo de diferentes aspectos del riesgo.

En Scrum, cualquier miembro del equipo puede identificar riesgos y el Product Owner puede actualizar los riesgos identificados en el Backlog Priorizado del Producto del riesgo ajustado. Los principios de Scrum sobre el control de proceso empírico y de desarrollo iterativo permiten que el Equipo Scrum pueda identificar constantemente los riesgos y sumarlos al Backlog Priorizado del Producto donde tales riesgos se pueden priorizar con otras historias de usuarios existentes en el backlog antes mencionada, a efectos de mitigarlos en futuros sprints. El Equipo Scrum tiene responsabilidades colectivas para la gestión de todos los riesgos para el sprint.

8. INICIO

Este capítulo incluye los procesos relacionados con el inicio de un proyecto: *Crear la visión del proyecto, Identificar al Scrum Master y stakeholder(s)*, *Formar el Equipo Scrum*, *Desarrollar épica(s)*, *Crear el Backlog Priorizado del Producto* y *Realizar la planificación del lanzamiento*.

El inicio, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)* es aplicable a los siguientes:

- Portafolios, programas y/o proyectos de cualquier industria
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria, desde pequeños proyectos o equipos con tan sólo seis miembros del equipo, hasta proyectos grandes y complejos que cuentan con cientos de miembros por equipo.

A fin de facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas por asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deben esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un Equipo Scrum que trabaja en un sprint para producir entregables que pudieran ser parte de un proyecto más grande, de un programa o portafolio. La información sobre cómo escalar Scrum en grandes proyectos está disponible en el capítulo 13 y lo relacionado a escalar Scrum para la empresa se puede encontrar en el capítulo 14.

La figura 8-1 proporciona una visión general de los procesos de la fase de Inicio, que son los siguientes:

8.1 Crear la visión del proyecto—En este proceso, el caso de negocio del proyecto se revisa para crear una declaración de la visión del proyecto (*Project Vision Statement*) que servirá de inspiración y proporcionará un enfoque de todo el proyecto. En este proceso se identifica al Product Owner.

8.2 Identificar al Scrum Master y stakeholders(s)—En este proceso, se identifica al Scrum Master y a los stakeholders utilizando criterios de selección específicos.

8.3 Formar el Equipo Scrum—En este proceso, se seleccionan a los miembros del Equipo Scrum. Normalmente, el Product Owner es el responsable principal de la selección de los miembros del equipo, pero a menudo lo hace en colaboración con el Scrum Master.

8.4 Desarrollar de épica(s)—En este proceso, la Declaración de la visión del proyecto (*Project Vision Statement*) sirve como base para el desarrollo de épicas. Las reuniones de grupo de usuarios pueden llevarse a cabo para discutir la(las) épica(s) apropiada(s).

8.5 Crear el Backlog Priorizado del Producto—En este proceso, la épica(s) es refinada, detallada y luego priorizada para el Backlog Priorizado del Producto del proyecto. En este punto también se establecen los criterios de terminado.

8.6 Realizar la planificación del lanzamiento—En este proceso, el equipo principal de Scrum revisa las historias de usuario en el Backlog Priorizado del Producto para desarrollar un cronograma de planificación del lanzamiento, que es esencialmente un programa de implementación por fases que se puede compartir con los stakeholders del proyecto. La duración de los sprints también se determina en este proceso.

8.1 Crear la visión del proyecto

ENTRADAS
1. Caso de negocio del proyecto*
2. Program Product Owner
3. Program Scrum Master
4. Program Stakeholder(s)
5. Program Product Backlog
6. Proyecto de prueba
7. Prueba del concepto
8. Visión de la empresa
9. Misión de la empresa
10. Estudio del mercado
11. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Reunión de visión del proyecto*
2. Sesiones JAD
3. Análisis FODA
4. Análisis de brecha

SALIDAS
1. Product Owner identificado*
2. Declaración de la visión del proyecto*
3. Acta constitutiva del proyecto
4. Presupuesto del proyecto

8.2 Identificar al Scrum Master y stakeholder(s)

ENTRADAS
1. Product Owner *
2. Declaración de la visión del proyecto*
3. Program Product Owner
4. Program Scrum Master
5. Program Stakeholder(s)
6. Requerimientos de las personas
7. Disponibilidad y compromiso de las personas
8. Matriz de recursos organizacionales
9. Matriz de las destrezas requeridas
10. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Criterios de selección*
2. Asesoramiento de expertos en recursos humanos
3. Capacitación y costos de capacitación
4. Costos de recursos

SALIDAS
1. Scrum Master identificado*
2. Stakeholder(s) identificado*

8.3 Formar el Equipo Scrum

ENTRADAS
1. Product Owner*
2. Scrum Master*
3. Declaración de la visión del proyecto*
4. Requisitos de las personas
5. Disponibilidad y compromiso de las personas
6. Matriz de recurso organizacional
7. Matriz de las destrezas requeridas
8. Requerimientos de recursos
9. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Selección del equipo Scrum*
2. Asesoramiento de expertos en recursos humanos
3. Costos del personal
4. Capacitación y costos de capacitación
5. Costos de recursos

SALIDAS
1. Equipo Scrum identificado*
2. Substitutos
3. Plan de colaboración
4. Plan de formación del equipo

8.4 Desarrollar épica(s)

ENTRADAS
1. Equipo Principal de Scrum*
2. Declaración de la visión del proyecto*
3. Stakeholder(s)
4. Program Product Backlog
5. Solicitudes de cambios aprobadas
6. Solicitudes de cambios no aprobadas
7. Riesgos del portafolio y del programa
8. Leyes y regulaciones
9. Contratos aplicables
10. Información de proyectos previos
11. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Reuniones del grupo de usuarios*
2. Talleres de historias de usuario
3. Reuniones del grupo de enfoque
4. Entrevistas al usuario o cliente
5. Cuestionarios
6. Técnicas de identificación de riesgos
7. Experiencia del Scrum Guidance Body

SALIDAS
1. Épica(s)*
2. Prototipos*
3. Cambios aprobados
4. Riesgos identificados

8.5 Crear el Backlog Priorizado del Producto

ENTRADAS
1. Equipo principal de Scrum*
2. Épica(s)*
3. Prototipos*
4. Stakeholder(s)
5. Declaración de la visión del proyecto
6. Program Product Backlog
7. Requerimientos del negocio
8. Solicitudes de cambios aprobadas
9. Riesgos identificados
10. Contratos Aplicables
11. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Métodos de priorización de historias del usuario*
2. Talleres de historias del usuario
3. Planificación de valor
4. Técnicas de evaluación del riesgo
5. Estimación del valor del proyecto
6. Métodos de estimación de historias de usuario
7. Experiencia Scrum Guidance Body

SALIDAS
1. Backlog Priorizado del Producto*
2. Criterios de terminado*

8.6 Realizar la planificación del lanzamiento

ENTRADAS
1. Equipo principal de Scrum*
2. Stakeholder(s)*
3. Declaración de la visión del proyecto*
4. Backlog Priorizado del Producto*
5. Criterios de terminado*
6. Program Product Owner
7. Program Scrum Master
8. Program Product Backlog
9. Requerimientos del negocio
10. Calendario de días festivos
11. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Sesiones de planificación del lanzamiento*
2. Métodos de priorización del lanzamiento*

SALIDAS
1. Cronograma de planificación del lanzamiento*
2. Duración del sprint*
3. Clientes meta para el lanzamiento
4. Backlog Priorizado del Producto refinado

Figura 8-1: Información general sobre el Inicio

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 8-2 muestra las entradas obligatorias, herramientas y salidas para los procesos en la fase de inicio.

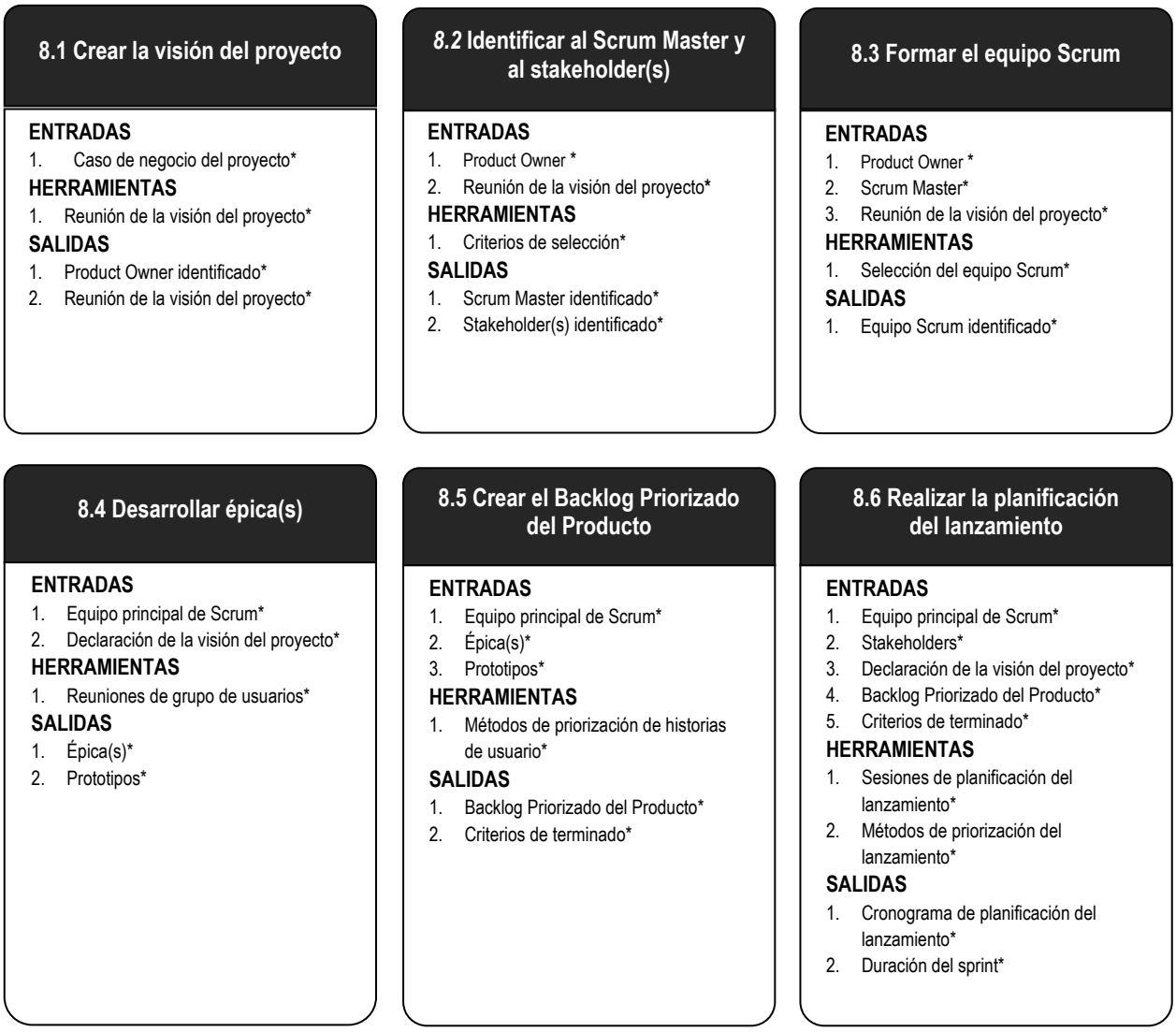


Figura 8-2: Resumen sobre el inicio (esenciales)

8.1 Crear la visión del proyecto

La figura 8-3 muestra todas las entradas, herramientas y salidas para el proceso de *Crear la visión del proyecto.*

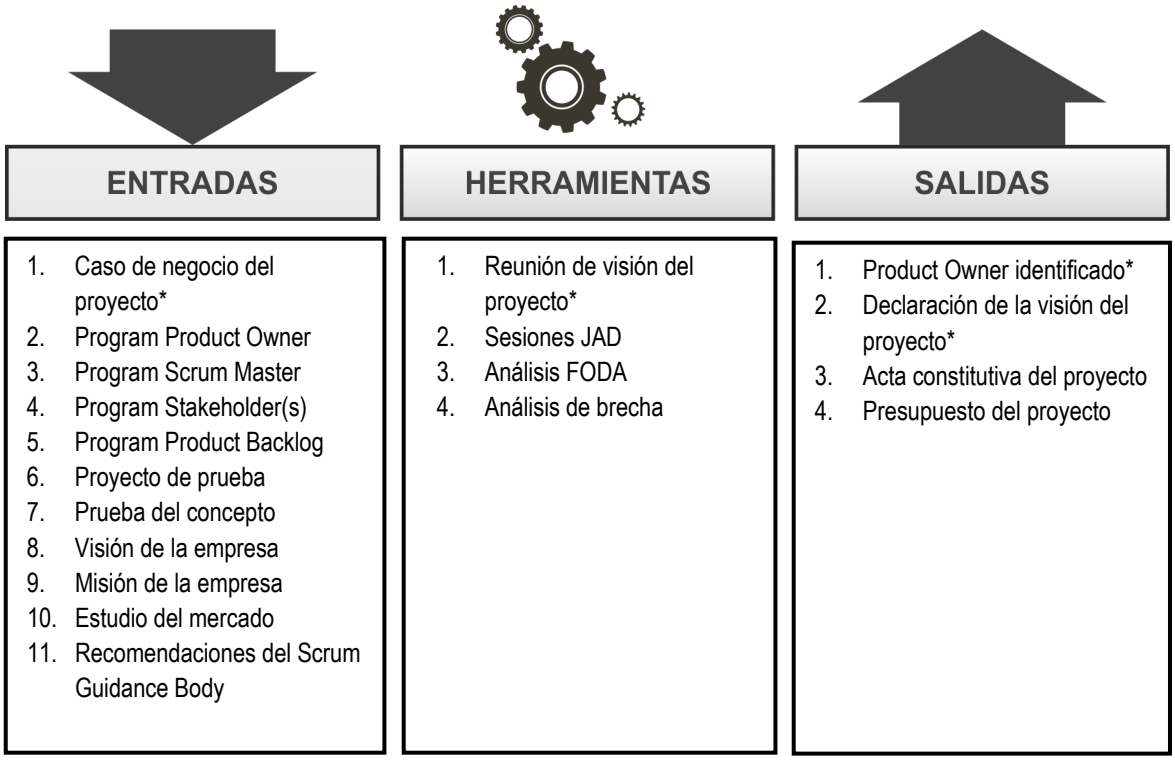


Figura 8-3: Crear la visión del proyecto – Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

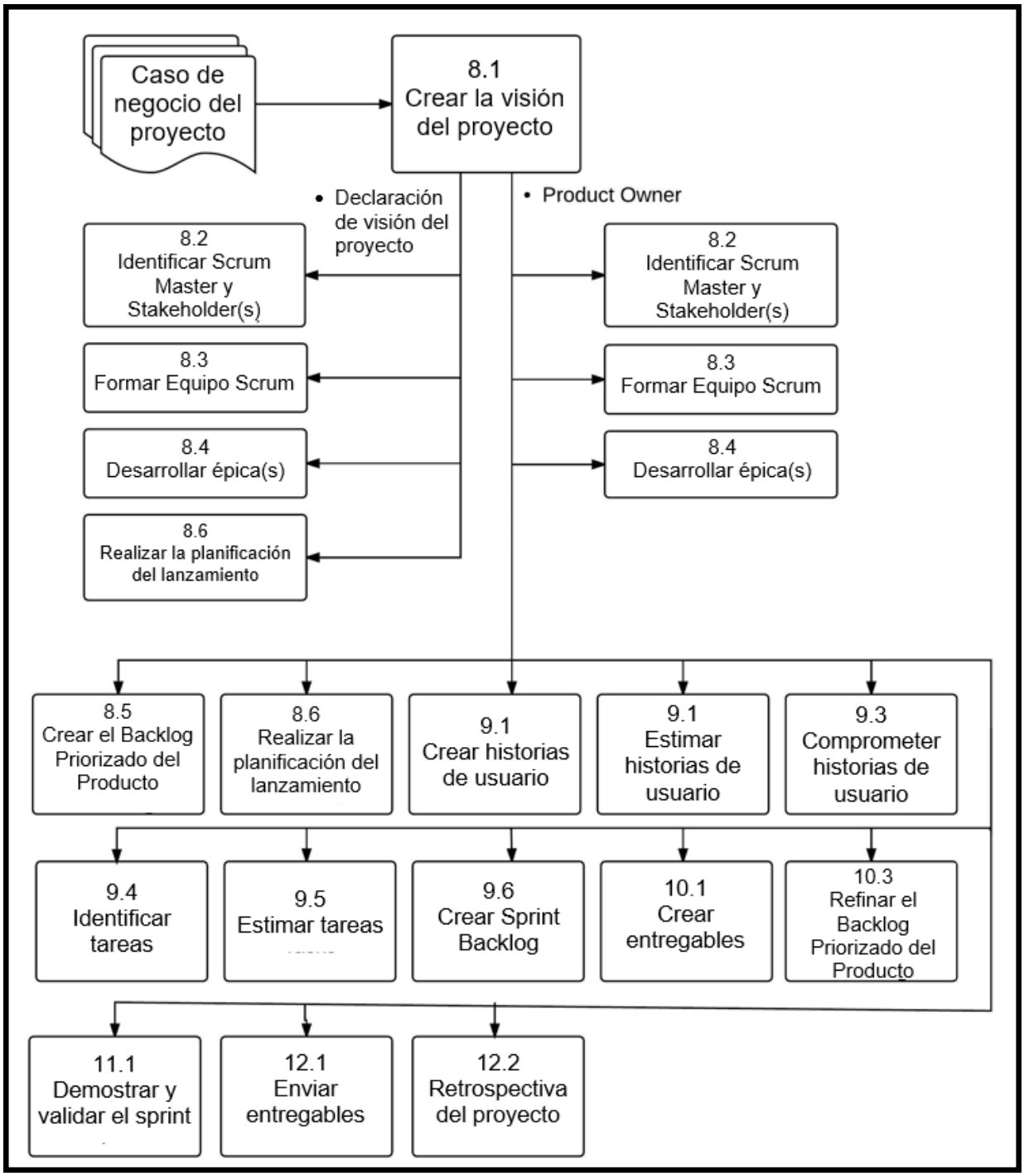


Figura 8-4: Crear la visión del proyecto - Diagrama del flujo de datos

8.1.1 Entradas

8.1.1.1 Caso de negocio del proyecto*

Un caso de negocio, o *Business Case*, puede ser un documento bien estructurado o simplemente una declaración verbal que expresa la razón para iniciar un proyecto. Puede ser formal y detallado, o informal y breve. Independientemente del formato, a menudo incluye información sustancial sobre los antecedentes del proyecto, los objetivos del negocio y los resultados deseados, un reporte de análisis FODA y de brecha, una lista de los riesgos identificados y las estimaciones de tiempo, esfuerzo y costo.

El proyecto se inicia con la presentación del caso de negocio del proyecto. Un caso de negocio se le presenta a los stakeholders y patrocinadores. De esta manera, los stakeholders comprenden los beneficios de negocio esperados de tal proyecto y los patrocinadores confirman que van a proporcionar los recursos financieros para el proyecto.

8.1.1.2 Program Product Owner

Descrito en la sección 3.7.3.1.

8.1.1.3 Program Scrum Master

Descrito en la sección 3.7.3.2.

8.1.1.4 Program Stakeholder(s)

Program Stakeholder(s) es un término colectivo que incluye a los clientes, usuarios y patrocinadores o *sponsors* de un programa. Estos influyen en todos los proyectos del programa durante todo el desarrollo del proyecto. El Program stakeholder también puede ayudar a definir la visión del proyecto y a proporcionar orientación en relación con el valor del negocio.

El Program stakeholder(s) debe interactuar con los Portfolio Stakeholders para asegurar la alineación del programa con las metas y objetivos del portafolio. También participan en el nombramiento del (los) stakeholder(s) para proyectos individuales y se aseguran de que la visión, los objetivos, los resultados y los lanzamientos de los proyectos individuales en el programa estén alineados con los del programa.

8.1.1.5 Program Product Backlog

El Program Product Owner desarrolla el Program Product Backlog que contiene una lista priorizada de requerimientos de alto nivel empresarial, así como requerimientos del proyecto, de preferencia escritos en forma de granes elementos del Program Backlog. Después las refina el Product Owner de proyectos

individuales a medida que elaboran y priorizan los backlogs de sus proyectos. Dichos Backlogs Priorizados del Producto cuentan con historias de usuario mucho más pequeñas pero detalladas que pueden ser estimadas y comprometidas por equipos individuales de Scrum.

El Program Product Owner refina constantemente el Program Product Backlog para garantizar que los nuevos requerimientos de negocio se añadan, y que los requisitos existentes estén debidamente documentados y priorizados. Esto asegura que a los requisitos más valiosos en el cumplimiento de los objetivos del programa se les esté dando alta prioridad y que los restantes reciban una prioridad más baja.

El Program Product Backlog creado presenta una imagen más grande de todos los proyectos que forman parte del programa. Por lo tanto, puede servir de orientación en relación con las metas del proyecto, el alcance, los objetivos y los beneficios esperados del negocio.

8.1.1.6 Proyecto de prueba

Si es factible, se puede llevar a cabo una demostración o proyecto de prueba como experimento para predecir y evaluar la viabilidad, el tiempo, el costo, los riesgos y los posibles efectos del proyecto. Esto ayuda a evaluar el ambiente de la práctica y guía el diseño real del proyecto antes del inicio del proyecto a gran escala.

8.1.1.7 Prueba de concepto

La prueba del concepto demuestra y verifica que la idea detrás del proyecto actual sea potencialmente viable en la vida real. A menudo esto se hace en la forma de un prototipo diseñado para determinar la viabilidad técnica y financiera, para ayudar a comprender los requerimientos y para ayudar en la evaluación de las decisiones de diseño al principio del proceso. Sin embargo, la prueba del concepto no necesita representar necesariamente los verdaderos entregables del proyecto.

8.1.1.8 Visión de la empresa

Entender la visión de la empresa ayuda a que el proyecto mantenga su enfoque en los objetivos de la organización y el futuro probable de la empresa. El Product Owner puede guiarse por la visión de la compañía para crear la declaración de la visión del proyecto.

8.1.1.9 Misión de la empresa

La misión de la empresa ofrece un framework para formular las estrategias de la empresa y orienta la toma

de decisiones generales en la compañía. Dicha misión debe enmarcarse de tal manera que su cumplimiento ayude a la organización a llevar a cabo su misión.

8.1.1.10 Estudio de mercado

El estudio de mercado es la investigación organizada, la recopilación, la comparación y el análisis de datos relacionados con las preferencias de los clientes sobre los productos. Generalmente incluye numerosos datos sobre las tendencias y la segmentación del mercado, así como los procesos de comercialización. El estudio de mercado podría incluir también un estudio analítico de los competidores, lo que proporciona una mejor comprensión de las fortalezas y debilidades de los competidores y puede ayudar a quienes toman decisiones a formular productos mejor posicionados.

8.1.1.11 Recomendaciones del Scrum Guidance Body

El Scrum Guidance Body (SGB) es un rol opcional. Por lo general, se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y demás parámetros claves de la organización. Estos objetivos guían el trabajo del Product Owner, del Scrum Master y del Equipo Scrum. El Scrum Guidance Body ayuda también a captar las mejores prácticas que deben utilizarse en todos los proyectos Scrum en la organización.

El Scrum Guidance Body no toma decisiones relacionadas con el proyecto. En cambio, actúa como estructura consultiva o asesora para todos los niveles jerárquicos la organización del proyecto: el portafolio, el programa y el proyecto. Los equipos Scrum tienen la opción de pedir recomendaciones al Scrum según sea necesario.

Es importante asegurarse de que la visión del proyecto esté en línea con las recomendaciones proporcionadas por el Scrum Guidance Body y que los procesos cumplan con las normas y directrices establecidas por dicho organismo.

8.1.2 Herramientas

8.1.2.1 Reunión de la visión del proyecto*

La reunión de la visión del proyecto es una reunión con el(los) Program Stakeholder(s), el Program Product Owner, el Program Scrum Master y el Chief Product Owner. Esta ayuda a identificar el contexto empresarial, los requerimientos de negocio y las expectativas de los stakeholders a fin de desarrollar una declaración de la visión del proyecto eficaz. Scrum cree en la participación y colaboración cercana con

todos los representantes de las empresas para obtener su sentido de compromiso con el proyecto y para ofrecer un valor más significativo.

8.1.2.2 Sesiones JAD

La sesión Joint Application Design (JAD), o sesión de diseño de aplicación conjunta, es una técnica de recopilación de requisitos. Es taller impartido y altamente estructurado que acelera el proceso de *Crear la visión del proyecto*, ya que permite al(los) stakeholder(s) y a otras personas que toman decisiones llegar a un consenso sobre el alcance, los objetivos y otras especificaciones del proyecto.

Esta técnica consiste en métodos para aumentar la participación del usuario, lo que acelera el desarrollo y la mejora de las especificaciones. El(los) stakeholder(s) relevante(s) del programa, el Program Product Owner, el Program Scrum Master y el Chief Product Owner pueden reunirse para delinear y analizar los resultados de negocio deseados y visualizar su visión para el proyecto Scrum.

8.1.2.3 Análisis FODA

El FODA (SWOT en inglés) es un enfoque estructurado para la planificación que ayuda a evaluar las **F**ortalezas**, O**portunidades, **D**ebilidades y **A**menazas relacionadas con un proyecto. Este tipo de análisis ayuda a identificar tanto los factores internos como los externos que podrían afectar el proyecto. Las fortalezas y debilidades son factores internos, mientras que las oportunidades y amenazas son factores externos. La identificación de estos factores ayuda a los stakeholders y a aquellos que toman decisiones a finalizar los procesos, las herramientas y las técnicas que se utilizarán para lograr los objetivos del proyecto. El análisis FODA permite la identificación temprana de las prioridades, los cambios potenciales y riesgos.

8.1.2.4 Análisis de brecha

El análisis de brecha (conocido en inglés como *Gap Analysis*) es una técnica que se utiliza para comparar el estado actual con el estado deseado. En una organización, esto involucra determinar y documentar la diferencia entre las capacidades actuales del negocio y el conjunto final de capacidades deseado. Normalmente, se inicia un proyecto para que una organización alcance una situación deseada, por lo que llevar a cabo un análisis de brecha ayudaría a quienes toman decisiones a determinar la necesidad de un proyecto.

Los principales pasos a seguir en el análisis de brecha se presentan en la figura 8-5.

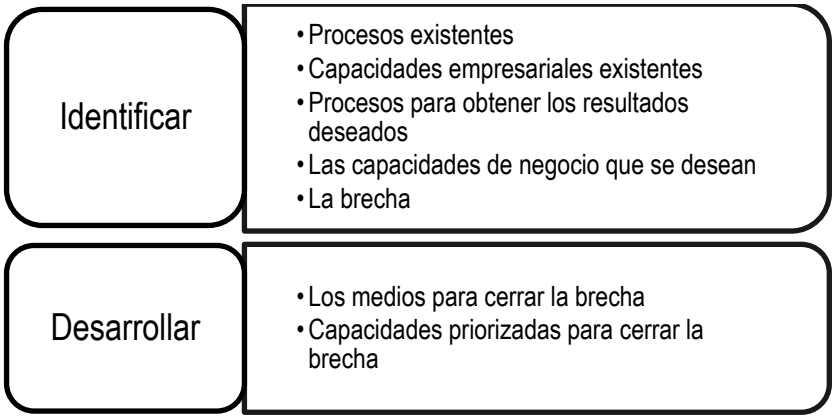


Figura 8-5: El proceso de Análisis de brecha

8.1.3 Salidas

8.1.3.1 Product Owner identificado*

Uno de los resultados de este proceso es identificar al Product Owner. El Product Owner es la persona responsable de lograr el máximo valor empresarial para el proyecto. También está a cargo de la articulación de requerimientos por parte de los clientes y de mantener la justificación del negocio para el proyecto. El Product Owner representa la voz del cliente.

El rol del Product Owner se describe en más detalle en la sección 3.4.

8.1.3.2 Declaración de la visión del proyecto*

El resultado clave del proceso de *Crear la visión del proyecto* es declaración bien estructurada de la visión del proyecto. Una buena visión del proyecto explica las necesidades empresariales que el proyecto busca cumplir en vez de cómo habrá cumplir con la necesidad.

La declaración de la visión del proyecto (*Project Vision Statement*) no debe ser demasiado específica y debe dejar espacio a la flexibilidad. Es posible que el conocimiento actual sobre el proyecto esté basado en suposiciones cambien conforme avanza el proyecto, por lo que es importante que la visión del proyecto sea lo suficientemente flexible como para adaptarse a estos cambios. La visión del proyecto debe centrarse en el problema y no en la solución.

Ejemplo:

VMFoods, una cadena de supermercados, quiere expandirse con un portal de comercio electrónico en línea y se ha comunicado con su empresa para crear el producto.

Visión del proyecto: Desarrollar una herramienta fácil de usar y un canal de ventas en línea para VMFoods que sea estéticamente agradable.

8.1.3.3 Acta constitutiva del proyecto (*Project Charter*)

Un acta constitutiva del proyecto (conocida en inglés como *Project Charter*) es una declaración oficial de los objetivos y resultados deseados del proyecto. En varias organizaciones, el acta constitutiva del proyecto es el documento que autoriza el proyecto oficial y formalmente, dándole al equipo la autoridad por escrito para iniciar el proyecto.

8.1.3.4 Presupuesto del proyecto

El presupuesto del proyecto es un documento financiero que incluye el costo del personal, materiales y otros gastos relacionados en un proyecto. Por lo general, dicho presupuesto es autorizado por el(los) patrocinador(es), o *sponsors* para asegurar que haya suficientes fondos disponibles. Una vez que se autoriza, el Product Owner y el Scrum Master estarán involucrados regularmente con la gestión del presupuesto del proyecto para garantizar que exista la disponibilidad necesaria de personal y recursos para el proyecto.

8.2 Identificar al Scrum Master y stakeholder(s)

La figura 8-6 muestra todas las entradas, las herramientas y las salidas para el proceso de *Identificar al Scrum Master y stakeholder(s)*

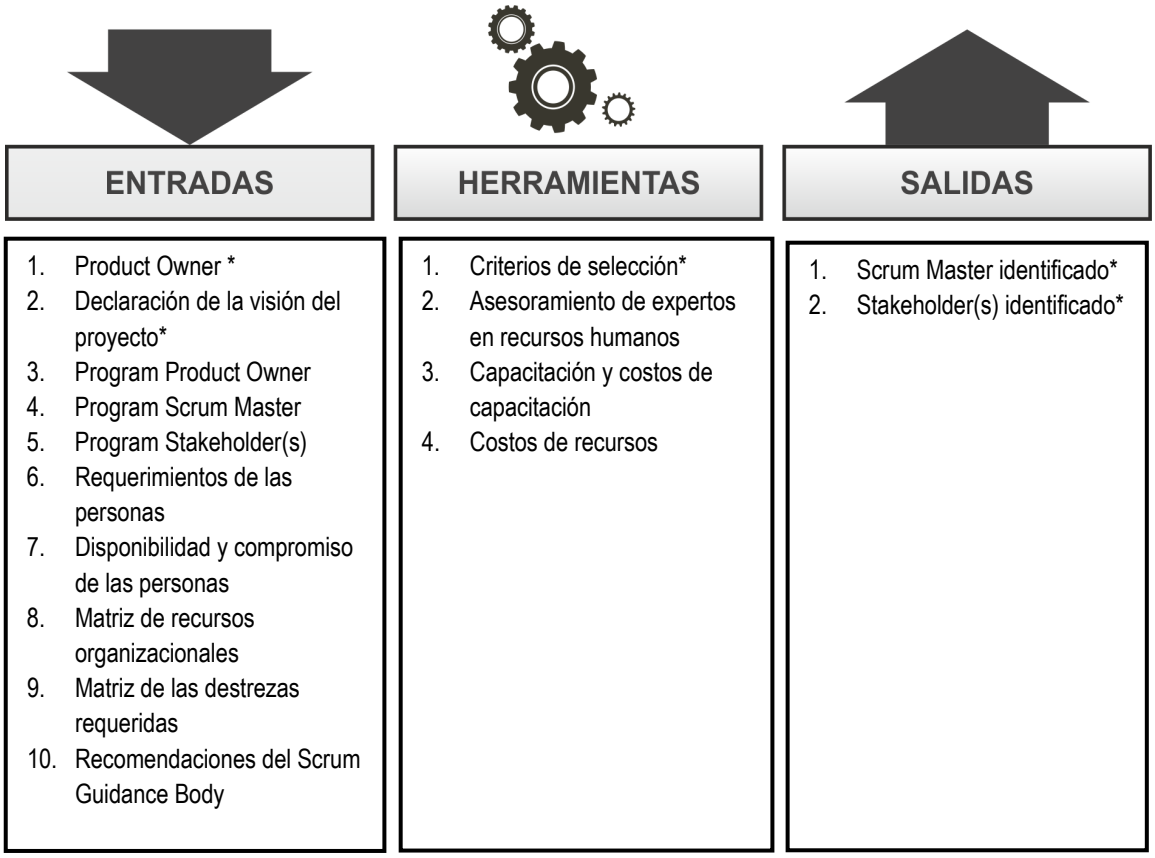


Figura 8-6: Identificar al Scrum Master y stakeholder(s)—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

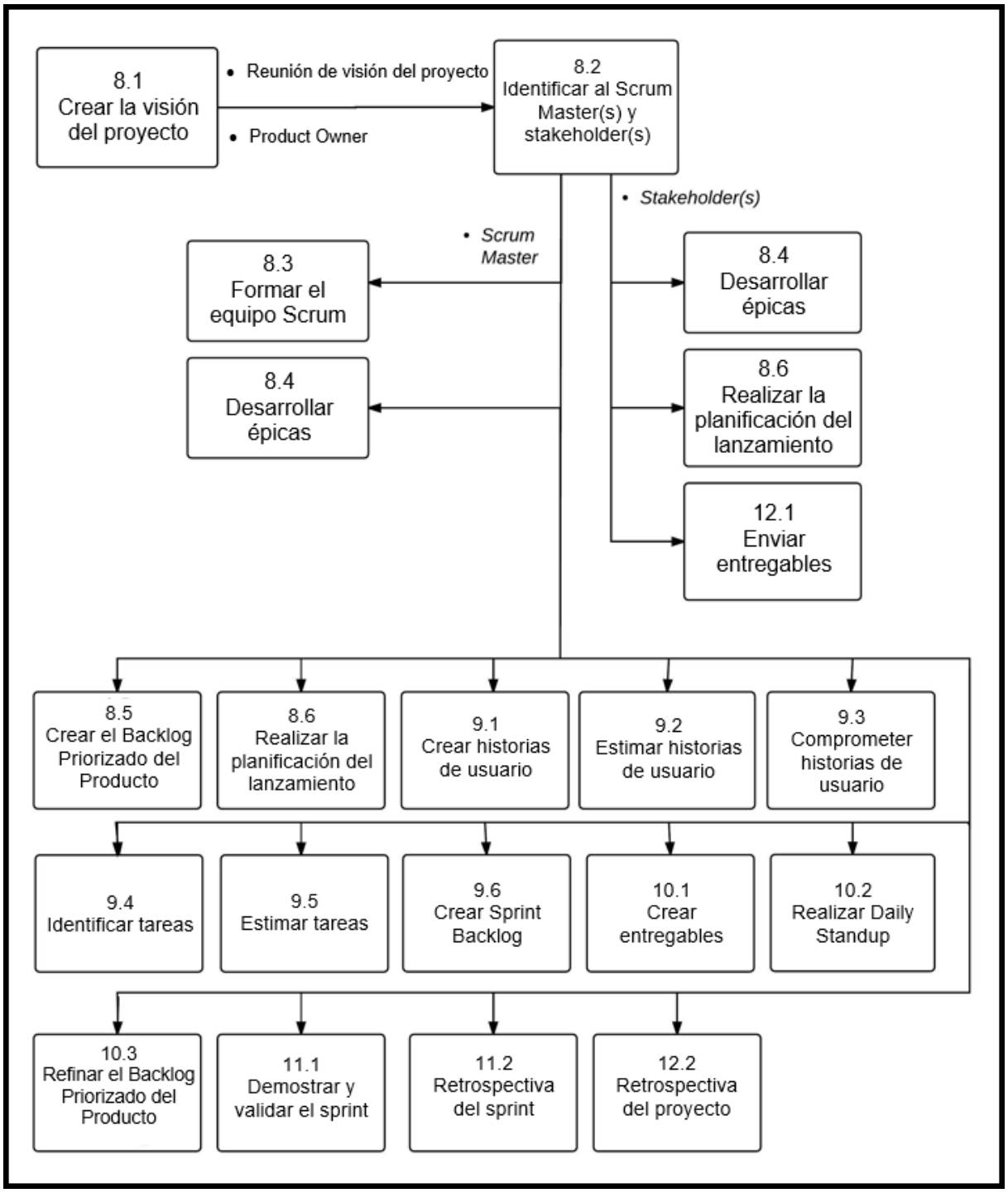


Figura 8-7: Identificar al Scrum Master y stakeholder(s)—Diagrama de flujo de datos

8.2.1 Entradas

8.2.1.1 Product Owner*

Descrito en la sección 8.1.3.1.

8.2.1.2 Declaración de la visión del proyecto*

Descrito en la sección 8.1.3.2.

8.2.1.3 Program Product Owner

Descrito en la sección 3.7.3.1.

8.2.1.4 Programa Scrum Master

Descrito en la sección 3.7.3.2

8.2.1.5 Program Stakeholder(s)

Descrito en la sección 8.1.1.4.

8.2.1.6 Requerimientos de las personas

Los requerimientos de las personas es uno de los pasos iniciales en la selección del Scrum Master y stakeholder(s). Es importante documentar los roles y responsabilidades de todos los que se estarían involucrados en la realización de las tareas del proyecto. Esto incluye a todas las personas involucradas en el proyecto, en cualquier capacidad, independientemente si su rol es central o no.

Por lo general, el Product Owner o el Scrum Master colaboran con el departamento de recursos humanos de la empresa para determinar y finalizar los requerimientos de personal de un proyecto.

8.2.1.7 Disponibilidad y compromiso de las personas

Antes de seleccionar al Scrum Master y stakeholder(s) debe confirmarse su disponibilidad. Solamente deben seleccionarse los miembros del equipo que estarán disponibles y que puedan comprometerse plenamente con el proyecto. La disponibilidad y el compromiso de las personas se representa comúnmente en forma de un calendario que muestre cuándo estarán disponibles los recursos humanos para trabajar a través de toda la duración del proyecto.

Para ser eficaces, los equipos Scrum deberían tener idealmente de seis a diez miembros; y la sustitución de las personas o el cambio de los miembros del equipo no es recomendable en un equipo principal de Scrum. Por lo tanto, es importante contar con personas en un equipo principal de Scrum que estén disponibles y plenamente comprometidas con el proyecto.

8.2.1.8 Matriz de recursos organizacionales

La matriz de recursos organizacionales es una representación jerárquica de una combinación de una estructura de organización funcional y una estructura organizativa proyectada. Las organizaciones matriciales reúnen a miembros de varios equipos del proyecto de diferentes departamentos funcionales, tales como alguien de tecnología, finanzas, comercialización, ventas, manufactura y otros departamentos para así crear equipos multifuncionales.

Los miembros del equipo en una organización matriz cumplen dos objetivos: funcionales y de proyecto. Los miembros del equipo son dirigidos por el(los) Product Owner(s) con respecto a las actividades de proyectos, mientras que los directores funcionales realizan actividades de gestión relacionadas con sus departamentos, como las evaluaciones de desempeño y la aprobación para ausentarse del trabajo.

8.2.1.9 Matriz de requerimientos de habilidades

La matriz de requerimientos de habilidades, también llamada framework de competencia, se utiliza para evaluar las carencias de habilidades y los requisitos de formación para los miembros del equipo. Una matriz de habilidades traza las habilidades, capacidades y el nivel de interés de los miembros del equipo en el uso de dichas habilidades y capacidades en un proyecto. Al utilizar esta matriz, la organización puede evaluar los vacíos de habilidades en los miembros del equipo e identificar a los empleados que van a necesitar más formación en un área o competencia particular.

8.2.1.10 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

8.2.2 Herramientas

8.2.2.1 Criterios de selección*

La selección adecuada de Scrum Master(s) y la identificación del(los) stakeholder(s) relevante(s) es crucial para el éxito de cualquier proyecto. En algunos proyectos puede haber condiciones previas que estipulen a determinados miembros del equipo y sus roles.

Cuando hay flexibilidad en la selección del(los) Scrum Master(s), se deben considerar los siguientes criterios de selección:

1. *Habilidades para resolver problemas*—Es uno de los principales criterios a considerar al seleccionar al(los) Scrum Master(s). El(los) Scrum Master(s) debe(n) tener las habilidades y la experiencia necesarias para ayudar a eliminar cualquier impedimento que enfrente el Equipo Scrum.
2. *Disponibilidad*—El Scrum Master debe estar disponible para programar, supervisar y organizar varias reuniones, incluyendo la reunión de planificación de del lanzamiento, el Daily Standup y otras reuniones relacionadas al sprint.
3. *Compromiso*—El Scrum Master debe estar muy comprometido a fin de asegurar que el Equipo Scrum cuente con un ambiente laboral conductivo para garantizar la entrega exitosa de proyectos Scrum.
4. *Estilo de liderazgo servicial*—Para más detalles, consulte la sección 3.10.4.1

En la identificación del(los) stakeholder(s) es importante recordar que estos incluyen a todos los clientes, usuarios y patrocinadores, quienes a menudo interactúan con el Product Owner, Scrum Master y Equipo Scrum para proveer entradas y facilitar la creación de los productos del proyecto. Los stakeholders influyen en el proyecto a lo largo de su ciclo de vida.

8.2.2.2 Asesoramiento de expertos recursos humanos

El asesoramiento de expertos de recursos humanos puede ser útil en la identificación del Scrum Master y stakeholder(s). El departamento de recursos humanos posee un conocimiento específico sobre los empleados de una organización y las diversas técnicas que pueden ayudar a identificar a un Scrum Master y stakeholder(s).

8.2.2.3 Capacitación y costos de capacitación

Scrum es un framework radicalmente diferente a los métodos tradicionales de gestión de proyectos. Los miembros del equipo no siempre pueden poseer los conocimientos o habilidades necesarias para trabajar en el ambiente de Scrum. El Product Owner debe evaluar las necesidades de capacitación de los miembros potenciales del equipo y facilitar la capacitación para eliminar brechas de conocimiento del equipo. El

Product Owner generalmente es el responsable de la evaluación y la selección de los miembros del equipo, pero a menudo lo hace en consulta con el Scrum Master, quien puede tener un conocimiento adicional de los recursos por trabajar con ellos en otros proyectos.

Se debe proporcionar una capacitación adecuada a los miembros del Equipo Scrum, tanto antes del inicio del trabajo y a medida que trabajan en sus proyectos. Los miembros del Equipo Scrum también deben estar dispuestos a aprender de los demás y de quienes tienen más experiencia en el equipo.

8.2.2.4 Costo de los recursos

Una de las principales consideraciones en la selección de las personas tiene que ver con las ventajas y desventajas relacionadas con el nivel de experiencia en comparación al salario. Hay otros factores relacionados a las personas que influyen en el costo que también se deben considerar. Idealmente, el(los) Scrum Master(s), los miembros del equipo y el(los) stakeholder(s) deben estar co-ubicados para que puedan comunicarse con frecuencia y facilidad. Si la co-ubicación no es posible y hay equipos en diferentes zonas geográficas, se deberán destinar recursos adicionales para facilitar la comunicación, el entendimiento de las diferencias culturales, para sincronizar el trabajo y fomentar el intercambio de conocimientos.

8.2.3 Salidas

8.2.3.1 Scrum Master identificado*

Un Scrum Master es un facilitador y un líder servicial que se asegura de que el Equipo Scrum cuente con un ambiente propicio para completar con éxito el proyecto. El Scrum Master guía, facilita y les enseña prácticas de Scrum a todos los involucrados en el proyecto; elimina impedimentos que enfrenta el equipo y se asegura de que se estén siguiendo los procesos de Scrum. Es responsabilidad del Product Owner identificar al Scrum Master en un proyecto Scrum.

El papel de Scrum Master se describe con más detalle en la sección 3.4

8.2.3.2 Stakeholder(s) identificado(s)*

Stakeholder(s) que es un término colectivo que incluye a clientes, usuarios y patrocinadores que con frecuencia interactúan con el equipo principal de Scrum e influyen en el proyecto durante todo el proceso de desarrollo de productos. Los beneficios colectivos que producto el proyecto es para los stakeholders.

La función del stakeholder se describe en la sección 3.3.2.

8.3 Formar el Equipo Scrum

La figura 8-8 muestra todas las entradas, las herramientas y las salidas para el proceso de *Formar el Equipo Scrum*.

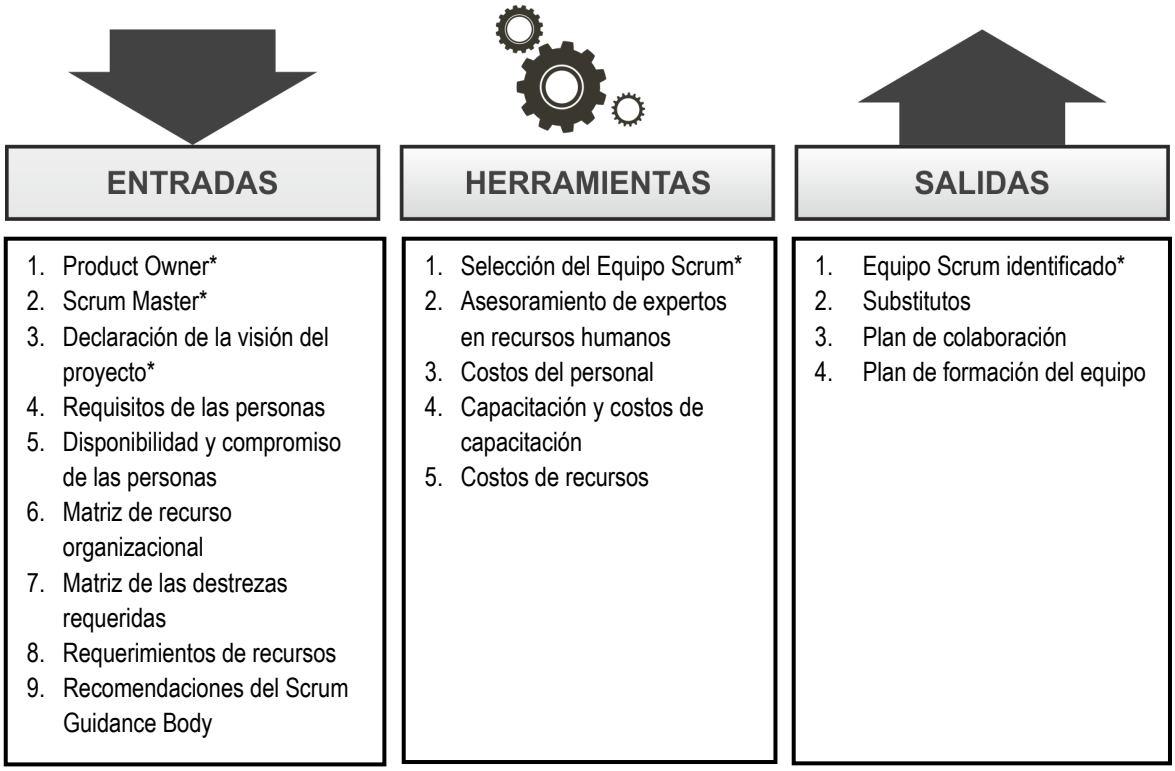


Figura 8-8: Formar el Equipo Scrum—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

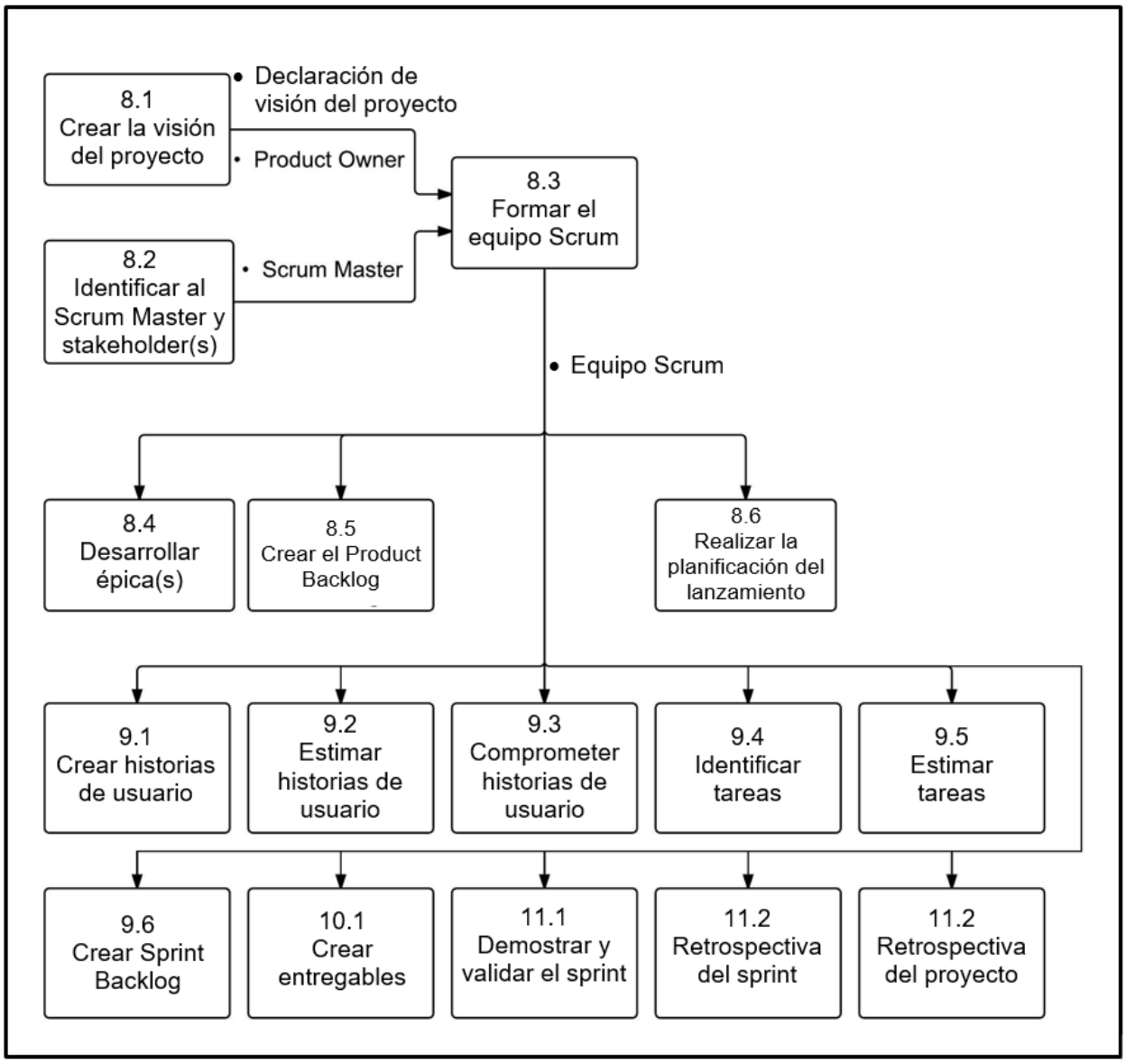


Figura 8-9: Formar el Equipo Scrum—Diagrama de flujo de datos

8.3.1 Entradas

8.3.1.1 Product Owner*

Descrito en la sección 8.1.3.1.

8.3.1.2 Scrum Master*

Descrito en la sección 8.2.3.1.

8.3.1.3 Declaración de la visión del proyecto*

Descrito en la sección 8.1.3.2.

8.3.1.4 Requerimientos de las personas

Descrito en la sección 8.2.1.8.

8.3.1.5 Disponibilidad y compromiso de las personas

Descrito en la sección 8.2.1.9.

8.3.1.6 Matriz de recursos organizacionales

Descrito en la sección 8.2.1.10.

8.3.1.7 Matriz de requerimientos de habilidades

Descrito en la sección 8.2.1.11.

8.3.1.8 Requisitos de recursos

Estos requisitos incluyen todos los recursos, humanos o no, necesarios para que el Equipo Scrum trabaje eficazmente. Estos recursos incluyen infraestructura de oficinas, espacios de reunión, equipos de trabajo, Scrumboards, etc. En el caso de los equipos virtuales deben de ponerse a consideración recursos adicionales, tales como herramientas de colaboración, videoconferencia, repositorios de documentos compartidos, servicios de traducción, etc.

8.3.1.9 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

8.3.2 Herramientas

8.3.2.1 Selección del Equipo Scrum*

El Equipo Scrum es la base de cualquier proyecto de Scrum y contar con los miembros adecuados es vital para la entrega exitosa de los proyectos Scrum. Los miembros del Equipo Scrum son generalistas/ especialistas, ya que cuentan con conocimiento en diversos campos y son expertos en al menos uno. Más allá de la experiencia en la materia, son las habilidades interpersonales de cada miembro del equipo lo que determinará el éxito de los equipos auto-organizados.

Los miembros ideales del Equipo Scrum son independientes, auto-motivados, se enfocan en el cliente y tienen un alto sentido de responsabilidad y colaboración. El equipo debe ser capaz de fomentar un ambiente de reflexión independiente y de tomar decisiones con el fin de extraer los mayores beneficios de la estructura.

8.3.2.2 Asesoramiento de expertos en recursos humanos

El asesoramiento de expertos de gerentes de recursos humanos puede ser valiosa al formar un Equipo Scrum. Este departamento posee un conocimiento especializado sobre los empleados de una organización y de las numerosas técnicas que pueden ayudar a los Product Owners, Scrum Masters y a los patrocinadores a identificar a los mejores miembros del equipo.

8.3.2.3 Costos del personal

Todos los costos asociados con los requisitos del personal deben evaluarse, analizarse, aprobarse y presupuestarse.

8.3.2.4 Capacitación y costos de capacitación

Los integrantes del Equipo Scrum tal vez no cuenten con las habilidades o conocimientos necesarios para llevar a cabo tareas especializadas. En tal caso, el Product Owner debe evaluar las necesidades de capacitación de los miembros potenciales del equipo y proporcionarles capacitación cuando se determine que faltan habilidades o conocimiento.

Para que una implementación de Scrum sea realmente eficaz, debe haber un alto nivel de conocimiento dentro de la organización sobre los principios y valores de Scrum. Este conocimiento ayudará a la ejecución exitosa de Scrum. El Equipo Scrum debe estar capacitado en las prácticas de Scrum y el Scrum Master debe desempeñar el rol de capacitador de este tema para el equipo. Dado a que la planificación de los sprints es un factor muy importante para el éxito, la capacitación ayudará a los miembros del equipo a entender cómo discutir e identificar metas alcanzables en un sprint. El Scrum Master tiene que aprovechar lo mejor del Equipo Scrum motivándolos y facilitando el proceso de desarrollo. Con la capacitación correcta, el Scrum Master puede ayudar al equipo a articular los problemas y desafíos que puedan enfrentar. Normalmente, cualquier problema o conflicto en el equipo lo resuelve el mismo equipo con la enseñanza y la ayuda del Scrum Master, según sea necesario. El Scrum Master debe atender problemas como son la baja moral o la falta de coordinación dentro del equipo. Es responsable de la eliminación de impedimentos en el equipo. Cuando sea necesario, el Scrum Master puede presentarle estos problemas externos e impedimentos a la gerencia para eliminarlos o resolverlos.

Los costos de formación y capacitación se abordan en el proceso *Identificar al Scrum Master y stakeholder(s)* en la sección 8.2.2.3.

8.3.2.5 Costo de los recursos

Los costos asociados a los requisitos no relacionados con las personas deben ser evaluados, analizados, autorizados y presupuestados. Un recurso, en el ambiente del proyecto, es todo aquello que se usa para realizar una tarea o actividad, incluyendo, pero sin limitarse a, equipos, materiales, servicios externos y el espacio físico.

8.3.3 Salidas

8.3.3.1 Equipo Scrum identificado*

El Equipo Scrum, también conocido como *equipo de desarrollo*, es un grupo o equipo de personas responsables de entender los requerimientos especificados por el Product Owner, la estimación de historias de usuario y la creación definitiva de los entregables del proyecto. Los equipos Scrum son interfuncionales y auto-organizados. El equipo decide la cantidad de trabajo al que se comprometerá en un sprint y determina la mejor manera de realizar las tareas. El equipo se compone de miembros interfuncionales que llevan a cabo todo el trabajo necesario para la creación de entregables potencialmente enviables, incluyendo el desarrollo, pruebas, garantía de calidad, etc.

La identificación del Equipo Scrum es responsabilidad del Product Owner, por lo general en consulta con el Scrum Master.

El rol del Equipo Scrum se describe con más detalle en la sección 3.6.

8.3.3.2 Substitutos (*Back-up Persons*)

Al seleccionar a los equipos, otro aspecto importante es la creación de substitutos para cada miembro del Equipo Scrum. Aunque la disponibilidad y compromiso por los miembros del equipo se confirman por adelantado, pueden surgir problemas tales como una enfermedad, emergencia familiar o simplemente el hecho de que un miembro deje de ser parte de la organización. Los equipos Scrum trabajan en pequeños grupos de seis a diez personas. El contar personas de respaldo, o *back-up*, asegura que no haya una disminución importante en la productividad debido a la falta de un miembro del equipo.

8.3.3.3 Plan de colaboración

La colaboración es un elemento muy importante en Scrum. La planificación de cómo interactúan y colaboran los tomadores de decisiones, los stakeholder y los miembros del equipo es de suma importancia. El plan de colaboración es una salida opcional que puede ser formal o informal. En ocasiones puede ser simplemente un entendimiento verbal entre los diversos stakeholders, ya en que Scrum se evita toda la documentación innecesaria. Sin embargo, para proyectos más grandes y complejos, especialmente aquellos con equipos en diferentes áreas geográficas, puede ser necesario poner en marcha un acuerdo más formal. El plan puede abordar la forma en que los miembros de equipo principal de Scrum, stakeholder(s) y otras personas involucradas en el proyecto Scrum, se comunican y colaboran durante todo el proyecto y puede también definir las herramientas o técnicas específicas que se utilizarán para este fin. Por ejemplo, en el caso de equipos distribuidos, tal vez sea necesario un acuerdo sobre cuándo y cómo se llevarán a cabo las reuniones, sobre qué tipo de herramientas de comunicación se utilizarán y quién debe participar en las diversas reuniones.

8.3.3.4 Plan de formación del equipo

Debido a que el Equipo Scrum es interfuncional, cada integrante debe participar activamente en todos los aspectos del proyecto. El Scrum Master debe identificar los problemas con los miembros del equipo y hacer frente a estos a fin de mantener un equipo eficaz.

Para lograr cohesión en el equipo, el Scrum Master debe asegurarse de que las relaciones entre los miembros del equipo sean positivas y que estén unificados en el cumplimiento de la totalidad del proyecto y de los objetivos de la organización, lo cual conduce a una mayor eficiencia y una mayor productividad.

En este contexto, es importante estudiar la sección 3.10, que habla sobre las teorías populares de recursos humanos y su relevancia para Scrum.

8.4 Desarrollar épica(s)

La figura 8-10 muestra todas las entradas, herramientas y salidas para el proceso de *Desarrollar épica(s)*.

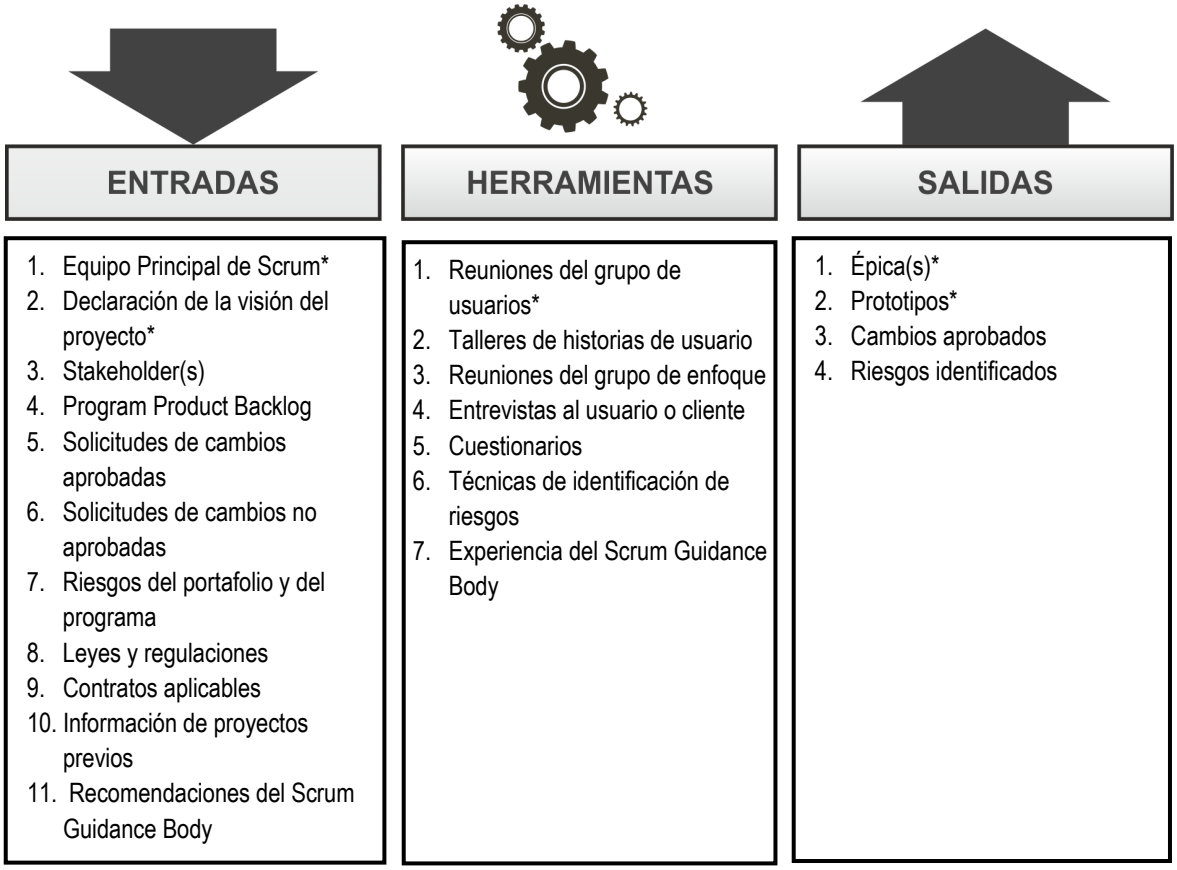


Figura 8-10: Desarrollar épica(s)—Entradas, herramientas y salidas

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

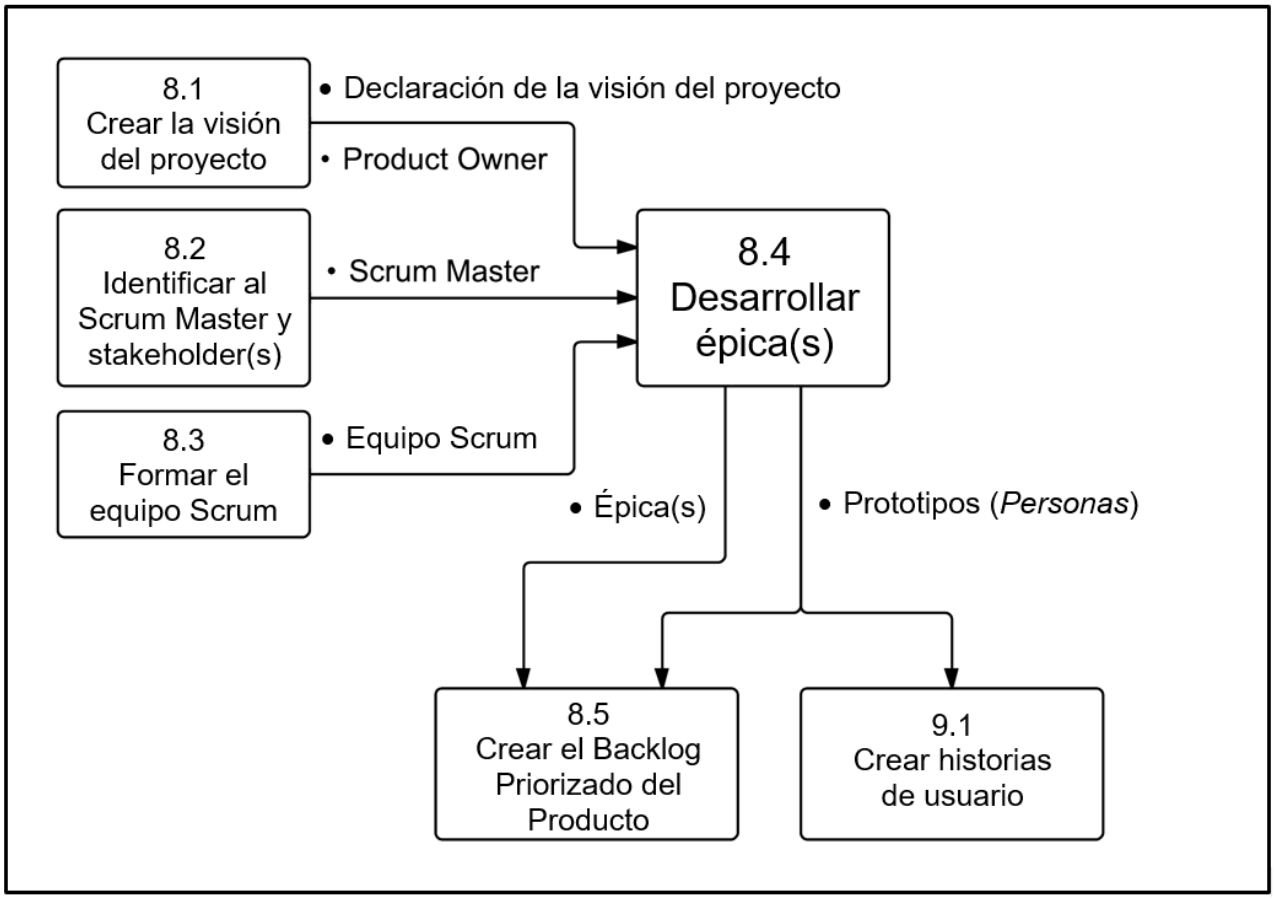


Figura 8-11: Desarrollar épica(s)—Diagrama de flujo

8.4.1 Entradas

8.4.1.1 Equipo principal Scrum *

El equipo principal de Scrum está conformado por el Equipo Scrum, el Scrum Master y el Product Owner, tal como se describe en la sección 3.3.1.

8.4.1.2 Declaración de la visión del proyecto*

Descrito en la sección 8.1.3.2.

8.4.1.3 Stakeholder(s)

Descrito en la sección 8.2.3.2.

8.4.1.4 Program Product Backlog

Descrito en la sección 8.1.1.6.

8.4.1.5 Solicitudes de cambio aprobadas

Las solicitudes de cambio aprobadas que se originan en programa o el portafolio son entradas que se añadirán a la lista de cambios aprobados de proyecto para su ejecución en sprints futuros. Cada cambio puede requerir su propia épica o historia de usuario y pudiera convertirse en una entrada al proceso de *Desarrollar de épica(s)*. En este proceso, las solicitudes de cambio aprobadas también pudieran ser el resultado de otros procesos de Scrum.

Las solicitudes de cambio y las solicitudes de cambio aprobadas se tratan en las secciones 6.3.1, 6.4.2.1 y 6.6.

8.4.1.6 Solicitudes de cambio no aprobadas

Las solicitudes para hacer cambios se presentan por lo general como solicitudes de cambio o *Change Requests*. Las solicitudes de cambio permanecen como no aprobadas hasta que se obtiene una aprobación formal. Las solicitudes de cambio no aprobadas en el proceso de *Desarrollar épica(s)* pudiera provenir de los procesos de *Crear entregables*, *Realizar el* Daily Standup y otros procesos.

Las solicitudes de cambio y las solicitudes de cambio no aprobadas se abordan en las secciones 6.3.1, 6.4.2.1 y 6.6.

8.4.1.7 Riesgos del programa y portafolio

Los riesgos relacionados con un portafolio o programa tendrán también un impacto en los proyectos que forman parte del portafolio o programa respectivo. Durante la evaluación del riesgo en portafolios y programas, si se determina que el riesgo puede afectar a un proyecto individual, la información relevante sobre el mismo debe comunicarse al Product Owner y al Equipo Scrum. Estos riesgos pudieran ser entradas para el proceso de *Desarrollar épica(s)* y pudieran tener un impacto global sobre cómo se lleva a cabo este proceso.

Los riesgos del programa y portafolio se describen en la sección 7.5.1.

8.4.1.8 Leyes y regulaciones

Dependiendo del proyecto, existen leyes y regulaciones impuestas por órganos de gobierno que afectan la planificación y la ejecución. Las leyes son externas a la organización y las impone una entidad gubernamental. Las regulaciones pueden ser internas o externas. Las regulaciones internas son aquellas que aplican dentro de la empresa, por lo general basadas en su política. Dichas regulaciones pueden estar relacionadas a los sistemas de gestión de la calidad, a regulaciones financieras, reglamentos de personal, etc. Las regulaciones externas son aquellas relacionadas a los estándares, normas y requisitos establecidos por el gobierno.

Las leyes y regulaciones se deben tomar en cuenta al momento de desarrollar épicas. Las épicas se basan en los requerimientos de negocio. Para cumplir con estos requerimientos, el equipo del proyecto tiene que cumplir con tanto con leyes y regulaciones internas y externas.

A veces, algunas de las leyes y regulaciones que afectan a múltiples proyectos Scrum pueden incluirse como parte de las recomendaciones del Scrum Guidance Body, como se analiza en la sección 8.1.1.12.

8.4.1.9 Contratos aplicables

Si todo el proyecto o partes del mismo se cumplen bajo un contrato, el contrato debe definir el alcance del trabajo y las condiciones específicas del mismo. El tipo de contrato que se utiliza influye en el riesgo del proyecto.

Algunos de los tipos de contratos más comunes que se utilizan en proyectos Scrum son los siguientes:

Contrato de entrega incremental (*Incremental Delivery Contract*)—Este contrato incluye puntos de inspección en intervalos frecuentes. Ayuda a que el cliente o los stakeholders tomen decisiones periódicas sobre el desarrollo a lo largo de un proyecto en cada punto de inspección. El cliente puede aceptar el desarrollo del producto, optar por detener su desarrollo o solicitar modificaciones al mismo.

Contrato Joint Venture—Este tipo de contratos se utiliza generalmente cuando dos o más partes se asocian para realizar el trabajo de un proyecto. Ambas partes involucradas en el proyecto lograrán algún

retorno sobre la inversión, ya que los ingresos o beneficios que sean generados, se habrán de compartir entre las partes.

Contrato de desarrollo en fases (*Development in Phases Contract*)—Este contrato facilita la disponibilidad de fondos cada mes o cada trimestre después de concluir satisfactoriamente un lanzamiento. Incentiva tanto al cliente como al proveedor y garantiza que el riesgo monetario del cliente esté limitado a un periodo específico, ya que los lanzamientos fracasados no reciben financiamiento.

Contrato de incentivos y sanciones—Este contrato se basa en el acuerdo de que el proveedor será recompensado con un incentivo económico si los productos del proyecto se entregan a tiempo, pero incurrirá en sanciones económicas si la entrega se realiza tarde.

Otros tipos de contratos más populares incluyen el pago por las características del contrato, el tiempo y los materiales del contrato, precio fijo y contrato de alcance fijo y el contrato de beneficio fijo.

Las épicas deben desarrollarse tomando en cuenta los términos y condiciones del tipo de contrato que se utilice.

8.4.1.10 Información de proyectos previos

La información y los conocimientos adquiridos en proyectos anteriores similares dentro de la organización son entradas valiosas para el desarrollo de épicas y la evaluación de riesgos. La información de proyectos previos podría incluir notas del Project Manager, registros del proyecto y comentarios hechos por el(los) stakeholder(s).

Algunos detalles y las mejores prácticas relacionadas con información de proyectos anteriores pueden estar disponibles a través de las recomendaciones Scrum Guidance Body.

8.4.1.11 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.12

Las recomendaciones del Scrum Guidance Body pueden incluir información sobre reglas, regulaciones, estándares y mejores prácticas para el desarrollo de épicas.

8.4.2 Herramientas

8.4.2.1 Reuniones del grupo de usuarios*

Las reuniones del grupo de usuarios incluyen a stakeholders relevantes (principalmente usuarios o clientes del producto). Estos brindan al equipo principal de Scrum información de primera mano sobre las expectativas del usuario. Esto ayuda a la formulación de los criterios de aceptación para el producto y proporciona información valiosa para el desarrollo de épicas. Las reuniones del grupo de usuarios son de vital importancia para evitar e trabajo costoso que pudiera resultar de la falta de claridad sobre las expectativas y exigencias. Estas reuniones también promueven un sentido de compromiso con el proyecto, así como un entendimiento común entre el equipo principal de Scrum y los stakeholders relevantes.

8.4.2.2 Talleres de historias de usuario

Los talleres de historias de usuarios se llevan a cabo como parte del proceso de *Desarrollar épica(s)*. El Scrum Master facilita estas sesiones, donde el equipo principal Scrum interviene y, en ocasiones, se recomienda incluir a otro stakeholders. Estos talleres ayudan al Product Owner a dar prioridad a los requerimientos y permiten que el equipo principal Scrum tenga una perspectiva compartida de los criterios de aceptación. Estos aseguran de que las épicas e historias de usuario describan la funcionalidad desde el punto de vista de los usuarios, que sean fáciles de entender y que se puedan calcular de forma segura. Dichos talleres son útiles en la comprensión de las expectativas del usuario para los entregables y son excelentes para la formación de equipos. También facilitan la preparación para la planificación del próximo sprint. Un taller de historias del usuario es una buena plataforma para discutir y aclarar todos los elementos de un producto y generalmente profundizan en detalles más pequeños para garantizar la claridad.

8.4.2.3 Reuniones del grupo de enfoque

En las reuniones del grupo de enfoque las personas se reúnen en una sesión guiada para ofrecer sus opiniones, percepciones o valoraciones sobre un producto, servicio o resultado deseado. Los miembros del grupo de enfoque tienen la libertad de hacerse preguntas de obtener aclaraciones sobre temas o conceptos específicos. Mediante el cuestionamiento, la crítica constructiva y la retroalimentación, los grupos de enfoque conducen a un producto de mejor calidad y con ello contribuyen a la satisfacción de las expectativas de los usuarios. En estas reuniones, los miembros del grupo de enfoque a veces llegan a un consenso en ciertas áreas, mientras que en otras áreas las opiniones pueden ser diferentes. Cuando los miembros del grupo tienen diferentes opiniones o puntos de vista, se hace el mayor esfuerzo para resolver las diferencias con el fin de llegar a un consenso.

Las sesiones de grupos de enfoque pueden ayudar a los equipos a crear ideas innovadoras, resolver problemas y dar sugerencias para mejorar. Estas reuniones generan ideas y retroalimentación de los usuarios potenciales y desarrolladores de productos. Estas reuniones se realizan normalmente para la

planificación, evaluación o mejora de un producto o servicio. El conocimiento obtenido a partir de estas reuniones también puede ayudar con el desarrollo de épicas e historias del usuario. En ocasiones, se hacen grupos de enfoque a cabo para resolver problemas que puedan surgir durante el desarrollo de las épicas.

8.4.2.4 Entrevistas al usuario o cliente

El incorporar a los stakeholders, incluyendo al patrocinador, usuarios y clientes del producto es importante para obtener el contexto y la visión necesarios para desarrollar épicas. El tiempo de calidad que se invierta en entrevistar a los usuarios y clientes resultará en un mejor alineamiento de los requisitos en las épicas con la visión general del proyecto, ofreciendo de esta forma un mayor valor.

Estas entrevistas ayudan a:

- Identificar y entender las necesidades y expectativas del stakeholder
- Obtener opiniones y hechos
- Comprender la perspectiva del stakeholder sobre el producto final
- Obtener retroalimentación sobre el producto iterado o parcialmente desarrollado

8.4.2.5 Cuestionarios

Una forma económica de obtener una perspectiva estadística cuantitativa y cualitativa de un gran número de usuarios o clientes es el uso de encuestas o cuestionarios. Un cuestionario es un instrumento de investigación con el fin de obtener información sobre un asunto o tema específico. Los cuestionarios pueden ser auto-administrados o administrados por un entrevistador.

Se debe tener mucho cuidado en el diseño de cuestionarios, la selección adecuada del público y en la determinación de un método apropiado de implementación de encuestas para evitar errores y prejuicios.

Durante el desarrollo de épicas, el Product Owner o el Scrum Master podrían llevar a cabo una encuesta para recopilar la información pertinente de stakeholders o el Equipo Scrum.

8.4.2.6 Técnicas de identificación de riesgos

Descrito en la sección 7.4.1.1

8.4.2.7 Experiencia del Scrum Guidance Body

Descrito en la sección 3.3.2

Al crear las épicas, la experiencia del Scrum Guidance Body pudieran relacionarse a las normas y regulaciones o a los estándares y mejores prácticas para crear épicas. Es posible que exista también un

equipo de expertos en la materia que pudiera ayudar al Product Owner a crear épicas. Este equipo podría incluir analistas empresariales, arquitectos líderes, Senior Developers, expertos de Scrum o demás personas con experiencia. Por lo general, este grupo de expertos no es el mismo equipo que permanecerá y trabajará en un proyecto en particular, ya que tienden a pasar de un proyecto a otro durante la "fase de venta" o "fase cero" con los clientes o usuarios.

8.4.3 Salidas

8.4.3.1 Épica(s)*

Las épicas se escriben en las etapas iniciales del proyecto, cuando la mayoría de las historias de usuario son funcionalidades de alto nivel o descripciones de productos que están ampliamente definidas. Las épicas son historias de usuario grandes sin refinar en el Backlog Priorizado del Producto.

Una vez que estas épicas aparecen en el Backlog Priorizado del Producto para completarse en el próximo sprint, se convierten en historias de usuario más pequeñas. Estas historias más pequeñas son generalmente funcionalidades simples, cortas y fáciles de implementar, o bloques de tareas que deben completarse en un sprint.

8.4.3.2 Prototipos* (*Personas*)

Los prototipos, conocidos en inglés como *personas*, son personajes ficticios altamente detallados que representan a la mayoría de los usuarios y demás stakeholders que pudieran no utilizar directamente el producto final. Los prototipos se crean para identificar las necesidades de los usuarios. La creación de prototipos específicos puede ayudar al equipo a entender mejor a los usuarios y sus necesidades y metas. Basado en un prototipo, el Product Owner puede priorizar de manera más efectiva las funciones para crear el Backlog Priorizado del Producto.

Creación de un prototipo: Esto implica la asignación de un nombre ficticio y de preferencia una foto para el personaje, como una fotografía de stock. El prototipo incluye atributos muy específicos como edad, género, nivel académico, ambiente, intereses y metas. También se puede incluir una cita que ilustre las necesidades del prototipo. A continuación, se muestra un ejemplo de un prototipo para un sitio web de viajes.

Ejemplo:
Vanessa tiene 39 años de edad y es residente de San Francisco. Le apasionan los viajes y después de haber tenido una exitosa carrera como abogada, ha decidido disfrutar de dicha pasión. Le gusta tener opciones disponibles al seleccionar sus viajes por avión y servicios de alojamiento para poder elegir el mejor a un precio accesible. Se frustra con los sitios web lentos y desordenados.

8.4.3.3 Cambios aprobados

Las solicitudes de cambio no aprobadas pueden ser aprobadas por el Product Owner durante el proceso de *Desarrollar épica(s)*, a veces con sugerencias que proporcionan los stakeholders relevantes. Tales cambios se clasifican como cambios aprobados y pueden ser priorizados e implementadas en los futuros sprints.

Las solicitudes de cambio y las solicitudes de cambio aprobadas se abordan en las secciones 6.3.1, 6.4.2.1 y 6.6.

8.4.3.4 Riesgos identificados

Al crear épicas, se pueden identificar nuevos riesgos, los cuales constituyen una salida importante de esta etapa. Estos riesgos contribuyen al desarrollo del Backlog Priorizado del Producto (conocido también como Backlog del Producto del riesgo ajustado).

La identificación de riesgos se describe en la sección 7.4.1.

8.5 Crear el Backlog Priorizado del Producto

La figura 8-12 muestra todas las entradas, las herramientas y salidas para el proceso de *Crear el Backlog Priorizado del Producto*.

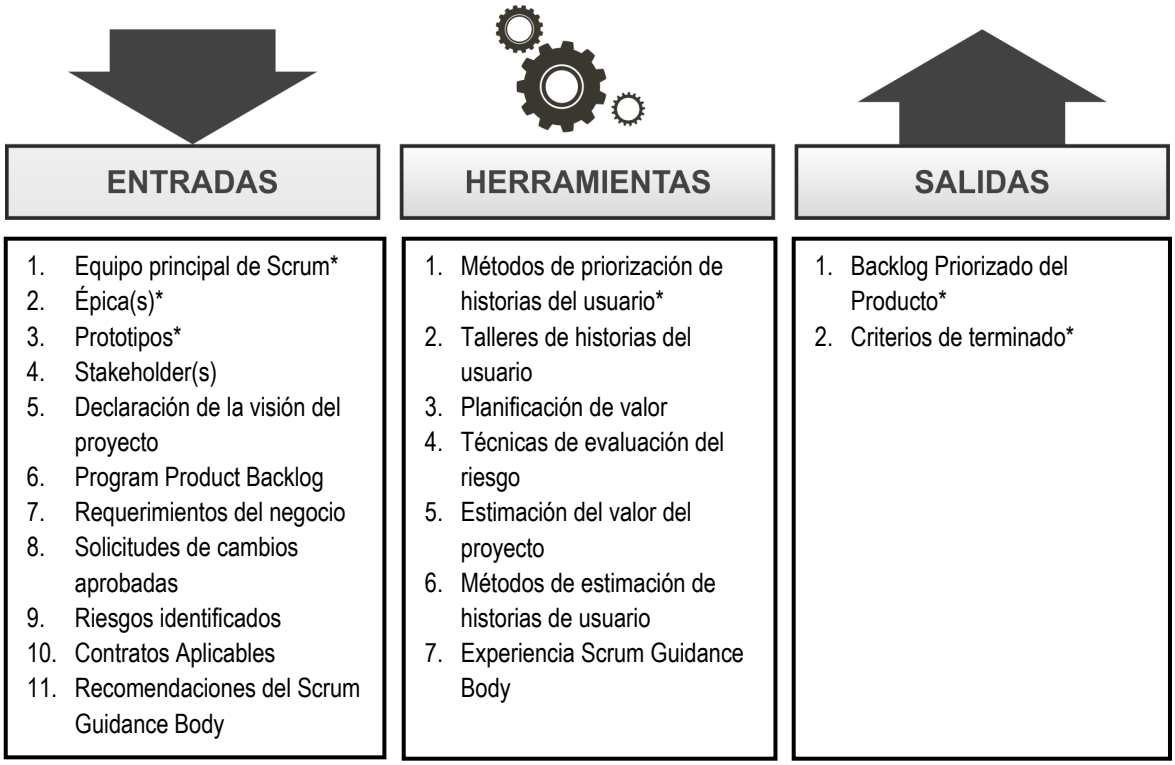


Figura 8-12: Crear el Backlog Priorizado del Producto—Entradas, herramientas y salidas

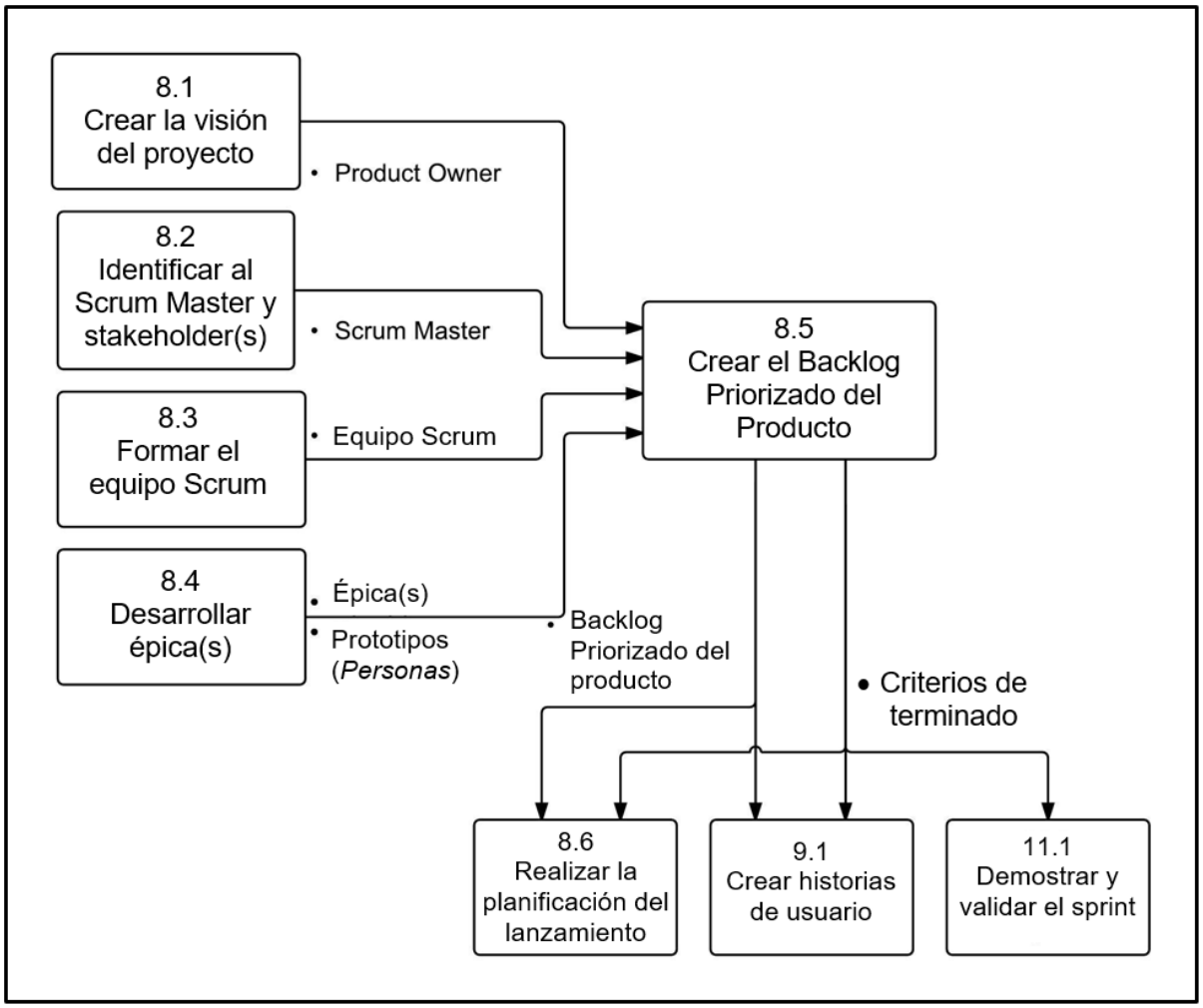


Figura 8-13: *Crear el Backlog Priorizado del Producto*—Diagrama de flujo de datos

8.5.1 Entradas

8.5.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

8.5.1.2 Épica(s)*

Descrito en la sección 8.4.3.1.

8.5.1.3 Prototipos*

Descrito en la sección 8.4.3.2.

8.5.1.4 Stakeholder(s)

Descrito en la sección 8.2.3.2.

8.5.1.5 Declaración de la visión del proyecto

Descrito en la sección 8.1.3.2.

8.5.1.6 Program Product Backlog

Descrito en la sección 8.1.1.6.

8.5.1.7 Requerimientos del negocio

La suma de todos los conocimientos adquiridos mediante las diversas herramientas tales como las entrevistas a los clientes o usuarios o los cuestionarios, las sesiones JAD, el análisis de brecha, el análisis FODA y otras reuniones, ayudan a desarrollar una mejor perspectiva sobre los requerimientos y ayuda en la creación del Backlog Priorizado del Producto.

8.5.1.8 Solicitudes de cambio aprobadas

Descrito en la sección 8.4.3.3.

8.5.1.9 Riesgos identificados

Descrito en la sección 8.4.3.4.

8.5.1.10 Contratos aplicables

Descrito en la sección 8.4.1.9.

8.5.1.11 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

Durante la creación del Backlog Priorizado del Producto, las recomendaciones del Scrum Guidance Body pueden incluir información sobre las normas, reglamentos, procedimientos y las mejores prácticas para el desarrollo del Backlog Priorizado del Producto.

8.5.2 Herramientas

8.5.2.1 Métodos de priorización de las historias de usuarios

A continuación, se presentan algunas de las técnicas que se utilizan para dar prioridad a las historias de usuarios o requerimientos en el Backlog Priorizado del Producto sobre la base del valor de negocio:

- **Esquema de priorización MoSCoW—**El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: "Debe tener" (*Must have*), "Debería tener" (*Should have*), "Podría tener" (*Could have*) y "No tendrá" (*Won't have*). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de "debería tener", siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de "gustaría que tuviera" siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.

- **Comparación por pares**—En esta técnica, la comparación por pares (*Paired Comparison*) es una técnica donde se prepara una lista de todas las historias de usuario en el Backlog Priorizado del Producto. Después, cada historia de usuario se toma en forma individual y se compara con otras

historias en la lista, una a la vez. Cada vez que se comparan dos historias de usuario, se toma una decisión en cuanto a cuál de las dos es más importante. Por medio de este proceso, se puede generar una lista priorizada de historias de usuario.

- **El método de los 100 puntos**—El método de los 100 puntos (*100-Point Method*) fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes. El objetivo es dar más peso a las historias de usuarios que son de mayor prioridad en comparación con las otras historias de usuario disponibles. Cada miembro del grupo asigna puntos a las diversas historias de usuarios, dando más puntos a las que opinan son más importantes. Al finalizar el proceso de votación, la priorización se determina calculando el total de puntos asignados a cada historia de usuario.

- **Análisis de Kano**

Descrito en la sección 4.5.2

8.5.2.2 Talleres de historias de usuarios

Descrito en la sección 8.4.2.2.

8.5.2.3 Planificación de valor

Descrito en la sección 4.5.2

8.5.2.4 Técnicas de evaluación de riesgos

Descrito en la sección 7.4.2.1.

8.5.2.5 Estimación de valor del proyecto

Descrito en la sección 4.5.1.

8.5.2.6 Métodos de estimación de historias de usuarios

Todas las herramientas para estimar historias de usuario en los procesos de *Estimar historias de usuario* (como se describe en la sección 9.2.2.3) se pueden utilizar para crear estimaciones de alto nivel para las épicas cuando se elabora el Backlog Priorizado del Producto Algunas herramientas importantes son:

1. Wideband Delphi
2. Planning Poker
3. Puño de cinco
4. Estimación por afinidad

8.5.2.7 Experiencia del Scrum Guidance Body

Descrito en la sección 8.4.2.7

Durante la creación del Backlog Priorizado del Producto, la experiencia del Scrum Guidance Body podría relacionarse con reglamentos y regulaciones documentadas o con normas y mejores prácticas para la creación de épicas. También puede haber un equipo de expertos en la materia que puede ayudar al Product Owner en el proceso de *Crear el Backlog Priorizado del Producto*. Este equipo podría incluir analistas empresariales, arquitectos líderes, Senior Developers, expertos de Scrum o demás personas con experiencia. Por lo general, este grupo de expertos no es el mismo equipo que permanecerá y trabajará en un proyecto en particular, ya que tienden a pasar de un proyecto a otro durante la "fase de venta" o "fase cero" con los clientes o usuarios.

8.5.3 Salidas

8.5.3.1 Backlog Priorizado del Producto*

El Product Owner desarrolla un Backlog Priorizado del Producto que contiene una lista priorizada de los requerimientos del negocio y de los proyectos escritos en forma de épica(s), que son historias de usuario de alto nivel. El Backlog Priorizado del Producto se basa en tres factores principales: valor, riesgo o incertidumbre y dependencias. También se le conoce como Backlog Priorizado del Producto del riesgo ajustado, ya que incluye riesgos identificados y evaluados relacionados al proyecto. También incluye cambios aprobados que pueden ser priorizados adecuadamente en el Backlog Priorizado del Producto (tal como se describe en la sección 6.3.1).

- **Valor**—Es la responsabilidad del Product Owner asegurar primero la entrega de los productos que ofrezcan el mayor valor. Incluso un producto de gran valor no puede ser parte del primer lanzamiento si hay otros productos de mayor valor que son suficientes para un primer lanzamiento.

- **Riesgo e incertidumbre**—Entre mayor incertidumbre exista, más riesgoso es el proyecto. Por lo tanto, es importante que se les dé mayor prioridad los productos de mayor riesgo en el Backlog Priorizado del Producto. Los productos que llevan un mayor nivel de riesgo también requerirán acciones de mitigación de riesgos. Cuando estas acciones se priorizan en comparación al backlog, el resultado es un Backlog Priorizado del Producto del riesgo ajustado. Tratar con riesgos al principio del proyecto no garantiza que el proyecto tenga éxito, pero sí mejorará la capacidad del equipo para hacer frente a los riesgos. Esto se describe en la sección 7.4.3.

- **Dependencias**—Generalmente no es posible crear un Backlog Priorizado del Producto donde no existan dependencias entre las historias de usuarios. Los requerimientos funcionales a menudo dependen de otros requerimientos funcionales incluso los no-funcionales. Estas dependencias pueden afectar la forma en que se priorizan las historias de usuario en el Backlog Priorizado del Producto. Dos de las formas más comunes para resolver las dependencias son dividir una sola historia en varias partes o combinar historias interdependientes.

- **Estimaciones**—Las estimaciones de alto nivel para la épica(s) también están disponibles en el Backlog Priorizado del Producto.

8.5.3.2 Criterios de terminado*

Los criterios de terminado son un conjunto de reglas que se aplican a todas las historias de usuarios. Es importante contar con una definición clara de terminado, ya que elimina la ambigüedad de los requisitos y ayuda a que el equipo se apegue a las normas obligatorias de calidad. Esta clara definición se utiliza para crear los criterios de terminado, que son un resultado del proceso de *Crear el Backlog Priorizado del Producto*. Una historia de usuario se considera terminada cuando se demuestra al Product Owner y es aprobada por este, quien juzga con base a los criterios de terminado y a los criterios de aceptación de las historias de usuario.

Ejemplo de criterios de terminado:

Proyecto: Diseñar las nuevas variantes de un popular auto deportivo en la empresa LRA, S.A.

Criterios de terminado:

- El diseño es aprobado por la división de Excelencia Técnica.
- El prototipo pasa todas las pruebas de túnel de viento por instrucciones por la división de Aerodinámica.
- El diseño es aprobado para la producción por la división de Propiedad Intelectual.
- Las expectativas de diseño de seguridad son corroboradas en el informe de Diseño de Seguridad de la División de Seguridad.
- El informe de estimación de costos del diseño es aprobado por la división de finanzas.

8.6 Realizar la planificación del lanzamiento

La figura 8-14 muestra todas las entradas, herramientas y salidas para el proceso de *Realizar la planificación del lanzamiento.*

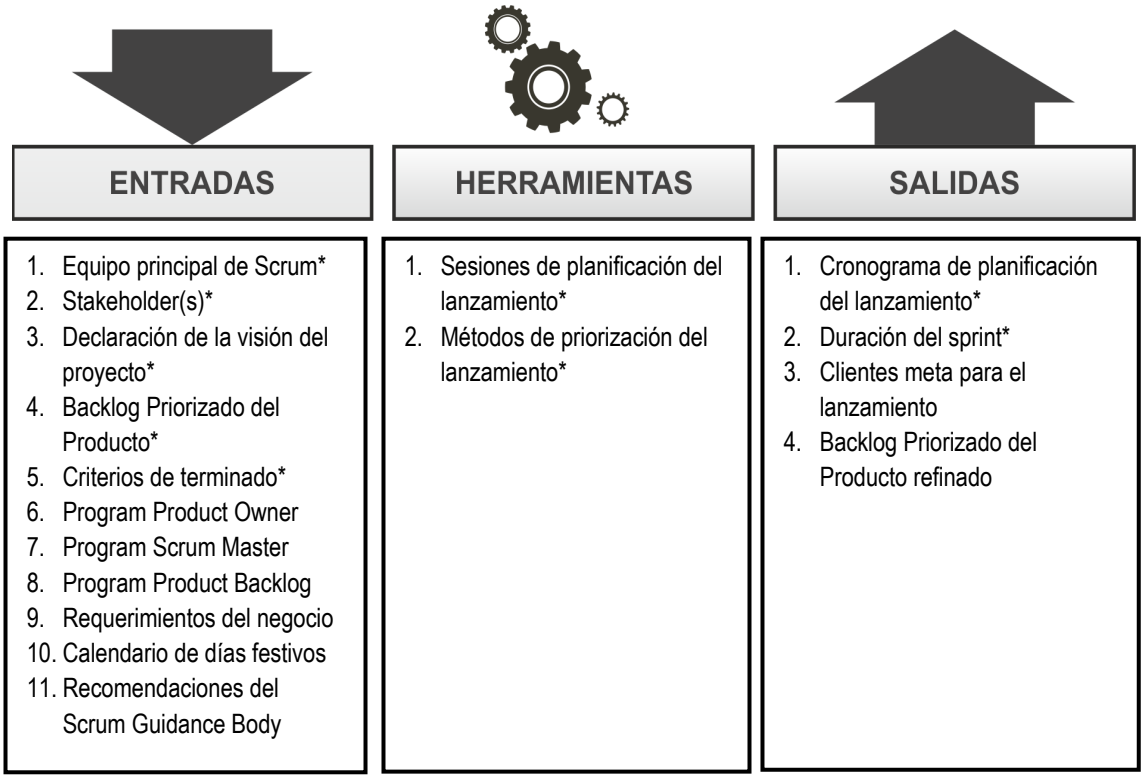


Figura 8-14: Realizar la planificación del lanzamiento—Entrada, herramientas y salidas

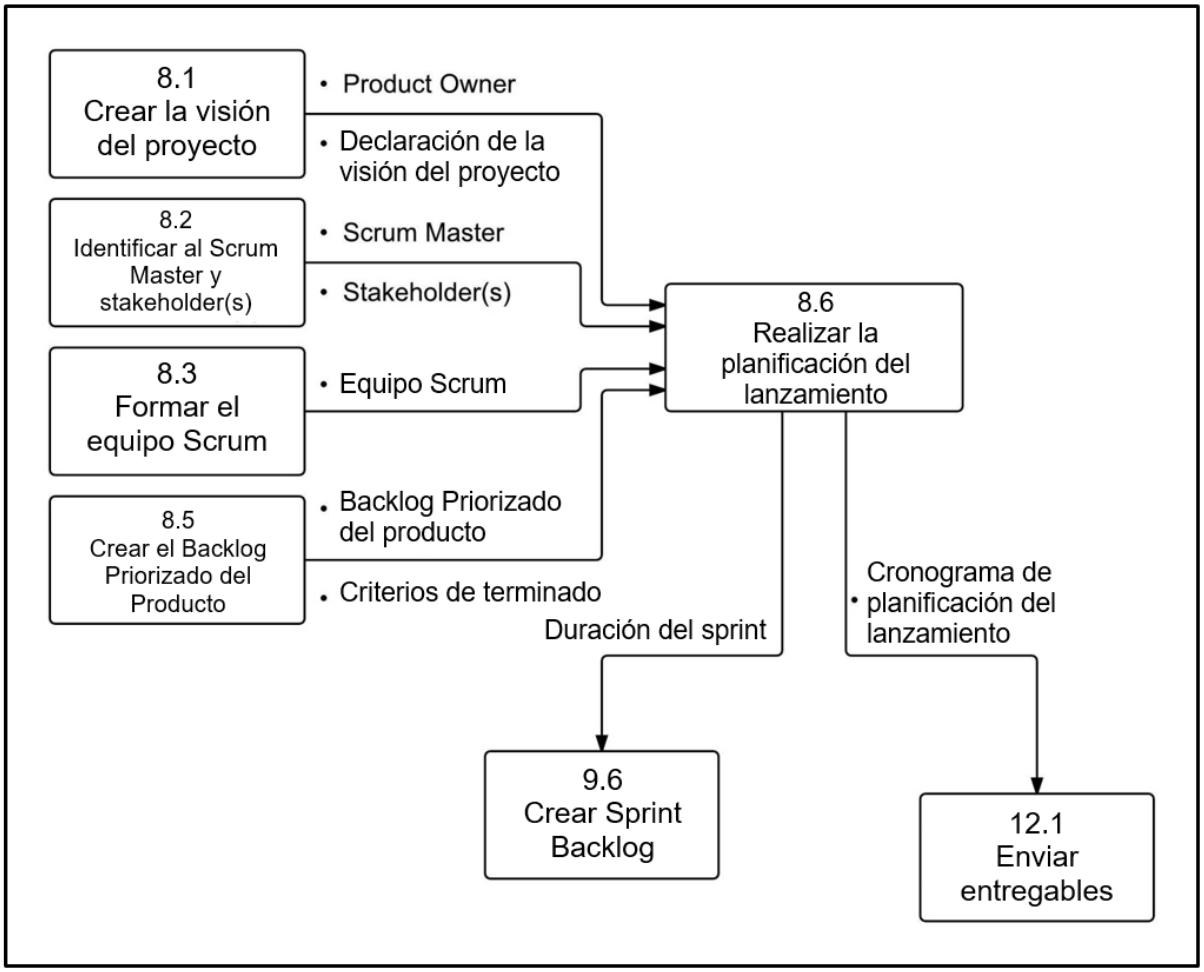


Figura 8-15: Realizar el plan de lanzamiento—Diagrama de flujo de datos

8.6.1 Entradas

8.6.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

8.6.1.2 Stakeholder(s)*

Descrito en la sección 8.2.3.2.

8.6.1.3 Declaración la visión del proyecto*

Descrito en la sección 8.1.3.2.

8.6.1.4 Backlog Priorizado del Producto*

Descrito en la sección 8.5.3.1.

8.6.1.5 Criterios de terminado*

Descrito en la sección 8.5.3.2.

8.6.1.6 Program Product Owner

Descrito en la sección 8.1.1.2.

8.6.1.7 Program Scrum Master

Descrito en la sección 8.1.1.3.

8.6.1.8 Program Product Backlog

Descrito en la sección 8.1.1.6.

8.6.1.9 Requerimientos del negocio

Descrito en la sección 8.5.1.7.

8.6.1.10 Calendario de días festivos

Es importante que el Equipo Scrum esté al tanto de las fechas importantes y de la disponibilidad de todos los integrantes del equipo. Esto se puede lograr mediante el uso de un calendario compartido que proporcione información sobre los días feriados, ausencias, ausencias, planes de viaje, eventos, etc. Este calendario ayudará al equipo en la planificación y ejecución de los sprints.

8.6.1.11 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11

En el proceso de *Realizar la planificación del lanzamiento, las* recomendaciones de Scrum Guidance Body pueden estar relacionadas a las normas, reglamentos, estándares y mejores prácticas para el desarrollo del Plan de Lanzamiento. Dicho órgano puede ser la mejor autoridad para definir las directrices relacionadas al valor del negocio, las expectativas de lanzamiento, estrategias de implementación, calidad y seguridad.

8.6.2 Herramientas

8.6.2.1 Sesiones de planificación del lanzamiento*

Las sesiones de planificación del lanzamiento se llevan a cabo para desarrollar un plan del lanzamiento. Dicho plan define cuándo las distintas series de funcionalidades útiles serán entregadas al cliente. En Scrum, el objetivo principal de una sesión de planificación de lanzamiento es hacer que el Equipo Scrum cuente con una visión general de los lanzamientos y del calendario de entrega del producto que están desarrollando para que puedan alinearse con las expectativas del Product Owner y los stakeholders relevantes (principalmente el patrocinador o *Project Sponsor*).

Muchas organizaciones tienen una estrategia relacionada el lanzamiento de los productos. Algunas organizaciones prefieren un despliegue continuo, donde se produce un lanzamiento después de la creación de la funcionalidad útil especificada. Otras organizaciones prefieren un despliegue por etapas, donde los lanzamientos se hacen en intervalos predefinidos. Dependiendo de la estrategia de la organización, la sesión de planificación del lanzamiento en los proyectos puede estar guiada por la funcionalidad, donde el objetivo es hacer un lanzamiento una vez que se ha desarrollado un conjunto predeterminado de funcionalidades; o bien, la planificación puede estar impulsada por la fecha en la que ocurre el lanzamiento en una fecha predefinida.

Dado a que el framework de Scrum promueve la toma de decisiones iterativa basada en información en vez de una planificación anticipada y detallada como en el estilo tradicional de gestión de proyectos en casada, las sesiones de planificación de lanzamiento no necesitan producto un plan detallado de lanzamiento para todo el proyecto. El plan de lanzamiento puede actualizarse constantemente a medida que la información relevante está disponible.

8.6.2.2 Métodos de priorización del lanzamiento*

Los métodos de priorización del lanzamiento se utilizan para desarrollar un plan del lanzamiento. Estos métodos son específicos a la industria y organización, y generalmente son determinados por la alta gerencia de la organización.

8.6.3 Salidas

8.6.3.1 Cronograma de planificación del lanzamiento*

Un cronograma de planificación del lanzamiento (*Release Planning Schedule*) es una de las salidas más importantes del proceso de *Realizar la planificación del lanzamiento*. Un cronograma de planificación del lanzamiento indica cuáles entregables serán entregados al cliente, así como los intervalos planificados y fechas para los lanzamientos. Tal vez no exista un lanzamiento programado al final de cada iteración del sprint. En ocasiones, un lanzamiento puede planificarse después de completar un grupo de iteraciones del sprint. Dependiendo de la estrategia de la organización, las sesiones de la planificación del lanzamiento en los proyectos pueden estar guiadas por la funcionalidad donde el objetivo es la entrega una vez que se ha desarrollado un conjunto predeterminado de funcionalidades, o bien, la planificación puede estar impulsada por la fecha, donde el lanzamiento se da en una fecha predefinida. El entregable de debe lanzar cuando ofrezca suficiente valor empresarial para el cliente.

8.6.3.2 Duración del sprint*

Con base en las diversas entradas, incluyendo los requerimientos del negocio y el cronograma de planificación del lanzamiento, el Product Owner y el Equipo Scrum deciden la duración del sprint para el proyecto. Una vez determinada, la duración del sprint generalmente permanece igual durante el proyecto.

Sin embargo, la duración del sprint puede cambiar si lo consideran apropiado el Product Owner y el Equipo Scrum. A principios del proyecto se puede seguir dando la experimentación para encontrar la mejor duración del sprint. Si más adelante en el proyecto hay un cambio en la duración del sprint, normalmente significa que se puede reducir debido a mejoras en el entorno del proyecto.

Un sprint pudiera tener a un time-box de una a seis semanas. Sin embargo, para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint en un time-box de cuatro semanas, a menos que existan proyectos con requisitos muy estables, donde los sprints pueden extenderse hasta seis semanas.

El impacto de un cambio esperado en la duración del sprint se describe en la sección 6.5.1

8.6.3.3 Clientes meta para el lanzamiento

No todos los lanzamientos estarán dirigidos a todos los stakeholders o usuarios. El(los) stakeholder(s) puede(n) optar por limitar ciertos lanzamientos a un subconjunto de usuarios. El plan de lanzamiento debe especificar cuáles son los clientes en quienes se va a enfocar el lanzamiento.

8.6.3.4 Backlog Priorizado del Producto refinado

El Backlog Priorizado del Producto, desarrollada en el proceso de *Crear Backlog Priorizado del Producto*, se puede refinar en este proceso. Es posible que haya más claridad sobre las historias de usuarios en el Backlog Priorizado del Producto después de que el equipo principal de Scrum lleve a cabo las sesiones de planificación del lanzamiento con el(los) stakeholder(s).

8.7 Diagrama de flujo de datos de fase

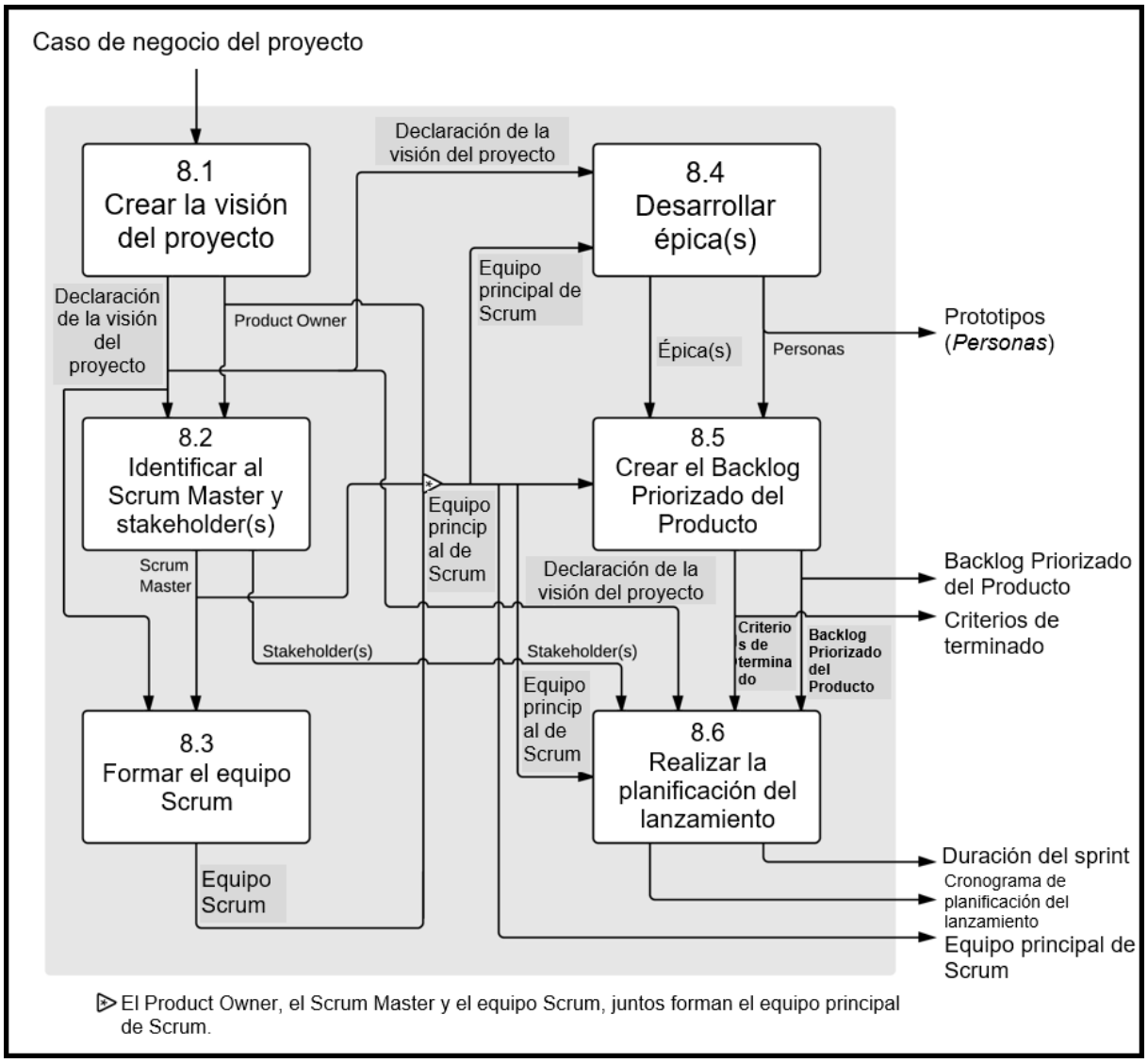


Figura 8-16: Fase de inicio—Diagrama del flujo de datos

9. PLANIFICACIÓN Y ESTIMACIÓN

La fase de Planificación y Estimación consiste en procesos relacionados a la planificación y estimación de tareas, los cuales incluyen: *Crear historias de usuario*, *Estimar historias de usuario*, *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y *Crear el Sprint Backlog*.

La *Planificación y Estimación*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

A fin de facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas por asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deben esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™,* no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un Equipo Scrum que trabaja en un sprint para producir entregables que pudieran ser parte de un proyecto más grande, de un programa o portafolio. La información sobre cómo escalar Scrum en grandes proyectos está disponible en el capítulo 13 y lo relacionado a escalar Scrum para la empresa se puede encontrar en el capítulo 14.

La figura 9-1 brinda una descripción general de los procesos en la fase de planificación y estimación, que son los siguientes:

9.1 Crear historias de usuario—En este proceso se crean las historias de usuario y sus respectivos criterios de aceptación. Las historias de usuario generalmente las escribe el Product Owner y se diseñan para garantizar que los requerimientos del cliente estén claramente representados y que todos los stakeholders las pueden entender completamente. Se pueden llevar a cabo talleres de redacción de historias de usuario donde se involucre a los miembros del Equipo Scrum en la creación de dichas historias. Tales historias se incorporan al Backlog Priorizado del Producto.

9.2 Estimar historias de usuario—En este proceso el Product Owner aclara las historias de usuario a fin de que el Scrum Master y el Equipo Scrum hagan una estimación sobre el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario.

9.3 Comprometer historias de usuario—En este proceso el Equipo Scrum se compromete a entregar al Product Owner las historias de usuario para un sprint. El resultado de este proceso serían historias de usuario comprometidas.

9.4 Identificar tareas—En este proceso las historias de usuario comprometidas se dividen en tareas específicas y se compilan en una lista de tareas.

9.5 Estimar tareas—En este proceso, el equipo principal de Scrum estima el esfuerzo necesario para completar cada tarea en la lista de tareas. El resultado de este proceso es una Effort Estimated Task List.

9.6 Crear el Sprint Backlog—En este, el equipo principal de Scrum lleva a cabo reuniones de planificación del sprint donde el grupo elabora un Sprint Backlog que contiene todas las tareas a completarse en el sprint.

9.1 Crear historias de usuario

ENTRADAS
1. Equipo principal de Scrum*
2. Backlog Priorizado del Producto*
3. Criterios de terminado*
4. Prototipos*
5. Stakeholder(s)
6. Épica(s)
7. Requerimientos del negocio
8. Leyes y regulaciones
9. Contratos aplicables
10. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Experiencia en la redacción de historias de usuario*
2. Talleres de historias del usuario
3. Reuniones del grupo de usuarios
4. Reuniones del grupo de enfoque
5. Entrevistas al cliente o usuario
6. Cuestionarios
7. Experiencia del Scrum Guidance Body

SALIDAS
1. Historias de usuarios*
2. Criterio de aceptación de historias del usuario*
3. Backlog Priorizado del Producto actualizado
4. Prototipos actualizados o refinados

9.2 Estimar historias de usuario

ENTRADAS
1. Equipo principal de Scrum*
2. Historias de usuarios*
3. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Reuniones de planificación del sprint
2. Reuniones de revisión del Backlog Priorizado del Producto
3. Métodos de estimación*

SALIDAS
1. Historias del usuario estimadas*
2. Backlog Priorizado del Producto actualizado
3. Criterios de aceptación del usuario actualizado

9.3 Comprometer historias de usuario

ENTRADAS
1. Equipo principal de Scrum*
2. Historias del usuario estimadas*
3. Duración del sprint*
4. Velocidad del sprint anterior
5. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Reuniones de planificación del sprint*
2. Técnicas de comunicación

SALIDAS
1. Historias de usuario comprometidas*

9.4 Identificar tareas

ENTRADAS
1. Equipo principal de Scrum*
2. Historias de usuario comprometidas*

HERRAMIENTAS
1. Reuniones de planificación del sprint*
2. Descomposición
3. Determinación de dependencia

SALIDAS
1. Lista de tareas
2. Historias de usuario comprometidas actualizadas
3. Dependencias

9.5 Estimar tareas

ENTRADAS
6. Equipo principal de Scrum*
7. Lista de tareas*
8. Criterios de aceptación de historia de usuario
9. Dependencias
10. Riesgos identificados
11. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
4. Reuniones de planificación del sprint*
5. Criterios de estimación*
6. Métodos de estimación*

SALIDAS
3. Effort Estimated Task List*
4. Lista de tareas actualizada

9.6 Crear el Sprint Backlog

ENTRADAS
1. Equipo principal de Scrum*
2. Effort Estimated Task List*
3. Duración del sprint*
4. Dependencias
5. Calendario del equipo

HERRAMIENTAS
1. Reuniones de planificación del sprint*
2. Herramientas de seguimiento del sprint
3. Parámetros de seguimiento del sprint

SALIDAS
1. Sprint Backlog*
2. Sprint Burndown Chart*

Figura 9-1: Resumen de la planificación y estimación

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 9-2 muestra las entradas obligatorias, las herramientas y las salidas de los procesos en la fase de planificación y estimación.

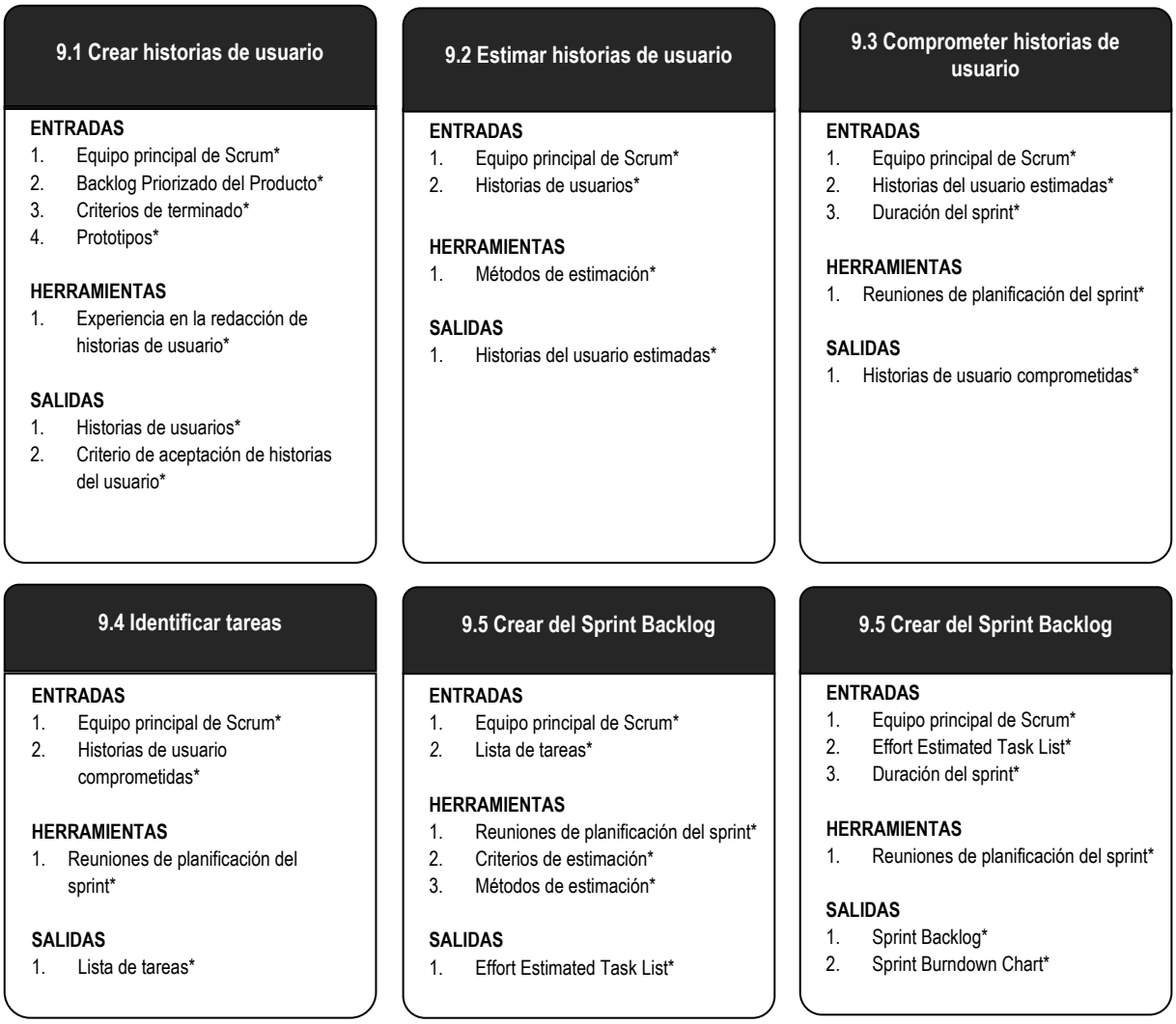


Figura 9-2: Resumen de la Planificación y Estimación (Esenciales)

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

9.1 Crear historias de usuario

La figura 9-3 muestra todas las entradas, las herramientas y las salidas para el proceso de *Crear historias de usuario*.

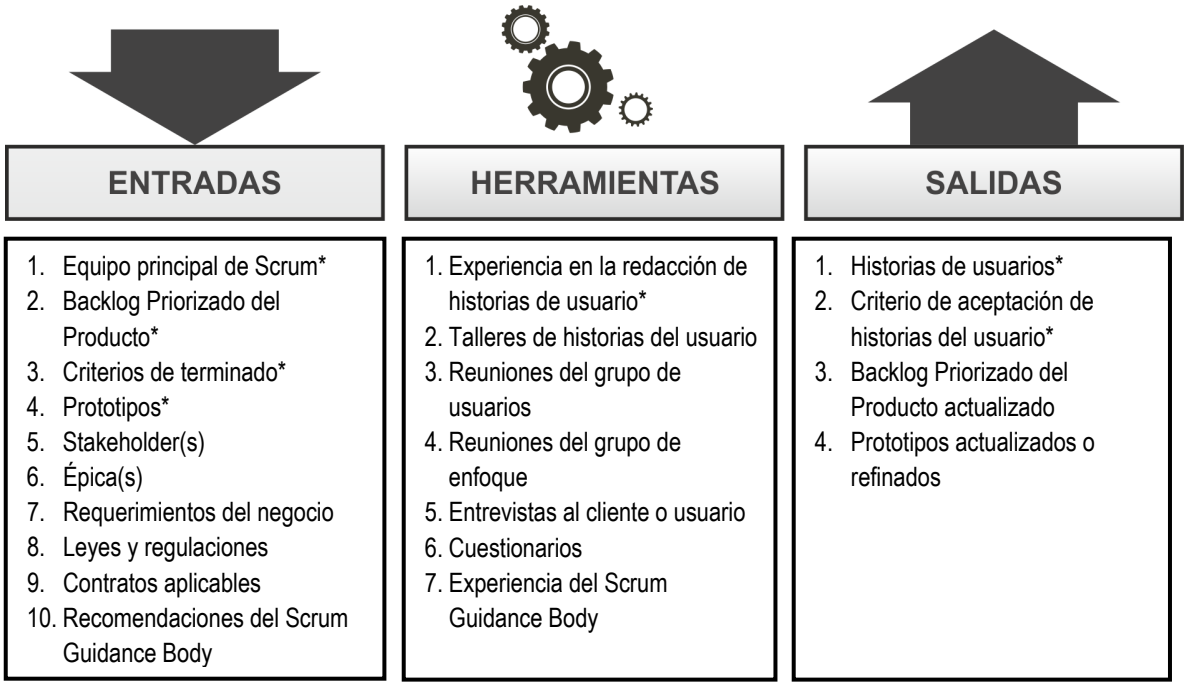


Figura 9-3: Crear historias de usuario—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

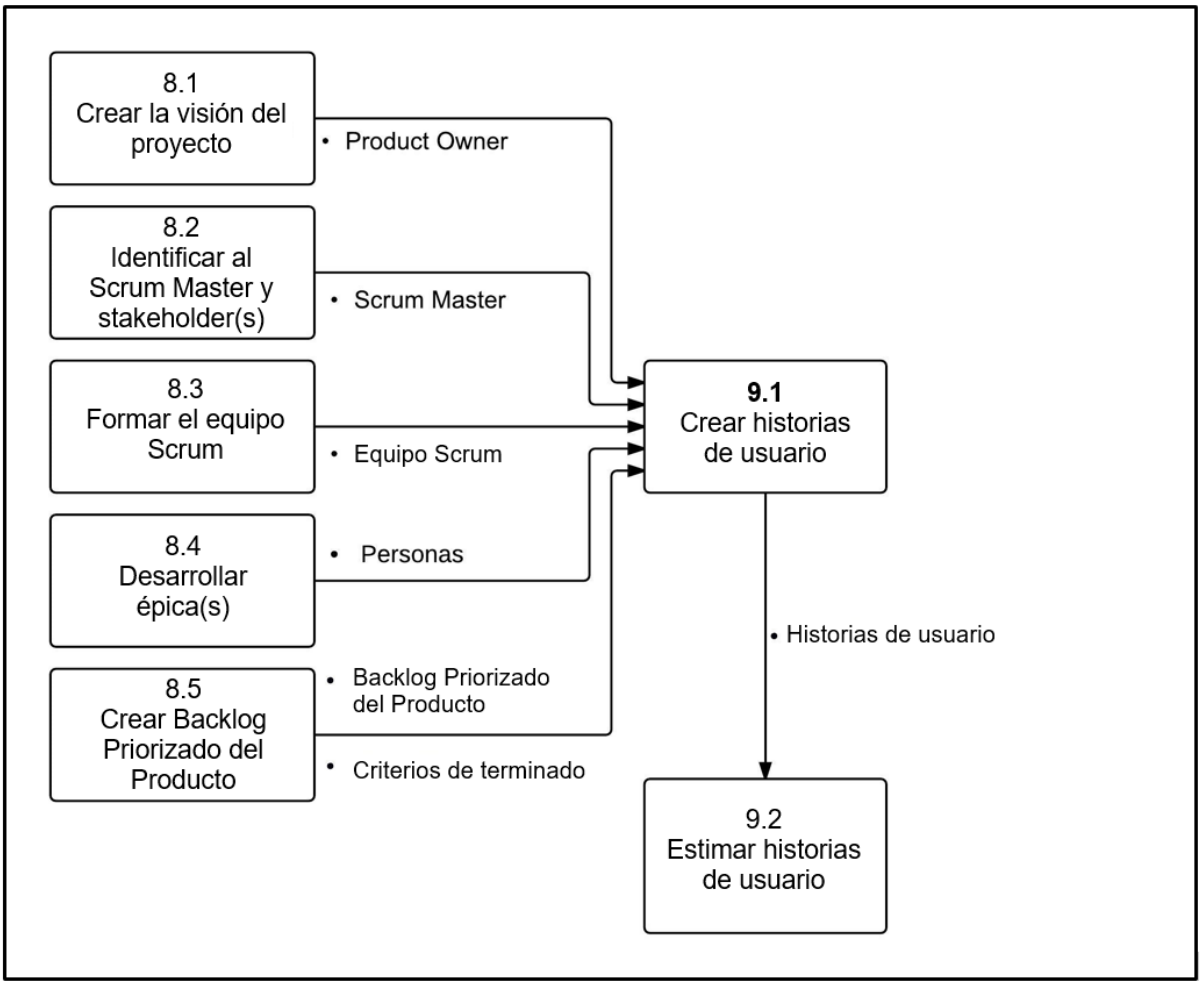


Figura 9-4: Crear historias de usuario—Diagrama de flujo de datos

9.1.1 Entradas

9.1.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

9.1.1.2 Backlog Priorizado del Producto*

Descrito en la sección 8.5.3.1.

9.1.1.3 Criterios de terminado*

Descrito en la sección 8.5.3.2.

9.1.1.4 Prototipos*

Descrito en la sección 8.4.3.2.

9.1.1.5 Stakeholder(s)

Descrito en la sección 8.2.3.2.

9.1.1.6 Épica(s)

Descrito en la sección 8.4.3.1.

9.1.1.7 Requerimientos del negocio

Descrito en la sección 8.5.1.7.

9.1.1.8 Leyes y regulaciones

Descrito en la sección 8.4.1.8.

9.1.1.9 Contratos aplicables

Descrito en la sección 8.4.1.9.

9.1.1.10 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de *Crear historias de usuario*, las recomendaciones del Scrum Guidance Body pueden incluir información sobre las normal, regulaciones, estándares y mejores prácticas necesarias para crear historias de usuario efectivas.

9.1.2 Herramientas

9.1.2.1 Experiencia en la redacción de historias de usuario*

El Product Owner —con base en su interacción con los stakeholders, en su conocimiento del negocio, en la experiencia y en las aportaciones del equipo—, desarrolla las historias de usuario que habrán de formar la Backlog Priorizado del Producto inicial para el proyecto. El Backlog Priorizado del Producto representa la suma total de lo que debe completarse en el proyecto. El objetivo de este ejercicio es crear historias de usuario elaboradas y refinadas que pudieran ser estimadas y comprometidas por el Equipo Scrum. En ocasiones, el Product Owner pudiera incluir a un analista empresarial para que ayude en la redacción de historias de usuario.

Aunque el Product Owner tiene la principal responsabilidad de escribir las historias de usuario y generalmente lleva a cabo esta actividad por sí mismo, se puede también llevar a cabo un taller de redacción de historias de usuario si así se desea.

9.1.2.2 Talleres de historias de usuario

Descrito en la sección 8.4.2.2.

9.1.2.3 Reuniones de grupo de usuarios

Descrito en la sección 8.4.2.1.

9.1.2.4 Reuniones de grupo de enfoque

Las reuniones de grupos de enfoque son una técnica cualitativa para cuantificar y entender las necesidades y expectativas del usuario respecto al producto que se propone. Se selecciona un pequeño grupo de usuarios para que formen parte del grupo de enfoque. Este grupo pudiera ser seleccionado en forma aleatoria de entre muchos usuarios o bien, puede seleccionarse específicamente para representar a todos los prototipos de cliente (*personas*) meta. Las reuniones de grupo de enfoque normalmente se apegan a cierto formato en el cual se le hacen preguntas que después analizan entre ellos. Cada reunión de grupo de enfoque puede contar con su propio reglamento de discusiones según lo hayan decidido los organizadores. Dichas reuniones generalmente se llevan a cabo en presencia de un moderador.

9.1.2.5 Entrevistas al cliente o usuario

Descrito en la sección 8.4.2.4.

9.1.2.6 Cuestionarios

Descrito en la sección 8.4.2.5.

9.1.2.7 Experiencia del Scrum Guidance Body

Descrito en la sección 8.4.2.7.

La experiencia Scrum Guidance Body podría referirse a las normas y regulaciones o a los estándares y mejores prácticas para crear historias de usuario. Pudiera haber también un equipo de expertos en la materia que pudiera ayudar al Product Owner o brindar orientación sobre cómo crear historias de usuario. Dicho equipo pudiera incluir a analistas empresariales, arquitectos líderes, desarrolladores experimentados, expertos en Scrum o demás personas con experiencia. Este grupo no es generalmente el mismo equipo que permanecerá y trabajará en el proyecto, ya que tienden a trasladarse de proyecto en proyecto brindando orientación a los equipos Scrum si así lo requieren.

Al crear historias de usuario, la experiencia del Scrum Guidance Body pudieran relacionarse a las normas y regulaciones o a los estándares y mejores prácticas para crear historias de usuario. Es posible que exista también un equipo de expertos en la materia que pudiera ayudar al Product Owner a crear historias de usuario. Este equipo podría incluir analistas empresariales, arquitectos líderes, Senior Developers, expertos de Scrum o demás personas con experiencia. Por lo general, este grupo de expertos no es el mismo equipo que permanecerá y trabajará en un proyecto en particular, ya que tienden a pasar de un proyecto a otro y brindan asesoría al Equipo Scrum cuando es necesario.

9.1.3 Salidas

9.1.3.1 Historias de usuario*

Las historias de usuario se apegan a una estructura específica predefinida y son una forma simple de documentar los requerimientos y funcionalidades que desea el usuario final. Una historia de usuario incluye tres elementos sobre el requerimiento: ¿Quién? ¿Qué? y ¿Por qué? Los requerimientos expresados en las historias de usuario son oraciones breves, sencillas y fáciles de entender. El formato estándar predefinido da como resultado en una comunicación mejorada entre los stakeholders, así como en mejores estimaciones por parte del equipo. Algunas historias de usuario tal vez sean demasiado extensas como para poderse manejar dentro un solo sprint. A estas amplias historias generalmente se les llama épicas. Una vez que las épicas surgen en el Backlog Priorizado del Producto para completarse en otro sprint, se fragmentan aún más en historias de usuario.

El Backlog Priorizado del Producto es una lista dinámica que se actualiza constantemente debido a la re-priorización de historias de usuarios nuevas, actualizadas, refinadas y en ocasiones eliminadas. Tales actualizaciones al backlog son generalmente el resultado de cambios en los requerimientos del negocio.

Véase también la sección 8.5.3.1 para conocer más sobre el Backlog Priorizado del Producto.

Formato de historia de usuario:

Como <rol/prototipo de cliente> yo debería <requerimiento> a fin de <beneficio>.

Ejemplo de historia de usuario:

Como administrador de una base de datos, yo debería contar con la capacidad de revertir una cantidad selecta de actualización de la base de datos a fin de que se restablezca a la versión deseada.

9.1.3.2 Criterios de aceptación de historias de usuario*

Cada historia de usuario cuenta con sus respectivos criterios de aceptación. Las historias de usuario son subjetivas de tal forma que los criterios de aceptación brindan la objetividad requerida para que las historias de usuario se consideren terminadas o no terminadas durante la revisión del sprint. Los criterios de aceptación brindan claridad al equipo respecto a los que se espera en una historia de usuario; eliminan la ambigüedad de los requerimientos, ayudando a la alineación de las expectativas. El Product Owner define y comunica los criterios de aceptación al Equipo Scrum. En las reuniones de revisión del sprint, los criterios de aceptación brindan el contexto para que el Product Owner decida si la historia de usuario se ha completado satisfactoriamente. Es responsabilidad del Scrum Master asegurar que el Product Owner no cambie los criterios de aceptación de una historia de usuario asignada a mitad de un sprint.

9.1.3.3 Backlog Priorizado del Producto actualizado

El Backlog Priorizado del Producto elaborado en el proceso de su creación, se actualiza con la información en las historias de usuario, épicas, estimaciones para historias de usuario, así como los criterios de aceptación de las mismas.

El Backlog Priorizado del Producto se describe en la sección 8.5.3.1.

9.1.3.4 Prototipos actualizados o refinados

Los prototipos (del inglés: *personas*) se elaboran en el proceso de *Desarrollar épica(s).* Al momento de redactar las historias de usuario, el Equipo Scrum puede llegar a una decisión colectiva de que dichos prototipos iniciales son inadecuados y deben refinarse. En caso de que exista la necesidad de refinar los prototipos, esto se lleva a cabo normalmente casi al final del proceso de *Crear historias de usuario*.

Los prototipos se describen en la sección 8.4.3.2.

9.2 Estimar historias de usuarios

La figura 9-5 muestra todas las entradas, las herramientas y las salidas para el proceso de *Estimar historias de usuario*.



Figura 9-5: Estimar historias de usuario—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

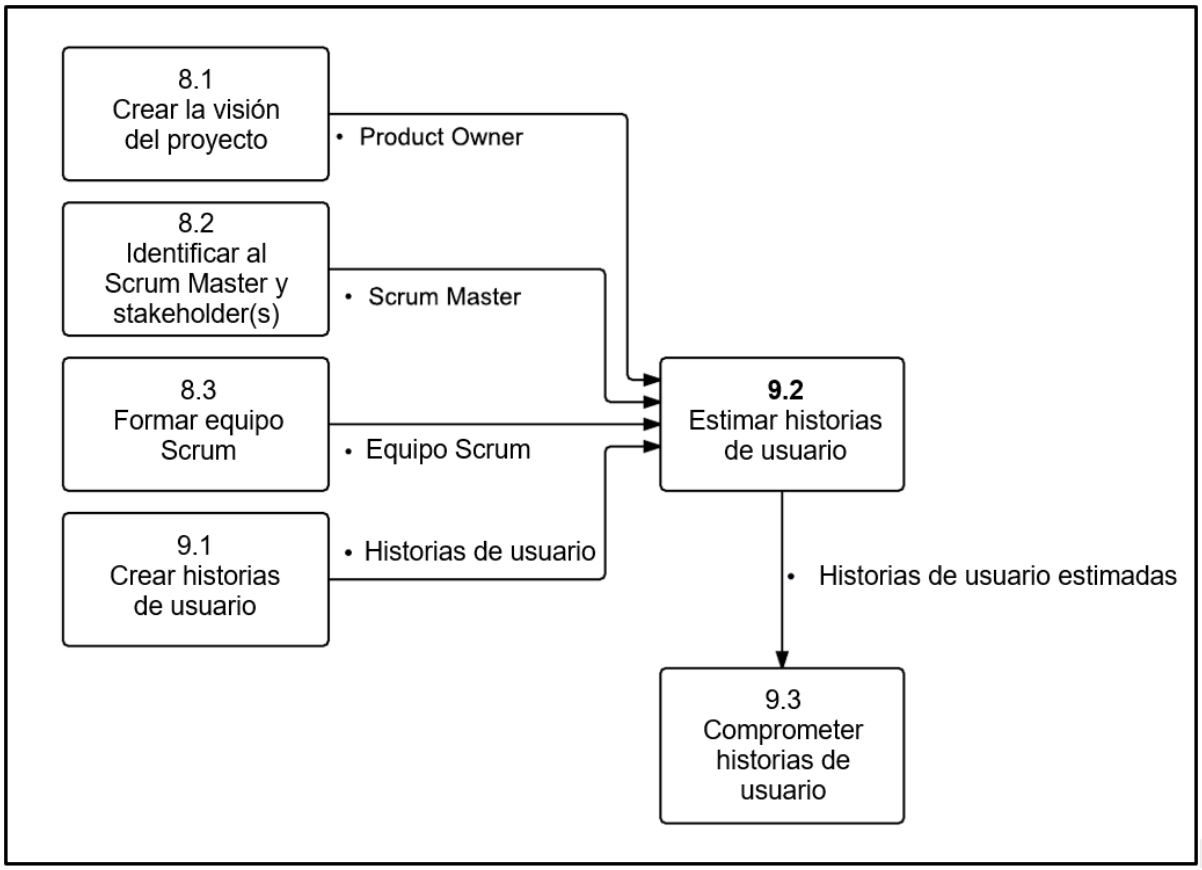


Figura 9-6: Estimar historias de usuario— Diagrama de flujo de datos

9.2.1 Entradas

9.2.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

9.2.1.2 Historias de usuario*

Descrito en la sección 9.1.3.1.

9.2.1.3 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de *Estimar historias de usuario*, las recomendaciones del Scrum Guidance Body pudieran incluir información sobre las normal, regulaciones, estándares y mejores prácticas necesarias para estimar eficazmente las historias de usuario.

9.2.2 Herramientas

9.2.2.1 Reuniones de planificación del sprint*

Durante las reuniones de planificación del sprint, el equipo principal de Scrum somete a consideración las historias de usuario. Si no se hizo durante la creación o el refinamiento del Product Backlog, cada historia de usuario se evalúa y se le asigna una estimación de alto nivel en puntos de historia relativos.

Véase también las secciones 9.3.2.1, 9.4.2.1, 9.5.2.1 y 9.6.2.1.

9.2.2.2 Reuniones de revisión del Backlog Priorizado del Producto

Las reuniones de revisión del Backlog Priorizado del Producto se llevan a cabo como parte del proceso de *Refinar el Backlog Priorizado del Producto*. A medida que se refinan las historias de usuario nuevas o actualizadas en el backlog, el Equipo Scrum asignará o actualizará estimaciones de alto nivel para cada historia de usuario. Se pueden utilizar el tamaño relativo (*Relative Sizing*), o los puntos de historia para estimar el tamaño general de una historia de usuario o característica. Este método asigna un punto de valor de historia con base en una evaluación general del tamaño de una historia de usuario tomando en cuenta el riesgo, la cantidad de esfuerzo necesario y el nivel de complejidad. Dicha evaluación la llevará a cabo el Equipo Scrum y se asignará un valor de punto de historia. Una vez hecha la evaluación en una historia de usuario en el Backlog Priorizado del Producto, el Equipo Scrum puede entonces evaluar otras historias de usuario relativas a esa primera historia.

9.2.2.3 Métodos de estimación*

Se pueden utilizar numerosos métodos de estimación para estimar historias de usuario. Algunas herramientas importantes son:

1. Wideband Delphi

El Wideband Delphi es una técnica de estimación basada en grupo para determinar la cantidad de trabajo necesario y el tiempo que tardará en completarse. Los individuos en el equipo proporcionan estimaciones anónimas para cada artículo y las estimaciones iniciales se trazan en una gráfica. Posteriormente, el equipo analiza los factores que influyeron en sus estimaciones y proceden a una segunda ronda de estimación. Este proceso se repite hasta que las estimaciones de los individuos quedan cerca una de la otra y se puede llegar a un consenso para la estimación final.

2. Planning Poker

El Planning Poker, conocido también como Estimation Poker, es un derivado de la técnica Wideband Delphi. Esta técnica de estimación implementa el consenso para estimar los tamaños relativos de las historias de usuario o el trabajo necesario para desarrollarlos.

En el Planning Poker, a cada miembro del equipo se le asigna una baraja. Cada carta está enumerada en forma secuencial y los números representan la complejidad del problema en términos de tiempo o esfuerzo, según lo estimado por el miembro del equipo. Los miembros del Equipo Scrum evalúan el artículo (tarea o historia de usuario) e intentan entenderlo mejor antes de brindar su estimación para su desarrollo. Después, cada miembro elige una carta de la baraja que represente su estimación para la historia de usuario. Si la mayoría, o todos los miembros del equipo seleccionan la misma carta, entonces el cálculo que indique la carta será el estimado para el artículo. Si no hay un consenso, entonces los integrantes del equipo discuten las razones de la selección de distintas cartas o estimaciones. Después del análisis seleccionan nuevamente las cartas. Esta secuencia continúa hasta que se entienden todas las presuposiciones; hasta que se resuelven los malentendidos o hasta que se llega a un consenso o a un acuerdo.

El Planning Poker promueve una mayor interacción y una mejor comunicación entre los participantes. Facilita el pensamiento independiente por parte de los participantes, evitando con ello el fenómeno del pensamiento en grupo.

3. Puño de cinco (*Fist of Five*)

El puño de cinco, o *Fist of Five*, es un mecanismo sencillo y rápido que se puede utilizar como práctica de estimación, así como técnica general de formación de consenso colectivo. Tras el debate inicial sobre sobre la estimación de un elemento, se les pide a los miembros del Equipo Scrum que voten en una escala de 1 a 5 utilizando sus dedos. Al utilizarse como herramienta de estimación, el número de dedos que se muestran indican el valor relativo de estimación. Los integrantes del equipo con

estimaciones atípicas (valores más altos o más bajos) explican al grupo el motivo de su estimación para su análisis. Una vez que el equipo ha debatido, se lleva a cabo otra ronda de *Fist of Five* o se toma una decisión colectiva.

El valor en el uso de esta técnica no está sólo en lograr un consenso, sino también en el debate, ya a que a cada miembro del equipo se le pide que explique el motivo de su estimación. También se les da la oportunidad de expresar cualquier problema o inquietud. Al utilizarse como técnica para lograr un consenso general, se discute inicialmente la propuesta o la decisión pendiente que está a consideración; después, los integrantes del equipo votan con base en su nivel de participación y deseo de debatir:

Un dedo: No estoy de acuerdo con la conclusión del grupo y tengo grandes inquietudes.

Dos dedos: No estoy de acuerdo con la conclusión del grupo y me gustaría hablar sobre algunos asuntos menores.

Tres dedos: No estoy seguro y me gustaría sumarme a la conclusión de consenso del grupo.

Cuatro dedos: Estoy de acuerdo con la conclusión del grupo y me gustaría discutir algunos asuntos menores.

Cinco dedos: Estoy totalmente de acuerdo con la conclusión del grupo.

4. Estimación por afinidad (*Affinity Estimation*)

La estimación por afinidad (del inglés: *Affinity Estimation*) es una técnica que se utiliza para estimar rápidamente un gran número de historias de usuarios con el uso de categorías. Utilizando notas adhesivas o fichas y cinta, cada equipo coloca las historias de usuario en la pared o en cualquier otra superficie en orden desde la más pequeña hasta la más grande. Para ello, cada integrante del equipo inicia con un subconjunto de historias de usuario de todo el Backlog Priorizado del Producto para colocarse por tamaño relativo. Esta colocación inicial se hace en silencio. Una vez que todos han colocado en la pared sus historias de usuario, el equipo las revisa y las puede mover según sea necesario. Esta segunda parte del ejercicio incluye discusiones. Por último, el Product Owner indicará en la pared algunas categorías de tamaño. Dichas categorías pueden ser pequeñas, medianas o grandes, o bien, pueden estar enumeradas utilizando valores de punto de la historia (*Point Story Values*) para indicar el tamaño relativo. Después el equipo reubicará las historias de usuario en dichas categorías en el paso final del proceso. Algunos de los beneficios claves de este método son que el proceso es muy transparente, visible para todos y fácil de llevar a cabo.

9.2.3 Salidas

9.2.3.1 Historias del usuario estimadas *

Después de que las historias de usuario han sido estimadas por el Equipo Scrum con el uso de las distintas técnicas de estimación analizadas en esta sección, se ponen a consideración para ser historias de usuario estimadas. Se puede utilizar el *Relative Sizing* o los puntos de historia para estimar el tamaño general de una historia de usuario o característica. Este método asigna un valor de punto de historia con base en una evaluación general del tamaño de una historia de usuario tomando en cuenta el riego, la cantidad de esfuerzo necesario y el nivel de complejidad. Esta evaluación la llevará a cabo el Equipo Scrum y se asignará un valor de punto de historia. Una vez hecha la evaluación en una historia de usuario en el Backlog Priorizada del Producto, el Equipo Scrum puede entonces evaluar otras historias de usuario relativas a esta primera historia. Debe destacarse que debido a que la calibración del punto de usuario por cada equipo será diferente, el número de puntos de historia completados pudiera utilizarse para hacer un comparativo.

9.3 Comprometer historias de usuario

La figura 9-7 muestra todas las entradas, las herramientas y las salidas del proceso de *Comprometer historias de usuario*.



Figura 9-7: Comprometer tareas—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

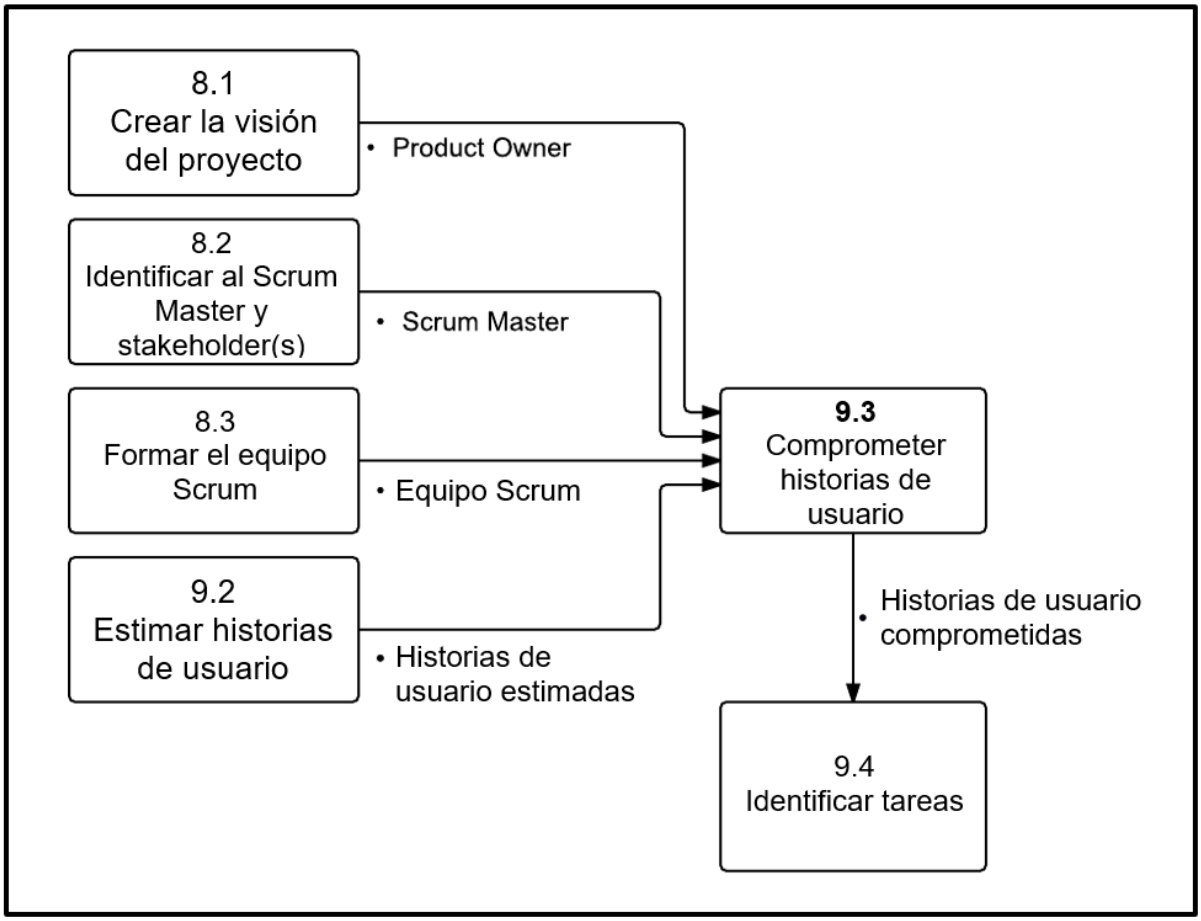


Figura 9-8: Comprometer historias de usuario—Diagrama de flujo de datos

9.3.1 Entradas

9.3.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

9.3.1.2 Historias de usuario estimadas*

Descrito en la sección 9.2.3.1.

9.3.1.3 Duración del sprint*

Descrito en la sección 8.6.3.2.

9.3.1.4 Velocidad del sprint anterior

La velocidad del sprint es la velocidad en la que el equipo puede completar el trabajo en un sprint. Por lo general se expresa en las mismas unidades que se utilizan para la estimación, normalmente en puntos de historia o tiempo ideal. Se lleva un registro de la velocidad del sprint del equipo para cada sprint y se utiliza como referencia para futuros sprints. La velocidad del sprint anterior se convierte en el factor más importante para determinar la cantidad del trabajo a la que se avocará el equipo en un subsecuente sprint. Cualquier cambio en la situación o en las condiciones a partir del último sprint, se toma en cuenta para garantizar un cálculo preciso de la velocidad del sprint para el siguiente.

9.3.1.5 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11

En el proceso de *Comprometer historias de usuario*, el Scrum Guidance Body puede incluir información sobre normas, regulaciones, estándares y las mejores prácticas necesarias para comprometer eficazmente las historias de usuario.

9.3.2 Herramientas

9.3.2.1 Reuniones de planificación de tareas*

En las reuniones de planificación de tareas, el Equipo Scrum se reúne para planificar el trabajo que se hará en el sprint. El equipo revisa las historias de usuario asignadas que encabezan el Backlog Priorizado del

Producto. El Product Owner se encuentra presente durante las reuniones en caso de ser necesaria una aclaración relacionada a las historias de usuario o en las prioridades. Para ayudar a garantizar que el equipo no se salga del tema, la reunión debe de tener un time-box con una duración estándar limitada de dos horas por semana de duración del sprint. Esto ayuda a prevenir la tendencia de desviarse hacia discusiones que deberían de realizarse en otras reuniones, tales como en las de planificación del lanzamiento o en reuniones de revisión del sprint. Como parte de esta reunión, todo el Equipo Scrum se comprometerá a entregar un subconjunto de historias de usuario del Backlog Priorizado del Producto en el sprint.

Véase también las secciones 9.2.2.1, 9.4.2.1, 9.5.2.1 y 9.6.2.1.1

9.3.2.2 Técnicas de comunicación*

Scrum promueve la comunicación precisa y eficaz primordialmente mediante la co-ubicación de equipos Scrum. Scrum favorece también las interacciones informales cara a cara en vez de la comunicación formal por escrito. Cuando un Equipo Scrum debe estar distribuido geográficamente, el Scrum Master debe asegurar que existan técnicas eficaces de comunicación para que los equipos se puedan auto organizar y trabajar eficientemente.

9.3.3 Salidas

9.3.3.1 Historias de usuario comprometidas*

El Equipo Scrum se compromete a un subconjunto de historias de usuario estimadas que consideran que se pueden completar en el siguiente sprint con base en la velocidad. Las historias de usuario comprometidas se seleccionarán siempre según las prioridades definidas por el Product Owner.

9.4 Identificar tareas

La figura 9-9 muestra todas las entradas, las herramientas y las salidas para el proceso de *Identificar tareas*.

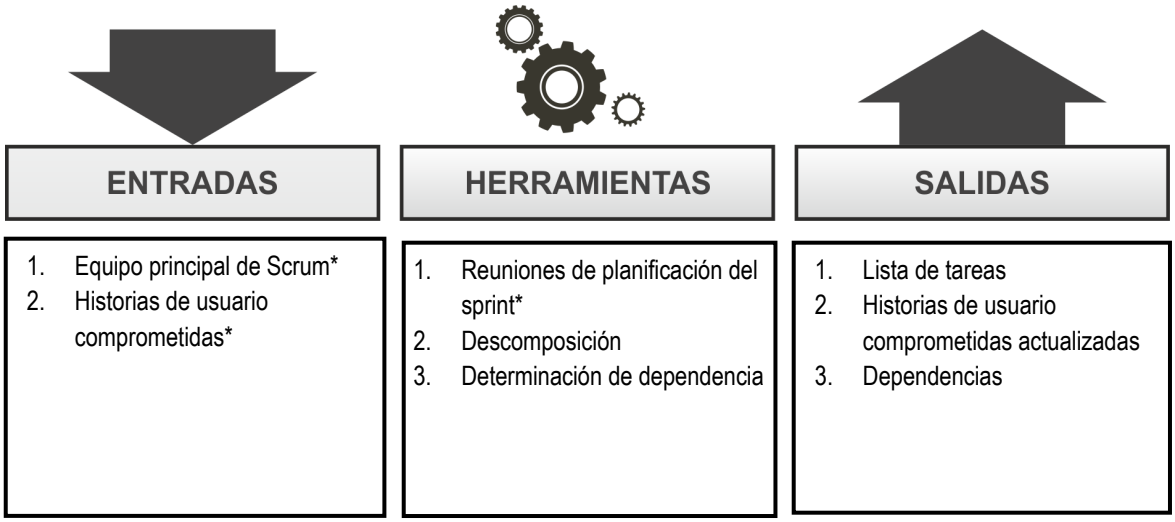


Figura 9-9: Identificar tareas—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

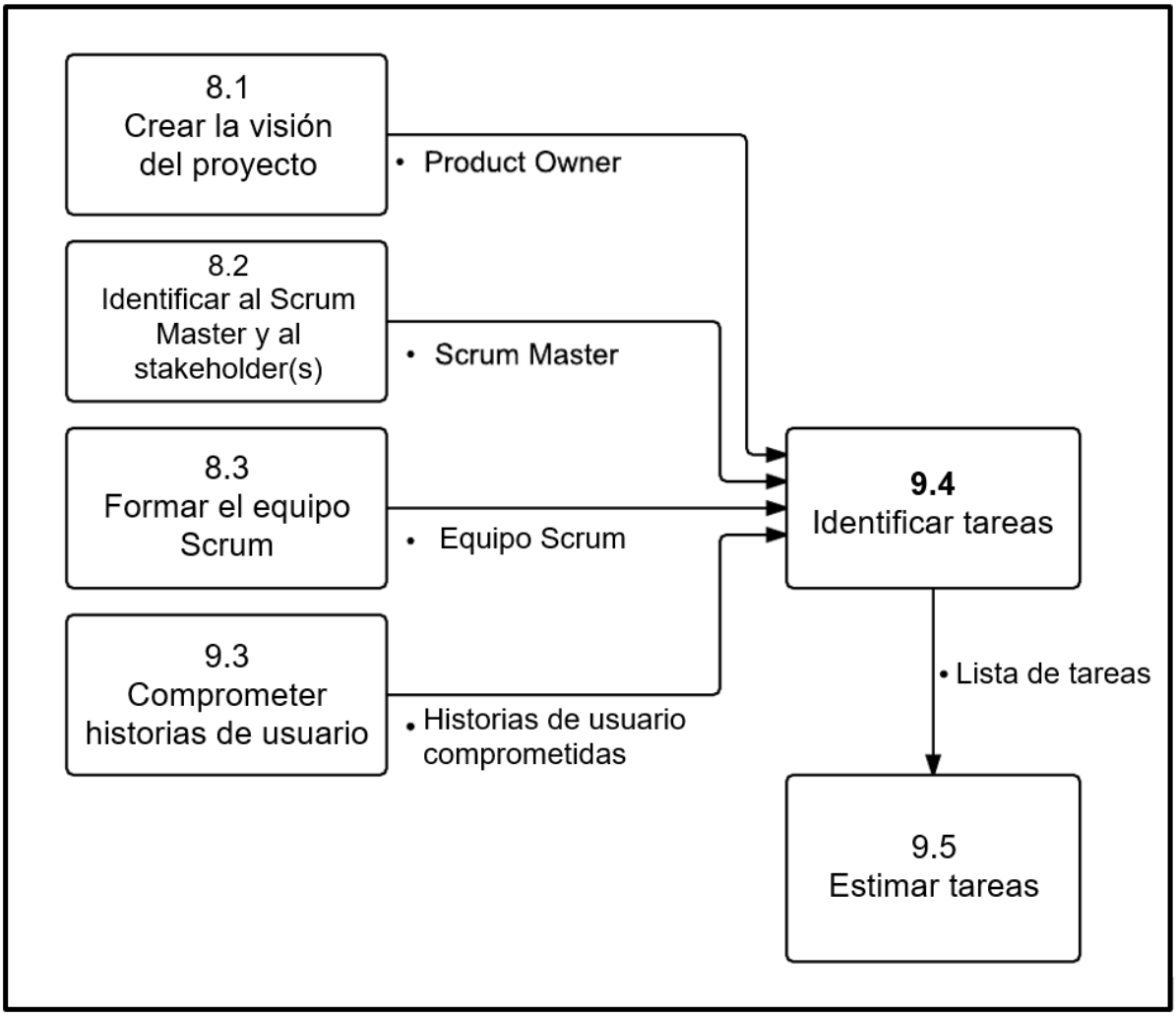


Figura 9-10: Estimar tareas—Diagrama de flujo de datos

9.4.1 Entradas

9.4.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

9.4.1.2 Historias de usuario comprometidas*

Descrito en la sección 9.3.3.1.

9.4.2 Herramientas

9.4.2.1 Reuniones de planificación del sprint*

En las reuniones de planificación del sprint, el Equipo Scrum se reúne para planear el trabajo a realizar en sprint. El equipo revisa cada historia de usuario comprometida para el sprint e identifica actividades accionables o las tareas necesarias para implementar los entregables necesarios para cumplir totalmente y cumplir los criterios de aceptación de usuarios de usuario. El Product Owner se encuentra presente durante esta reunión en caso de que se necesita una aclaración relacionada a las historias de usuario comprometidas a fin de ayudar al equipo a tomar decisiones sobre diseño.

Véase también las secciones 9.2.2.1, 9.3.2.1, 9.5.2.1 y 9.6.2.1.

9.4.2.2 Descomposición*

La descomposición (también conocida como “segmentación”) es una herramienta donde las tareas de alto nivel se dividen en niveles inferiores y más detallados. Los miembros del Equipo Scrum segmentan las historias de usuario en tareas. Las historias de usuario del Backlog Priorizado del Producto deben estar suficientemente segmentadas a un nivel que le brinde al Equipo Scrum la información adecuada para crear entregables a partir de las tareas mencionadas en la lista de tareas.

9.4.2.3 Determinación de dependencia

Una vez que el Equipo Scrum ha seleccionado las historias de usuario para un determinado sprint, deben considerar las dependencias, incluyendo aquellas relacionadas a la disponibilidad de personal, así como cualquier dependencia técnica. Documentar adecuadamente las dependencias ayuda a los equipos Scrum a determinar el orden relativo en el cual deben ejecutarse las tareas para crear los entregables del sprint.

Las dependencias destacan también la relación e interacción entre las tareas dentro del Equipo Scrum que trabaja en un determinado Sprint y con otros equipos Scrum en el proyecto.

Existen numerosos tipos de dependencias: obligatorias y discrecionales; internas y externas, o alguna combinación de estas. Por ejemplo, una dependencia puede ser tanto obligatoria como externa.

- **Dependencias obligatorias**—Dependencias que son inherentes en la naturaleza del trabajo (como una limitación física) o pueden darse debido a obligaciones contractuales o por requerimientos legales. Por ejemplo, el trabajo en el primer piso no puede iniciar hasta que se hayan terminado los cimientos del edificio. A las dependencias obligatorias también se les llama comúnmente como "lógica dura" (*hard logic*).

- **Dependencias discrecionales**—Las dependencias discrecionales son aquellas que se colocan en el flujo de trabajo por decisión propia. Normalmente, el Equipo Scrum determina cuáles son las dependencias discrecionales con base en experiencias anteriores o las mejores prácticas en un campo o dominio específico. Por ejemplo, el equipo puede optar por completar una tarea antes de trabajar en otra ya que sería una mejor práctica, aunque no obligatoria. Por ejemplo, el equipo pudiera optar por construir marcos de puertas y ventanas antes de toda la pared esté en pie.

- **Dependencias externas**—Las dependencias externas son aquellas que están relacionadas a las tareas, actividades o productos fuera del alcance del trabajo a realizar por el Equipo Scrum, pero que son necesarias para concluir una tarea del proyecto o crear un entregable del proyecto. Las dependencias externas generalmente están fuera del alcance del Equipo Scrum. Por ejemplo, si el Equipo Scrum no está a cargo de la adquisición de los materiales necesarios para la construcción de paredes, entonces dichos materiales y las tareas relacionadas a su adquisición se considerarían dependencias externas.

- **Dependencias internas**—Las dependencias internas son aquellas entre las tareas, productos o actividades que están bajo el control del Equipo Scrum y dentro de un enfoque de trabajo a ser ejecutado por dicho equipo. Por ejemplo, la instalación de paneles de yeso debe ser completada antes de pintar la pared. Este es un ejemplo de una dependencia interna ya que ambas tareas son parte del proyecto. En este caso, también es obligatorio, ya que se basa en una limitación física. No es posible pintar la pared antes qué la pared esté seca.

9.4.3 Salidas

9.4.3.1 Lista de tareas*

Es una lista integral que contiene todas las tareas a las que se ha comprometido el Equipo Scrum en el actual sprint. Contiene descripciones de cada tarea, así como las estimaciones obtenidas durante el proceso de *Identificar tareas*. La lista de tareas, o *Task List*, debe incluir cualquier prueba o actividad de

integración a fin de que el incremento del producto del sprint puede integrarse con éxito en los entregables de previos sprints.

Aun cuando las tareas generalmente se basan en tareas, el nivel de granularidad al que se segmentan las tareas lo decide el Equipo Scrum.

9.4.3.2 Historias de usuario comprometidas actualizadas

Durante este proceso se actualizan las historias de usuario. Las actualizaciones pueden incluir modificaciones a las estimaciones de las historias de usuario con base en la creación de tareas y en factores de complejidad analizados durante la reunión de planificación del sprint. Las historias de usuario comprometidas se describen en la sección 9.3.3.1.

9.4.3.3 Dependencias

Las dependencias describen la relación y la interacción entre las distintas tareas en un proyecto y pueden clasificarse como obligatorias o discrecionales; internas o externas, tal como se describe en la sección 9.4.2.3.

Existen numerosas formas de identificar, definir y presentar las tareas y sus dependencias. Dos métodos comunes incluyen el uso de diagramas de flujo de producto y diagramas de Gantt.

9.5 Estimar tareas

La figura 9-11 muestra todas las entradas, las herramientas y las salidas para el proceso de *Estimar tareas*.

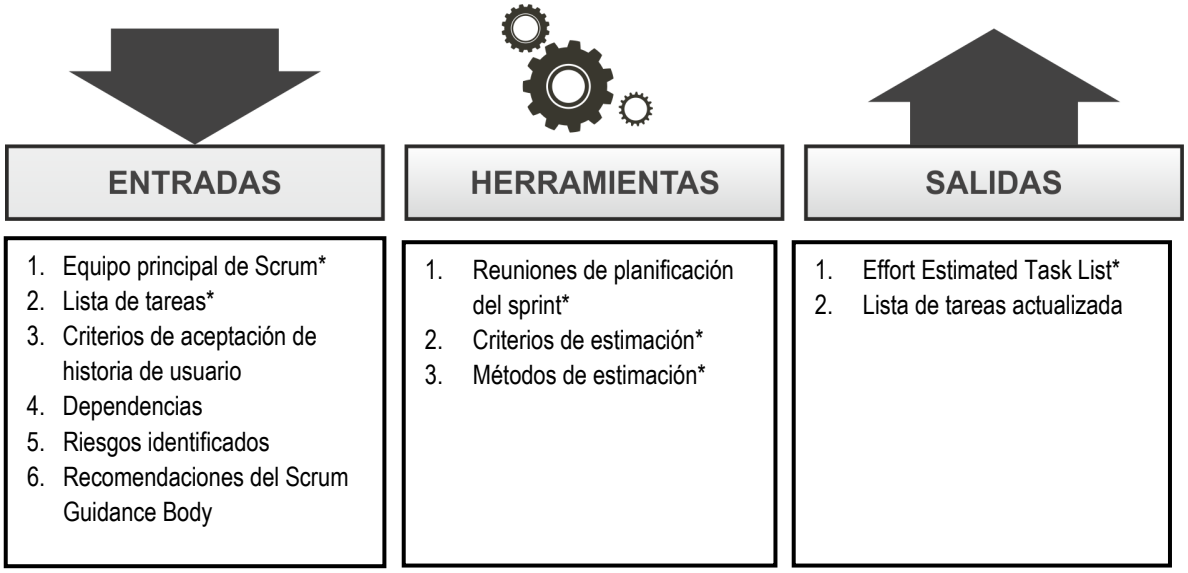


Figura 9-9: Estimar tareas—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

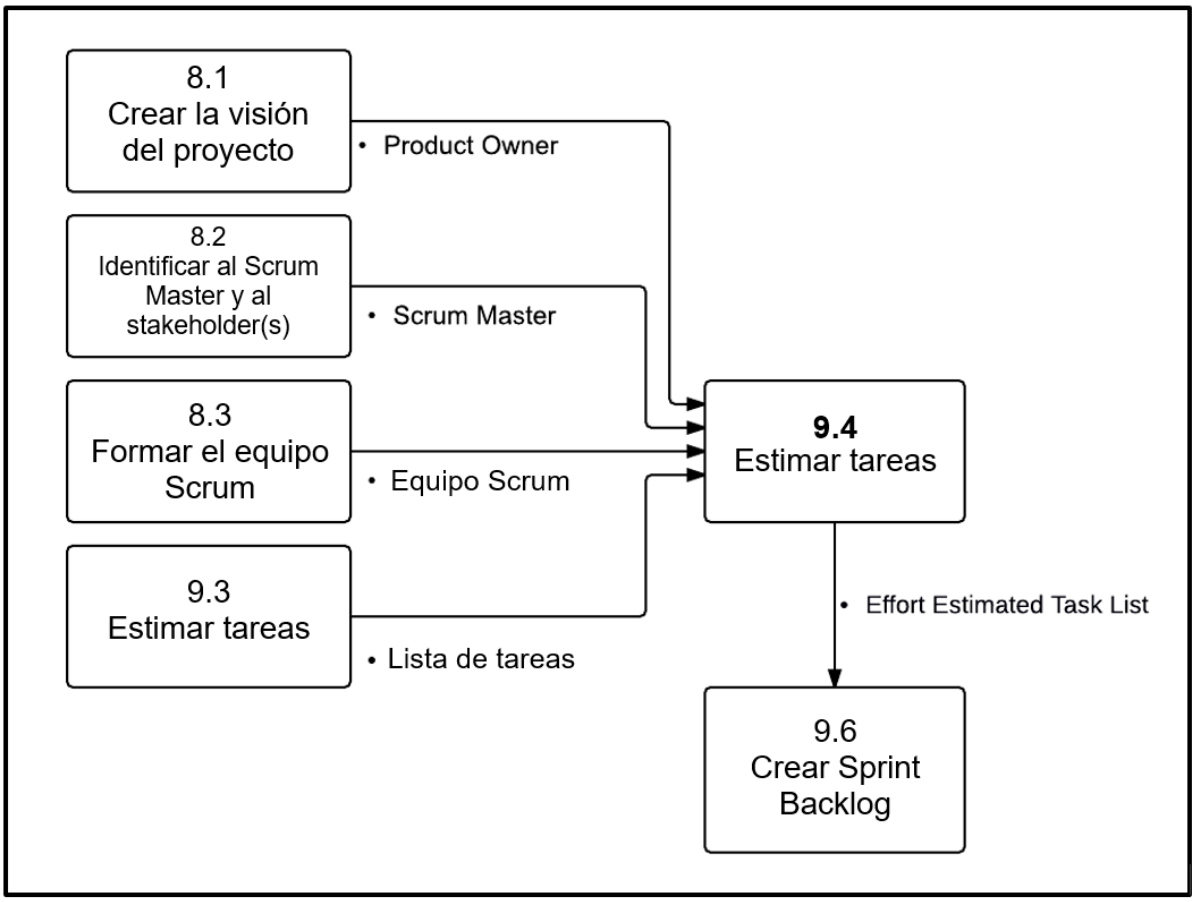


Figura 9-12: Estimar tareas—Diagrama de flujo de datos

9.5.1 Entradas

9.5.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

9.5.1.2 Lista de tareas*

Descrito en la sección 9.4.3.1.

9.5.1.3 Criterios de aceptación de historias de usuario*

Descrito en la sección 9.1.3.2.

El Equipo Scrum debe asegurarse de que los criterios de aceptación definidos sean apropiados para las historias de usuario y brinden claridad sobre los requerimientos para el Equipo Scrum. Las pruebas de aceptación son la evaluación de la capacidad con la que cuenta el entregable completado para cumplir sus criterios de aceptación. Esto proporciona información al Product Owner para ayudarle a tomar una decisión sobre la aprobación o rechazo de entregable.

Al momento de desarrollar los criterios de aceptación de historias de usuario, se debe tener en cuenta lo siguiente:

- Los criterios de aceptación no deben ser confusos, ambiguos o demasiado generalizados.
- Los criterios de aceptación definidos deben garantizar que el equipo pueda verificar que los resultados estén alineados con las metas y objetivos de la organización que patrocina.

9.5.1.4 Dependencias

Descrito en la sección 9.4.3.3.

9.5.1.5 Riesgos identificados

Descrito en la sección 8.4.3.4.

9.5.1.6 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de *Estimar tareas*, las recomendaciones del Scrum Guidance Body puede incluir información sobre normas, regulaciones, estándares y mejores prácticas para estimar eficazmente las tareas en la lista de tareas.

9.5.2 Herramientas

9.5.2.1 Reuniones de planificación del sprint*

Como parte de las reuniones de planificación del sprint, el Equipo Scrum estima el esfuerzo necesario para completar una tarea o serie de tareas y estimar el esfuerzo en cuestión de personal y recursos necesario para llevar a cabo las tareas en un determinado sprint. Los miembros del Equipo Scrum utilizan la lista de tareas para estimar la duración y el esfuerzo para las historias de usuario que serán completadas en el sprint.

Uno de los beneficios clave de esta técnica es que permite al equipo contar con una perspectiva compartida de las historias de usuario y los requerimientos de forma que pueda estimar de forma viable el esfuerzo requerido.

Véase también las secciones 9.2.2.1, 9.3.2.1, 9.4.2.1 y 9.6.2.1.

9.5.2.2 Criterios de estimación

Los criterios de estimación pueden expresarse de muchas formas. Dos ejemplos comunes son los puntos de historia y el tiempo ideal. Los valores de puntos de historia se utilizan para representar el esfuerzo relativo o comparativo para completar tareas. Mientras que el tiempo ideal normalmente describe el número de horas que los miembros de un Equipo Scrum trabajan exclusivamente en el desarrollo de los entregables del proyecto sin incluir ningún tiempo dedicado a otras actividades o a trabajo ajeno al proyecto. Los criterios de estimación le facilitan al Equipo Scrum estimar el esfuerzo y le permiten evaluar y atender las ineficiencias cuando es necesario.

9.5.2.3 Métodos de estimación*

Descrito en la sección 9.2.2.3.

Los mismos métodos de estimación utilizados para estimar historias de usuario se pueden aplicar también a las tareas.

9.5.3 Salidas

9.5.3.1 Effort Estimated Task Last*

La llama Effort Estimated Task List (lista de tareas del esfuerzo estimado) es una lista de tareas asociadas con las historias de usuario incluidas en un sprint. Típicamente la precisión de las estimaciones varía dependiendo de las habilidades del equipo El esfuerzo estimado se expresa en términos de los criterios de estimación acordados por el equipo. El Equipo Scrum utiliza la Effort Estimated Task List durante las reuniones de planificación del sprint a fin de crear el Sprint Backlog y el Sprint Burndown Chart. Se utiliza también para determinar cuándo el equipo necesita reducir su compromiso o asumir historias de usuario adicionales durante la planificación del sprint.

9.5.3.2 Lista de tareas actualizada*

La lista de tareas, desarrollada como parte del proceso de *Identificar tareas*, incluyendo las estimaciones iniciales de historias de usuario que deben revisarse con base en actividades de estimación más detalladas llevadas a cabo en el proceso de *Estimar tareas*. También pueden ser re-estimaciones que resulten después de revisar los primeros sprints o cambiar el entendimiento colectivo del Equipo Scrum sobre los requerimientos de las historias de usuario.

9.6 Crear el Sprint Backlog

La figura 9-12 muestra todas las entradas, herramientas y salidas del proceso de *Crear el Sprint Backlog*.

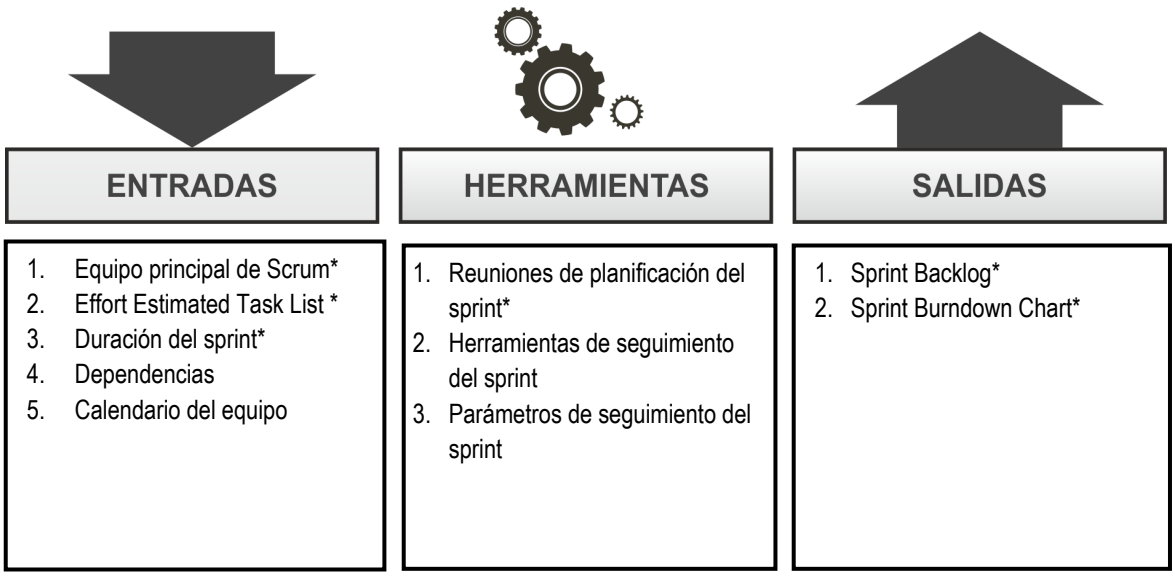


Figura 9-10: Crear el Sprint Backlog—Entradas, herramientas y salidas

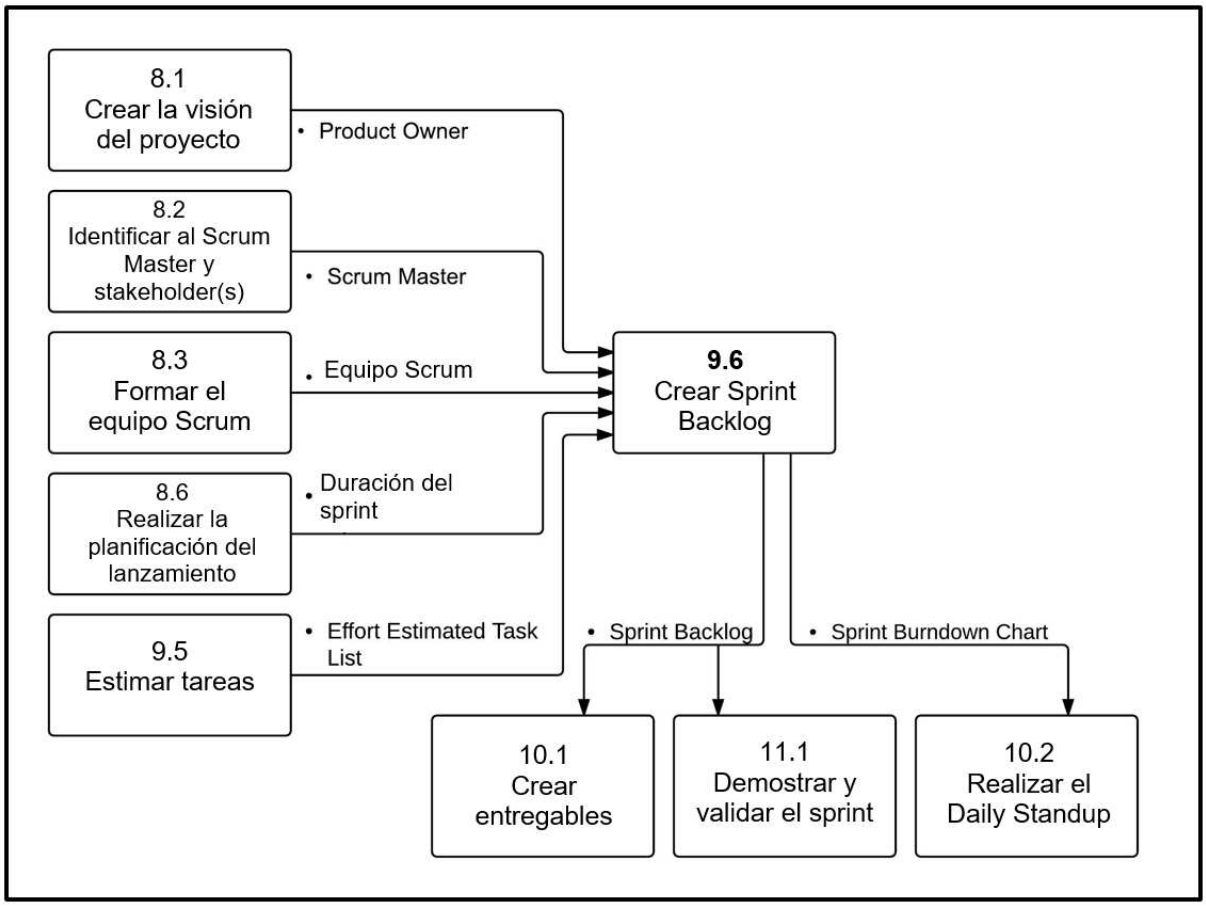


Figura 9-14: Crear el Sprint Backlog—Diagrama de flujo de datos

9.6.1 Entradas

9.6.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1

9.6.1.2 Effort Estimated Task List

Descrito en la sección 9.5.3.1

9.6.1.3 Duración del sprint*

Descrito en la sección 8.6.3.2.

9.6.1.4 Dependencias

Descrito en la sección 9.4.3.3.

9.6.1.5 Calendario del equipo

El calendario del equipo contiene información sobre la disponibilidad de los miembros, incluyendo información relacionada a las vacaciones de los empleados, licencias, ausencias temporales y días festivos.

Uno de los principales objetivos del uso del calendario del equipo es dar seguimiento al trabajo que realiza cada integrante del equipo durante el proyecto. Ayuda al equipo no solo a planificar y llevar a cabo eficientemente los sprints, sino que ayuda a alinearlos con las fechas de lanzamiento.

9.6.2 Herramientas

9.6.2.1 Reuniones de planificación del sprint

Durante las reuniones de planificación del sprint, el Equipo Scrum se compromete las historias de usuario para un sprint e identifica y estima las tareas. Cada miembro del Equipo Scrum también utiliza la Effort Estimated Task List para seleccionar las tareas en las que planean trabajar en el sprint con base en sus habilidades y experiencia. El Equipo Scrum elabora también el Sprint Backlog y el Sprint Burndown Chart utilizando las historias de usuario y la lista antes mencionada durante las reuniones de planificación del sprint.

Véase también las secciones 9.2.2.1, 9.3.2.1, 9.4.2.1 y 9.5.2.1.

9.6.2.2 Herramientas de seguimiento del sprint

Es importante rastrear el proceso de un sprint y saber dónde se encuentra el Equipo Scrum en términos de conclusión de tareas el Sprint Backlog. Se puede utilizar una variedad de herramientas para dar seguimiento al trabajo de un sprint, pero una de las más comunes es Scrumboard, conocido también como tablero de tareas o gráfica de proceso. La versión más básica de un Scrumboard se divide en tres secciones: Por hacer, o *To Do* (conocido también como "Trabajo no iniciado"); Trabajo en progreso (*In Progress*) y Trabajo concluido (*Completed Work*). Se colocan notas adhesivas en la categoría correspondiente para representar cada tarea o historia de usuario a fin de reflejar el estado actual de las actividades. Se reacomodan en la siguiente categoría confirme avanza el trabajo.

9.6.2.3 Parámetros de seguimiento del sprint

Los parámetros que se utilizan en los proyectos Scrum incluyen: velocidad, valor empresarial entregado y cantidad de historias.

Velocidad: Representa el número de historias de usuario o de funcionalidades entregadas en un solo sprint.

Valor empresarial entregado: Mide el valor de las historias de usuario entregadas desde la perspectiva del negocio.

Número de historias: Describe cuántas historias de usuario se entregan como parte de un solo sprint. Se puede expresar en términos de un simple conteo o de un conteo ponderado.

9.6.3 Salidas

9.6.3.1 Sprint Backlog*

La lista de tareas que llevará a cabo el Equipo Scrum en el siguiente sprint se denomina Sprint Backlog. Es común que el Sprint Backlog esté representado en un Scrumboard o tablero de tareas, el cual proporciona una constante representación visual del estado de las historias de usuario en el backlog. En el Sprint Backlog también se incluye cualquier riesgo asociado a las varias tareas. Cualquier actividad de mitigación de riesgos para atender los riesgos identificados también se incluirían como tareas en el Sprint Backlog.

Una vez que el Equipo Scrum finaliza y se compromete al Sprint Backlog no se deben agregar nuevas historias de usuario; sin embargo, las tareas que pudieron haberse pasado por alto o ignoradas de las historias de usuario comprometidas pudieran ser agregadas. Si durante un sprint surgen nuevos requerimientos, estos serán agregados al Backlog Priorizado del Producto e incluidos en un futuro sprint.

9.6.3.2 Sprint Burndown Chart*

El Sprint Burndown Chart es una gráfica que muestra la cantidad de trabajo pendiente en el actual sprint. El Sprint Burndown Chart inicial se acompaña de un Planned Burndown. El Sprint Burndown Chart debe actualizarse al final de cada día conforme se concluye el trabajo. Dicha gráfica muestra el progreso que ha realizado el Equipo Scrum y permite también la detección de estimaciones que pudieron haberse hecho incorrectamente. Si el Sprint Burndown Chart muestra que el Equipo Scrum no va por el rumbo correcto en la conclusión a tiempo del sprint, el Scrum Master debe identificar cualquier obstáculo o impedimento de una conclusión satisfactoria e intentar eliminarlos.

Una tabla relacionada es el Sprint Burnup Chart. A diferencia del Sprint Burndown Chart —que muestra la cantidad de trabajo pendiente—, el Sprint Burnup Chart sprint muestra el trabajo concluido como parte del sprint.

9.7 Diagrama de flujo de datos de fase

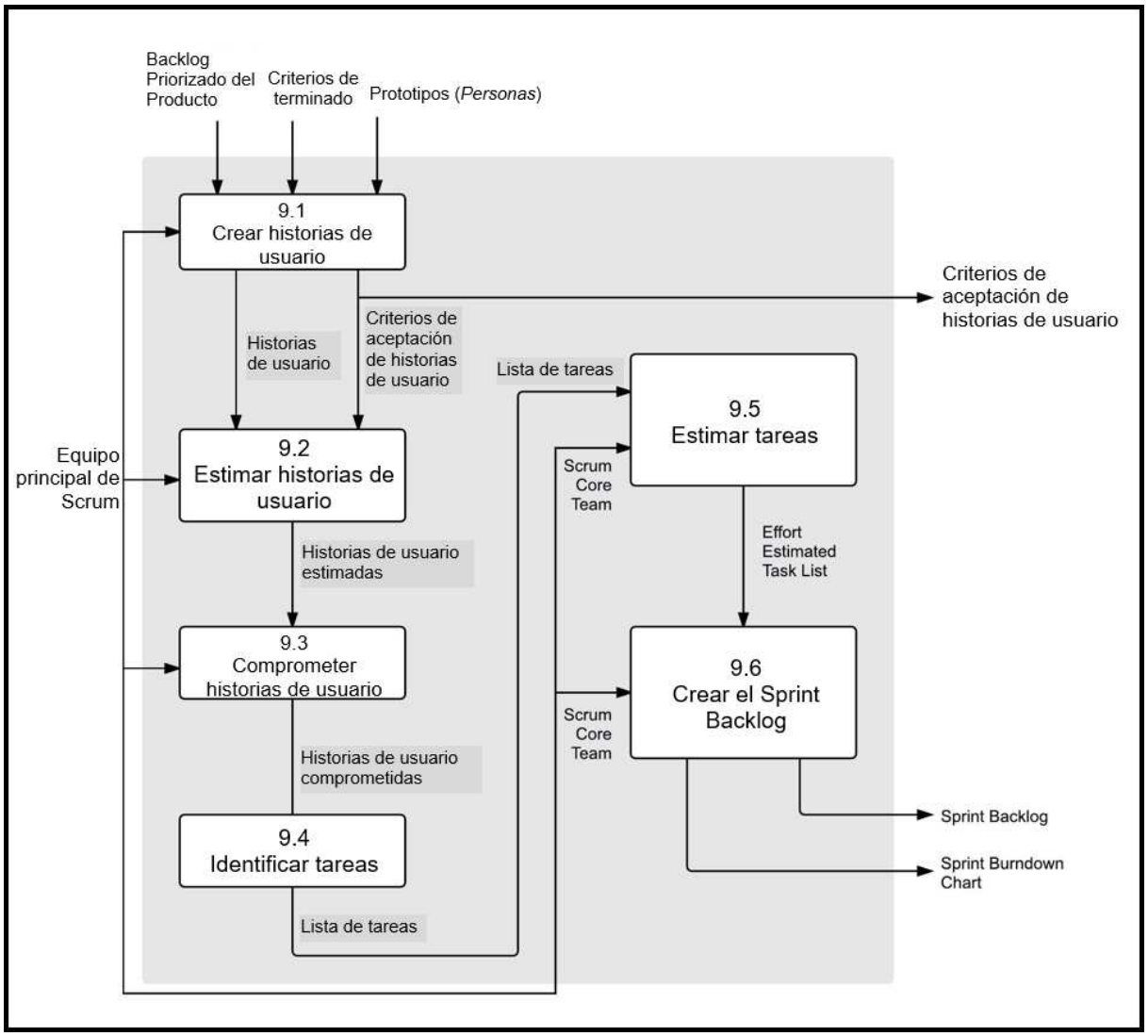


Figura 9-5: Fase de planificación y estimación—Diagrama de flujo de datos

10. IMPLEMENTACIÓN

La fase de implementación está relacionada a la ejecución de las tareas y actividades para crear el producto de un proyecto. Estas actividades incluyen la creación de varios entregables, realizar Daily Standups y el refinamiento (revisiones, ajustes y actualización periódica) del Backlog Priorizado del Producto en intervalos frecuentes.

La *implementación*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™,* no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un Equipo Scrum que trabaja en un sprint para producir entregables que pudieran ser parte de un proyecto más grande, de un programa o portafolio. La información sobre cómo escalar Scrum en grandes proyectos está disponible en el capítulo 13 y lo relacionado a escalar Scrum para la empresa se puede encontrar en el capítulo 14.

La figura 10-1 brinda una visión general de los procesos de la fase implementación, los cuales son los siguientes:

10.1 Crear entregables—En este proceso, el Equipo Scrum trabaja en las tareas del Sprint Backlog para crear los entregables del sprint. Generalmente se utiliza un Scrumboard para dar seguimiento al trabajo y a las actividades que se llevan a cabo. Los problemas que enfrenta el Equipo Scrum pueden actualizarse en el Impediment Log (o registro de impedimentos).

10.2 Realizar Daily Standup—En este proceso se lleva a cabo diariamente una reunión altamente focalizada con un time-box asignado y denominada: Daily Standup Meeting. Es un foro para que el Equipo Scrum se ponga al día sobre sus progresos y sobre cualquier impedimento que pudieran estar enfrentando.

10.3 Refinamiento del Backlog Priorizado del Producto—En este proceso constantemente se actualiza y refina el Backlog Priorizado del Producto. Se puede celebrar una reunión de revisión del Backlog Priorizado del Producto, donde los cambios y actualizaciones al backlog se analizan y se incorporan al Backlog Priorizado del Producto, según corresponda.

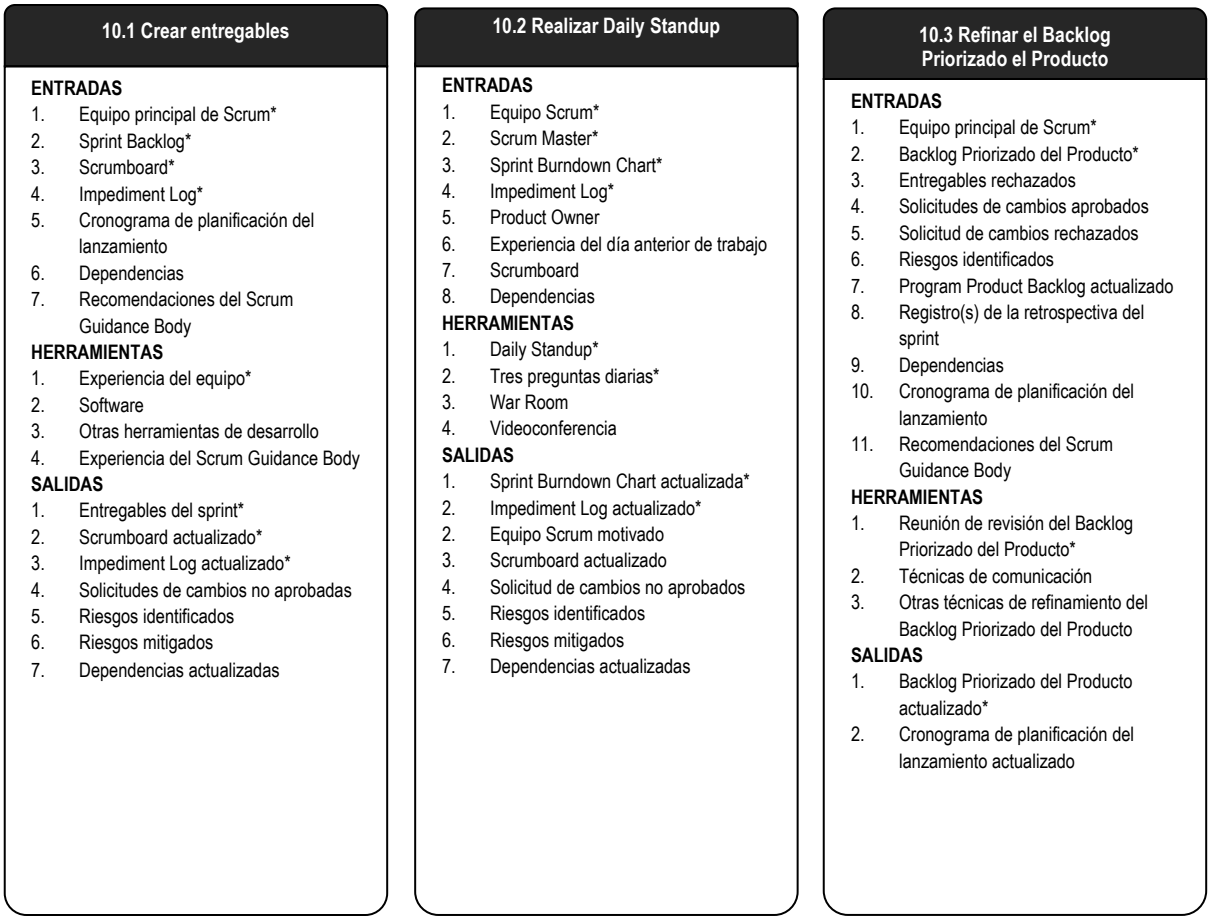


Figura 10-1: Descripción general de la Implementación

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 10-2 muestra las entradas, herramientas y salidas obligatorias para los procesos en la fase de implementación.

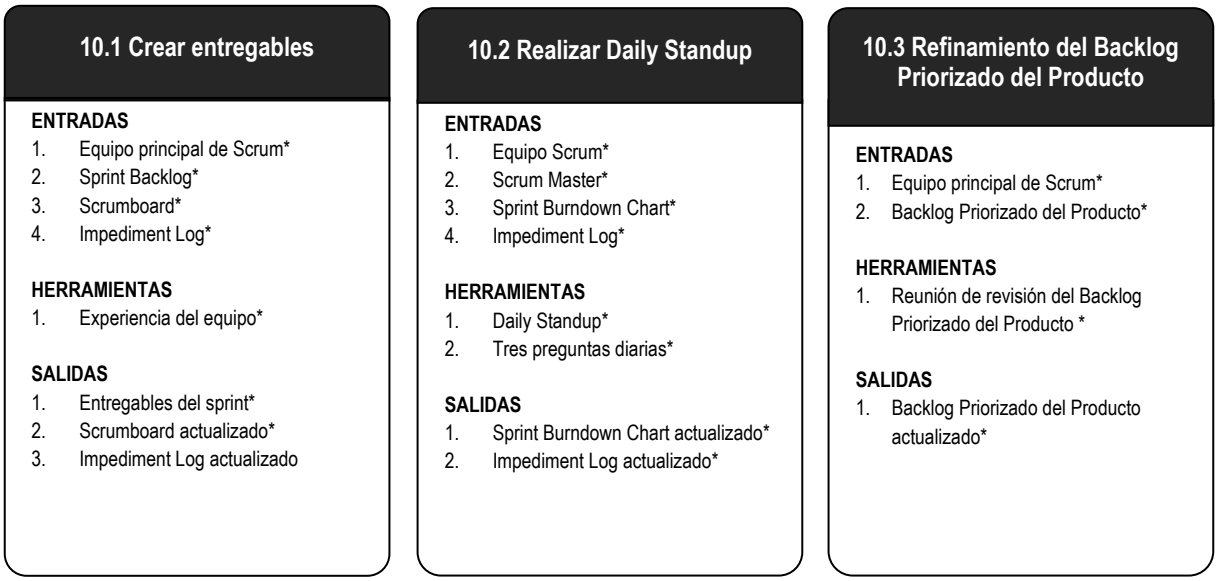


Figura 10-2: Resumen de la implementación (Esenciales)

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

10.1 Crear entregables

La figura 10-3 muestra todas las entradas, las herramientas y salidas para el proceso de *Crear entregables*.

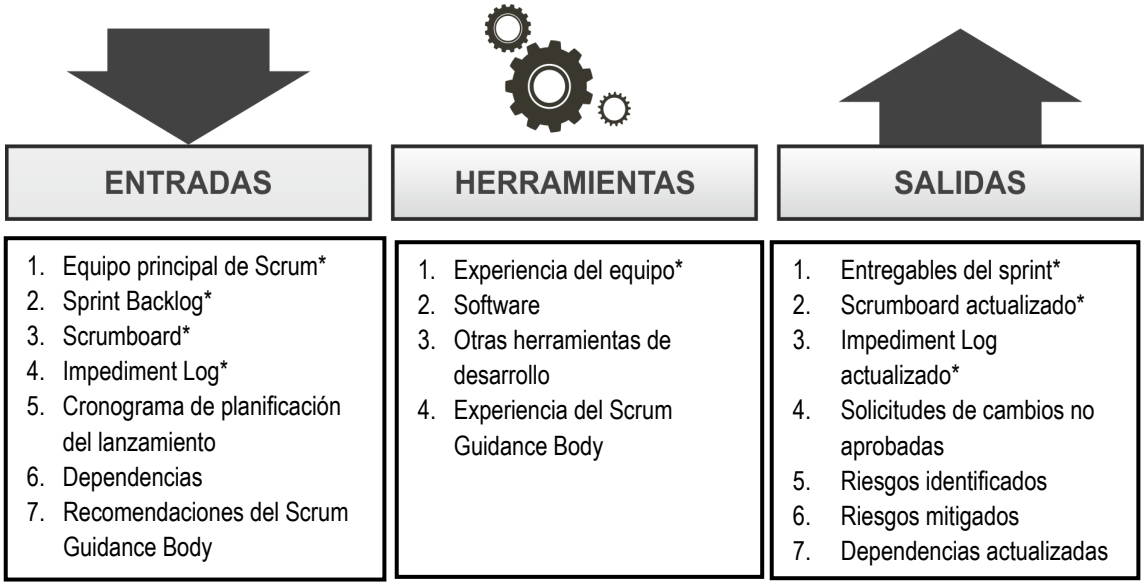


Figura 10-3: Crear entregables—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

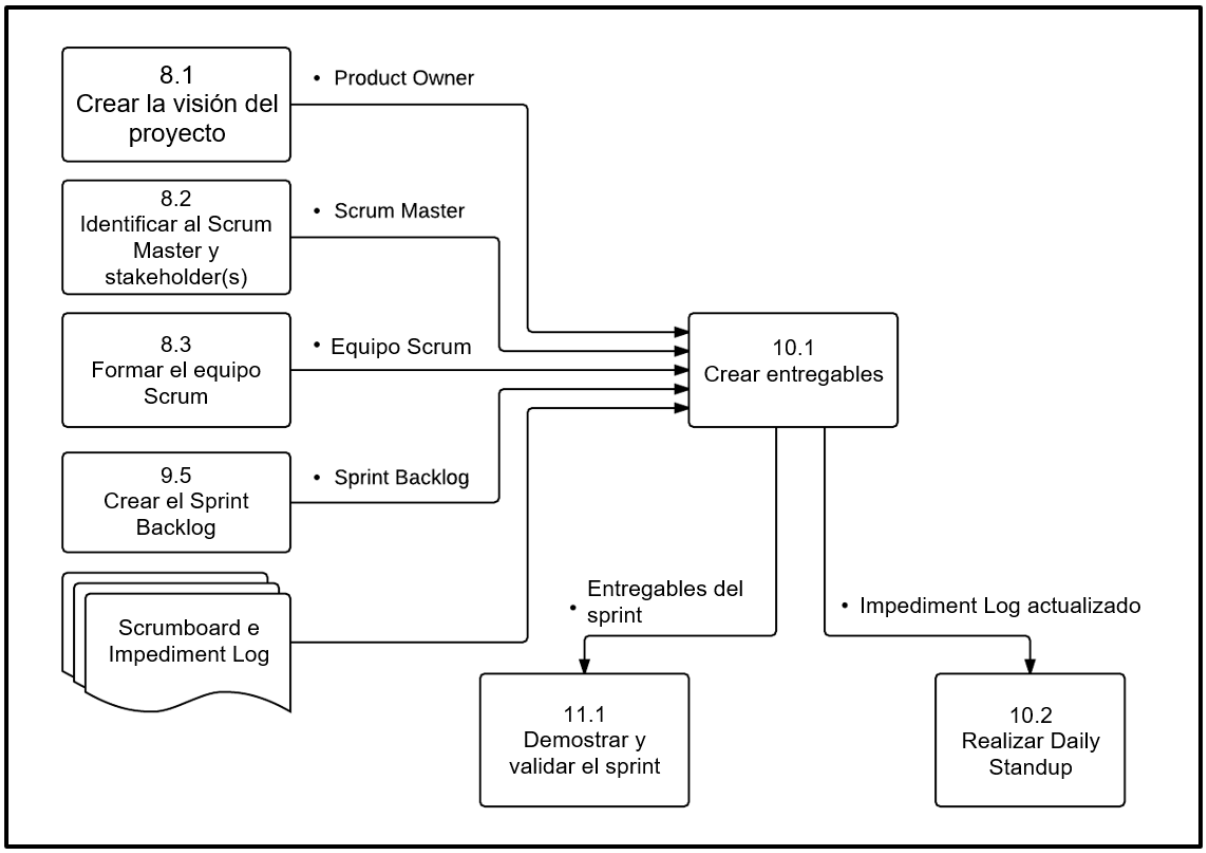


Figura 10-4: Crear entregables—Diagrama de flujo de datos

10.1.1 Entradas

10.1.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

10.1.1.2 Sprint Backlog*

Descrito en la sección 9.5.3.1.

10.1.1.3 Scrumboard*

En Scrum, la transparencia proviene de las herramientas visiblemente abiertas tales como el Scrumboard, donde se muestra el avance del equipo. El equipo utiliza un Scrumboard para planificar y dar seguimiento al progreso durante cada sprint. El tablero contiene cuatro columnas para indicar el progreso de las tareas estimadas para el sprint: una columna “por hacer” para las tareas que aún no se inician; una columna “en progreso” para las tareas que ya iniciaron, pero no se han concluido; una columna “en prueba” para las tareas concluidas pero que están en proceso de evaluación y una columna de “terminado” para las tareas que se han concluido y evaluado satisfactoriamente. Al principio de cada sprint, todas las tareas se colocan en la columna “por hacer” y avanzan según su progreso.

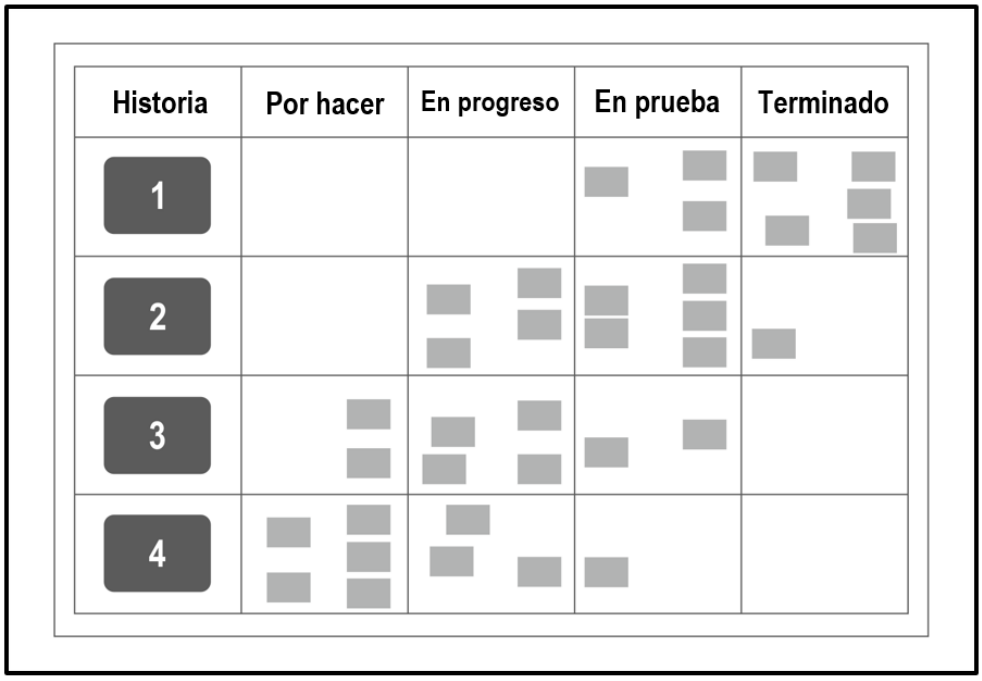


Figura 10-5: Scrumboard

El Scrumboard de preferencia debe mantenerse manualmente en papel o en un pizarrón, aunque también se puede hacer de manera electrónica en una hoja de cálculo.

El Equipo Scrum debe cambiar y agregarle al Scrumboard según sea necesario, de tal forma que brinde información visual y control sobre el trabajo en acción según lo acordado y lo comprometido por el equipo.

10.1.1.4 Impediment Log*

Un impedimento es cualquier obstáculo o barrera que reduce la productividad del Equipo Scrum. Los impedimentos deben identificarse, resolverse y eliminarse si el equipo sigue trabajando eficazmente. Los impedimentos pueden ser internos en un equipo, tales como un flujo de trabajo ineficiente o la falta de comunicación, o bien, pudieran ser externos. Algunos ejemplos de impedimentos externos pudieran ser problemas relacionados a licencias de software o requisitos de documentación innecesaria. El framework de Scrum, con su transparencia inherente, facilita la rápida y fácil identificación de impedimentos. Si no se identifican o se atienden los impedimentos, pudiera ser muy costoso. El Scrum Master debe registrar formalmente los impedimentos en un Impediment Log y se pueden analizar durante las Daily Standups y en las reuniones de revisión del sprint según sea necesario.

10.1.1.5 Cronograma de planificación del lanzamiento

Descrito en la sección 8.6.3.1.

10.1.1.6 Dependencias

Descrito en la sección 9.3.3.3.

10.1.1.7 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de creación de entregables, las recomendaciones del Scrum Guidance Body pudieran incluir las mejores prácticas para crear eficazmente los entregables, incluyendo métodos preferenciales para realizar las revisiones, la evaluación, documentación, etc.

10.1.2 Herramientas

10.1.2.1 Experiencia del equipo*

Hace referencia a la experiencia colectiva de los miembros del Equipo Scrum para entender las historias de usuario y las tareas en el Sprint Backlog a fin de crear los entregables finales. La experiencia del equipo se utiliza para evaluar las entradas necesarias a fin de ejecutar el trabajo planificado del proyecto. El juicio y la experiencia se aplican en todos los aspectos técnicos y administrativos del proyecto durante el proceso de creación de entregables. Los miembros del Equipo Scrum cuentan con la autoridad y la responsabilidad para determinar las mejores formas de convertir los elementos del Backlog Priorizado del Producto en productos finales, sin solicitar la participación de ningún stakeholder fuera del equipo. De ser necesario, el Scrum Guidance Body cuenta con experiencia disponible.

10.1.2.2 Software

Las herramientas automatizadas de software se pueden utilizar para programar, recopilar información y para la distribución. Las herramientas virtuales de colaboración también son esenciales en proyectos donde el Equipo Scrum no esté co-ubicado. Hay una variedad de herramientas automatizadas de software que permiten dar seguimiento al proceso, a recopilar información y para la distribución y que contribuyen a acelerar el proceso.

10.1.2.3 Otras herramientas de desarrollo

Con base en los requerimientos específicos del proyecto y en las especificaciones de la industria, se pueden implementar otras herramientas de desarrollo según sea necesario.

1. **Refactorización (*Refactoring*)**

   La refactorización es una herramienta específica para proyectos de software. El objetivo de esta técnica es mejorar el mantenimiento del código existente y hacerlo más sencillo, más conciso y más flexible. Refactorizar significa mejorar el diseño del código actual sin cambiar el comportamiento del mismo. Implica lo siguiente:

   - Eliminación de código repetitivo y redundante
   - Separar los métodos y las funciones en rutinas más pequeñas
   - Definir claramente las variables y los nombres de los métodos
   - Simplificar el diseño del código
   - Hacer que el código sea más fácil de entender y de modificar

La refactorización constante optimiza el diseño del código poco a poco durante cierto periodo. En última instancia, la refactorización da como resultado en un código más limpio y más fácil de mantener, preservando a la vez todas las funcionalidades.

2. **Patrones de diseño**

   Los patrones de diseño proporcionan una manera formal de registrar una resolución a un problema de diseño en un campo específico de especialización. Dichos patrones registran tanto el proceso que se utiliza como la resolución, misma que puede reutilizarse después para mejorar la tomar de decisiones y la productividad.

10.1.2.4 Experiencia del Scrum Guidance Body

Descrito en la sección 8.4.2.7.

En los procesos de *Crear entregables* y *Estimar historias de usuario*, el Scrum Guidance Body pudiera relacionarse a las normas y regulaciones documentadas, directrices de desarrollo, o bien, a los estándares y mejores prácticas (por ejemplo, orientación sobre cómo realizar revisiones o pruebas). Pudiera haber también un equipo de expertos en la materia que brinden orientación al Equipo Scrum sobre la creación de entregables. Dicho equipo puede incluir a arquitectos líderes, Senior Developers, expertos en seguridad o demás personas con experiencia.

10.1.3 Salidas

10.1.3.1 Entregables del sprint*

Al final de cada sprint se completa un incremento de producto o entregable. El entregable debe incluir todas las características y funcionalidades definidas en las historias de usuario que se incluyen en el sprint y deben haber sido evaluadas satisfactoriamente.

10.1.3.2 Scrumboard actualizado

El Scrumboard se actualiza con regularidad a medida que el equipo completa las tareas. Sin embargo, al final del sprint, el Scrumboard se reinicia o se borra y se crea un nuevo para el siguiente sprint.

10.1.3.3 Impediment Log actualizado

Descrito en la sección 10.1.1.4.

10.1.3.4 Solicitudes de cambio no aprobadas

Descrito en la sección 8.4.1.6.

10.1.3.5 Riesgos identificados

Descritos en la sección 8.4.3.4.

10.1.3.6 Riesgos mitigados

A medida que el Equipo Scrum ejecuta el trabajo de creación de entregables según las historias de usuario en el Backlog Priorizado del Producto, lleva a cabo acciones de mitigación que fueron definidas para atender cualquier riesgo identificado anteriormente. Durante todo el proceso de *Crear entregables*, el equipo documenta cualquier nuevo riesgo identificado y se toman las acciones para mitigarlo. El registro de los riesgos del proyecto (del inglés: *record of project risks*) es un documento vivo que actualiza constantemente el equipo a lo largo del proyecto a fin de reflejar el estado actual de todos los riesgos.

En la sección 7.4.3 se describe información adicional sobre la gestión de riesgos.

10.1.3.7 Dependencias actualizadas

Descritas en la sección 9.3.3.3.

10.2 Realizar el Daily Standup

La figura 10-6 muestra todas las entradas, las herramientas y las salidas del proceso para *Realizar el Daily Standup*.

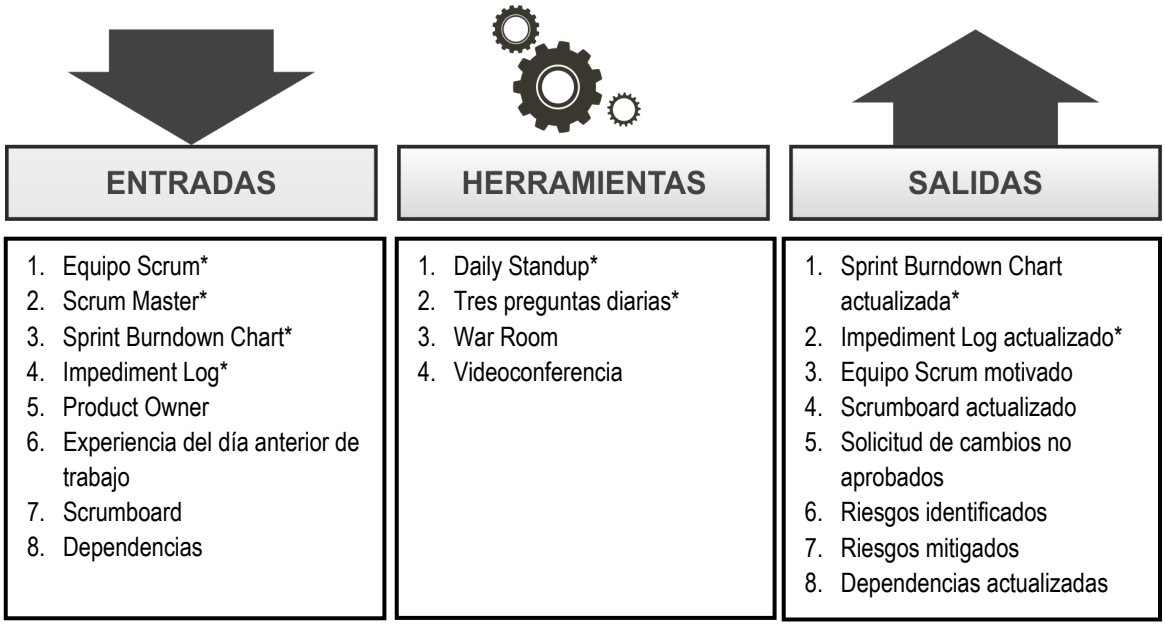


Figura 10-6: Realizar el Daily Standup—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

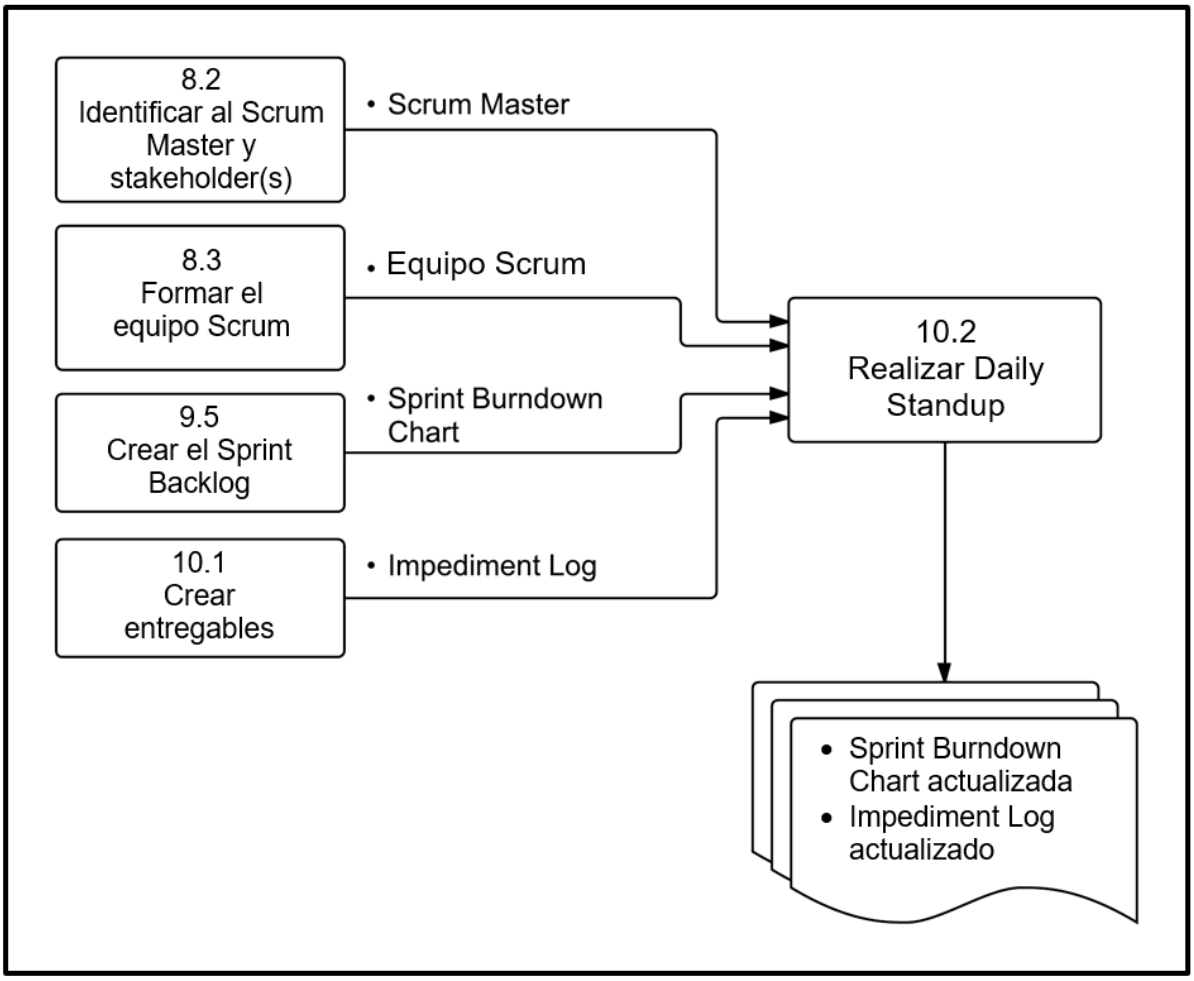


Figura 10-7: Realizar el Daily Standup—Diagrama de flujo de datos

10.2.1 Entradas

10.2.1.1 Equipo Scrum*

Descrito en la sección 8.3.3.1.

10.2.1.2 Scrum Master*

Descrito en la sección 8.2.3.1.

10.2.1.3 Sprint Burndown Chart*

Descrito en la sección 9.5.3.2.

10.2.1.4 Impediment Log*

Descrito en la sección 10.1.1.4.

10.2.1.5 Product Owner

Descrito en la sección 8.1.3.1.

10.2.1.6 Experiencia del día anterior de trabajo

Los miembros del Equipo Scrum mantienen al tanto a sus compañeros durante las reuniones de Daily Standup. Esta sesión se denomina *Standup* ("de pie"), ya que los miembros permanecen de pie durante toda la reunión. Los integrantes del equipo discuten los logros y la experiencia del día anterior. Esta experiencia es una entrada importante al Daily Standup.

10.2.1.7 Scrumboard

Descrito en la sección 10.1.1.3.

10.2.1.8 Dependencias

Descritas en la sección 9.3.3.3.

10.2.2 Herramientas

10.2.2.1 Daily Standup*

El Daily Standup es una breve reunión diaria con un time-box de 15 minutos. Los miembros del equipo se reúnen para dar un reporte sobre su progreso en el sprint y planificar las actividades del día. La duración de la reunión es muy corta y se busca que todos los integrantes del Equipo Scrum estén presentes. Sin embargo, la reunión no se cancela o se retrasa si uno o más miembros no pueden asistir.

En la reunión cada miembro del Equipo Scrum da respuesta a las tres preguntas diarias que se mencionan en la sección 10.2.2.2. Se recomiendan los debates entre el Scrum Master y el equipo o entre los miembros del Equipo Scrum, aunque tales discusiones suceden después de la reunión a fin de garantizar que el Daily Standup sea breve.

10.2.2.2 Tres preguntas diarias*

Las tres preguntas diarias se utilizan en los Daily Standups, organizados por el Scrum Master, donde cada miembro del Equipo Scrum brinda información en forma de respuesta a tres preguntas específicas:

1. ¿Qué he hecho desde la última reunión?
2. ¿Qué tengo planeado hacer antes de la siguiente reunión?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Al enfocarse en estas tres preguntas, todo el equipo puede tener una comprensión clara de la situación de trabajo. En ocasiones se pueden discutir otros elementos, pero se mantiene al mínimo en razón del tiempo asignado que tiene la reunión.

Es muy recomendable que, de ser posible, los miembros del equipo den respuesta a las dos primeras preguntas en forma cuantificable en vez de dar largas respuestas cualitativas. Los miembros del equipo pueden organizar reuniones adicionales después del Daily Standup a fin de abordar temas que requieran de mayor discusión.

10.2.2.3 War Room

En Scrum, es preferible que el equipo este co-ubicado, donde todos los miembros trabajen en la misma ubicación. “Salón de guerra”, o *War Room* en inglés, es el término que comúnmente se utiliza para describir a dicho lugar. Normalmente está diseñado de tal manera que los miembros del equipo puedan moverse libremente, trabajar y comunicarse fácilmente, ya que se encuentran ubicados en proximidades inmediatas. Típicamente se distribuyen fichas, notas adhesivas o demás herramientas sencillas o sofisticadas a fin de facilitar el flujo de trabajo, la colaboración y la solución de problemas.

El salón en ocasiones es ruidoso debido a las conversaciones del equipo, aunque tales conversaciones contribuyen al progreso del equipo. Un buen War Room no tiene cubículos y permite a todo el equipo sentarse junto, garantizando una comunicación cara a cara, lo cual conduce a la formación del equipo y a la apertura. El salón de guerra, o War Room, también es ideal para llevar a cabo los Daily Standups.

El (los) stakeholder(s) de otros equipos de Scrum pudieran también llegar al War Room y discutir temas relevantes.

10.2.2.4 Videoconferencia

En situaciones de la vida real, tal vez no sea posible que todo el Equipo Scrum esté co-ubicado. En tales casos, resulta importante utilizar herramientas de videoconferencia que permitan la comunicación cara a cara.

10.2.3 Salidas

10.2.3.1 Sprint Burndown Chart actualizado*

Descrita en la sección 9.5.3.2.

10.2.3.2 Impediment Log actualizado*

Descrito en la sección 10.1.1.4.

10.2.3.3 Equipo Scrum motivado

Los Daily Standups propagan la idea de que cada miembro del equipo es importante y es un importante colaborador, lo cual mejora la moral individual y colectiva. Ello, aunado al concepto de la auto-organización del equipo, mejora la motivación general, conduciendo a un mejor desempeño del equipo y mejorando la calidad de los entregables que se producen.

El Equipo Scrum se describe en la sección 8.3.3.1.

10.2.3.4 Scrumboard actualizado

Descrito en la sección 10.1.1.3.

10.2.3.5 Solicitudes de cambio no aprobadas

Descritas en la sección 8.4.1.6.

10.2.3.6 Riesgos identificados

Descritos en la sección 8.4.3.4.

10.2.3.7 Riesgos mitigados

Descritos en la sección 10.1.3.6.

10.2.3.8 Dependencias actualizadas

Descritas en la sección 9.3.3.3.

10.3 Refinar el Backlog Priorizado del Producto

La figura 10-8 muestra todas las entradas, las herramientas y las salidas para el proceso de *Refinar el Backlog Priorizado del Producto*.

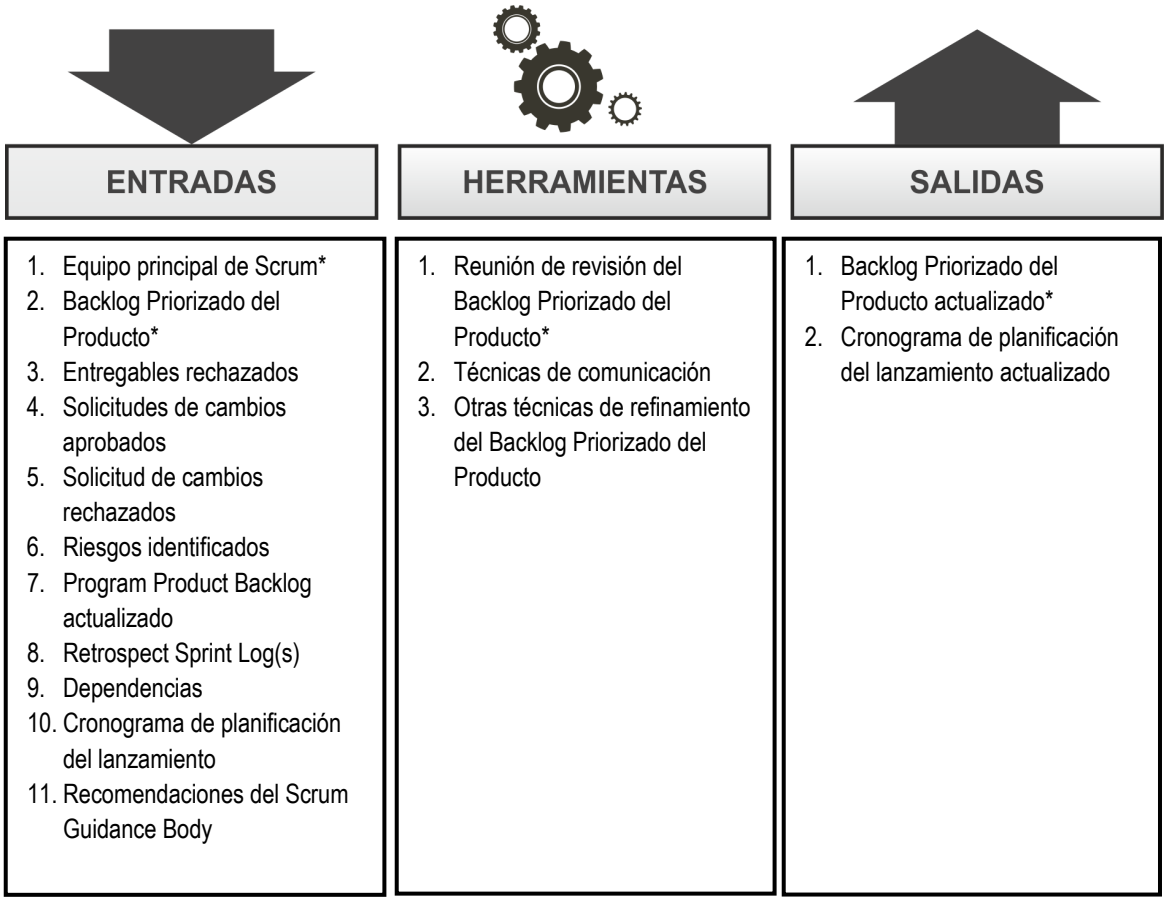


Figura 10-8: Refinar el Backlog Priorizado del Producto—Entradas, herramientas y salidas

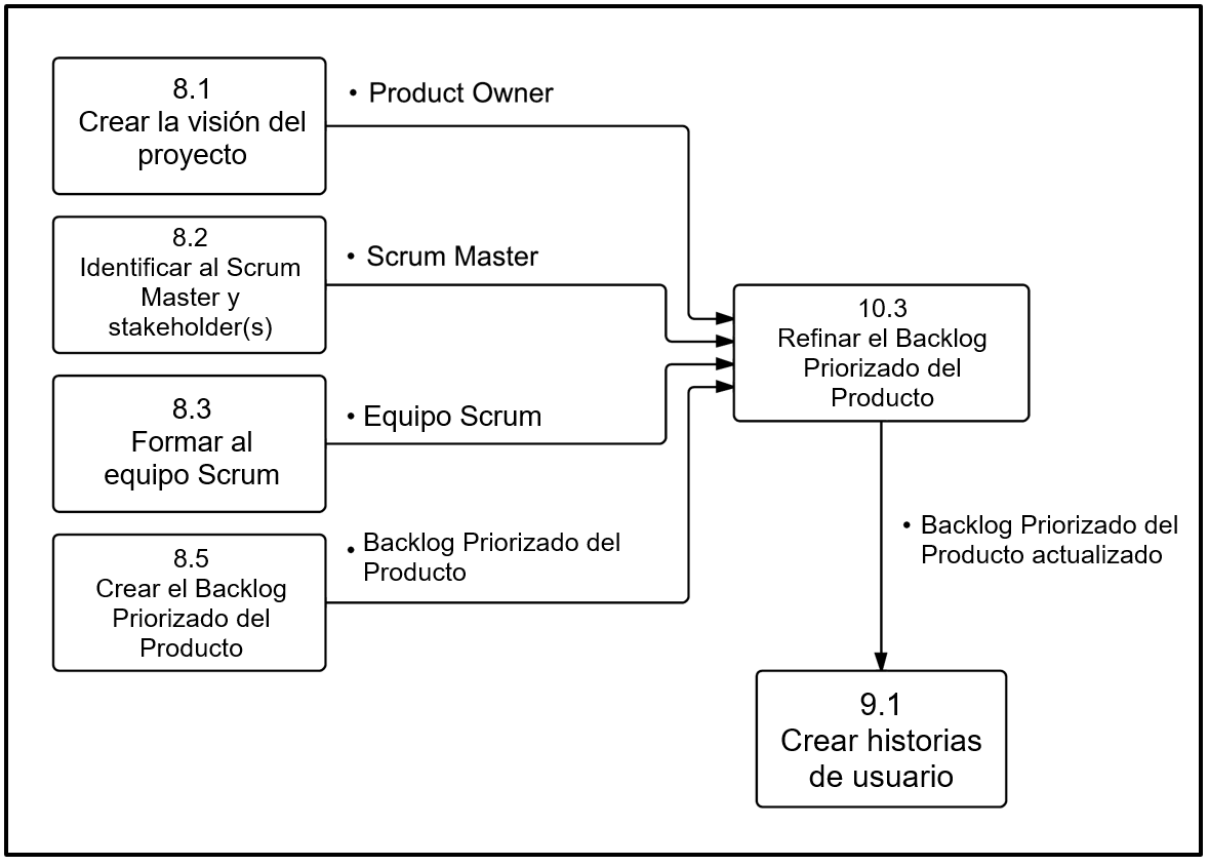


Figura 10-9: Refinar el Backlog Priorizado del Producto—Diagrama de flujo de datos

10.3.1 Entradas

10.3.1.1 Equipo principal de Scrum*

Descrito en las secciones 8.1.3.1, 8.2.3.1 y 8.3.3.1.

10.3.1.2 Backlog Priorizado del Producto*

Descrita en la sección 8.5.3.1.

10.3.1.3 Entregables rechazados

En los casos donde un entregable no cumpla con los criterios de aceptación, este se considera como entregable rechazado (*Rejected Deliverable*). Los entregables rechazados generalmente no se mantienen en una lista por separado, simplemente permanecen en el Backlog Priorizado del Producto y no se marcan como terminados, de forma que se puedan volver a priorizar en el proceso de Refinamiento del Backlog Priorizado del Producto y se consideren para su desarrollo en el siguiente sprint.

10.3.1.4 Solicitudes de cambio aprobadas

Descrito en la sección 8.4.1.5.

10.3.1.5 Solicitudes de cambio no aprobadas

Descrito en la sección 8.4.1.6.

10.3.1.6 Riesgos identificados

Descritos en la sección 8.4.3.4.

10.3.1.7 Program Product Backlog actualizado

Al igual que Project Product Backlog, el Program Product Backlog también puede someterse a un refinamiento constante a fin de incorporar cambios y nuevos requerimientos. Los cambios al Program Product Backlog pueden resultar a partir de cambios en las condiciones externas como internas. Las condiciones externas pueden incluir situaciones de cambio empresariales, tendencias en la tecnología o requerimientos de cumplimientos legales. Los factores internos que afectan el Program Product Backlog pueden estar relacionados a modificaciones a las políticas o a las estrategias organizaciones, a riesgos identificados o demás factores. Los cambios en los requerimientos en el Program Product Backlog pueden

tener un impacto del Project Product Backlogs en proyectos subyacentes, a fin de que se puedan tomar en cuenta durante el proceso de *Refinar el Backlog Priorizado del Producto.*

10.3.1.8 Retrospect Sprint Log(s)

Descrito en la sección 11.3.3.4.

10.3.1.9 Dependencias

Descrito en la sección 9.3.3.3.

10.3.1.10 Cronograma de planificación del lanzamiento

Descrito en la sección 8.6.3.1.

10.3.1.11 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de *Refinamiento del Backlog Priorizado del Producto*, las recomendaciones del Scrum Guidance Body pueden incluir las mejores prácticas sobre cómo entender y recopilar sistemáticamente los requerimientos de los stakeholders y de los equipos de Scrum; priorizarlos adecuadamente en el Product Backlog y comunicar las actualizaciones a todas las personas relevantes involucradas en un proyecto Scrum.

10.3.2 Herramientas

10.3.2.1 Reuniones de revisión del Backlog Priorizado del Producto*

El Product Owner puede tener varias reuniones por separado con los stakeholders relevantes, con el Scrum Master y con el Equipo Scrum a fin de asegurarse de contar con la información suficiente para actualizar el Backlog Priorizado del Producto en el proceso de su refinamiento.

La intención de las reuniones de revisión del Backlog Priorizado del Producto es asegurarse de que se entiendan las historias de usuario y los criterios de aceptación y de que el Product Owner la escriba adecuadamente reflejando los requerimientos y prioridades actuales del stakeholder (cliente); que todos en el Equipo Scrum entiendan las historias de usuario y que las historias de usuario de mayor prioridad sean bien refinadas de tal forma que el Equipo Scrum las pueda estimar adecuadamente y avocarse a ellas.

Las reuniones de revisión del Backlog Priorizado del Producto garantizan también que se eliminen historias de usuario irrelevantes y que se incorporen las solicitudes de cambio aprobadas o los riesgos identificados en el Backlog Priorizado del Producto.

10.3.2.2 Técnicas de comunicación

Scrum promueve la comunicación precisa y eficaz primordialmente mediante la co-ubicación de equipos Scrum. Scrum favorece también las interacciones informales cara a cara en vez de la comunicación formal por escrito. Cuando un Equipo Scrum debe estar distribuido geográficamente, el Scrum Master debe asegurar que existan técnicas eficaces de comunicación para que los equipos se puedan auto organizar y trabajar eficientemente.

10.3.2.3 Otras técnicas de refinamiento del Backlog Priorizado del Producto

Otras herramientas de refinamiento del Backlog Priorizado del Producto incluyen muchas de las mismas herramientas implementadas en los siguientes procesos:

- *Desarrollar épica(s)*—Descrito en la sección 8.4.2.
- *Crear el Backlog Priorizado del Producto*—Descrito en la sección 8.5.2.
- *Realizar la planificación del lanzamiento*—Descrito en la sección 8.6.2.
- *Crear historias de usuario*—Descrito en la sección 9.1.2.
- *Estimar historias de usuarios*—Descrito en la sección 9.2.2.
- *Comprometer historias de usuario*—Descrito en la sección 9.3.2.
- *Identificar tareas*—Descrito en la sección 9.4.2.
- *Estimar tareas*—Descrito en la sección 9.5.2.

10.3.3 Salidas

10.3.3.1 Backlog Priorizado del Producto actualizado*

Descrito en la sección 8.5.3.1.

El Backlog Priorizado del Producto se puede actualizar con nuevas historias de usuario, nuevas solicitudes de cambio, nuevos riesgos identificados o historias de usuario actualizadas o bien, la re-priorización de las historias de usuario ya existentes.

10.3.3.2 Cronograma de planificación del lanzamiento actualizado

Descrito en la sección 8.6.3.1.

El cronograma de planificación del lanzamiento (del inglés: *Release Planning Schedule*) puede actualizarse a fin de que refleje el impacto de las historias de usuario nuevas o actualizadas en el Backlog Priorizado del Producto.

10.4 Diagrama de flujo de datos de fase

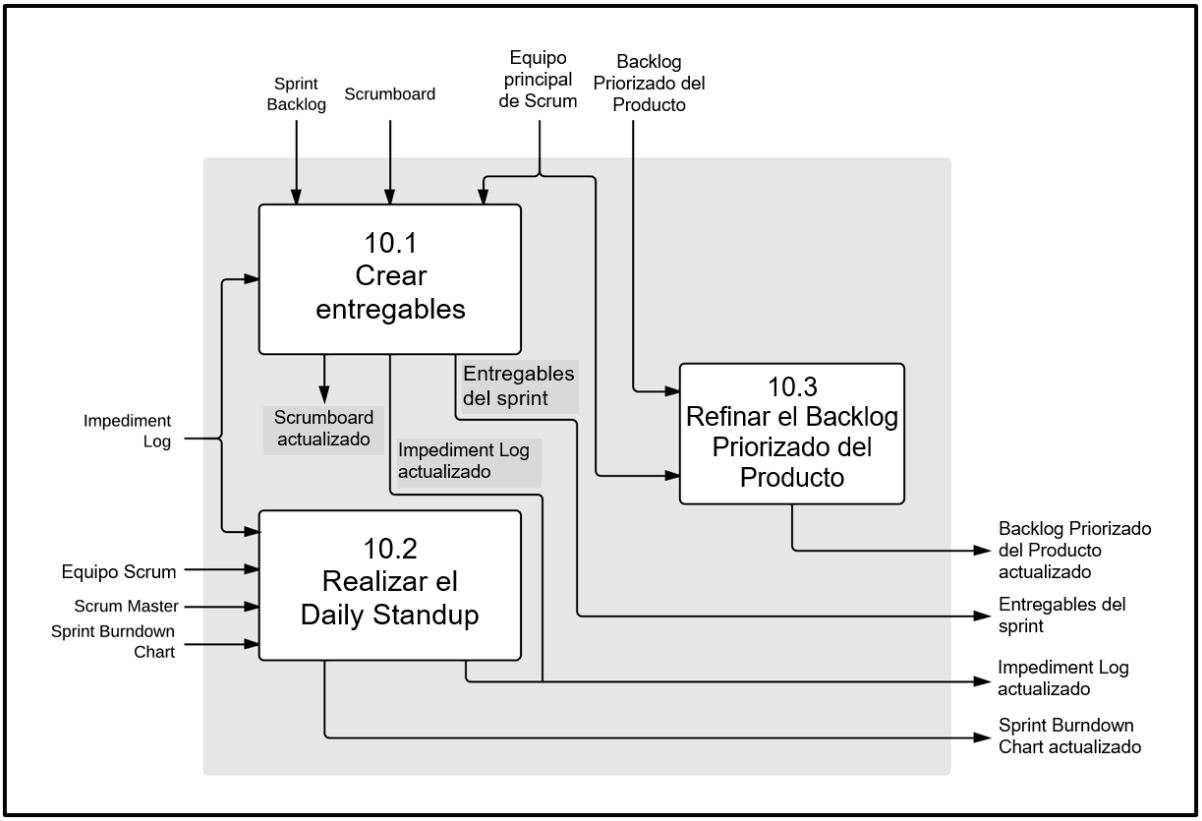


Figura 10-10: Fase de implementación—Diagrama de flujo de datos

11. REVISIÓN Y RETROSPECTIVA

La fase de Revisión y Retrospectiva cubre los relacionado a la revisión de los entregables y al trabajo que se ha realizado y determina las formas para mejorar las prácticas y métodos implementados para realizar el trabajo del proyecto. En las grandes organizaciones, el proceso de revisión y retrospectiva también puede incluir el convocar a reuniones de Scrum de Scrums.

La *Revisión y Retrospectiva*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™,* no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un Equipo Scrum que trabaja en un sprint para producir entregables que pudieran ser parte de un proyecto más grande, de un programa o portafolio. La información sobre cómo escalar Scrum en grandes proyectos está disponible en el capítulo 13 y lo relacionado a escalar Scrum para la empresa se puede encontrar en el capítulo 14.

La figura 11-1 brinda una descripción general de los procesos de la fase *Revisión y Retrospectiva*, los cuales son los siguientes:

11.1 Demostrar y validar el sprint—En este proceso, el Equipo Scrum demuestra los entregables del sprint al Product Owner y a los stakeholders relevantes durante una reunión de revisión del sprint. El propósito de esta reunión es lograr la aprobación y aceptación del Product Owner respecto al producto o servicio.

11.2 Retrospectiva de sprint—En este proceso, el Scrum Master y el Equipo Scrum se reúnen para discutir las lecciones aprendidas durante el sprint. Dicha información se documenta como lecciones aprendidas que pudieran implementarse en futuros sprints. Generalmente, como consecuencia de esta reunión, se pudieran obtener *Agreed Actionable Improvements* (mejoras aceptadas) o recomendaciones actualizadas del Scrum Guidance Body.

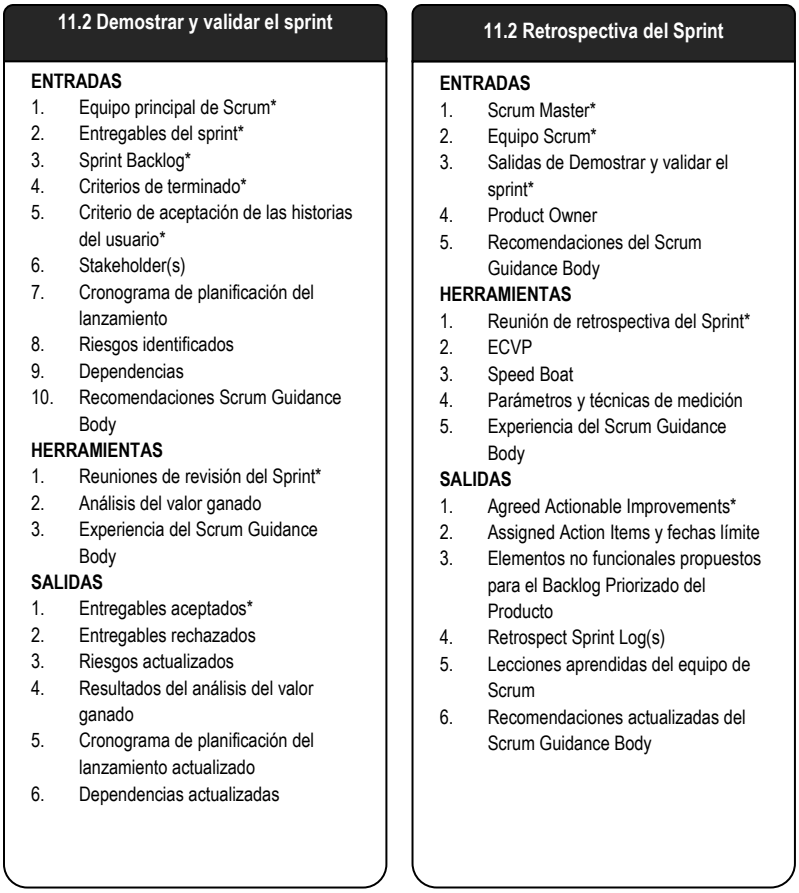


Figura 11-1: Descripción general de la Revisión y retrospectiva

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 11-2 muestra las entradas, herramientas y salidas obligatorias para los procesos en la fase *Revisión y Retrospectiva.*

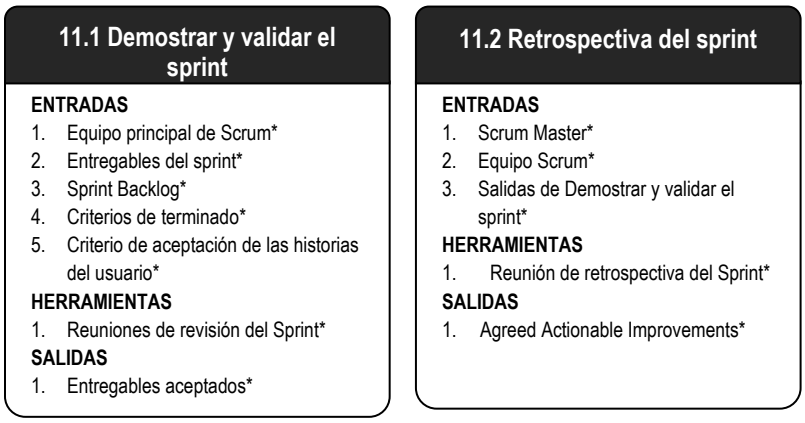


Figura 11-2: Descripción general de la Revisión y retrospectiva (esenciales)

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

11.1 Demostrar y validar el sprint

La figura 11-5 muestra todas las entradas, las herramientas y las salidas para el proceso de *Demostrar y validar el sprint*.

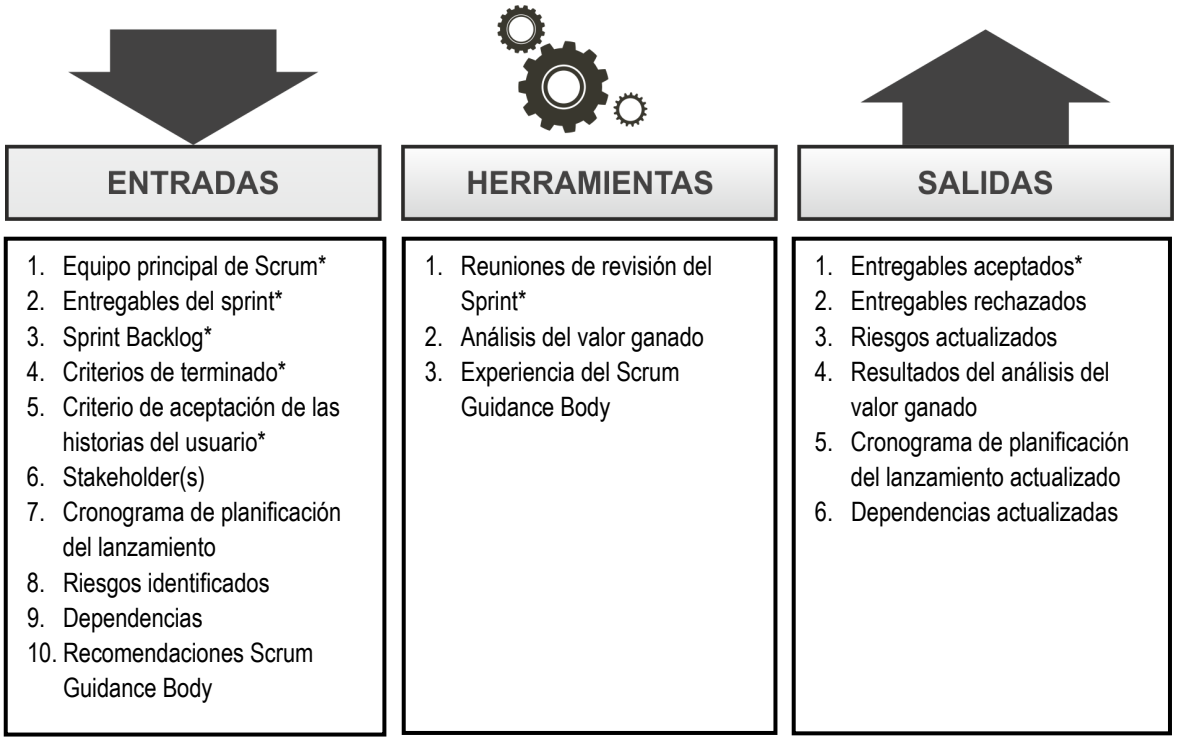


Figura 11-3: Demostrar y validar el sprint—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

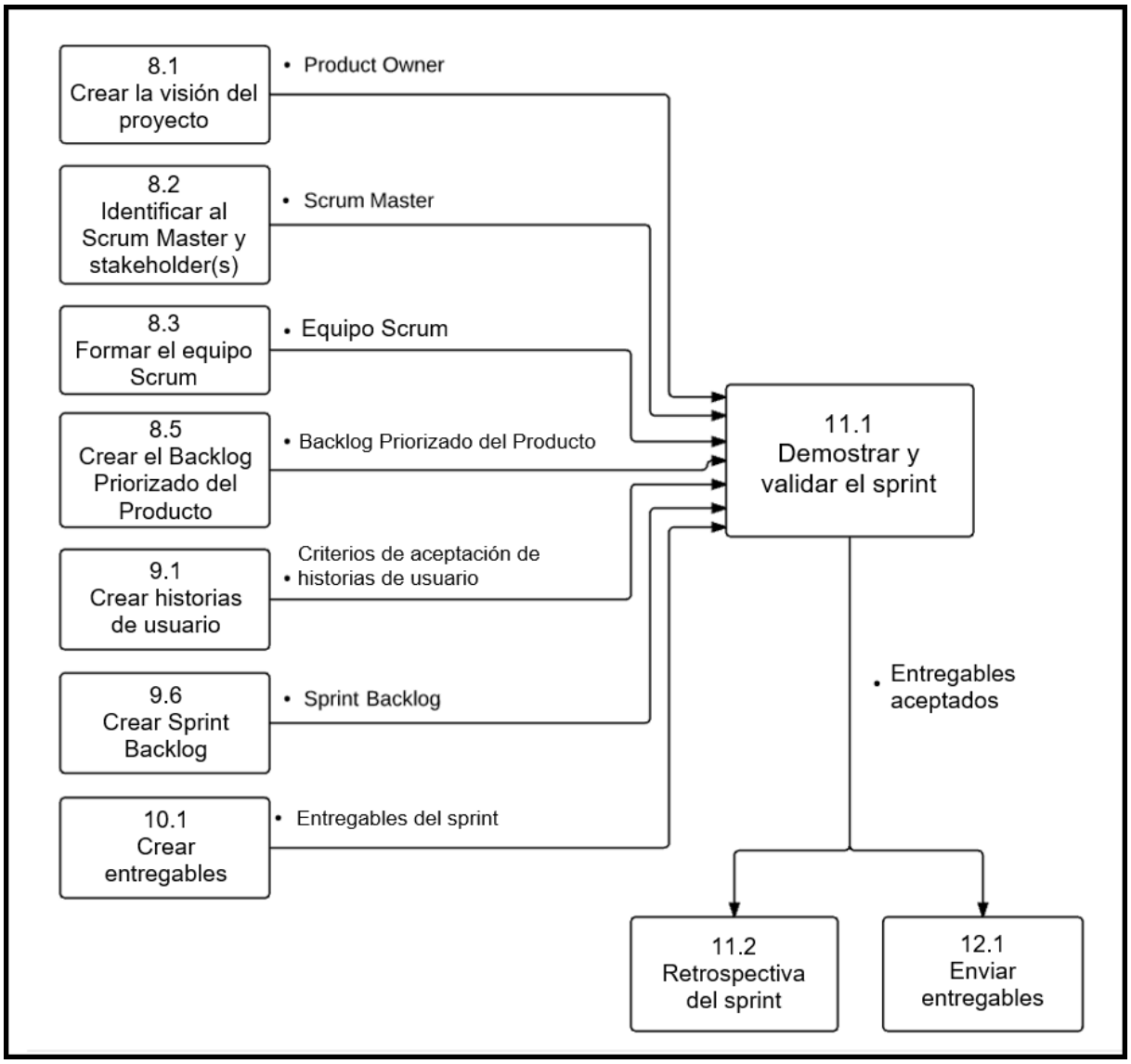


Figura 11-4: Demostrar y validar el sprint—Diagrama de flujo de datos

11.1.1 Entradas

11.1.1.1 Equipo principal de Scrum*

Descrito en la sección 8.4.1.1.

11.1.1.2 Entregables del sprint*

Descritos en la sección 10.1.3.1

11.1.1.3 Sprint Backlog*

Descrita en la sección 9.5.3.1.

11.1.1.4 Criterios de terminado*

Descritos en la sección 8.5.3.2.

11.1.1.5 Criterios de aceptación de las historias de usuario*

Descritos en la sección 9.4.1.3.

11.1.1.6 Stakeholder(s)

Descrito en la sección 8.2.3.2.

11.1.1.7 Cronograma de planificación del lanzamiento

Descrito en la sección 8.6.3.1.

11.1.1.8 Riesgos identificados

Descrito en la sección 8.4.3.4.

11.1.1.9 Dependencias

Descritas en la sección 9.3.3.3

11.1.1.10 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11

En el proceso de *Demostrar y validar el sprint*, las recomendaciones del Scrum Guidance Body pueden incluir las mejores prácticas sobre cómo llevar a cabo las reuniones de revisión del sprint y evaluar los resultados del análisis del valor ganado. Pudiera haber también una orientación sobre como compartir experiencias con otras personas en el equipo principal de Scrum, así como demás equipos Scrum en el proyecto.

11.1.2 Herramientas

11.1.2.1 Reunión de revisión del sprint*

Los miembros del equipo principal de Scrum y los stakeholders relevantes participan en las reuniones de revisión del sprint para aceptar los entregables que cumplan con los criterios de aceptación de las historias de usuario y rechazar los entregables no aceptables. Tales reuniones se convocan al final de cada sprint. El Equipo Scrum demuestra los logros del sprint, incluyendo las nuevas funcionalidades o los productos elaborados. Esto brinda una oportunidad para que el Product Owner y el(los) stakeholder(s) inspeccionen lo que se ha completado hasta el momento y determinen si deban realizarse cambios en el proyecto o en los procesos en sprints posteriores.

11.1.2.2 Análisis del valor ganado (*Earned Value Analysis*)

Descrito en la sección 4.6.1

11.1.2.3 Experiencia del Scrum Guidance Body

Descrito en la sección 8.4.2.7.

En el proceso de *Demostrar y validar el sprint*, la experiencia del Scrum Guidance Body podría estar relacionado a las mejores prácticas documentadas sobre cómo llevar a cabo reuniones de revisión del sprint. Pudiera también haber expertos que pudieran brindar orientación sobre como facilitar de una mejor forma las reuniones de revisión del sprint.

11.1.3 Salidas

11.1.3.1 Entregables aceptados*

Los entregables que cumplen con los criterios de aceptación de las historias de usuario son aceptados por el Product Owner. El objetivo de un sprint es crear entregables potencialmente enviables o incrementos del producto que cumplan con los criterios de aceptación definidos por el cliente y el Product Owner. Estos se consideran entregables aceptados que pudieran ser entregados al cliente y así se desea. Se mantiene una lista de entregables aceptados y se actualiza después de cada reunión de revisión del sprint. Si un entregable no cumple con los criterios de aceptación definidos, no se considera aceptado y generalmente se llevará a un sprint posterior para corregir cualquier problema. Esto no es muy recomendable, ya que el objetivo de cada sprint es que los entregables cumplan con los criterios de aceptación.

11.1.3.2 Entregables rechazados

Si los entregables no cumplen con los criterios de aceptación, tales entregables se rechazan. Las historias de usuario asociados a tales entregables rechazados se agregan al Backlog Priorizado del Producto para que tales entregables puedan ser considerados como parte de un sprint posterior.

11.1.3.3 Riesgos actualizados

Descrito en la sección 8.4.3.4.

11.1.3.4 Resultados del análisis del valor ganado

Descrito en la sección 4.6.1.

11.1.3.5 Cronograma de la planificación del lanzamiento actualizado

Descrito en la sección 10.3.3.2.

11.1.3.6 Dependencias actualizadas

Descritas en la sección 9.3.3.3.

11.2 Retrospectiva del sprint

La figura 11-5 muestra todas las entradas, las herramientas y las salidas para el proceso *Retrospectiva del sprint*.

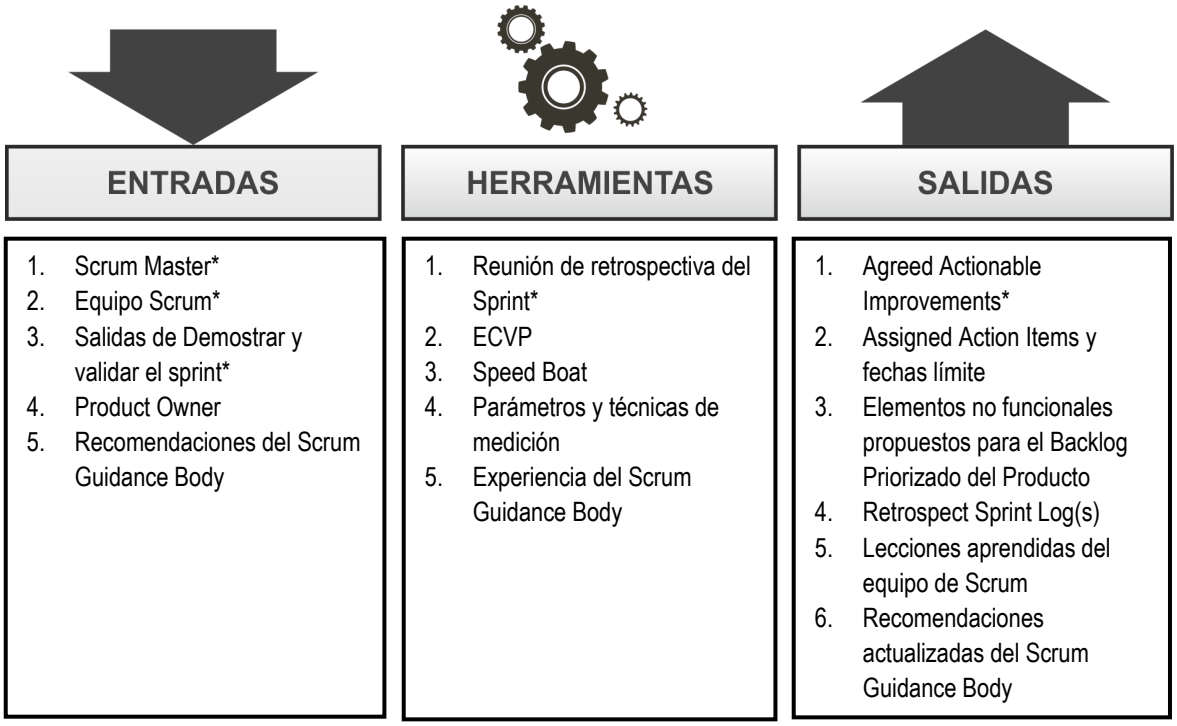


Figura 11-5: Retrospectiva del sprint—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

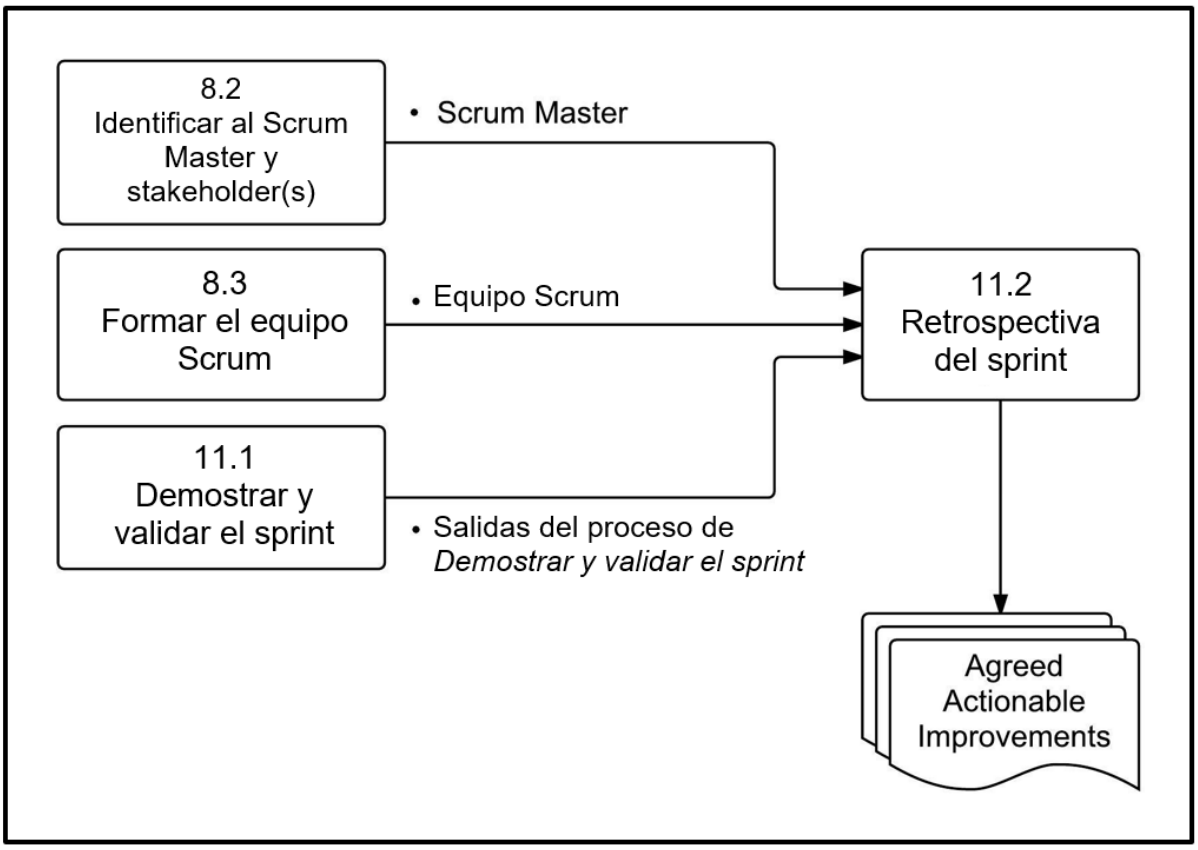


Figura 11-6: Retrospectiva del sprint—Diagrama de flujo de datos

11

11.2.1 Entradas

11.2.1.1 Scrum Master*

Descrito en la sección 8.2.3.1.

11.2.1.2 Equipo Scrum*

Descrito en la sección 8.3.3.1.

11.2.1.3 Salidas del proceso de Demostrar y validar el sprint*

Descrito en la sección 11.2.3.

Las salidas del proceso de *Demostrar y validar el sprint* ofrecen información útil al momento de llevar a cabo el proceso de retrospectiva del sprint.

11.2.1.4 Product Owner

Descrito en la sección 8.1.3.1.

11.2.1.5 Recomendaciones del Scrum Guidance Body

El Scrum Guidance Body puede brindar directrices para llevar a cabo las reuniones de retrospectiva del sprint, incluyendo sugerencias sobre herramientas a utilizarse, así como documentación o los entregables que se esperan de las reuniones.

11.2.2 Herramientas

11.2.2.1 Reunión de retrospectiva del sprint*

La reunión de retrospectiva del sprint es un elemento importante del framework de "inspección-adaptación" de Scrum y es el último paso en un sprint. Todos los miembros del Equipo Scrum asisten a la reunión, misma que organiza y modera el Scrum Master. Se recomienda que asista el Product Owner, aunque no es obligatorio. Un integrante del equipo se desempeña como secretario y documenta las discusiones y los elementos para acciones a futuro. Es esencial celebrar esta reunión es un entorno abierto y relajado a fin de fomentar la completa participación de todos los miembros del equipo. Las discusiones en la reunión de

retrospectiva del sprint abarcan tanto lo que salió mal como lo que salió bien. Los objetivos primordiales de la reunión son identificar tres elementos específicos:

1) Las cosas que el equipo necesita seguir haciendo: mejores prácticas
2) Las cosas que el equipo necesita empezar a hacer: mejoras en el proceso
3) Las cosas que el equipo necesita dejar de hacer: problemas de proceso y embotellamiento

Estas áreas se analizan y se crea una lista de mejoras accionables aceptadas (*Agreed Actionable Improvements*).

11.2.2.2 Explorador—Comprador—Vacacionista—Prisionero (ECVP)

Es un ejercicio que se puede llevar a cabo al inicio de la reunión de retrospectiva del sprint para entender la mentalidad de los participantes y establecer el tono de la reunión. Se les pide a los asistentes que indiquen de forma anónima lo que mejor representa su punto de vista en la reunión.

- Explorador (*Explorer*)—Quiere participar y aprender todo lo que se discutió en la retrospectiva
- Comprador (*Shopper*)—Quiere escuchar todo y elegir lo que entiende de la retrospectiva
- Vacacionista (*Vacationer*)—Quiere relajarse y ser turista en la retrospectiva
- Prisionero (*Prisoner*)—Quiere estar en otro lugar y asiste a la retrospectiva por ser obligatorio

El Scrum Master reúne las respuestas; prepara y comparte la información con el grupo.

11.2.2.3 Speed Boat

El Speed Boat es una técnica que se puede utilizar para llevar a cabo la reunión de retrospectiva del sprint. Los miembros juegan el rol de la tripulación en una lancha. La lancha debe llegar a una isla: simbólicamente la visión del proyecto. Los asistentes utilizan notas adhesivas para llevar un registro de motores y anclas. Los “motores” ayudan a llegar a la isla, mientras que las “anclas” son cosas que están obstaculizando la llegada. Este ejercicio tiene un time-box de unos cuantos minutos. Una vez que se documentan todos los elementos, se reúne la información, se discute y se prioriza por medio de un proceso de votación. Se reconocen los motores y se planifican las acciones de mitigación para las anclas dependiendo de la prioridad.

11.2.2.4 Parámetros y técnicas de medición

Se pueden utilizar varios indicadores para medir y contrastar el desempeño del equipo en el actual sprint en comparación al rendimiento de sprints anteriores. Algunos ejemplos de dichos parámetros incluyen:

- Velocidad del equipo—Número de puntos de historia en un determinado sprint

- Tasa de terminado satisfactorio—Porcentaje de puntos de historia que se han terminado en comparación a los que se han asignado
- Eficacia de la estimación—Número o porcentaje de discrepancia entre el tiempo estimado y el tiempo actual en las tareas e historias de usuario.
- Clasificación de retroalimentación de revisiones—Se puede solicitar la retroalimentación de los stakeholders utilizando clasificaciones cuantitativas o cualitativas, brindando una medida del desempeño del equipo.
- Clasificación moral del equipo—Resultados de una auto evaluación de la moral de los miembros del equipo.
- Retroalimentación de los compañeros—Se pueden implementar mecanismos de retroalimentación de 360 grados para solicitar una crítica constitutiva e información sobre el rendimiento del equipo.
- Progreso para la liberación o lanzamiento—El valor de negocio que se brinda en cada lanzamiento, así como el valor representado por el actual proceso encaminado a un lanzamiento. Esto contribuye a la motivación del equipo y al nivel de satisfacción de trabajo.

11.2.2.5 Experiencia del Scrum Guidance Body

También se describe en la sección 8.4.2.7.

En el proceso de *Retrospectiva del sprint*, la experiencia del Scrum Guidance Body podría estar relacionada a las mejores prácticas sobre cómo llevar a cabo reuniones de retrospectiva del sprint. Pudiera también haber expertos que pudieran brindar orientación sobre cómo utilizar las herramientas en el proceso de retrospectiva del sprint para entregar Agreed Actionable Improvements (*mejoras accionables aceptadas*) para los futuros sprints.

11.2.3 Salidas

11.2.3.1 Agreed Actionable Improvements*

Los llamados *Agreed Actionable Improvements* son el principal resultado del proceso de *Retrospectiva del sprint*. Forman parte de la lista de *actionable items* (elementos accionables) que ha elaborado el equipo para hacer frente a los problemas y mejorar los procesos a fin de mejorar también su desempeño en futuros sprints.

11.2.3.2 Assigned Action Items y fechas límite

Una vez que se han elaborado y refinado, los *Assigned Action Items* (elementos de acción asignados) y las fechas límite, el Equipo Scrum puede considerar los puntos de acción para implementar las mejoras. Cada elemento de acción contará con una fecha límite de conclusión.

11.2.3.3 Elementos no funcionales propuestos para el Backlog Priorizado del Producto

Cuando se desarrolla el Backlog priorizado del Producto inicial, este se basa en historias de usuario y en funcionalidades requeridas. Por lo general, los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint o en las reuniones de retrospectiva del sprint. Estos elementos deben añadirse al Backlog Priorizado del Producto a medida que se descubren. Algunos ejemplos de requerimientos no funcionales son: tiempos de respuesta, limitaciones en la capacidad y problemas en materia de seguridad.

11.2.3.4 Retrospect Sprint Log(s)

Los registros de la retrospectiva del sprint, o Retrospect Sprint Log(s), son registros de las opiniones, debates y elementos accionables planteados en la reunión de retrospectiva del sprint. El Scrum Master puede facilitar la creación de dicho registro con la aportación de los miembros del equipo principal de Scrum. La recopilación de todos los registros de la retrospectiva del sprint se convierte en el diario del proyecto y detalla los éxitos del mismo, los problemas y las resoluciones. Los registros son documentos públicos disponibles a todos en la organización.

11.2.3.5 Lecciones aprendidas del Equipo Scrum

Se espera que el Equipo Scrum (equipo auto organizado y empoderado) aprenda de los errores cometidos durante el sprint y que estas lecciones aprendidas ayuden a mejorar su desempeño en futuros sprints. Estas lecciones aprendidas también pueden documentarse en las recomendaciones del Scrum Guidance Body para compartirse con otros equipos de Scrum.

Puede haber varias lecciones positivas aprendidas como parte de un sprint. Estas lecciones positivas aprendidas son parte clave de la retrospectiva y deben compartirse adecuadamente dentro del equipo y con el Scrum Guidance Body conforme el equipo avanza hacia una auto-mejora continua.

11.2.3.6 Recomendaciones del Scrum Guidance Body actualizadas

Como resultado de la reunión de retrospectiva del sprint, se pueden hacer sugerencias para revisar o mejorar las recomendaciones del Scrum Guidance Body. Si dicho órgano acepta las sugerencias, estas se incorporarán como actualizaciones a la documentación del Scrum Guidance Body.

11.3 Diagrama de flujo de datos de fase

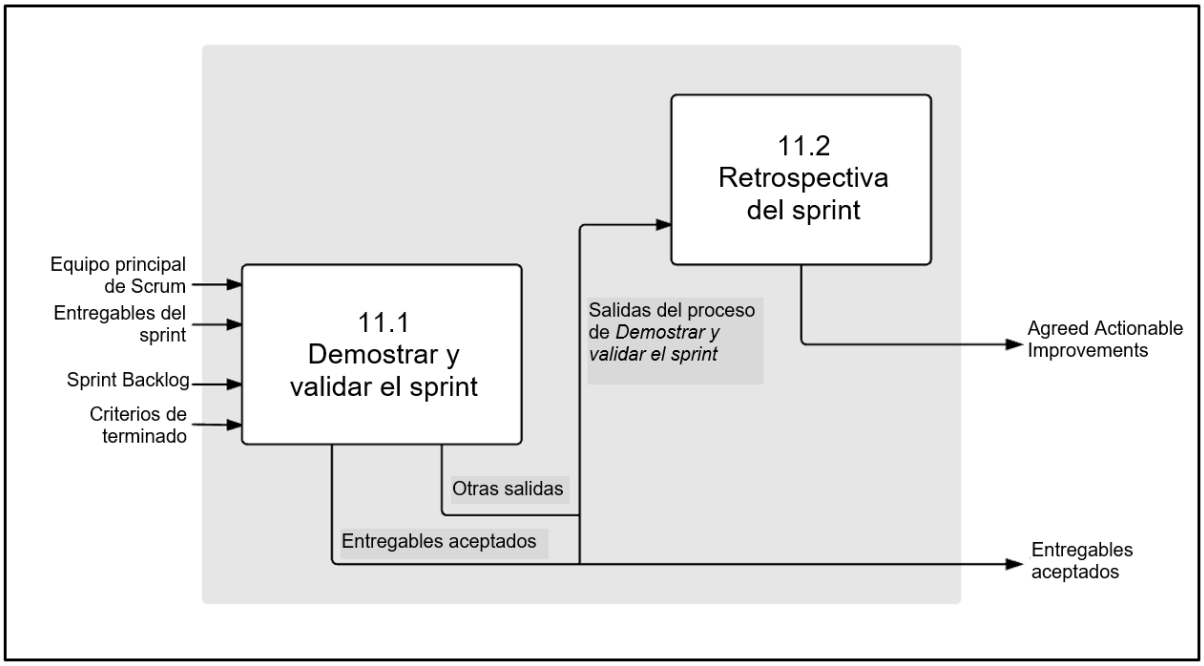


Figura 11-6: Fase de revisión y retrospectiva—Diagrama de flujo de datos

12. LANZAMIENTO

La fase de lanzamiento (*Release*) hace énfasis en la entrega al cliente de los entregables aceptados, así como en la identificación, documentación e internalización de las lecciones aprendidas durante el proyecto.

El *Lanzamiento*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™,* no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un Equipo Scrum que trabaja en un sprint para producir entregables que pudieran ser parte de un proyecto más grande, de un programa o portafolio. La información sobre cómo escalar Scrum en grandes proyectos está disponible en el capítulo 13 y lo relacionado a escalar Scrum para la empresa se puede encontrar en el capítulo 14.

La figura 12-1 brinda una descripción general de los procesos de la fase de lanzamiento, los cuales son los siguientes:

12.1 Enviar entregables—En este proceso se hace la entrega o la transición de los entregables aceptados a los stakeholders relevantes. La conclusión satisfactoria del sprint se documenta en un *Working Deliverables Agreement* (o Acuerdo de entregables funcionales).

12.2 Retrospectiva del proyecto—En este proceso, en el cual se concluye el proyecto, los stakeholders de la organización y los miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones que se aprendieron. Generalmente, dichas lecciones permiten documentar las Agreed Actionable Improvements e implementarlas en futuros proyectos.

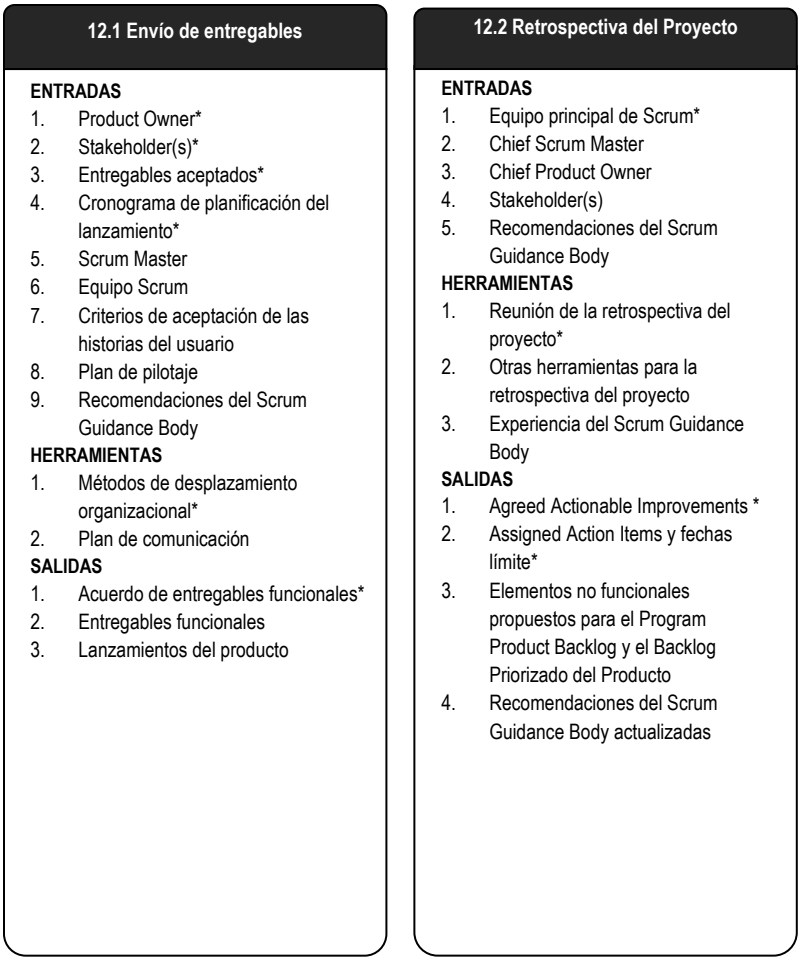


Figura 12-1: Descripción general del lanzamiento

La figura 12-2 muestra las entradas obligatorias, herramientas y salidas para los procesos en fase de lanzamiento.

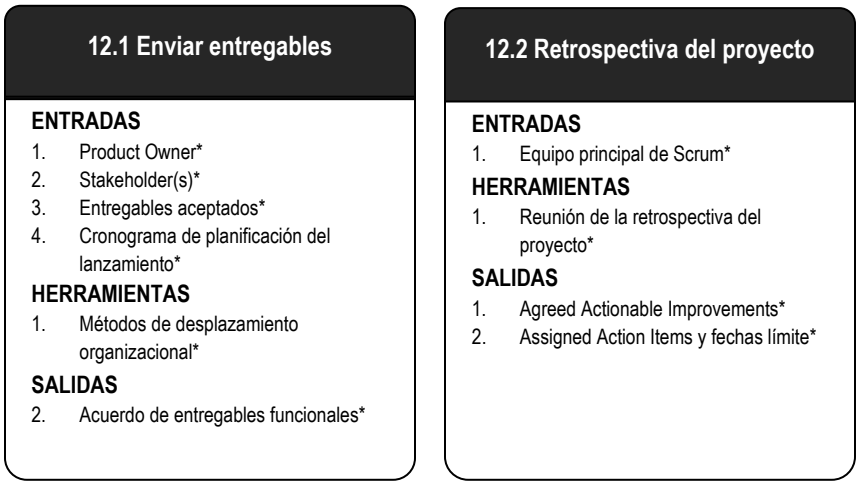


Figura 12-2: Resumen del lanzamiento (Esenciales)

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

12.1 Enviar entregables

La figura 12-3 muestra todas las entradas, las herramientas y las salidas para el proceso de *Enviar entregables*.



Figura 12-3: Enviar entregables—Entradas, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

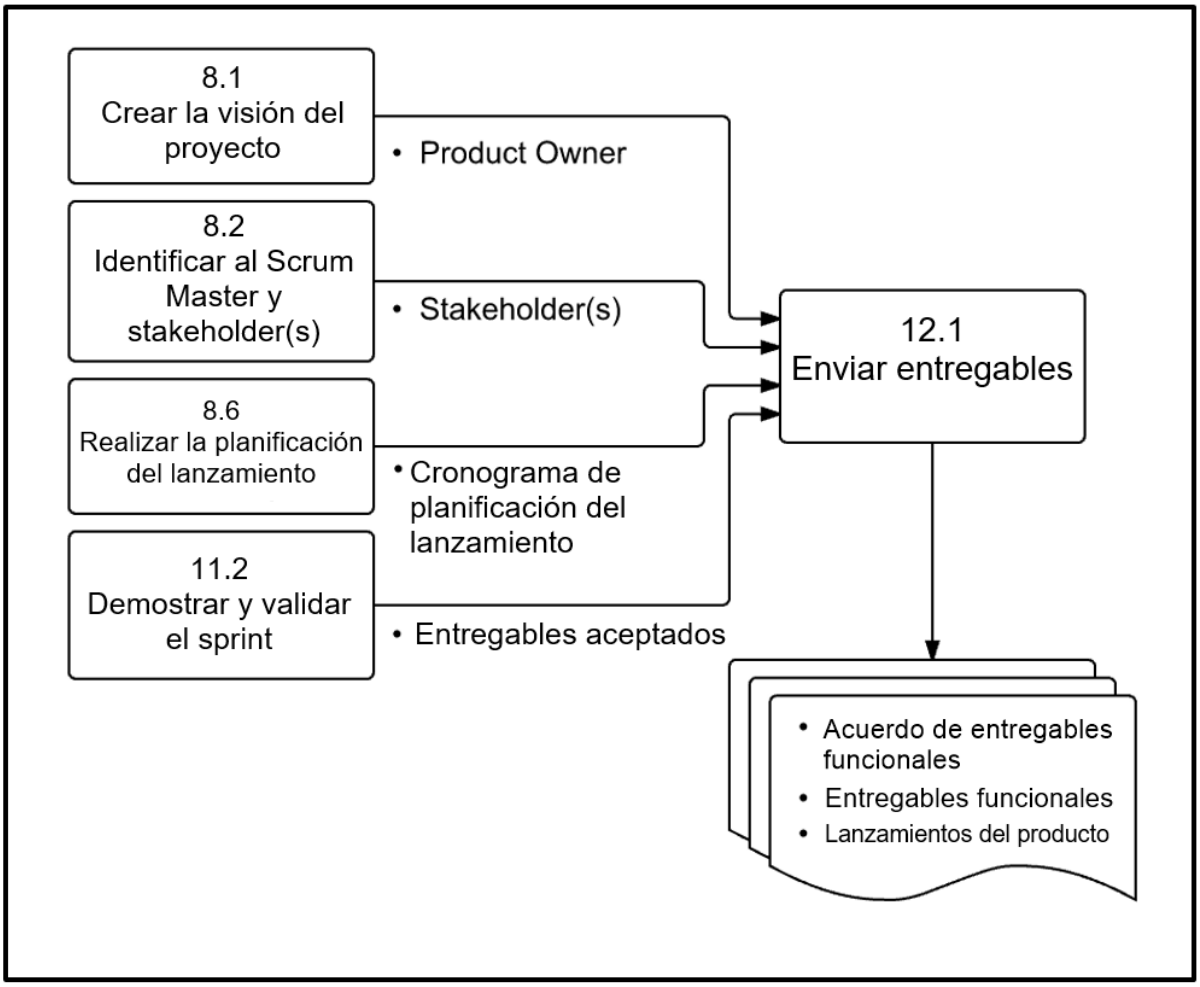


Figura 12-4: Envío de entregables—Diagrama de flujo de datos

12.1.1 Entradas

12.1.1.1 Product Owner*

Descrito en la sección 8.1.3.1.

12.1.1.2 Stakeholder(s)*

Descrito en la sección 8.2.3.2.

12.1.1.3 Entregables aceptados*

Descrito en la sección 11.2.3.1.

12.1.1.4 Cronograma de planificación del lanzamiento

Descrito en la sección 8.6.3.1.

12.1.1.5 Scrum Master

Descrito en la sección 8.2.3.1.

12.1.1.6 Equipo Scrum

Descrito en la sección 8.3.3.1.

12.1.1.7 Criterios de aceptación de las historias de usuario

Descrito en la sección 9.1.3.2.

12.1.1.8 Plan de pilotaje

El plan pilotaje (conocido en inglés como: *Piloting Plan*) es una entrada opcional que se puede utilizar para trazar un desplazamiento piloto a detalle. El alcance y los objetivos del desplazamiento, la base de usuarios seleccionada para la implementación, un cronograma del desplazamiento, los planes de transición, la preparación requerida del usuario, los criterios de evaluación para el desplazamiento y otros elementos clave relacionados al desplazamiento, se identifican en el plan de pilotaje y se comparten con los stakeholders.

12.1.1.9 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.12.

En el proceso de envío de entregables, el Scrum Guidance Body puede brindar recomendaciones y directrices para el desplazamiento de productos. Estas son las mejores prácticas que se deben tomar en cuenta al implementar un producto al cliente con el fin de maximizar el valor entregado.

12.1.2 Herramientas

12.1.2.1 Métodos de desplazamiento organizacional*

Los mecanismos de desplazamiento de cada organización tienden a ser diferentes basados en su respectiva industria, en los usuarios meta y el posicionamiento. Dependiendo del producto a entregarse, el desplazamiento puede ser remoto o puede incluir el envío físico o la transición de un artículo. Debido a que la implementación tiende a implicar un alto nivel de riesgo, las organizaciones suelen tener mecanismos de implementación bien definidos y establecidos, con procesos detallados para garantizar el cumplimiento de todas las normas aplicables y medidas de garantía de calidad. Estas podrían incluir aprobaciones finales de los representantes específicos de gestión, mecanismos de aprobación de usuario y directrices para la funcionalidad mínima de un lanzamiento.

12.1.2.2 Plan de comunicación

En muchos proyectos existe un plan de comunicación. El plan especifica los registros que se deben crear y mantener durante todo el proyecto. Se utiliza una variedad de métodos para transmitir a los stakeholders información importante sobre el proyecto. El plan de comunicación define los métodos e indica quién es responsable de las distintas actividades de comunicación. A medida que se evalúan los entregables, el estado de las actividades de examinación se comunica mediante el plan de comunicación según lo determinado por el Product Owner y el patrocinador. Un mecanismo común de comunicación es una representación visual que muestra información importante y en un formato fácil de interpretar, publicado en un lugar accesible y que se mantiene al día con la información más actualizada.

12.1.3 Salidas

12.1.3.1 Acuerdo de entregables funcionales (*Working Deliverables Agreement*) *

Los entregables que cumplen con los criterios de aceptación, reciben el cierre formal del negocio y la aprobación formal por parte del cliente o del patrocinador. Obtener la aceptación formal del cliente es fundamental para el reconocimiento de los ingresos. La responsabilidad de obtener dicha aceptación se define en las políticas de la empresa y no es necesariamente la responsabilidad del Product Owner.

12.1.3.2 Entregables funcionales

Esta salida es el entregable final enviable (del inglés: *shippable deliverable*) para el cual fue sancionado el proyecto. A medida que se crean nuevos incrementos de los productos, estos se integran continuamente a los incrementos anteriores, por lo que hay un producto potencialmente entregable disponible en todo momento a lo largo del proyecto.

12.1.3.3 Lanzamientos del producto*

Los lanzamientos de productos deben incluir lo siguiente:

- Contenido del lanzamiento—Esto consiste en información esencial sobre los entregables que pudiera ayudar al equipo de atención al cliente.
- Notas de lanzamiento—Las notas de lanzamiento deben incluir criterios de envío externos de cara al mercado para los productos que serán entregados.

12.2 Retrospectiva del proyecto

La figura 12-5 muestra todas las entradas, las herramientas y las salidas para el proceso de *Retrospectiva del proyecto*.

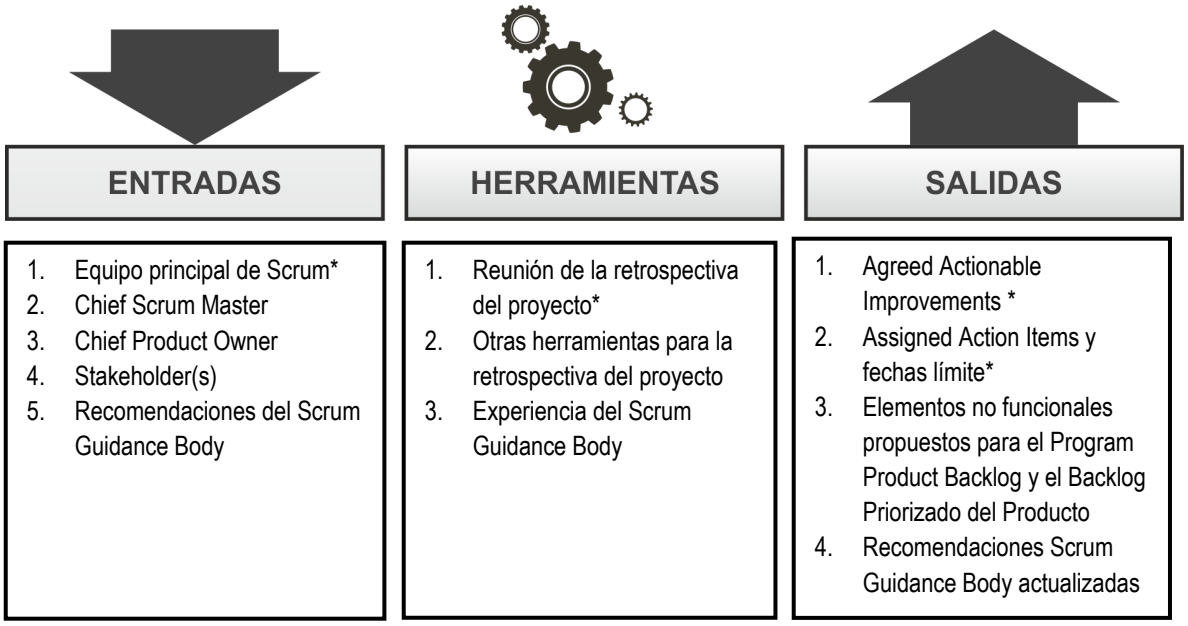


Figura 12-5: Retrospectiva del proyecto—Entrada, herramientas y salidas

Nota: Los asteriscos (*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

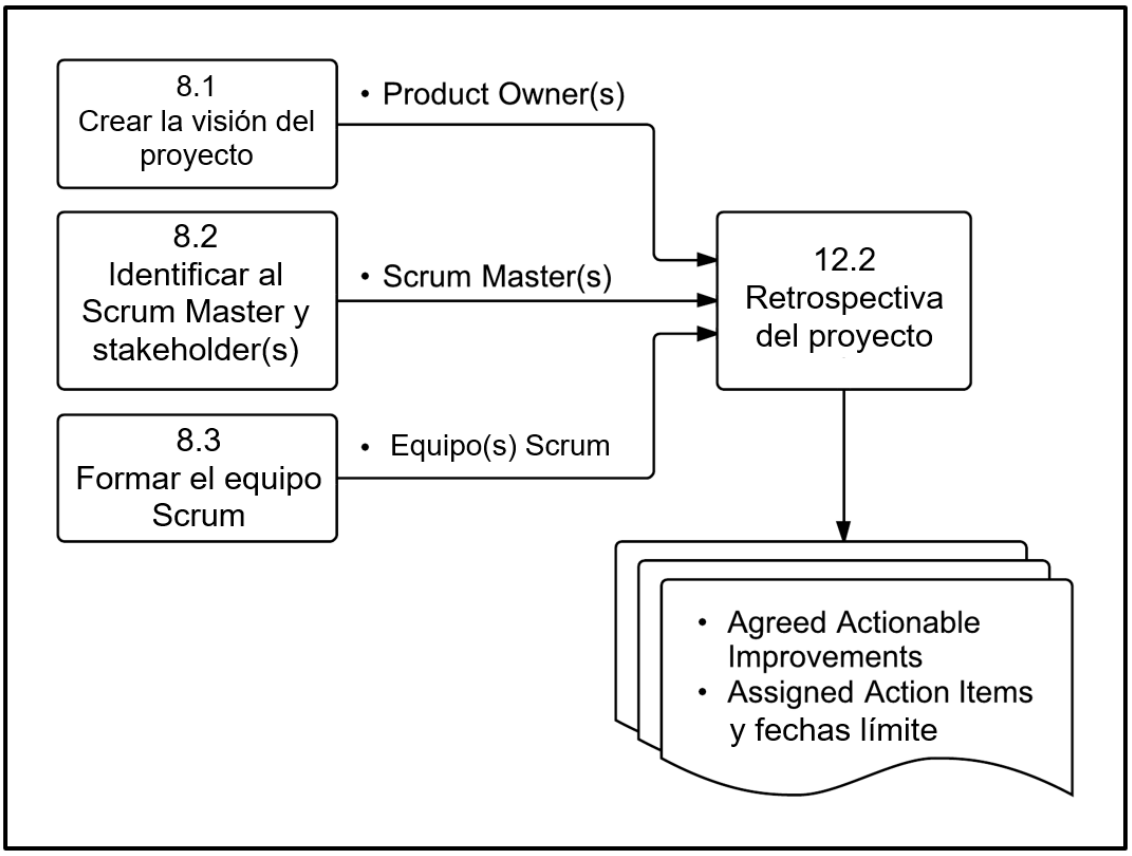


Figura 12-6: Retrospectiva del proyecto—Diagrama de flujo de datos

12.2.1 Entradas

12.2.1.1 Equipo(s) principal de Scrum*

Descrito en la sección 8.4.1.1.

12.2.1.2 Chief Scrum Master

Descrito en la sección 8.2.1.6.

12.2.1.3 Chief Product Owner

Descrito en la sección 8.1.1.5.

12.2.1.4 Stakeholder(s)

Descrito en la sección 8.2.3.2.

12.2.1.5 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

En el proceso de *Retrospectiva del proyecto*, las recomendaciones del Scrum Guidance Body pueden incluir un repositorio de plantillas internas que apoyan los proyectos futuros y brindan orientación para la realizar la reunión de retrospectiva del proyecto. La orientación que se brinda puede estar relacionada a procedimientos administrativos, auditorias, evaluaciones y criterios de transición del proyecto. Generalmente incluyen también la forma en la que la organización habrá de mantener el conocimiento base de las lecciones aprendidas y la información de todos los proyectos.

12.2.2 Herramientas

12.2.2.1 Reunión de retrospectiva del proyecto*

La reunión de retrospectiva del proyecto es una reunión para determinar las formas en las que la colaboración y eficacia del equipo puede mejorarse en futuros proyectos. También se analizan las oportunidades positivas, negativas y potenciales para mejorar. Esta reunión no tiene un time-box asignado y se puede llevar a cabo en forma presencial o en formato virtual. Entre los asistentes se encuentran: el equipo del proyecto, el Chief Scrum Master, el Chief Product Owner y el (los) stakeholder(s). Durante la reunión, se documentan las lecciones aprendidas y los participantes buscan oportunidades para mejorar los procesos y atender las ineficiencias.

12.2.2.2 Otras herramientas para la retrospectiva del proyecto

Algunas de las herramientas utilizadas en el proceso de retrospectiva del sprint también se pueden utilizar en este proceso. A continuación, se ofrecen ejemplos:

- Ejercicios de Explorador—Comprador—Vacacionista—Prisionero (ECVP) (véase la sección 11.2.2.2)
- Speed Boat (véase la sección 11.2.2.3)
- Parámetros y técnicas de medición

12.2.2.3 Experiencia del Scrum Guidance Body

Analizado en la sección 8.4.2.7.

En el proceso de *Retrospectiva del proyecto*, la responsabilidad principal del Scrum Guidance Body es asegurar que las lecciones aprendidas en cada proyecto no se hayan perdido y que estén integradas en la organización.

Asimismo, un Scrum Guidance Body puede aportar su experiencia en diversos ámbitos, entre ellos: la calidad, recursos humanos y Scrum, que pudiera ser útil en el proceso de *Retrospectiva del proyecto.* Además, dicho órgano puede ofrecer sugerencias respecto a cómo se debe llevar a cabo la reunión de retrospectiva del proyecto.

12.2.3 Salidas

12.2.3.1 Agreed Actionable Improvements *

Descrito en la sección 11.3.3.1.

12.2.3.2 Assigned Action Items y fechas límite*

Descrito en la sección 11.3.3.2.

12.2.3.3 Elementos no funcionales propuestos para el Program Product Backlog y el Backlog Priorizado del Producto

Cuando se desarrolla inicialmente el Program Product Backlog o el Backlog Priorizado del Producto, estos se basan en las historias de usuario y en las funcionalidades requeridas. Por lo general los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint, la retrospectiva del sprint o en las reuniones de retrospectiva del proyecto. Dichos elementos deben agregarse al Program Product Backlog (para el programa) y en el Backlog Priorizado del Producto (para el proyecto) conforme se van encontrando. Algunos ejemplos de requerimientos no funcionales son: tiempos de respuesta, limitaciones en la capacidad y asuntos en materia de seguridad.

12.2.3.4 Recomendaciones del Scrum Guidance Body actualizadas

Descritos en la sección 11.2.3.6

12.3 Diagrama de flujo de datos de fase

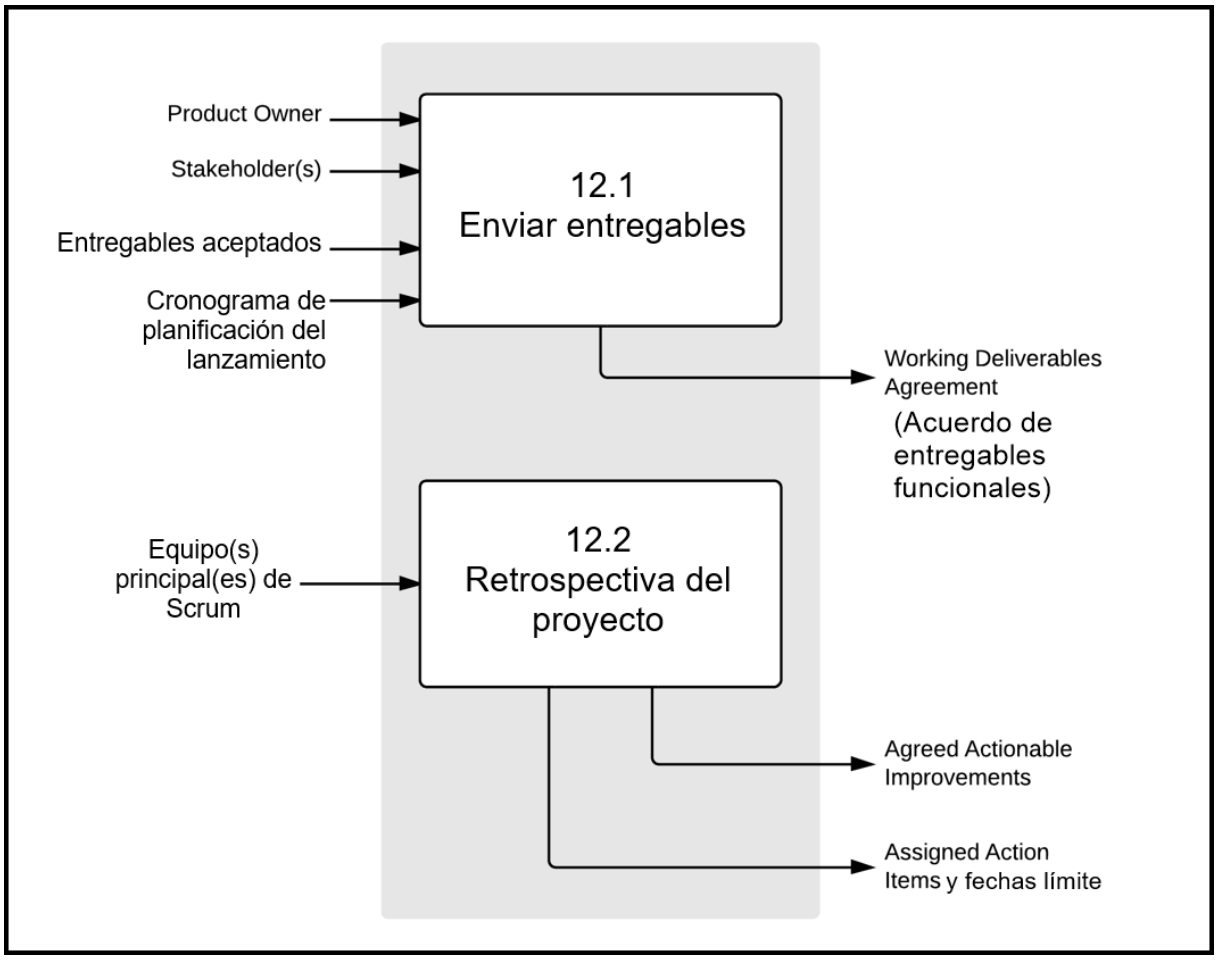


Figura 12-7: Fase de lanzamiento—Diagrama de flujo de datos

13. ESCALAMIENTO DE SCRUM EN GRANDES PROYECTOS

Este capítulo hace énfasis en aspectos adicionales de proyectos Scrums que aplican únicamente a grandes proyectos. Escalar Scrum en grandes proyectos, tal como se define en la *Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)* aplica a lo siguiente:

- Grandes proyectos en cualquier industria
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum no solo se aplica eficazmente en pequeños proyectos en cualquier industria, sino también en proyectos grandes y complejos con cientos de personas y varios equipos.

Además del impacto que tienen los grandes proyectos en los procesos fundamentales de Scrum en los capítulos del 8 al 12, este capítulo introduce tres procesos adicionales que aplican a nuevos grandes proyectos.

Para facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el Equipo Scrum y aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los Product Owners, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™,* no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un equipo de un proyecto grande que coordina actividades de varios equipos Scrum en un proyecto grande para producto incrementos/entregables potencialmente enviables. La información adicional sobre el uso de Scrum para cualquier proyecto, no solo en los granes, está disponible en los capítulos del 2 al 7, mismos que cubren los principios y aspectos de Scrum.

Grandes proyectos vs. Típico proyecto Scrum

Los procesos fundamentales de Scrum que se definen en los capítulos del 8 al 12 son válidos para todo proyecto Scrum. Los conceptos mencionados en dichos capítulos son suficientes para gestionar proyectos Scrum con tan solo algunos equipos Scrum, típicamente de uno a tres. El impacto de estos procesos fundamentales de Scrum que solo aplican a grandes proyectos se describen al final de este capítulo.

Al gestionar grandes proyectos que generalmente cuentan con cuatro o más equipos Scrum —además de los procesos definidos en los capítulos del 8 al 12—, tal vez sean necesarios algunos procesos adicionales para abordar la coordinación adicional y los esfuerzos de sincronización. La definición de "grandes proyectos" dependerá de la empresa y de la complejidad del proyecto emprendido. La principal regla de un proyecto grande en comparación a uno pequeño es contar con varios Scrum Masters y/o Product Owners.

Algunos de los motivos tras la necesidad de definir procesos adicionales para grandes proyectos son:

- Mayor interacción y dependencias entre los equipos Scrum, ya que la complejidad aumenta en los grandes proyectos.
- La necesidad de colaborar en un equipo de Product Owners.
- La necesidad de manejar conflictos, resolver problemas y fijar prioridades entre todos los equipos Scrum.
- La necesidad de la especialización, ya que algunos equipos Scrum pueden requerir de recursos especializados para realizar tareas específicas en un proyecto grande, y tal vez no todos los equipos Scrum cuenten con tales habilidades.
- La necesidad de definir ciertas normas y estándares que deben cumplir todos los equipos Scrum (ejemplo: estándares de seguridad dentro de una empresa o normas legales o gubernamentales en industrias específicas). Estas tal vez tengan que ser definidas en el Scrum Guidance Body.
- Los requerimientos para establecer un ambiente en un proyecto grande que después utilizarán todos los equipos Scrum.
- La necesidad de coordinar los resultados de varios equipos Scrum a fin de crear el lanzamiento de en un proyecto grande.

Los procesos adicionales relacionados al escalamiento de Scrum en grandes proyectos son los siguientes:

13.1 Crear componentes de grandes proyectos—Este proceso define la forma en la que los Product Owners trabajan en conjunto y de cómo varios equipos de Scrum trabajan juntos. También se identifican componentes comunes, así como recursos comunes y especializados.

13.2 Realizar y coordinar sprints— Por lo general, este proceso solo es relevante en grandes proyectos y aborda aspectos específicos que deben ser considerados durante cada sprint. De ser necesario, se pueden llevar a cabo reuniones de Scrum de Scrums a fin de coordinar los esfuerzos entre los distintos equipos de Scrum.

13.3 Preparar el lanzamiento de grandes proyectos—En algunos proyectos grandes, pudiera tener sentido empresarial llevar a cabo un sprint especial con anticipación a fin de preparar el lanzamiento del producto (que será decidido por el equipo del proyecto con base en las necesidades del negocio). Este proceso aborda dicho sprint preparatorio.

La figura 13-1 muestra todas las entradas, herramientas y salidas de los procesos en el Escalamiento de Scrum en grandes proyectos.

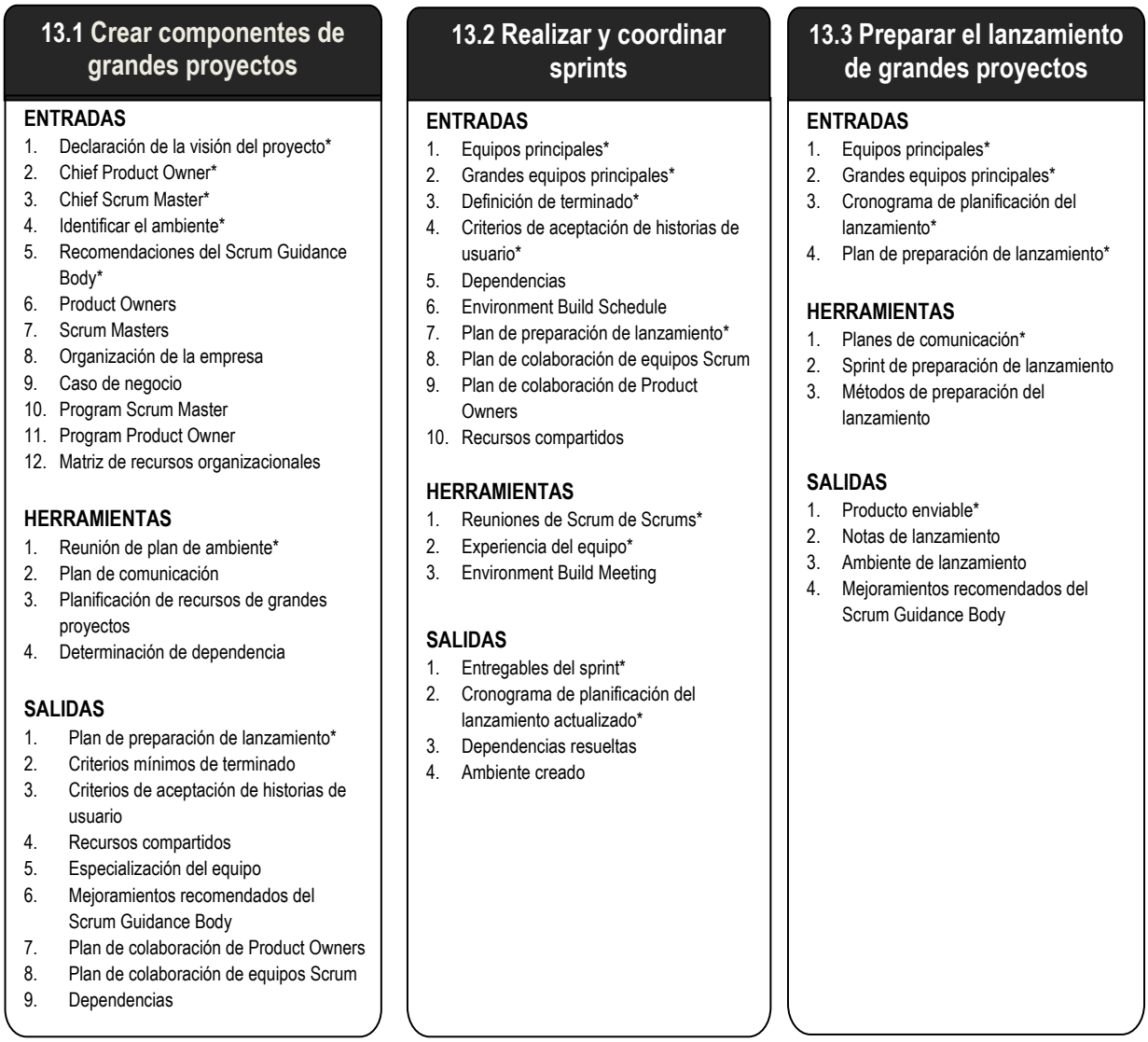


Figura 13-1: Descripción general del Escalamiento de Scrum en grandes proyectos

La figura 13-2 muestra las entradas, herramientas y salidas obligatorias de los procesos en el Escalamiento de Scrum en grandes proyectos.

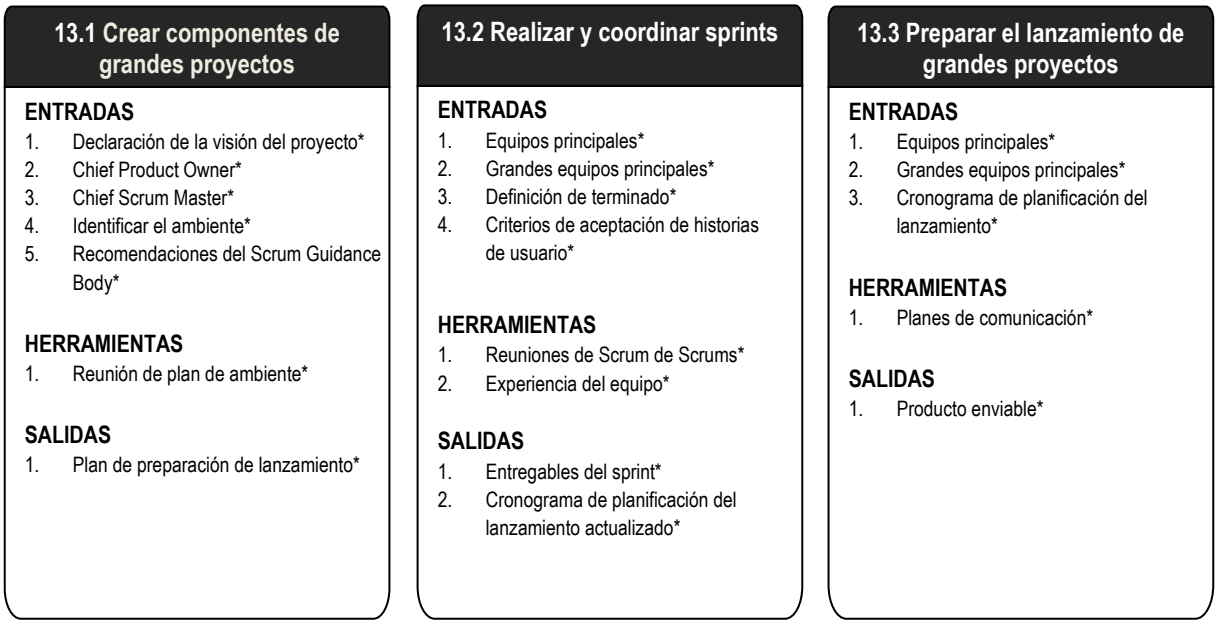


Figura 13-2: Descripción general de Escalamiento de Scrum en grandes proyectos (esenciales)

13.1 Crear componentes de grandes proyectos

La figura 13-3 muestra todas las entradas, herramientas y salidas del proceso de *Crear componentes de grandes proyectos*.

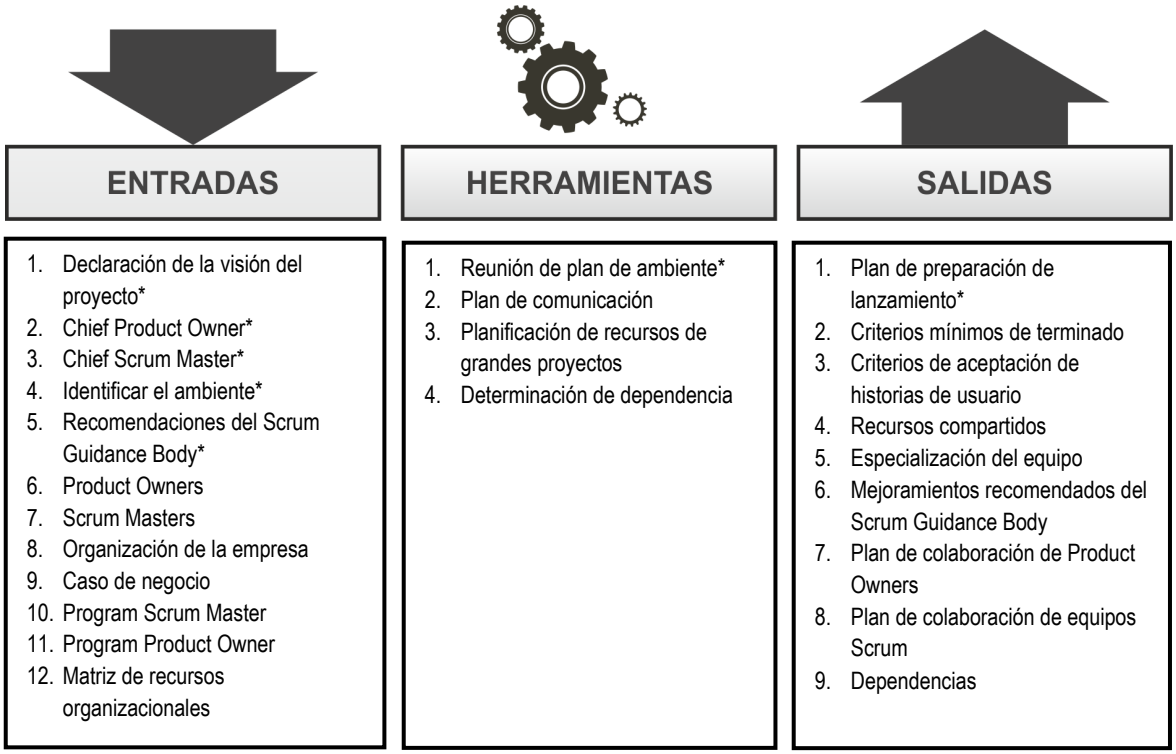


Figura 13-3: Crear componentes de grandes proyectos—Entradas, herramientas y salidas

La figura 13-4 muestra toda la relación del proceso de *Crear componentes de grandes proyectos* en los procesos fundamentales de Scrum en la fase de Inicio.

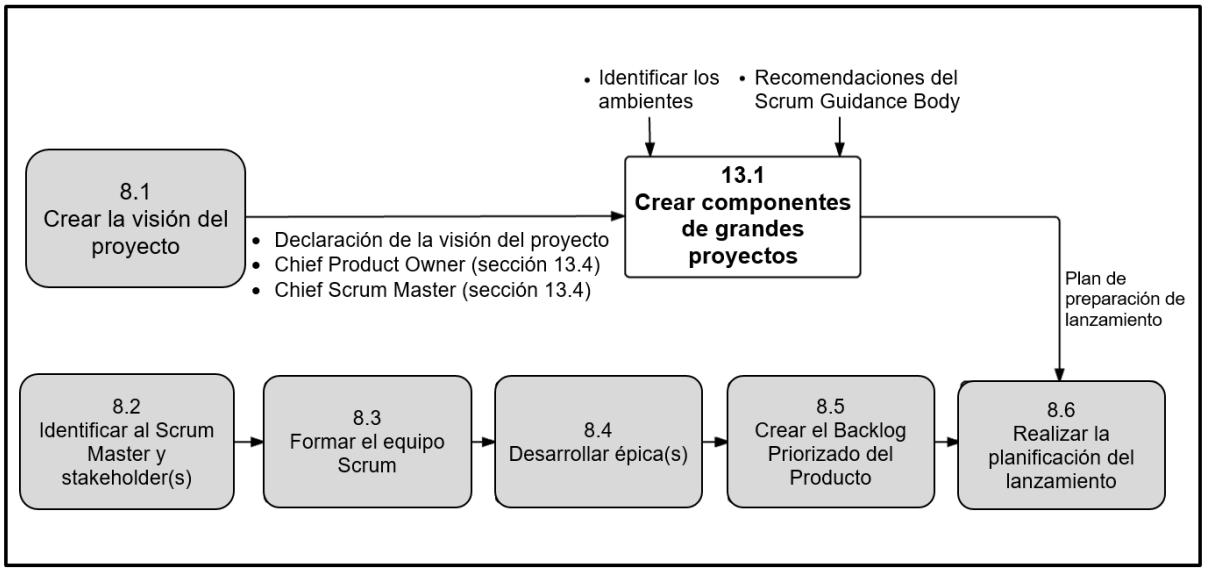


Figura 13-4: Crear componentes de grandes proyectos—Diagrama de flujo de datos

13.1.1 Entradas

13.1.1.1 Declaración de la visión del proyecto*

Descrito en la sección 8.1.3.2.

13.1.1.2 Chief Product Owner*

Descrito en la sección 3.4.2.

13.1.1.3 Chief Scrum Master*

Descrito en la sección 3.5.1.

13.1.1.4 Identificar el ambiente*

En un proyecto grande, es importante identificar el tipo y la cantidad de ambientes necesarios, ya que los varios equipos Scrum iniciarán y concluirán sus sprints el mismo día. Algunos ejemplos de ambiente incluyen: áreas de pruebas y desarrollo de software, recursos/áreas de trabajo físico o bien, límites procesales de cada Equipo Scrum.

13.1.1.5 Recomendaciones del Scrum Guidance Body

Descrito en la sección 8.1.1.11.

Para grandes proyectos, el Scrum Guidance Body es un recurso importante de información para brindar orientación y promover las mejores prácticas a fin de mejorar las posibilidades de éxito.

13.1.1.6 Product Owners

El rol básico de un Product Owner es el mismo para proyectos grandes y pequeños y se describe en la sección 3.4. Incluye la colaboración de Product Owners con sus respectivos equipos Scrum.

La principal diferencia en un proyecto grande es que el Product Owner ya no toma las decisiones prioritarias diarias, sino que brinda únicamente aportes y recomendaciones al Chief Product Owner. Además, las interacciones con los stakeholders se dividen entre todos los Product Owners. Cada Product Owner trabaja

con un equipo específico según los roles definidos. Todo esto se define en el Plan de Colaboración de Product Owners.

Los Product Owners colaboran con el Chief Scrum Master, con los Chief Product Owners, Scrum Masters y otros Product Owners para desarrollar la lista de componentes y recursos necesarios en común para todos los equipos a lo largo del proyecto. También ayudan a diseñar y aprobar el Plan de preparación del lanzamiento (*Release Readiness Plan*) que puede incluir una prueba de integración de fin a fin (*End to End Integration Test*).

13.1.1.7 Scrum Masters

El rol de un Scrum Master es el mismo en proyectos grandes y pequeños y se describe en la sección 3.5.

Los grandes proyectos generalmente cuentan con varios Scrum Masters, cada uno para facilitar y garantizar un ambiente de trabajo productivo en sus respectivos equipos Scrum; un Scrum Master puede trabajar con más de un Equipo Scrum. La identificación de Scrum Masters se lleva a cabo en el proceso de *Identificar al Scrum Masters y stakeholder(s)* (véase la sección 8.2).

La colaboración entre los varios equipos Scrum, que facilitan los Scrum Masters, se define en el Plan de colaboración de Equipos Scrum (véase la sección 13.1.3.8).

Los Scrum Masters colaboran con el Chief Scrum Master, el Chief Product Owner y otros Scrum Masters y Product Owners para desarrollar la lista de componentes y recursos necesarios en común para todos los equipos durante el proyecto. También bridan los aportes para la creación del Plan de preparación de lanzamiento (*Release Readiness Plan*).

13.1.1.8 Organización de la empresa

Las organizaciones que busquen implementar Scrum en grandes proyectos debe adoptar el framework de Scrum. Para los grandes proyectos, la organización debe contar con la capacidad de apoyar el esfuerzo mediante su compromiso con los recursos necesarios. De no ser posible, deben hacerse planes para obtener más recursos tales como personal, herramientas y espacio laboral. Es importante que la planificación empresarial para el uso de Scrum esté preparada para cambiar su cultura laboral y sus hábitos a fin de obtener los beneficios del uso de Scrum.

13.1.1.9 Caso de negocio

Descrito en las secciones 4.1.1, 4.4.2 y 8.1.1.1.

Aunque el caso de negocio como tal no es distinto al de un proyecto pequeño, con el aumento en el tamaño del proyecto, es de suma importancia contar con una sólida justificación de negocio. El impacto que tienen los bienes en un negocio —y en especial la mala selección de un proyecto—, es considerablemente más alto debido a la mayor cantidad de dinero y demás recursos involucrados en el proyecto.

13.1.1.10 Program Scrum Master

Descrito en la sección 3.5.2.

13.1.1.11 Program Product Owner

Descrito en la sección 3.4.3.

13.1.1.12 Matriz de recursos organizacionales (*Organizational Resource Matrix*)

En un entorno de grandes proyectos Scrum con varios equipos Scrums que compiten por la asignación de recursos y priorización de tareas, es importante gestionar los recursos organizacionales en forma óptima para lograr los objetivos generales del proyecto. Los Product Owners deben gestionar la creación de valor mediante planes organizacionales. Dichos planes deben abordar los componentes que serán desarrollados, las habilidades, los costos y demás recursos necesarios para desarrollarlos; así como las actuales velocidades de los equipos Scrum para calcular la duración del proyecto, los requerimientos de comunicación y otras interfaces que deben mantener dichos equipos.

Descrito en la sección 8.2.1.10.

13.1.2 Herramientas

13.1.2.1 Reunión de plan de ambiente*

Para definir un programa/calendario de cómo los equipos Scrum compartirán ambientes, se utiliza una reunión de plan de ambiente o *Environment Plan Meeting*. Por ejemplo, con equipos distribuidos en distintas zonas con distintos horarios, es posible hacer pruebas las 24 horas del día para maximizar el uso de los ambientes. Por ello, es de suma importancia crear un cronograma para determinar los horarios de pruebas para cada equipo. Para proyectos de software, el plan de ambiente, o *Environment Plan* también puede incluir información sobre cómo y qué código se promueve en cada ambiente.

13.1.2.2 Plan de comunicación

Los planes de comunicación son esenciales en un proyecto grande, ya que la mala comunicación, o la falta de comunicación entre los numerosos equipos Scrum puede ser perjudicial en los efectos colaborativos y resulte en el fracaso del proyecto. Los planes de comunicación deben de cubrir mensajes clave, métodos de comunicación, canales o mecanismos para comunicar los mensajes clave a los stakeholders apropiados, incluyendo la forma en la que el Chief Product Owner y otros Product Owners habrán de comunicar las prioridades a los equipos Scrum, las responsabilidades de comunicación, la clasificación de información delicada, los tiempos de actividades de comunicación y los procesos para evaluar la efectividad de la comunicación. Los planes de comunicación también deben de incluir el tiempo y la frecuencia de las reuniones de Scrum de Scrums (SoS) y la forma en la que dichas reuniones se llevarán a cabo.

Cada Equipo Scrum puede contar con un plan de comunicación que especifique los registros que deben ser creados y mantenidos durante el proyecto, así como una variedad de métodos para transmitir información importante del proyecto a los stakeholders. Especificará también quién es responsable de las varias actividades de comunicación. Véanse la sección 12.1.2.2.

13.1.2.3 Planificación de recursos de grandes proyectos

La planificación de recursos para grandes proyectos es esencial debido a la complejidad de la asignación de varios tipos de recursos a los numerosos equipos Scrum que trabajan en paralelo. Habrá necesidades en conflicto por la escasez de recursos y el Chief Product Owner y otros Product Owners planificarán la entrega del mayor valor en el menor tiempo posible. Por lo tanto, en la planificación de recursos en un proyecto grande, se deben de tomar en cuenta los varios costos relacionados a recursos, tales como personal, capacitación, hardware y software, servicios externos y espacio físico.

El Chief Product Owner y otros Product Owners tal vez tengan que coordinarse con recursos externos para adquirir recursos y aumentar el personal (ejemplo: tal vez sea necesario contratar recursos externos para que trabajen con el equipo existente de tiempo completo y tal vez tengan que interactuar con el equipo de gestión del vendedor existente dentro de la empresa). Cuando se contratan recursos externos, el Chief Product Owner y el equipo deben de apegarse a las políticas de la empresa respecto a los recursos externos y los vendedores.

En los grandes proyectos, el Chief Product Owner tal vez tenga que tomar en cuenta la planificación de recursos externos a fin de atender las necesidades de los equipos especializados y la necesidad de fijar el ambiente de los varios equipos Scrum que trabajan en forma paralela. El Chief Product Owner y los Product Owners pueden colaborar con los Scrum Masters y los equipos Scrum para definir las habilidades especiales que se requieren en los grandes proyectos, así como el número de recursos requeridos, los equipos Scrum que necesiten habilidades especiales y la estimación de las asignaciones.

13.1.2.4 Determinación de dependencia

Descrito en la sección 9.4.2.3.

En los grandes proyectos, las dependencias debidamente identificadas ayudan a los equipos Scrum a determinar cuáles decisiones pueden afectar a otros equipos. Pueden también influir el orden relativo en el que un solo Equipo Scrum ejecuta sus respectivas tareas para crear entregables del sprint.

13.1.3 Salidas

13.1.3.1 Plan de preparación del lanzamiento

Debido a que en cada sprint se produce un producto potencialmente enviable u otros entregables, en un proyecto pequeño, un lanzamiento se puede hacer después de cada sprint cuando tenga lógica empresarial hacerlo. En un proyecto grande, pudiera haber motivos empresariales válidos para no completar en su totalidad las actividades necesarias para la preparación del lanzamiento en cada sprint. Por ejemplo, un equipo de proyecto pudiera optar por llevar a cabo una serie de pruebas de desempeño (*Performance Tests*) caras y tardadas solo cuando esté cerca del lanzamiento o bien, llevar a cabo una serie de pruebas de integración de fin a fin (*End-to-End Integration Tests*) al acercarse un lanzamiento. De cualquier forma, deben cumplirse los criterios de terminado para sprints regulares.

Si se deben de tomar tales decisiones empresariales, será necesario un sprint de preparación de lanzamiento por separado (véase la sección 13.3, *Preparar el lanzamiento de grandes proyectos*). Tales decisiones empresariales con las respectivas justificaciones de negocio se describen en un Plan de preparación de lanzamiento. Dicho plan detalla los pasos a seguir por cada Equipo Scrum, así como el proyecto en su conjunto para confirmar que se hayan cumplido los requerimientos mínimos para el lanzamiento y que el producto o los componentes estén listos para el lanzamiento.

13.1.3.2 Criterios mínimos de terminado

Descrito en la sección 5.4.3. Los criterios de terminado se describen en la sección 8.5.3.2.

Para los grandes proyectos, existen aspectos adicionales que pudieran ser tan relevantes como los criterios de terminado, como el caso del traslado hacia un ambiente específico o el cumplimiento de ciertos criterios internos antes de avanzar a las pruebas de aceptación de uso, o *Use Acceptance Testing* (UAT).

Además, los criterios de terminado en los grandes proyectos deben ser considerados como criterios “mínimos” de terminado. Los Product Owners pueden definir criterios de terminado adicionales para tipos específicos de historias de usuario (ejemplo: para todas las historias de usuario que lleven a la creación de o cambios de un sitio web). Si están definidos, dichos criterios de terminado también deben ser tomados en

cuenta por los equipos Scrum que desarrollan los productos. Los criterios de terminado pueden describirse como listas de verificación (*checklists*) que pueden utilizarse en los niveles del sprint y del lanzamiento. Por lo tanto, además de los criterios mínimos de terminado, los Product Owners —en forma individual y sus equipos Scrum—, pueden definir criterios de terminado adicionales si así lo desean o si lo consideran necesario.

13.1.3.3 Criterios de aceptación de usuario

Descrito en la sección 9.1.3.2.

13.1.3.4 Recursos compartidos

Los recursos compartidos pueden ser el personal, el ambiente y el equipamiento que necesitan todos o algunos equipos Scrum que trabajan en un proyecto. En un proyecto grande, los recursos compartidos pueden estar limitados; son recursos que necesitan al mismo tiempo todos o parte de los equipos Scrum. En este contexto, el Chief Product Owner, el Chief Scrum Master, así como otros Product Owners y Scrum Masters deben desarrollar un método para utilizar estos recursos compartidos. Las responsabilidades de los entregables del sprint en un proyecto grande se distribuyen entre los varios equipos Scrum con distintas prioridades y disponibilidades cada uno. Un ejemplo de un método para la asignación de recursos compartidos pudiera ser garantizar que los recursos se destinen primero a las características de valor más importantes/elevadas y a los equipos que trabajan en ellas. Cuando las solicitudes competentes tienen prioridades muy similares, el Chief Product Owner debe decidir sobre la asignación de recursos con base en los últimos requerimientos del negocio.

13.1.3.5 Especialización del equipo

En un proyecto grande tal vez sea necesaria la especialización del equipo. Existen tres dimensiones de especialización de equipo.

La primera dimensión es la necesidad de cumplir con una tarea específica. Por ejemplo, un equipo especializado pudiera ser un equipo de integración que cuente con conocimiento especializado sobre la integración continua. Dicho conocimiento puede ser de especial importancia si existe un sprint de preparación de lanzamiento (*Release Readiness Sprint*).

La segunda dimensión es la necesidad de que los integrantes individuales del equipo cuenten con habilidades especiales. En teoría, todos los equipos Scrum son generalistas y especialistas, ya que tienen conocimiento sobre varias áreas y son expertos por lo menos en una. Sin embargo, este no pudiera ser el caso en un proyecto grande. Los miembros de equipos especializados deben tener habilidades específicas, tales como conocimiento de temas específicos como seguridad, lo cual no pudiera existir en todos los

equipos en grandes proyectos donde es necesario. Para un proyecto grande, sería demasiado costoso capacitar a todos en todas las áreas. Estos expertos con conocimiento y habilidades especiales pueden trabajar como miembros temporales del equipo en distintos equipos y en ocasiones pudiera ser necesario contratarlos de recursos externos. Agregar a un nuevo integrante al equipo afectará su velocidad.

La tercera dimensión es que pudiera haber limitaciones en la flexibilidad del equipo. Como se menciona anteriormente, en un proyecto grande sería demasiado costoso capacitar a todos en todas las áreas. Ello significa que cada equipo contará con una o más áreas que conozcan, algunas áreas en las que podrán trabajar con cierto aporte y capacitación, así como otras áreas en las que no puedan desempeñarse. Cuando se trata de la planificación del sprint, cada equipo contará con una sub-serie de historias de usuario que naturalmente les pertenecen, con algunas en las que pueden trabajar y con algunas en las que no pueden trabajar ya que no cuentan con el conocimiento o las habilidades.

El proyecto tendrá que tomar el riesgo de no poder terminar todas las historias de usuario más prioritarias en un sprint debido a estas limitaciones y tal vez tenga que desarrollar algunas historias de usuario de menor prioridad mientras espera la disponibilidad de miembros del equipo con habilidades especializadas.

13.1.3.6 Mejoramientos recomendados del Scrum Guidance Body

Como resultado de la planificación de grandes proyectos, se pueden hacer sugerencias para modificar o mejorar las recomendaciones del Scrum Guidance Body. Si dicho órgano acepta tales sugerencias, estas se incorporan como actualizaciones a la documentación del Scrum Guidance Body.

13.1.3.7 Plan de colaboración de Product Owners

El Plan de colaboración de Product Owners debe definir la forma en la que los múltiples Product Owners colaboran con el Chief Product Owner. Debe de definir, como mínimo, la cantidad de equipos Scrum que puede manejar un Product Owner (con base en experiencia, tiempo, conocimiento del tema), cómo se dividirá el trabajo entre los Product Owners para obtener los requerimientos de los stakeholders, cómo se actualizará el Backlog Priorizado del Producto con nuevos requerimientos o cambios en los mismos y cómo colaborarán los Product Owners con múltiples equipos Scrum. Debe señalarse que cada Equipo Scrum colaborará con solamente un Product Owner; sin embargo, un solo Product Owner puede manejar más de un Equipo Scrum si es necesario.

13.1.3.8 Plan de colaboración de Equipos Scrum

El Plan de colaboración de Equipos Scrum define la forma en la que los varios equipos Scrum colaboran entre sí a fin de proveer el más alto valor en el menor tiempo posible. Dicho plan debe incluir información sobre las áreas (dominios) especializados asignados a los equipos cualificados; la forma en la que los

equipos ayudarán en el refinamiento y estimación del Backlog Priorizado del Producto (ejemplo: decidir qué integrantes del equipo participarán en las sesiones de refinamiento y ejercicios de estimación de alto nivel), y sobre cómo los equipos organizarán las reuniones de Scrum de Scrums.

Este plan también puede incluir información sobre cómo será entrenado cada Equipo Scrum (ejemplo: ¿Habrá un coach por separado además de su Scrum Master? En el caso de equipos distribuidos ¿habrá un Scrum Master en cada ubicación? ¿Cómo colaborarán los miembros del equipo con los Scrum Masters co-ubicados y los no co-ubicados?).

13.1.3.9 Dependencias

Descrito en la sección 9.4.3.3.

13.2 Realizar y coordinar sprints

La figura 13-5 muestra todas las entradas, herramientas y salidas del proceso de *Realizar y coordinar sprints*.

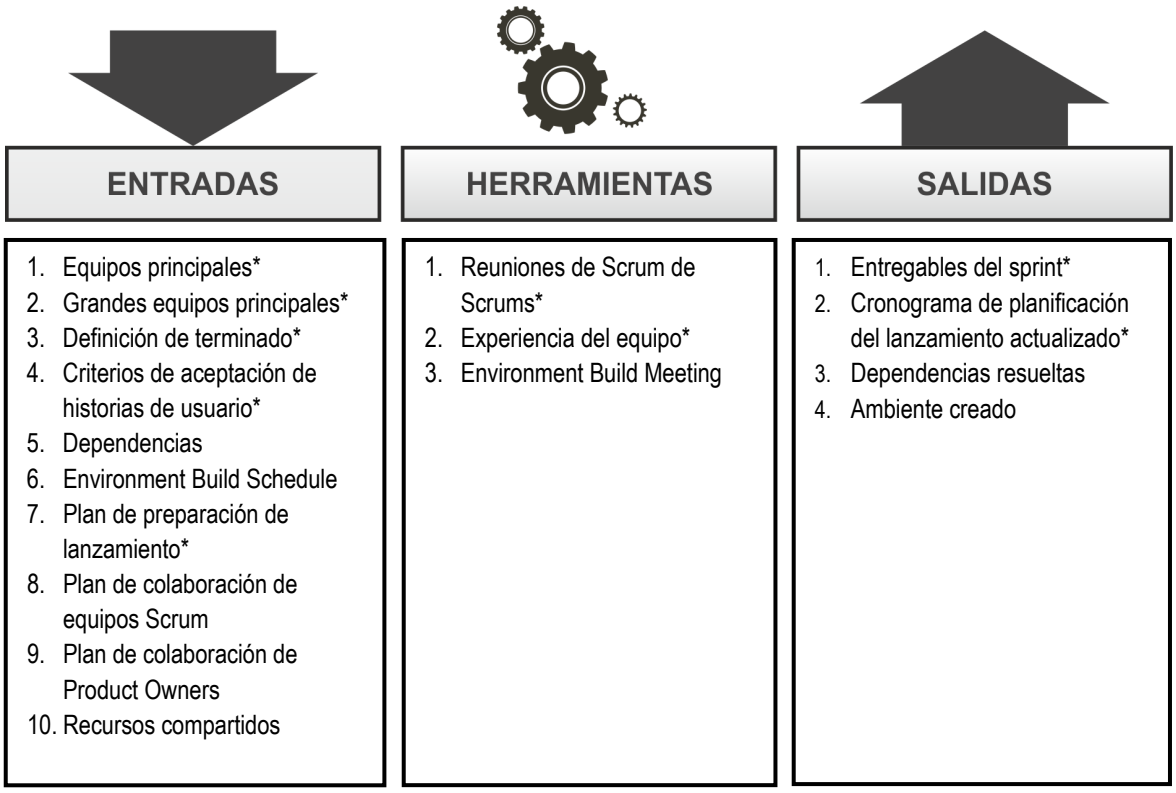


Figura 13-5: Realizar y coordinar sprints—Entradas, herramientas y salidas

La figura 13-6 muestra toda la relación del proceso de *Realizar y coordinar sprints* con los procesos fundamentales de Scrum.

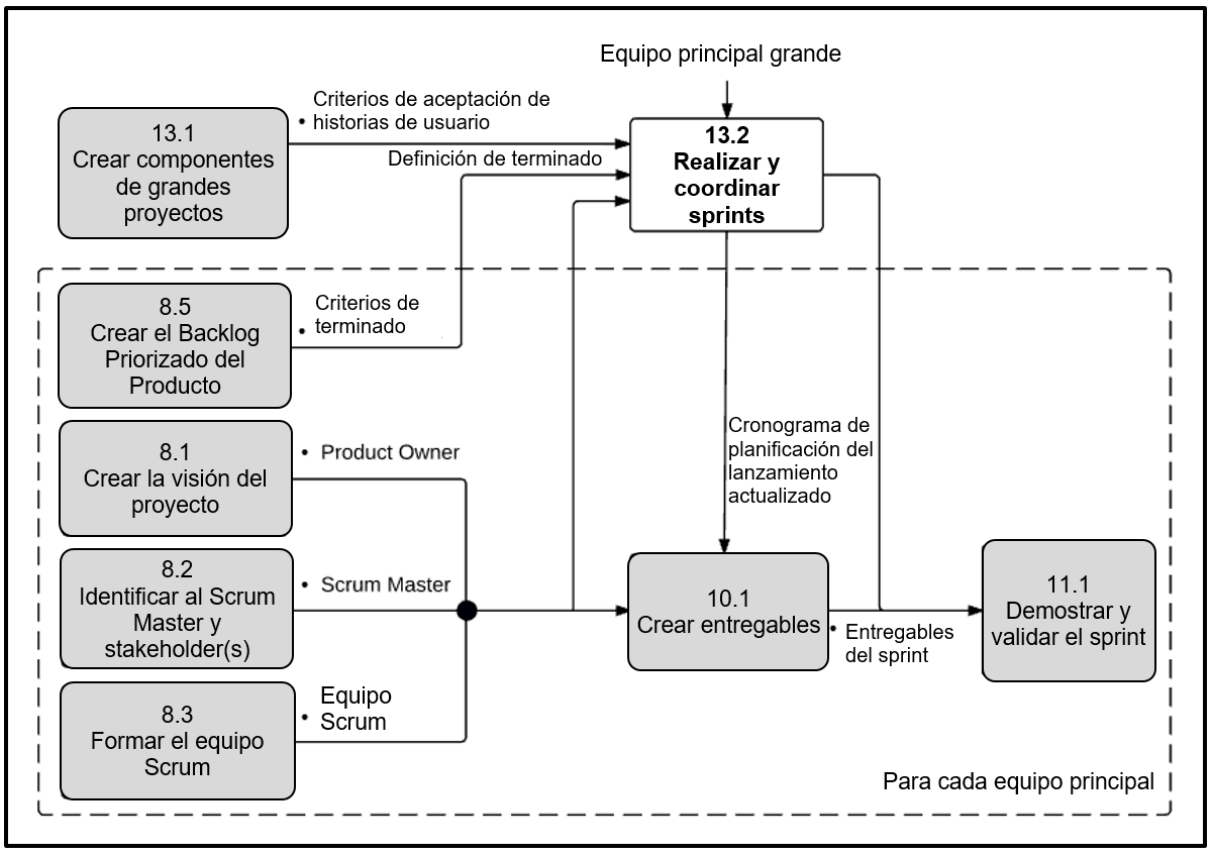


Figura 13-6: Realizar y coordinar sprints—Diagrama de flujo de datos

13.2.1 Entradas

13.2.1.1 Equipos principales*

Los múltiples equipos principales integrados por el Scrum Master, el Product Owner y el Equipo Scrum se describen en las secciones 8.4.1.1 y 3.3.1.

13.2.1.2 Grandes equipos principales*

Los grandes equipos principales se componen por el Chief Product Owner, el Chief Scrum Master, los Scrum Masters, Productos Owners y miembros seleccionados de los equipos Scrum en los grandes proyectos. No sería práctico tener un equipo principal grande que incluya a todos los integrantes de todos los equipos. Por lo tanto, los equipos pueden seleccionar a un miembro que debe representarlos en el equipo principal grande (*Large Core Team*).

El Chief Scrum Master elimina impedimentos y asegura un ambiente productivo en el proyecto a todos los equipos Scrum involucrados en el mismo. El Chief Product Owner prepara y mantiene el Backlog Priorizado del Producto general de los grandes proyectos y lo utiliza para coordinar el trabajo por medio de los Product Owners y Equipos Scrum. Los Product Owners son responsables de decidir la prioridad de las características y componentes que desarrollarán los equipos Scrum que forman parte de los grandes proyectos.

El Chief Scrum Master, el Chief Product Owner, Scrum Masters, Product Owners y los miembros seleccionados de los equipos Scrum colaboran para desarrollar la lista de componentes y recursos necesarios en común para todos los equipos durante el proyecto.

13.2.1.3 Definición de terminado

Descrito en las secciones 8.5.3.2 y 5.4.3.

Si hay un Sprint de preparación de lanzamiento (*Release Readiness Sprint*), sus criterios de terminado probablemente sean diferentes a los de otros sprints. Los criterios de terminado se definen para garantizar que los entregables del sprint sean “potencialmente enviables”. El Sprint de preparación de lanzamiento aborda todos los asuntos que no fueron atendidos en sprints regulares debido a decisiones empresariales deliberadas y se anotan en el Plan de preparación de lanzamiento (*Release Readiness Plan*).

13.2.1.4 Criterios de aceptación de historias de usuario

Descrito en la sección 9.1.3.2.

13.2.1.5 Dependencias

Descrito en las secciones 9.3.2.4 y 9.3.3.3.

13.2.1.6 Environment Schedule

El *Environment Schedule* es un horario/calendario que indica cómo los equipos Scrum compartirán los ambientes. Indica los días y periodos asignados a cada equipo para el uso de cada ambiente.

13.2.1.7 Plan de preparación de lanzamiento

Descrito en la sección 13.1.3.1.

13.2.1.8 Plan de colaboración de Equipos Scrum

Descrito en la sección 13.1.3.9.

13.2.1.9 Plan de colaboración de Product Owners

Descrito en la sección 13.1.3.7

13.2.1.10 Recursos compartidos

Los recursos compartidos pueden ser el personal, el ambiente y el equipamiento que necesitan todos o algunos equipos Scrum que trabajan en un proyecto. En un proyecto grande, los recursos compartidos pueden estar limitados; son recursos que necesitan al mismo tiempo todos o parte de los equipos Scrum. Véase la sección 13.1.3.4.

13.2.2 Herramientas

13.2.2.1 Reunión de Scrum de Scrums

Una reunión de Scrum de Scrums es un elemento importante en el escalamiento de Scrum para grandes proyectos. En la reunión típicamente hay un representante de cada Equipo Scrum, por lo general el Scrum Master, aunque también es común que cualquiera en el Equipo Scrum asista a la reunión en caso de ser necesario. Esta reunión generalmente la organiza el Chief Scrum Master y la intención es enfocarse en áreas de coordinación e integración entre los distintos equipos Scrum.

De preferencia estas reuniones son breves (aunque generalmente sin un time-box para permitir el intercambio de información entre los equipos) donde un representante de cada Equipo Scrum se reúne para compartir el estado de los equipos respectivos. Las reuniones Scrum de Scrum (SoS) se llevan a cabo en intervalos predeterminados o cuando lo requieran los equipos Scrum para facilitar el intercambio de información entre los distintos equipos. Se pueden monitorear los problemas, las dependencias y los riesgos que afectan a los múltiples equipos Scrum, lo cual ayuda a integrar y coordinar mejor el trabajo de los varios equipos que trabajan en un proyecto grande. Es responsabilidad del Chief Scrum Master (u otro Scrum Master que organice las reuniones de Scrum de Scrums) asegurarse de que todos los representantes cuenten con un ambiente conductivo para el intercambio abierto y honesto de información, que incluya la retroalimentación de los representantes de otros equipos. Para los grandes proyectos que involucran a una cantidad considerable de equipos, se pueden convocar a múltiples niveles de dichas reuniones para compartir el estado de los equipos respectivos.

Cada representante de los equipos Scrum proporcionará actualizaciones sobre su equipo. Dichas actualizaciones generalmente se presentan en forma de respuesta a cuatro preguntas específicas:

1) ¿En qué ha estado trabajando mi equipo desde la última reunión?
2) ¿Qué hará mi equipo hasta la próxima reunión?
3) ¿Qué esperaban otros equipos que mi equipo hiciera y que aún no se ha hecho?
4) ¿Qué planea hacer nuestro equipo que pudiera afectar a otros equipos?

Las respuestas a estas cuatro preguntas proporcionan información que permite a cada equipo entender claramente la situación de trabajo de todos los demás equipos.

13.2.2.2 Experiencia del equipo*

Descrito en la sección 10.1.2.1.

Asimismo, para los grandes proyectos esto también incluye el conocimiento de la experiencia en equipo del Chief Product Owner y Chief Scrum Master. Este último es responsable de evaluar las opciones para desarrollar las historias de usuario para cada sprint.

13.2.2.3 Environment Meeting (*Reunión de ambiente*)

Esta reunión se lleva a cabo para identificar el tipo y la cantidad de ambientes necesarios para desarrollar, administrar y evaluar los entregables del proyecto. En esta reunión también se discuten los recursos necesarios para establecer los ambientes requeridos.

13.2.3 Salidas

13.2.3.1 Entregables del sprint*

Los entregables del sprint son los productos potencialmente enviables desarrollados por los equipos Scrum al final de cada sprint. Los entregables deben contar con todas las características y funcionalidades definidas en las historias de usuario incluidas en los sprints y deben ser evaluadas satisfactoriamente.

13.2.3.2 Cronograma de planificación del lanzamiento actualizado

El plan de preparación de lanzamiento (*Release Readiness Plan*) detalla los pasos que cada Equipo Scrum y el proyecto en su conjunto deben de seguir para confirmar que se hayan cumplido los requerimientos mínimos para el lanzamiento. Este se puede actualizar con cambios identificados en este proceso.

13.2.3.3 Dependencias resueltas

Las dependencias entre las historias de usuario, las tareas y los recursos necesarios para convertir las historias de usuario en entregables requieren que los grandes proyectos cuenten con un plan de acción de dependencias (*Dependency Action Plan*). Este plan cubre todas las acciones necesarias para gestionar todo tipo de dependencias: obligatorias, discrecionales, internas y externas.

13.2.3.4 Ambiente(s)

Es la identificación y documentación de todos los ambientes necesarios para desarrollar y evaluar los entregables del proyecto.

13.3 Preparar el lanzamiento de grandes proyectos

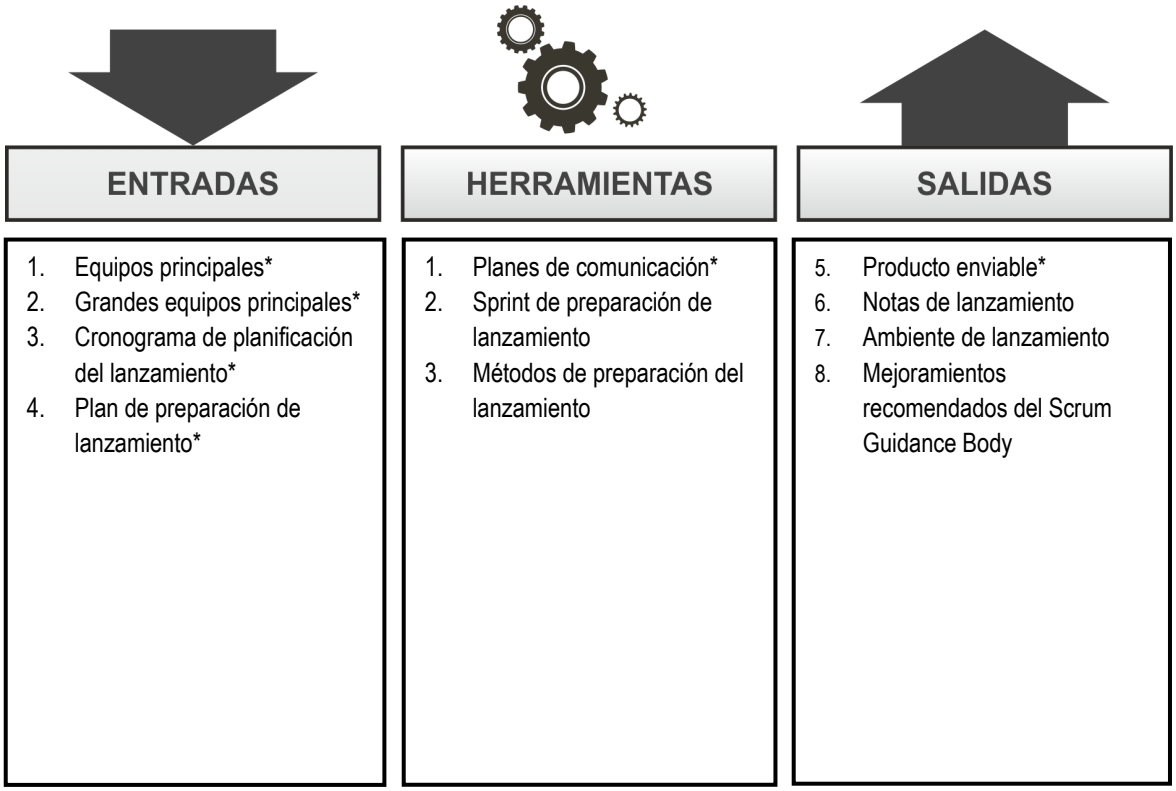


Figura 13-7: Preparar el lanzamiento de grandes proyectos—Entradas, herramientas y salidas

La figura 13-8 muestra toda la relación del proceso de *Preparar el lanzamiento de grandes proyectos* con los procesos fundamentales de Scrum.

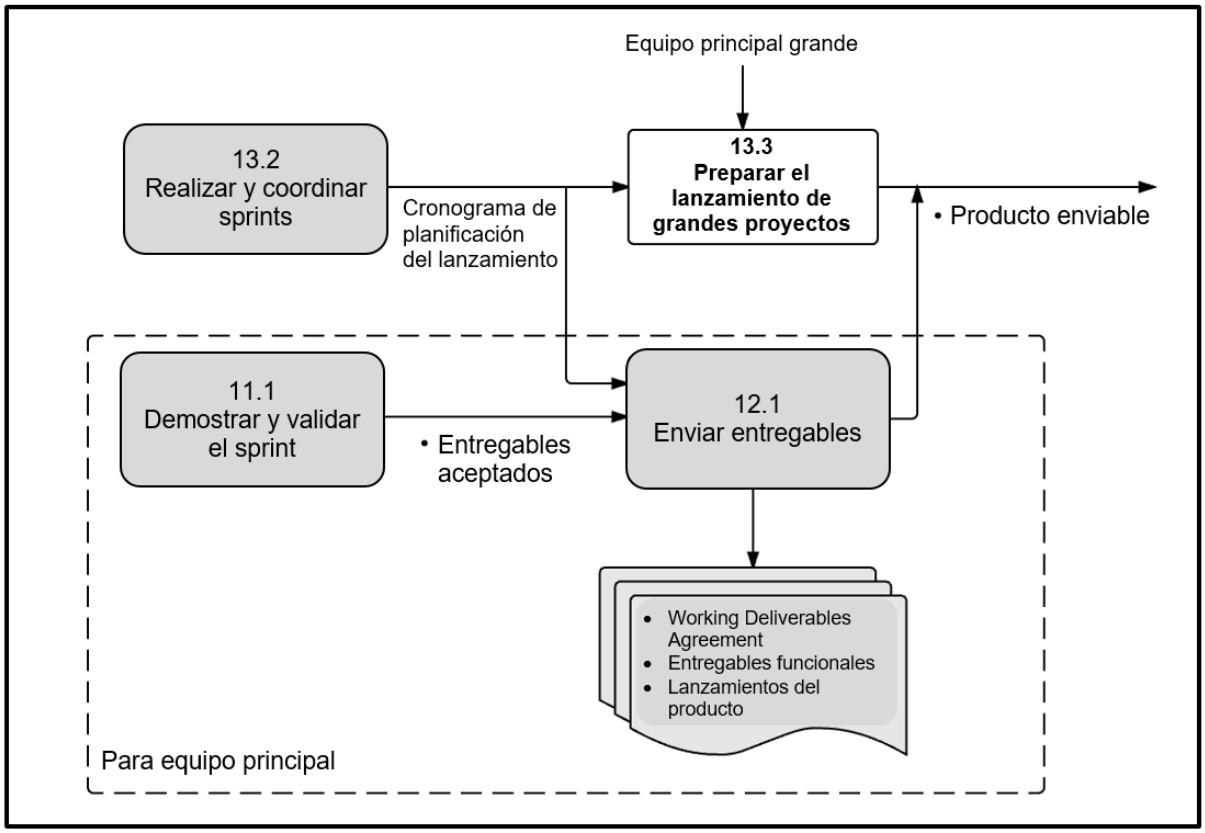


Figura 13-8: Preparar el lanzamiento de grandes proyectos—Diagrama de flujo de datos

13.3.1 Entradas

13.3.1.1 Equipos principales*

Descrito en las secciones 8.4.1.1 y 3.3.1.

13.3.1.2 Grandes equipos principales*

Descrito en la sección 13.2.1.2.

13.3.1.3 Cronograma de planificación del lanzamiento*

Descrito en la sección 8.6.3.1
.

13.3.1.4 Plan de preparación de lanzamiento

Descrito en la sección 13.1.3.1.

13.3.2 Herramientas

13.3.2.1 Plan de comunicaciones*

Descrito en las secciones 12.1.2.2. y 13.1.2.1.

13.3.2.2 Sprint de preparación del lanzamiento (*Release Readiness Sprint*)

Si hay necesidad de llevar a cabo tareas específicas para preparar un lanzamiento y confirmar que se hayan cumplido los requerimientos mínimos para el lanzamiento, dichas tareas se llevan a cabo en un sprint de preparación del lanzamiento (*Release Readiness Sprint*). De ser necesario, dicho sprint solo se hace una vez por cada lanzamiento como último sprint previo al lanzamiento. En un sprint de preparación de lanzamiento no se desarrollan historias de usuario del Backlog Priorizado del Producto. En cambio, se llevan a cabo las tareas identificadas en el Plan de preparación de lanzamiento (véase la sección 13.1.3.1).

Es importante recordar que dichas tareas se llevan a cabo en un sprint de preparación del lanzamiento como resultado de una decisión empresarial. Esto no cambia en lo absoluto la obligación de cumplir con los criterios de aceptación y de terminado al final de cada sprint.

Un sprint de preparación de lanzamiento (*Release Readiness Sprint*) **no** es obligatorio. Aunque no es obligatorio, en muchos proyectos grandes se toma la decisión empresarial de contar con dicho sprint.

13.3.2.3 Métodos de preparación de lanzamiento

Los métodos de preparación de lanzamiento son métodos que se utilizan para ejecutar las tareas identificadas en el Plan de preparación de lanzamiento (*Release Readiness Plan*) a fin de tener los entregables listos para ser enviados/lanzados. Estos métodos pueden ser específicos en cada proyecto, pero con más posibilidades de ser válidos en un portafolio o al menos al nivel del programa. Se pueden definir en el Scrum Guidance Body.

13.3.3 Salidas

13.3.3.1 Producto enviable*

Un producto enviable es un entregable o incremento de producto que cumple con los criterios de aceptación definidos por el cliente y el Product Owner. Estará listo para ser enviado o lanzado al final de un sprint de preparación del lanzamiento (*Release Readiness Sprint*).

13.3.3.2 Notas de lanzamiento

Las notas de lanzamiento (*Release Notes*) son los documentos proporcionados al cliente junto con el lanzamiento del producto. Incluyen criterios externos de envío o relacionados al mercado para que el producto sea enviado.

13.3.3.3 Ambiente de lanzamiento

El ambiente de lanzamiento (*Release Environment*) es necesario para ayudar al lanzamiento al campo.

13.3.3.4 Mejoramientos recomendados del Scrum Guidance Body

Descrito en la sección 13.1.3.6.

13.4 Impacto de los grandes proyectos en los procesos fundamentales de Scrum

Aunque los procesos fundamentales de Scrum descritos en los capítulos del 8 al 12 siguen siendo válidos en los grandes proyectos, existen efectos anticipados específicos que deben señalarse. La tabla 13-1 describe el impacto de los grandes proyectos en los procesos fundamentales de Scrum.

Proceso	Descripción del impacto(s) del proceso en un proyecto grande
8.1 Crear la visión del proyecto	**Salida adicional: Chief Product Owner identificado*** En un proyecto grande, el Chief Product Owner se identifica como una salida en este proceso. También se identifican múltiples Product Owners, mientras que en los proyectos pequeños solo requieren de un solo Product Owner. **Salida adicional: Chief Scrum Master identificado*** Al igual que el Chief Product Owner, el Chief Scrum Master también debe ser identificado en un proyecto grande.
8.2 Identificar al Scrum Master y stakeholder(s)	**Salida: Stakeholders identificados** Además de identificar a los stakeholders del proyecto, como parte de este proceso también se deben identificar algunos de los miembros clave de los equipos Scrum como principales stakeholders en los grandes proyectos. Estos stakeholders serán las entradas en el proceso de *Crear componentes de grandes proyectos*.
8.3 Formar el Equipo Scrum	**Entrada adicional: Chief Product Owner** Descrito en la sección 3.4.2. Para los grandes proyectos, el Chief Product Owner participa en la formación del Equipo Scrum y tienen injerencia sobre los miembros del equipo. El Chief Product Owner ayuda a los intereses de los grandes proyectos en su conjunto, mientras que los Product Owners se enfocan en el nivel de equipos individuales. **Entrada adicional: Chief Scrum Master** Descrito en la sección 3.5.1. Para los grandes proyectos, el Chief Scrum Master participa en la formación del Equipo Scrum y tienen injerencia sobre los miembros del equipo. El Chief Scrum Master ayuda a los intereses de los grandes proyectos en su conjunto, mientras que los Scrum Masters se enfocan en el nivel de equipos individuales. **Entrada adicional: Recursos compartidos** Descrito en la sección 13.1.3.4. El conocimiento de cualquier recurso compartido disponible a los equipos Scrum será una entrada necesaria en la formación de equipos individuales de Scrum. **Entrada adicional: Especialización del equipo** Descrito en la sección 13.1.3.5.

Proceso	Descripción del impacto(s) del proceso en un proyecto grande
8.4 Desarrollar épica(s)	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. Es una de las principales entradas del proceso de *Crear componentes de grandes proyectos*. Define la forma en la que los múltiples Product Owners colaboran con el Chief Product Owner. Aborda la forma en que trabajan con los stakeholders para obtener requerimientos, hacer actualizaciones al Backlog Priorizado del Producto y el trabajo con los múltiples equipos Scrum (seguirá habiendo solo un Product Owner como contacto directo con cada Equipo Scrum; pero deben decidir cómo dividir a los equipos Scrum entre los Product Owners y con cuántos equipos trabajará cada Product Owner).
8.5 Crear el Backlog Priorizado del Producto	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. Es una entrada importante en este proceso ya que el Plan de colaboración de Product Owners define cómo los Product Owners actualizan el Backlog Priorizado del Producto. **Entrada adicional: Dependencias** Descrito en la sección 9.4.3.3.
8.6 Realizar la planificación del lanzamiento	**Entrada adicional: Chief Scrum Master** Descrito en la sección 3.5.1. **Entrada adicional: Plan de preparación del lanzamiento (*Release Readiness Plan)*** Descrito en la sección 13.1.3.1. El Plan de preparación del lanzamiento incluye información importante que afecta el impacto de la planificación general del lanzamiento.
9.1 Crear historias de usuario	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. **Entrada adicional: Plan de colaboración de equipos Scrum** Descrito en la sección 13.1.3.8.
9.2 Estimar historias de usuario	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. **Entrada adicional: Plan de colaboración de equipos Scrum** Descrito en la sección 13.1.3.8.
9.3 Comprometer historias de usuario	Sin cambio
9.4 Identificar tareas	Sin cambio
9.5 Estimar tareas	Sin cambio
9.6 Crear el Sprint Backlog	Sin cambio

Proceso	Descripción del impacto(s) del proceso en un proyecto grande
10.1 Crear entregables	Sin cambio
10.2 Realizar Daily Standup	Sin cambio
10.3 Refinar el Backlog Priorizado del Producto	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. El plan de colaboración de Product Owners define cómo los Product Owners hacen actualizaciones el Backlog Priorizado del Producto. **Entrada adicional: Plan de colaboración de equipos Scrum** Descrito en la sección 13.1.3.8. El Plan de colaboración de equipos Scrum define la forma en que los equipos participan en el refinamiento del Backlog Priorizado del Producto. Este plan también define cuáles representantes de equipo estarían involucrados en el proceso de refinamiento (*grooming*) y cómo serían seleccionados. **Entrada adicional: Especialización del equipo** Descrito en la sección 13.1.3.5. **Salida adicional: Plan de preparación del lanzamiento actualizado** Los cambios al Product Backlog que se hacen durante el refinamiento (*grooming*) del Product Backlog pueden afectar el Plan de preparación del lanzamiento (véase la sección 13.1.3.1).
11.1 Demostrar y validar el sprint	Este proceso se hace por separado para cada Equipo Scrum en forma individual. Para cada equipo, su respectivo Producto Owner aprueba las historias de usuario; sin embargo, esto puede ser algo complejo debido a las inter-dependencias. En algunas ocasiones el Chief Product Owner puede asistir a cada reunión de revisión del sprint en cada equipo. En ciertos casos pudiera haber una validación de fin a fin que pudiera causar que ciertas cosas parezcan terminadas en la validación del equipo y que no cumplan con la aceptación final cuando se vean en la validación de fin a fin (*end-to-end validation*).
11.2 Retrospectiva del sprint	**Entrada adicional: Plan de colaboración de Product Owners** Descrito en la sección 13.1.3.7. Para los grandes proyectos, el refinamiento del Product Backlog es algo difícil de hacer; puede crear problemas y causar pérdida de tiempo. Por lo tanto, se recomienda que el refinamiento (*grooming*) sea un tema específico de la retrospectiva. Un aspecto específico del *grooming* es ver el inter-funcionamiento de los múltiples Product Owners y cómo interactúan con los múltiples equipos para el refinamiento. **Entrada adicional: Plan de colaboración de equipos Scrum** Descrito en la sección 13.1.3.8. Para los grandes proyectos, el refinamiento del Product Backlog es algo difícil de hacer; puede crear problemas y causar pérdida de tiempo. Por lo tanto, se recomienda que el refinamiento (*grooming*) sea un tema específico de la retrospectiva. Un aspecto específico del *grooming* es ver el inter-funcionamiento de los múltiples equipos Scrum y cómo interactúan con los múltiples Product Owners para el refinamiento.

Proceso	Descripción del impacto(s) del proceso en un proyecto grande
12.1 Enviar entregables	**Entrada adicional: Chief Product Owner** Descrito en la sección 3.4.2. **Entrada adicional: Chief Scrum Master** Descrito en la sección 3.5.1. **Entrada adicional: Notas de lanzamiento** Para los grandes proyectos, las notas de lanzamiento del proceso de *Preparar el lanzamiento de grandes proyectos* se convierten en entrada en este proceso.
12.2 Retrospectiva del proyecto	**Entrada adicional: Chief Product Owner** Descrito en la sección 3.4.2. **Entrada adicional: Chief Scrum Master** Descrito en la sección 3.5.1.

Tabla 13-1: Resumen del impacto de los grandes proyectos en los procesos fundamentales de Scrum

14. ESCALAMIENTO DE SCRUM PARA LA EMPRESA

Este capítulo incluye los procesos relacionados al escalamiento de Scrum para le empresa: *Crear componentes del programa o portafolio*, *Revisar y actualizar el Scrum Guidance Body*, *Crear y refinar el Program o Portfolio Backlog*, *Coordinar los componentes del programa o portafolio* y *Retrospectiva de los lanzamientos del programa o portafolio*.

El escalamiento de Scrum para la empresa, tal como se define en la *Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)* aplica a:

- Portafolios, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se entregará a los stakeholders;
- Proyectos de cualquier tamaño o complejidad.

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum no solo se aplica eficazmente en pequeños proyectos en cualquier industria, sino también en proyectos grandes y complejos con cientos de personas y varios equipos.

Para facilitar la mejor aplicación del framework de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas con asteriscos (*) son obligatorias, o consideradas importantes para el éxito, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que aquellas personas que empiezan a aprender sobre el framework y los procesos de Scrum se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los practicantes con experiencia en Scrum, se esfuercen por alcanzar un conocimiento más a fondo sobre la información en todo este capítulo.

La figura 14-1 ofrece una descripción general del proceso del Escalamiento de Scrum para la empresa:

14.1 Crear componentes de programa o portafolio—En este proceso, el Program Product Owner o el Portfolio Product Owner, así como los stakeholders clave identifican componentes comunes y recursos necesarios para el programa o portafolio. Los criterios mínimos de terminado se definen y se identifican a todos los stakeholders.

14.2 Revisar y actualizar el Scrum Guidance Body—En este proceso, las recomendaciones del Scrum Guidance Body se revisan constantemente por parte de sus miembros y se actualizan cuando sea necesario. En este proceso, también se atienden los cambios en los integrantes del Scrum Guidance Body.

14.3 Crear y refinar el backlog del programa o portafolio—En este proceso, se elabora, se actualiza y se refina el backlog del programa o portafolio. Se pueden hacer recomendaciones de mejoramiento por parte del Scrum Guidance Body y se pueden modificar los plazos con base en los cambios en los requerimientos y/o procesos del proyecto en el programa o portafolio.

14.4 Coordinar los componentes del programa o portafolio—En este proceso se coordinan los componentes del programa o portafolio. Se atienden las dependencias entre proyectos; se discuten los impedimentos comunes y se comparten las mejores prácticas. En ocasiones, se hacen recomendaciones de mejoramiento por parte del Scrum Guidance Body.

14.5 Retrospectiva de lanzamientos del programa o portafolio—En este proceso, el Program Product Owner o el Portfolio Product Owner y los stakeholders clave se reúnen para hacer una retrospectiva sobre el lanzamiento de un programa o portafolio e internalizar las lecciones aprendidas. Por lo general dichas lecciones aprendidas llevan a mejoras aceptadas (*Agreed Actionable Improvements*) para ser implementadas a futuro.

14.1 Crear componentes de programa o portafolio

ENTRADAS
1. Misión y visión de la compañía*
2. Portfolio Product Owner*
3. Portfolio Scrum Master*
4. Program Product Owner*
5. Program Scrum Master*
6. Matriz de recursos organizacionales
7. Recomendaciones del Scrum Guidance Body
8. Stakeholders clave

HERRAMIENTAS
1. Planes de comunicación*
2. Planes de Recursos Humanos de la empresa*
3. Análisis de Stakeholder

SALIDAS
1. Criterios mínimos de terminado*
2. Criterios de aceptación de historias de usuario*
3. Recursos compartidos*
4. Stakeholders identificados*
5. Mejoramientos recomendados del Scrum Guidance Body

14.2 Revisar y actualizar el Scrum Guidance Body

ENTRADAS
1. Regulaciones*
2. Mejoramientos recomendados del Scrum Guidance Body *
3. Miembros del Scrum Guidance Body

HERRAMIENTAS
1. Criterios de selección de miembros*
2. Benchmarking
3. Reuniones del Scrum Guidance Body

SALIDAS
1. Recomendaciones actualizadas del Scrum Guidance Body*
2. Actionable Escalations
3. Membresía actualizada del Scrum Guidance Body
4. Actualizaciones rechazadas a las recomendaciones del Scrum Guidance Body

14.3 Crear y refinar el backlog del programa o portafolio

ENTRADAS
1. Misión y visión de la compañía*
2. Portfolio Backlog Priorizado*
3. Program Backlog Priorizado*
4. Portfolio Product Owner*
5. Portfolio Scrum Master*
6. Program Product Owner*
7. Program Scrum Master*
8. Recomendaciones del Scrum Guidance Body
9. Políticas de la empresa
10. Estándares de la industria
11. Resultado de la evaluación/benchmarking

HERRAMIENTAS
1. Reuniones de revisión del Backlog Priorizado del Programa o el Portafolio *
2. Técnicas de comunicación*
3. Métodos de priorización de historias de usuario
4. Taller de historia de usuario
5. Entrevistas con el usuario o cliente
6. Cuestionarios

SALIDAS
1. Backlog priorizado del programa o el portafolio actualizado*
2. Mejoramientos recomendados del Scrum Guidance Body *
3. Fechas límite de implementación actualizadas para los proyectos*
4. Prototipos
5. Riesgos identificados

14.4 Coordinar los componentes del programa o portafolio

ENTRADAS
1. Definición de terminado*
2. Dependencias conocidas*
3. Backlog Priorizado del Programa o Portafolio*
4. Portfolio Product Owner*
5. Portfolio Scrum Master*
6. Program Product Owner*
7. Program Scrum Master*
8. Entregables potencialmente enviables de los proyectos
9. Impediments Logs
10. Backlogs Priorizados del Producto
11. Lecciones aprendidas del equipo Scrum
12. Cronogramas de planificación del lanzamiento

HERRAMIENTAS
1. Reunión de Scrum de Scrums (SoS) y Scrum de Scrum de Scrums*
2. Técnicas de comunicación

SALIDAS
1. Impediments Logs actualizados*
2. Dependencias actualizadas*
3. Mejoramientos recomendados del Scrum Guidance Body

14.5 Retrospectiva de lanzamientos del programa o portafolio

ENTRADAS
1. Portfolio Product Owner*
2. Portfolio Scrum Master*
3. Program Product Owner*
4. Program Scrum Master*
5. Stakeholders
6. Recomendaciones del Scrum Guidance Body

HERRAMIENTAS
1. Reunión de retrospectiva del programa o portafolio*
2. Experiencia del Scrum Guidance Body

SALIDAS
1. Agreed Actionable Improvements*
2. Assigned Action Items y fechas límite*
3. Mejoramientos recomendados del Scrum Guidance Body

Figura 14.1: Descripción general del Escalamiento de Scrum para la empresa

La figura 14-2 muestra las entradas, herramientas y salidas obligatorias de los procesos en el Escalamiento de Scrum para la empresa.

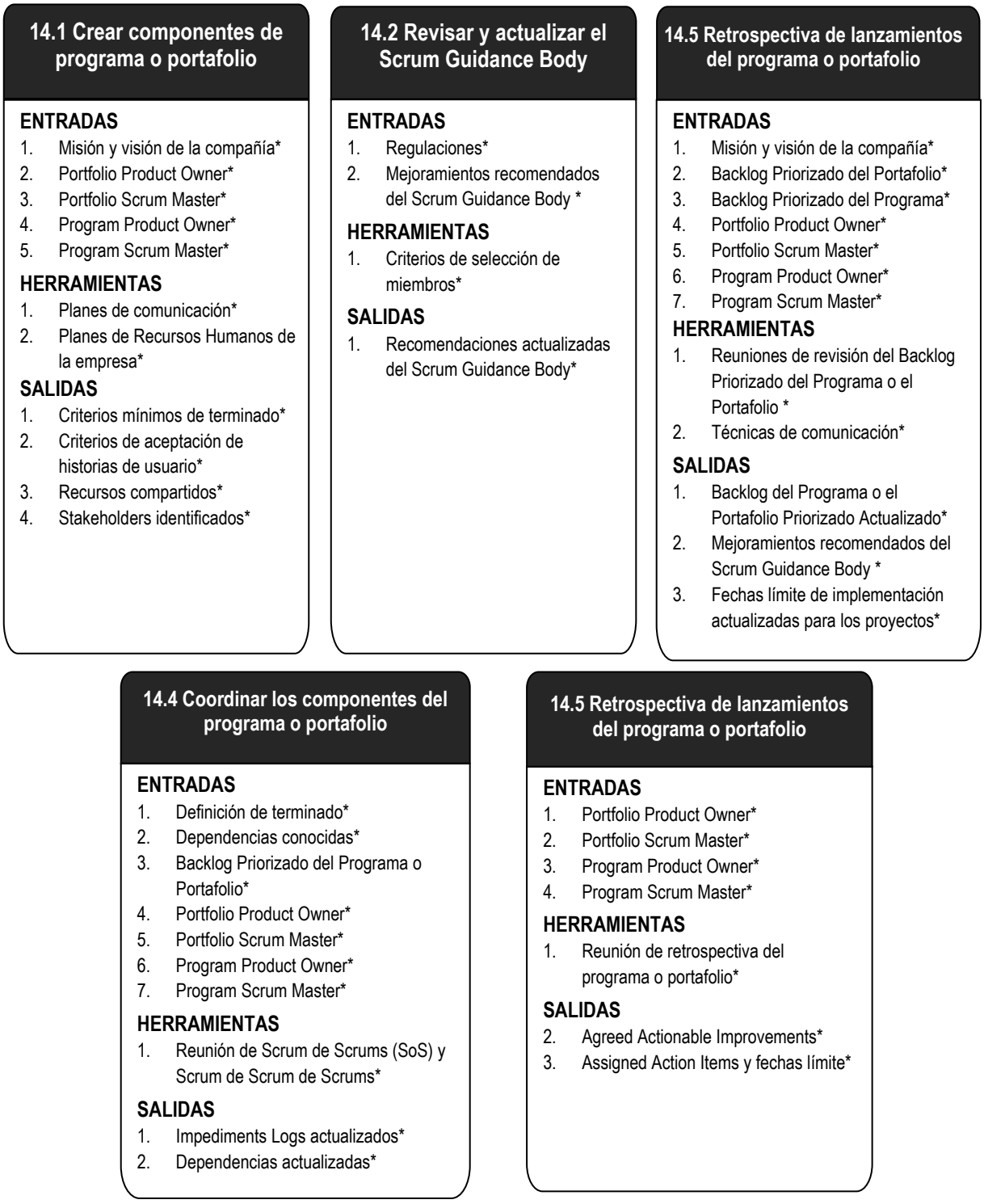


Figura 14.2: Descripción del Escalamiento de Scrum para la empresa

14.1 Crear componentes de programa o portafolio

La figura 14-3 muestra todas las entradas, herramientas y salidas del proceso de *Crear componentes de programa o portafolio.*

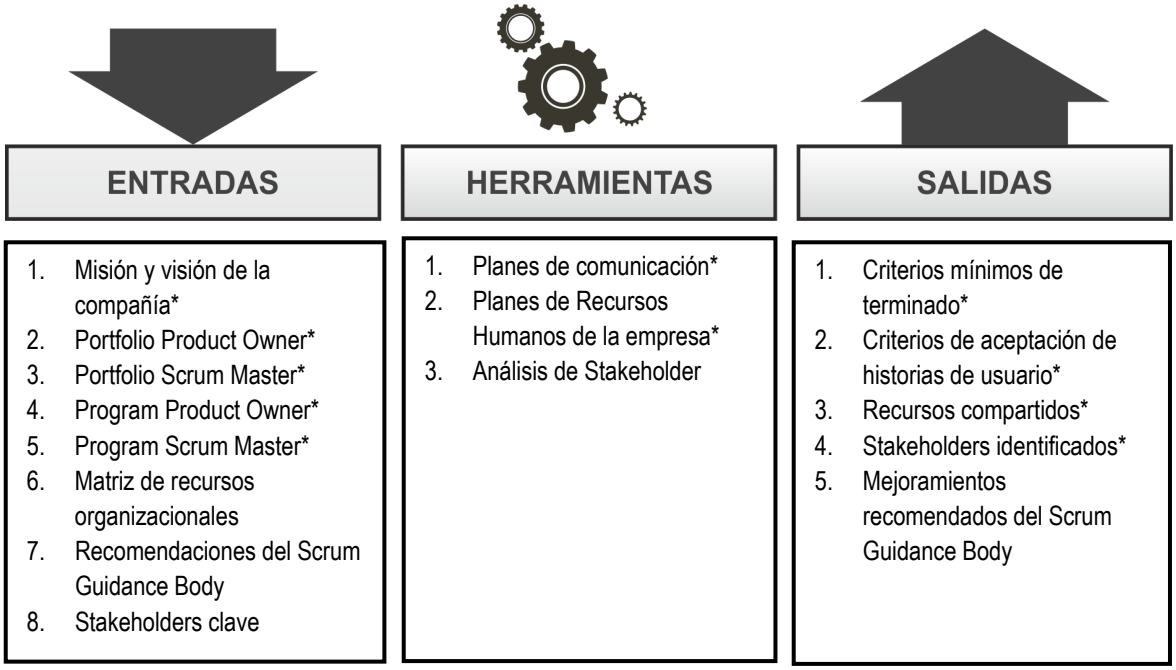


Figura 14-3: Crear componentes de programa o portafolio—Entradas, herramientas y salidas

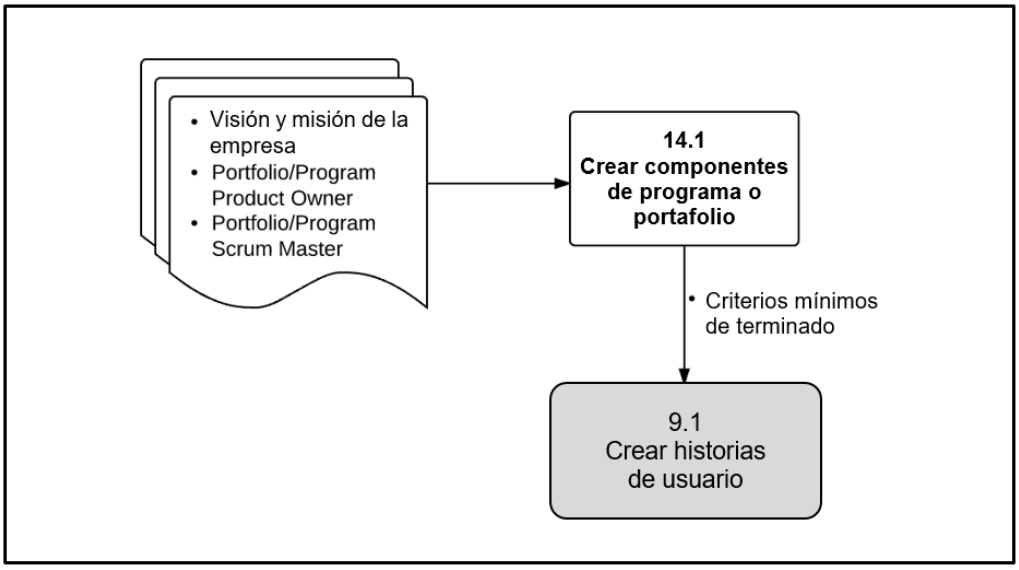


Figura 14-4: Crear componentes de programa o portafolio—Diagrama de flujo de datos

14.1.1 Entradas

14.1.1.1 Visión y misión de la empresa*

La visión y misión de la empresa se describen en las secciones 8.1.1.8 y 8.1.1.9, respectivamente.

Tanto la visión como la misión de la empresa son importantes para cualquier proyecto, pero más aún para los programas, especialmente al nivel del portafolio. Los programas y portafolios deben estar guiados por la misión y visión general de la empresa, ya que esto asegura la unidad del esfuerzo en toda la organización.

14.1.1.2 Portfolio Product Owner*

Descrito en la sección 3.4.4.

14.1.1.3 Portfolio Scrum Master*

Descrito en la sección 3.5.3.

14.1.1.4 Program Product Owner*

Descrito en la sección 3.4.3.

En un proyecto, el Program Product Owner es uno de los múltiples stakeholders. Al nivel del programa, el Program Product Owner juega un papel similar al del Product Owner en un proyecto. Guía y es responsable de la creación y refinamiento (*grooming*) de los componentes del programa.

14.1.1.5 Program Scrum Master*

Descrito en la sección 3.5.2.

En un proyecto, el Program Scrum Master es uno de los múltiples stakeholders. Al nivel del programa, el Program Scrum Master juega un papel similar al del Scrum Master en un proyecto. Es un facilitador, resuelve problemas y elimina impedimentos al nivel del programa.

14.1.1.6 Matriz de recursos organizacionales

Descrito en la sección 8.2.1.10.

14.1.1.7 Recomendaciones del Scrum Guidance Body

Descrito en las secciones 8.1.1.11 y 13.1.1.7.

Las recomendaciones del Scrum Guidance Body son de especial importancia en los niveles del programa o portafolio, ya que la orientación adecuada es necesaria para una importante cantidad de proyectos potenciales.

14.1.1.8 Stakeholders clave

Aunque en este proceso se identifican a muchos stakeholders, ya se sabrá quienes serán varios stakeholders clave. Por ejemplo, los stakeholders clave al nivel del portafolio son miembros del consejo ejecutivo de una empresa, organizaciones gubernamentales; y los stakeholders clave al nivel del programa son los patrocinadores del programa o de proyectos asociados, así como altos ejecutivos.

Descrito en las secciones 1.4.3.1, 3.3.2 y 6.4.2.1.

14.1.2 Herramientas

14.1.2.1 Plan(es) de comunicación*

Descrito en la sección 12.1.2.2.

Los planes de comunicación deben definir la forma de diseminar información en toda la organización y en todo el portafolio y los programas. También deben definir cómo y cuándo comunicar el tipo de información que será utilizado. El portafolio proporciona la guía para el plan de comunicación de los programas. De igual forma, el programa brinda orientación sobre los planes de comunicación de los proyectos.

14.1.2.2 Planes de Recursos Humanos de la empresa*

Los planes de Recursos Humanos de la empresa proporcionan amplia información sobre el personal que estará disponible en los varios proyectos, programas y portafolios. También proporcionan información sobre los planes de contratación del personal necesario para futuras actividades.

14.1.2.3 Análisis de Stakeholders

El análisis estándar de stakeholders se utiliza para identificar a los stakeholders al nivel del programa y portafolio. Los detalles adicionales relacionados a los stakeholders del programa o portafolio pueden ser identificados como prototipos (*personas*) en el proceso de *Crear y refinar el backlog del programa o portafolio*.

14.1.3 Salidas

14.1.3.1 Criterios mínimos de terminado*

Descrito en la sección 5.4.3.

14.1.3.2 Criterios de aceptación de historias de usuario*

Descrito en la sección 9.1.3.2.

14.1.3.3 Recursos compartidos*

Descrito en la sección 13.1.3.4.

14.1.3.4 Stakeholders identificados*

Descrito en la sección 8.2.3.2.

Los stakeholders clave en los niveles de programa o portafolio son una entrada en este proceso. Los stakeholders adicionales se identifican en este proceso.

14.1.3.5 Mejoramientos recomendados por el Scrum Guidance Body

Descrito en la sección 13.1.3.6.

Como resultado del proceso de *Crear componentes de programa o portafolio*, se pueden proporcionar sugerencias o retroalimentación para mejoramientos potenciales del Scrum Guidance Body (SGB). Tales mejoras recomendadas serían analizadas y aceptadas o rechazadas por dicho órgano (véase la sección 14.2: proceso de *Revisar y actualizar el Scrum Guidance Body*). Si el SGB acepta las sugerencias, estas se incorporan como actualizaciones a la documentación del Scrum Guidance Body.

14.2 Revisar y actualizar el Scrum Guidance Body

La figura 14-5 muestra todas las entradas, herramientas y salidas del proceso de *Revisar y actualizar el Scrum Guidance Body*.

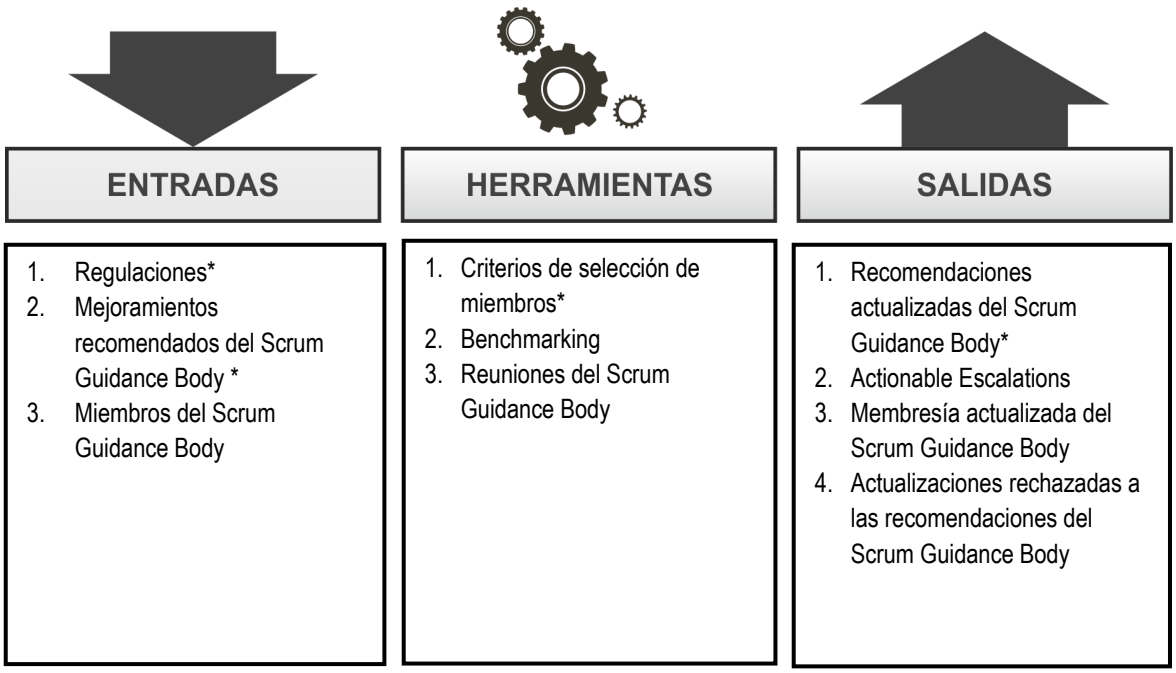


Figura 14-5: Revisar y actualizar el Scrum Guidance Body—Entradas, herramientas y salidas

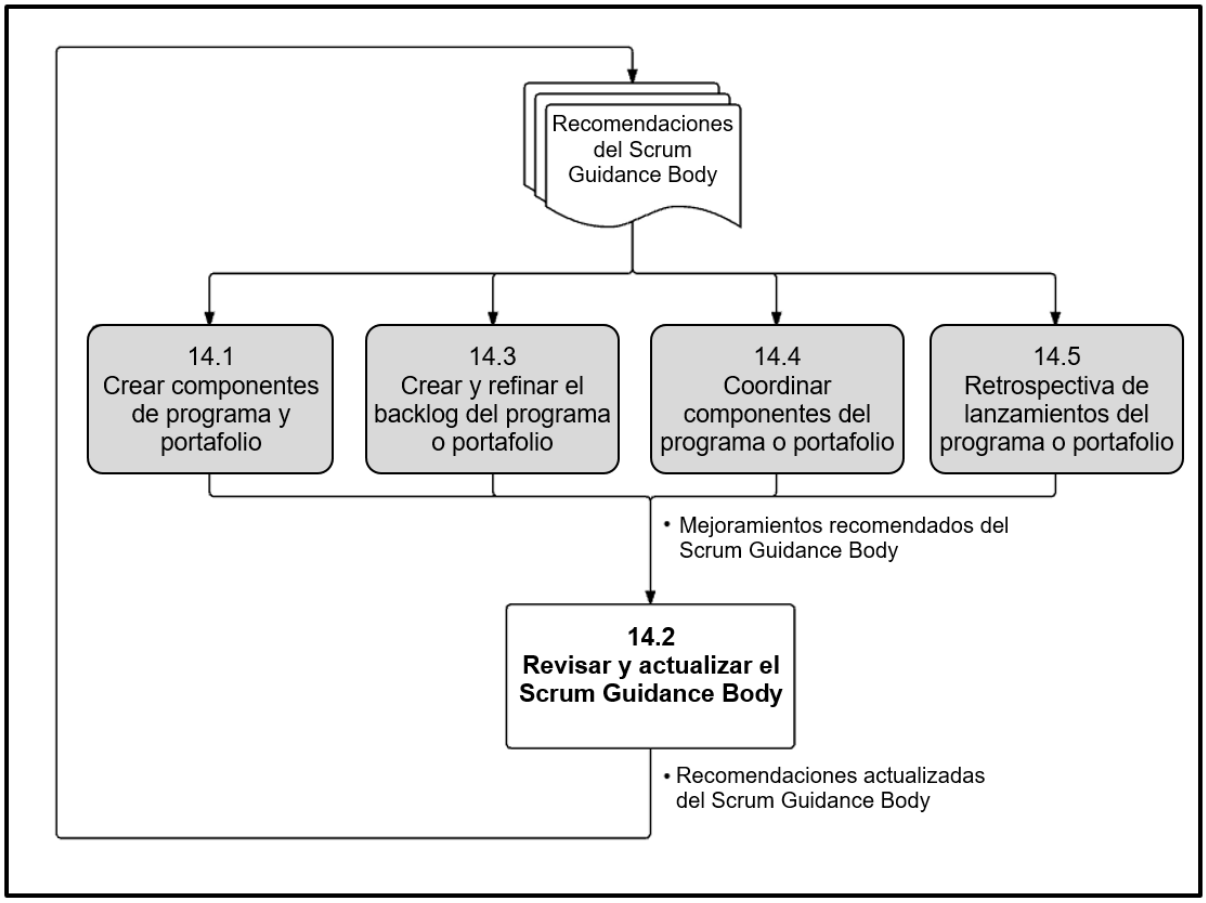


Figura 14-6: Revisar y actualizar el Scrum Guidance Body—Diagrama de flujo de datos

14.2.1 Entradas

14.2.1.1 Regulaciones*

Las regulaciones incluyen cualquier normativa federal, estatal o local a la que se deba acatar el programa o portafolio. Las historias de usuario creadas para cumplir con las regulaciones gubernamentales dentro de un periodo estipulado se incluyen en el backlog del portafolio o del programa.

En ocasiones tal vez sea necesario actualizar las recomendaciones del Scrum Guidance Body para reflejar nuevas regulaciones.

14.2.1.2 Mejoramientos recomendados del Scrum Guidance Body*

Descrito en la sección 13.1.3.6.

Como resultado de las retrospectivas de Scrum y otros procesos, se pueden hacer sugerencias y presentar retroalimentación para revisar o mejorar las recomendaciones del Scrum Guidance Body. Si este órgano aceptar tales sugerencias o la retroalimentación, estas se incorporan como actualizaciones a las recomendaciones del Scrum Guidance Body.

14.2.1.3 Miembros del Scrum Guidance Body

Los miembros del Scrum Guidance Body (SGB) pueden ser expertos de Scrum, Scrum Masters selectos, Product Owners y miembros del equipo (en todos los niveles). Sin embargo, debe haber un límite en la cantidad de miembros que puede tener el SGB a fin de asegurar que permanezca relevante y no tenga una naturaleza prescriptiva.

14.2.2 Herramientas

14.2.2.1 Criterios de selección de miembros*

Los stakeholders desarrollan los criterios de selección de miembros para definir quiénes serán los miembros del Scrum Guidance Body, cuáles serán sus respectivos roles y responsabilidades, la cantidad de integrantes, así como la experiencia y las habilidades necesarias.

Cada empresa puede tener sus propios criterios de selección de miembros del Scrum Guidance Body, sin embargo, se recomienda que cada integrante tenga experiencia en Scrum y que exista un límite en la cantidad de miembros que puede tener un SGB a fin de asegurar que permanezca relevante y no tenga una naturaleza prescriptiva.

14.2.2.2 Benchmarking

Una empresa debe referenciar (*benchmark*) con regularidad sus propias prácticas en comparación a otras industrias a fin de estar al día con la competencia. El *benchmarking* es el proceso de comprar los procesos empresariales de una organización y los parámetros de desempeño con aquellos en las empresas líderes en la misma industria o en otras.

14.2.2.3 Reuniones del Scrum Guidance Body

El Scrum Guidance Body se reúne con frecuencia para debatir sobre la posible necesidad de actualizar las recomendaciones de dicho órgano (por ejemplo: recomendaciones mejoradas de las retrospectivas y demás procesos, actualización de regulaciones, etc.). La frecuencia de estas reuniones la decide el Scrum Guidance Body con base en las necesidades específicas de la empresa.

14.2.3 Salidas

14.2.3.1 Recomendaciones actualizadas del Scrum Guidance Body*

Descrito en la sección 11.2.3.6.

Como resultado de la revisión del Scrum Guidance Body, tal vez resulte necesario actualizar las recomendaciones del Scrum Guidance Body.

14.2.3.2 Actionable Escalations

El Scrum Guidance Body puede determinar que algunas de las políticas de la empresa no permiten a los equipos obtener los máximos beneficios de la aplicación de Scrum. En tal caso, se debe accionar un escalamiento para obtener la autorización en el cambio de política.

14.2.3.3 Membresía actualizada del Scrum Guidance Body

Como resultado de una evaluación en la membresía del Scrum Guidance Body, se pueden incluir a nuevos miembros en dicho órgano y se pueden remover miembros existentes.

14.2.3.4 Recomendaciones rechazadas del Scrum Guidance Body

Los mejoramientos recomendados del Scrum Guidance Body no siempre son aceptados. Si la mejora recomendada es rechazada por los miembros del Scrum Guidance Body, se le proporciona a la parte interesada la retroalimentación respecto al motivo del rechazo.

14.3 Crear y refinar el backlog del programa o portafolio

La figura 14-7 muestra todas las entradas, herramientas y salidas del proceso de *Crear y refinar el backlog del programa o portafolio*.

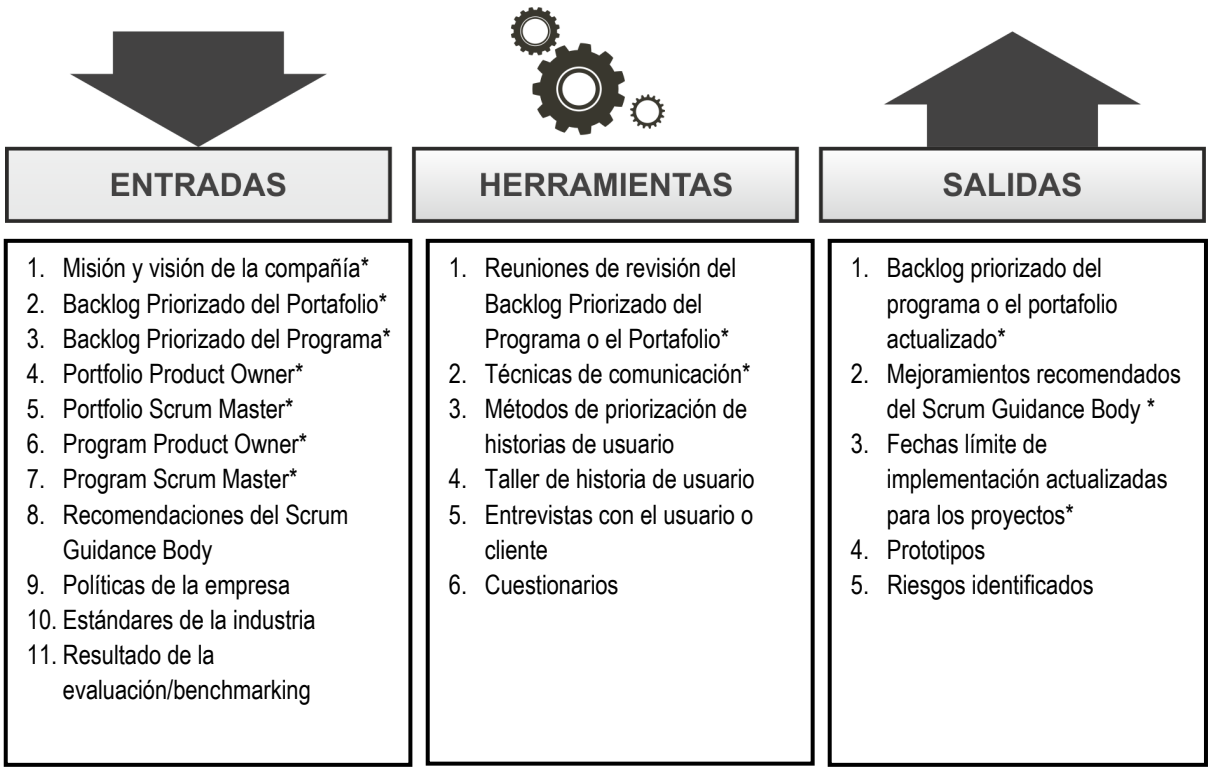


Figura 14-7: Crear y refinar el backlog del programa o portafolio—Entradas, herramientas y salidas

La figura 14-8 muestra toda la relación del proceso de *Crear y refinar el backlog del programa o portafolio* con los procesos fundamentales de Scrum.

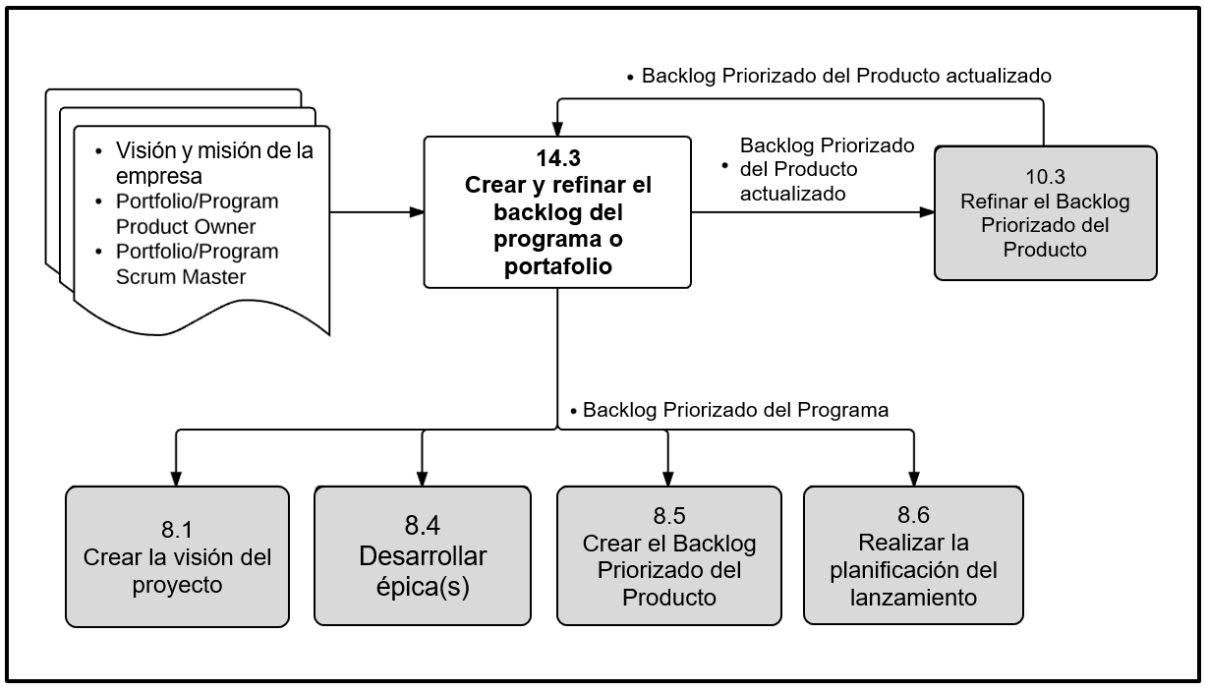


Figura 14-8: Crear y refinar el backlog del programa o portafolio—Diagrama de flujo de datos

14.3.1 Entradas

14.3.1.1 Visión y misión de la empresa

Descrito en las secciones 8.1.1.8 y 8.1.1.9.

14.3.1.2 Backlog Priorizado del Portafolio*

El Backlog Priorizado del Portafolio tiene la misma función al nivel del portafolio que el Backlog Priorizado del Programa al nivel del programa. Los elementos en el Backlog Priorizado del Producto proporcionan las entradas en los varios Backlogs Priorizados del Programa y —mediante los Backlogs Priorizados del Programa—, a los Backlogs Priorizados del Producto en proyectos individuales. Tal como lo que se describe para el Backlog Priorizado del Programa en la sección 8.1.1.5, de ser necesario, solo se hace un refinamiento mínimo en las historias de usuario en este nivel, ya que el refinamiento se hace en los proyectos en sus Backlogs Priorizados del Producto específicos.

14.3.1.3 Backlog Priorizado del Programa*

Descrito en la sección 8.1.1.5.

El Backlog Priorizado del Programa tiene la misma función al nivel del programa que el Backlog Priorizado del Producto al nivel del proyecto. Define los requerimientos del programa y sus prioridades.

Sin embargo, existen algunas diferencias:

La creación de los entregables respectivos y su aceptación se hace en los proyectos del programa. Los criterios de terminado o de aceptación de cada elemento en el Product Backlog o de cada historia de usuario tal vez tengan que se definidos en el nivel del programa. Los proyectos deben apegarse a dichos criterios, aunque pueden agregar sus propios criterios.

La duración de un sprint es distinta en cada proyecto y generalmente varía de proyecto a proyecto en un programa. Asimismo, la velocidad de los distintos equipos normalmente no es la misma. Por lo tanto, no es necesario tener historias de usuario muy granulares al nivel del programa. El refinamiento de historias de usuario al nivel del programa solo se extiende lo suficiente para garantizar que se entiendan las respectivas historias de usuario y se puedan definir los criterios de aceptación tangibles del programa.

14.3.1.4 Portfolio Product Owner*

Descrito en la sección 3.4.4.

El Portfolio Product Owner tiene una función similar al del Program Product Owner en un programa. Está a cargo y es responsable de llevar a cabo la creación y el refinamiento del Portfolio Product Backlog.

14.3.1.5 Portfolio Scrum Master*

Descrito en la sección 3.5.3.

En un proyecto, el Program Scrum Master es uno de varios stakeholders. Aquí, al nivel del portafolio, el Portfolio Scrum Master tiene una función similar a la del Program Scrum Master en un programa.

14.3.1.6 Program Product Owner*

Descrito en la sección 3.4.3.

El Program Product Owner es uno de varios stakeholders en un proyecto. Al nivel del programa, el Program Product Owner tiene una función similar al del Product Owner en un proyecto. Está a cargo y es responsable de la creación y refinamiento del Backlog Priorizado del Producto del Programa.

14.3.1.7 Program Scrum Master*

Descrito en la sección 3.5.2.

El Program Scrum Master es uno de varios stakeholders en un proyecto. Al nivel del programa, el Program Scrum Master tiene una función similar a la del Scrum Master en un proyecto. Es un facilitador, resuelve problemas y elimina impedimentos al nivel del programa.

14.3.1.8 Recomendaciones del Scrum Guidance Body

Descrito en las secciones 8.1.1.11 y 10.3.1.11.

Al momento de crear y refinar un backlog priorizado del programa o portafolio, las recomendaciones del Scrum Guidance Body presentan las mejores prácticas que deben tomarse en consideración al nivel del programa o portafolio.

14.3.1.9 Políticas de la empresa

Las políticas de la empresa son una serie de principios, reglas y directrices formuladas o adoptadas por una organización. Cambiar las políticas de una empresa afectaría las historias de usuario creadas, ya que fueron creadas en apego a las políticas existentes.

14.3.1.10 Estándares de la industria

Los nuevos estándares de la industria o los cambios a los estándares existentes deben ser implementados a fin de mantener un producto o servicio viable. Por lo tanto, las historias de usuario relacionadas deben ser incluidas en el Backlog Priorizado del Programa y/o del portafolio según corresponda.

En ocasiones, las recomendaciones del Scrum Guidance Body deben ser modificadas para que reflejen los estándares nuevos o modificados.

14.3.1.11 Resultados de la evaluación/benchmarking

Ante todo, los resultados de la evaluación/benchmarking necesitarán de una actualización de las mejores prácticas de las recomendaciones del Scrum Guidance Body. También pueden ayudar a establecer un estándar mínimo al crear un producto o servicio y llevar a criterios de terminado modificados. En ocasiones pueden también proporcionar el impulso para que un Program Product Owner o Portfolio Product Owner desarrolle nuevas historias de usuario para implementar las mejores prácticas.

14.3.2 Herramientas

14.3.2.1 Reuniones de revisión del Backlog Priorizado del Programa o el Portafolio*

La participación en las reuniones de revisión del backlog de programa o portafolio es muy diferente de la participación en las reuniones de revisiones al nivel del proyecto. Los equipos Scrum participarán en las sesiones de refinación (*grooming*) al nivel del proyecto. Al nivel del programa o portafolio, hay representación de cada proyecto en el programa o de cada programa en el portafolio. Para simplificar la reunión, generalmente se recomienda que, al nivel del programa o portafolio, solo haya un representante de cada proyecto o programa.

Consulte las secciones 10.3.2.1 y 6.5.1.2.

14.3.2.2 Técnicas de comunicación*

Descrito en la sección 10.3.2.2.

14.3.2.3 Métodos de priorización de historias de usuario

Descrito en la sección 8.5.2.1.

Al nivel del programa o portafolio, generalmente existe una menor cantidad de requerimientos/historias de usuario que al nivel del proyecto. El porcentaje de historias de usuario con un muy tangible impacto sobre el valor, en la necesidad de negocio o en los usuarios, normalmente es mucho menor que al nivel de proyecto. Esto significa que la selección de técnicas que son útiles al nivel de programa o portafolio serán más pequeñas.

Por ejemplo, el Análisis de Kano tiene limitaciones, ya que no habría "calidad requerida" (*dissatisfiers*) o "calidad atractiva" (*exciters*). Sin una cantidad considerable de stakeholders o usuarios especiales, el Método de los 100 puntos tendría valor limitado. La técnica MoSCoW también tiene sus limitaciones, ya que no habría ninguna característica de "No tendrá" (*Won't have*) en los niveles de programa y portafolio.

La comparación por pares (*Paired Comparison*) funciona bien en los niveles de programa y portafolio.

14.3.2.4 Taller de historias de usuario

Descrito en la sección 8.4.2.2.

En comparación a los proyectos, los talleres de historias de usuario para programas y portafolios producirán solo historias de usuario de más alto nivel como sus salidas y el número de historias también será menor. Sin embargo, las reuniones seguirán proporcionando valor, puesto que habrá la participación de representantes de distintos proyectos en un programa o de distintos programas en un portafolio. Esto garantiza que los requerimientos se entiendan y estén bien definidos.

14.3.2.5 Entrevistas con el usuario o cliente

Descrito en la sección 8.4.2.4.

14.3.2.6 Cuestionarios

Descrito en la sección 8.4.2.5.

14.3.3 Salidas

14.3.3.1 Backlog priorizado del programa o del portafolio actualizado*

El Backlog Priorizado del Programa o del portafolio puede ser actualizado con nuevas historias de usuario, con nuevas solicitudes de cambio, riesgos identificados, historias de usuario actualizadas o la re-priorización de las historias de usuario existentes.

Consulte la sección 10.3.3.1.

14.3.3.2 Mejoramientos recomendados del Scrum Guidance Body*

Descrito en la sección 13.1.3.6.

Como resultado del proceso de *Crear y refinar el backlog del programa o portafolio*, se pueden presentar sugerencias o retroalimentación para los mejoramientos potenciales del Scrum Guidance Body (véase la sección 14.2: proceso de *Revisar y actualizar el Scrum Guidance Body*). Si dicho órgano acepta tales sugerencias, estas se incorporan como actualizaciones a la documentación del Scrum Guidance Body.

14.3.3.3 Fechas límite de implementación actualizadas para los proyectos*

Las fechas límite de implementación en los proyectos pueden ser actualizadas para que se reflejen el impacto de historias de usuario nuevas o modificadas que deban cambiar o introducir requerimientos nuevos.

14.3.3.4 Prototipos (*Personas*)

Descrito en la sección 8.4.3.2.

14.3.3.5 Riesgos identificados

Descrito en la sección 8.4.3.4.

14.4 Coordinar componentes del programa o portafolio

La figura 14-9 muestra todas las entradas, herramientas y salidas del proceso *Coordinar componentes del programa o portafolio*.

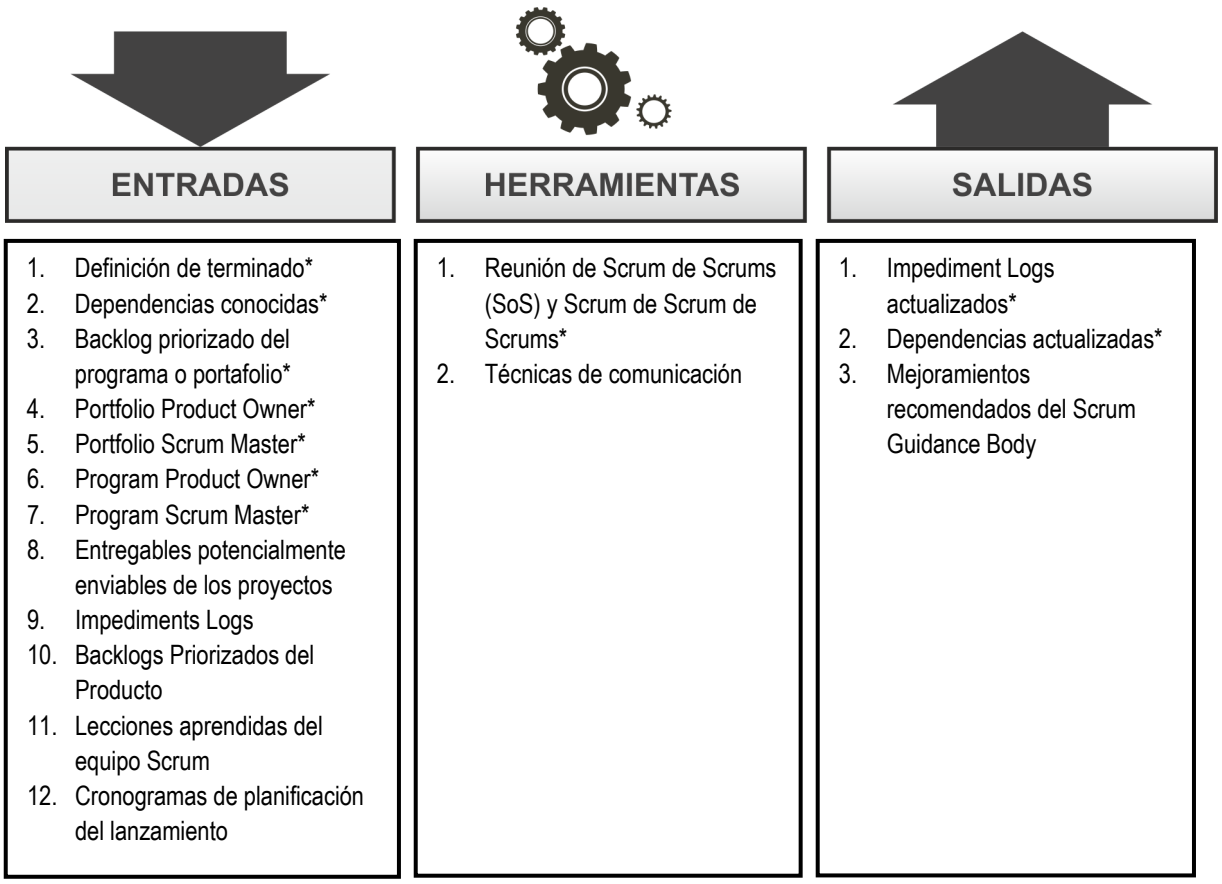


Figura 14-9: Coordinar componentes del programa o portafolio—Entradas, herramientas y salidas

La figura 14-10 muestra el flujo de datos del proceso de *Coordinar componentes del programa y portafolio*.

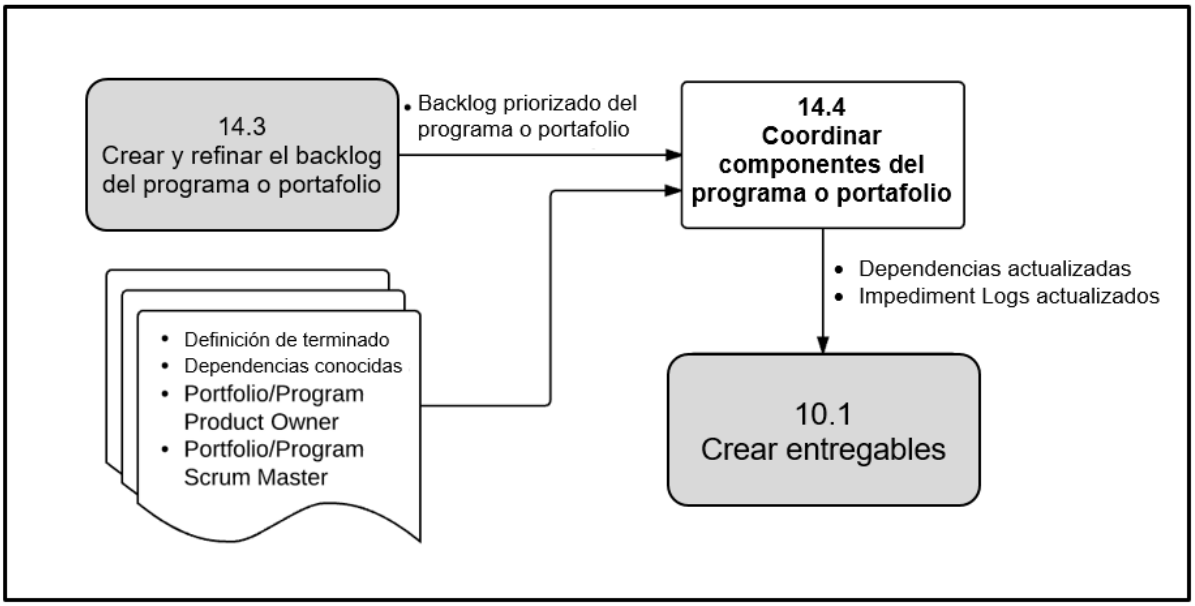


Figura 14-2: Coordinar componentes del programa o portafolio—Diagrama de flujo de datos

14.4.1 Entradas

14.4.1.1 Definición de terminado*

Descrito en la sección 5.4.3.

La definición de terminado que se define al nivel de programa o portafolio puede utilizarse como criterios mínimos de terminado para los proyectos en toda la empresa.

14.4.1.2 Dependencias conocidas*

En los casos de proyectos y/o productos inter-relacionados dentro de la empresa, pudiera haber algunas dependencias identificadas. Por lo tanto, debe haber una coordinación entre los proyectos para gestionar las dependencias. Dichas dependencias pudieran incluir:

- Fechas idénticas de lanzamiento para proyectos inter-relacionales.
- Dependencias entre lanzamientos
- Dependencias en características inter-relacionadas

Consulte las secciones 9.4.2.3 y 13.1.2.4.

14.4.1.3 Backlog priorizado del programa o portafolio*

Descrito en las secciones 14.3.1.2 y 14.3.1.3.

14.4.1.4 Portfolio Product Owner*

Descrito en las secciones 3.4.4. y 14.3.1.4.

14.4.1.5 Portfolio Scrum Master*

Descrito en las secciones 3.5.3 y 14.3.1.5.

14.4.1.6 Program Product Owner*

Descrito en las secciones 3.5.2 y 14.3.1.6.

14.4.1.7 Program Scrum Master*

Descrito en las secciones 3.5.2 y 14.3.1.7.

14.4.1.8 Entregables potencialmente enviables de los proyectos

Los entregables potencialmente enviables (*Potentially Shippable Deliverables*) de los proyectos son entradas valiosas para la coordinación al nivel del programa o portafolio. Al final de los sprints en proyectos, se completan los entregables o incrementos de producto. Las historias de usuario incluidas en dichos incrementos cumplen con los criterios en la definición de terminado, así como con sus respectivos criterios de aceptación.

Con cada sprint completado en cualquier de los proyectos asociados, se aclara la forma en la que avanzan los proyectos respectivos. Este conocimiento no solo permite las constantes proyecciones actualizadas sobre si todos los proyectos habrán de cumplir con las fechas límites requeridas, sino que brinda también información vital para abordar las dependencias entre proyectos.

14.4.1.9 Impediment Logs

Los impedimentos que enfrentan los proyectos individuales pueden ser relevantes en otros proyectos. Por lo tanto, los Impediment Logs (registros de impedimentos) deben compartirse entre proyectos y/o programas.

Descrito en la sección 8.5.3.1. Consulte la sección 13.2.1.2 sobre los relacionado a compartir los Impediment Logs entre los equipos en un proyecto grande.

14.4.1.10 Backlogs Priorizados del Producto

Descrito en la sección 8.5.3.1.

Estos serían los Backlogs Priorizados del Producto al nivel del proyecto.

14.4.1.11 Lecciones aprendidas de los equipos Scrum

Descrito en la sección 11.2.3.5.

14.4.1.12 Cronogramas de planificación de lanzamiento

Descrito en la sección 8.6.3.1.

Estos cronogramas, aunque son tentativos y están sujetos al cambio, son de suma importante para evaluar si habrá posibilidades de que los respectivos proyectos cumplan con los plazos requeridos y son de especial importancia respecto a las dependencias.

14.4.2 Herramientas

14.4.2.1 Reunión de Scrum de Scrums (SoS)

Descrito en la sección 13.2.2.1.

Esta reunión se utiliza en forma similar a su uso en grandes proyectos. Al nivel del programa, los representantes de todos los proyectos en el programa se reúnen en intervalos frecuentes en las reuniones de Scrum de Scrums (SoS).

14.4.2.2 Reunión de Scrum de Scrum de Scums (SoSoS)

Al nivel del portafolio, los representantes de todos los programas independientes (*stand-alone projects*) en el portafolio se reúnen en intervalos frecuentes. También se puede decir que se reúnen los representantes de cada Scrum de Scrum. Por lo tanto, este nivel adicional de reuniones se denomina Scrum de Scrum de Scrums (SoSoS). La figura 14-11 ilustra el concepto de Scrum de Scrums (SoS) y las reuniones de Scrum de Scrum de Scrums.

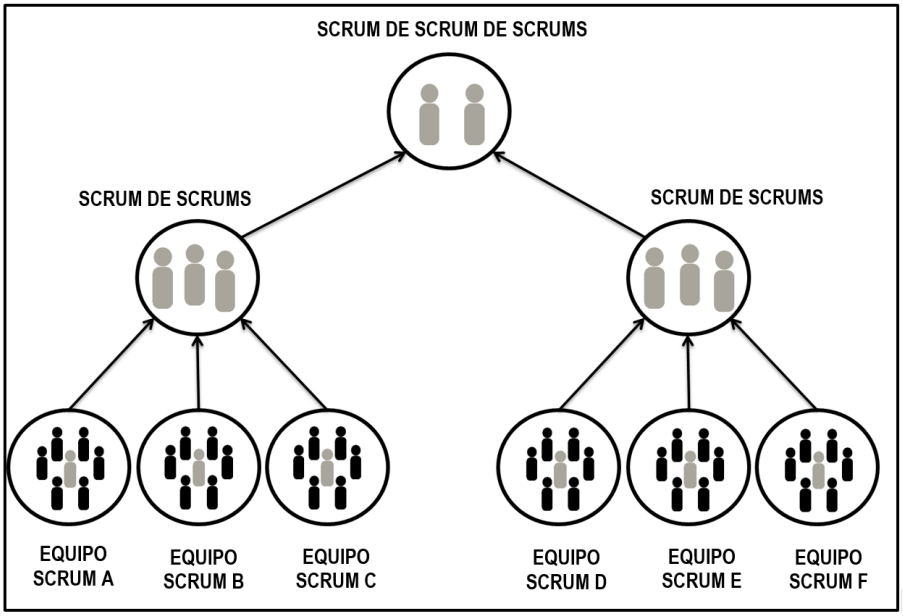


Figura 14-11: Reunión de Scrum de Scrums (SoS)

En este ejemplo, hay seis equipos Scrum que trabajan simultáneamente. Los equipos A, B y C están trabajando en las partes de un proyecto relacionado, mientras que los equipos D, E y F están trabajando en porciones de otro proyecto relacionado. Una reunión de Scrum de Scrum se lleva a cabo para coordinar las interdependencias entre los proyectos relacionados. Una reunión de Scrum de Scrum de Scrums se llevará a cabo para coordinar y gestionar las dependencias en todos los proyectos.

14.4.2.3 Técnicas de comunicación

Descrito en la sección 10.3.2.2.

Algunas de las técnicas de comunicación que se pueden utilizar al nivel del programa o portafolio son los tableros de mensajes (*message boards*) y la mensajería instantánea (*instant messaging*).

14.4.3 Salidas

14.4.3.1 Impediment Logs actualizados*

Como resultado del Scrum de Scrums (SoS) o de la reunión de Scrum de Scrum de Scrums (SoSoS), tal vez sea necesario actualizar los Impediment Logs.

Consulte la sección 10.1.1.4.

14.4.3.2 Dependencias actualizadas*

Como resultado de la coordinación de componentes del programa o portafolio, tal vez sea necesario actualizar las dependencias conocidas con dependencias nuevas o cambiar las ya existentes.

Consulte las secciones 9.4.3.3 y 13.1.3.9.

14.4.3.3 Mejoramientos recomendados del Scrum Guidance Body

Descrito en la sección 13.1.3.6.

Como resultado del proceso de *Coordinar componentes del programa o portafolio*, se pueden presentar sugerencias o retroalimentación para mejoramientos potenciales del Scrum Guidance Body. Estos mejoramientos recomendados serán analizados y aceptados o rechazados por dicho órgano (véase la sección 14.2: proceso de *Revisar y actualizar el Scrum Guidance Body*). Si Guidance Body acepta dichas sugerencias, estas se incorporan como actualizaciones a la documentación del Scrum Guidance Body.

14.5 Retrospectiva de los lanzamientos del programa o portafolio

La figura 14-12 muestra todas las entradas, herramientas y salidas del proceso de *Retrospectiva de los lanzamientos del programa o portafolio*.

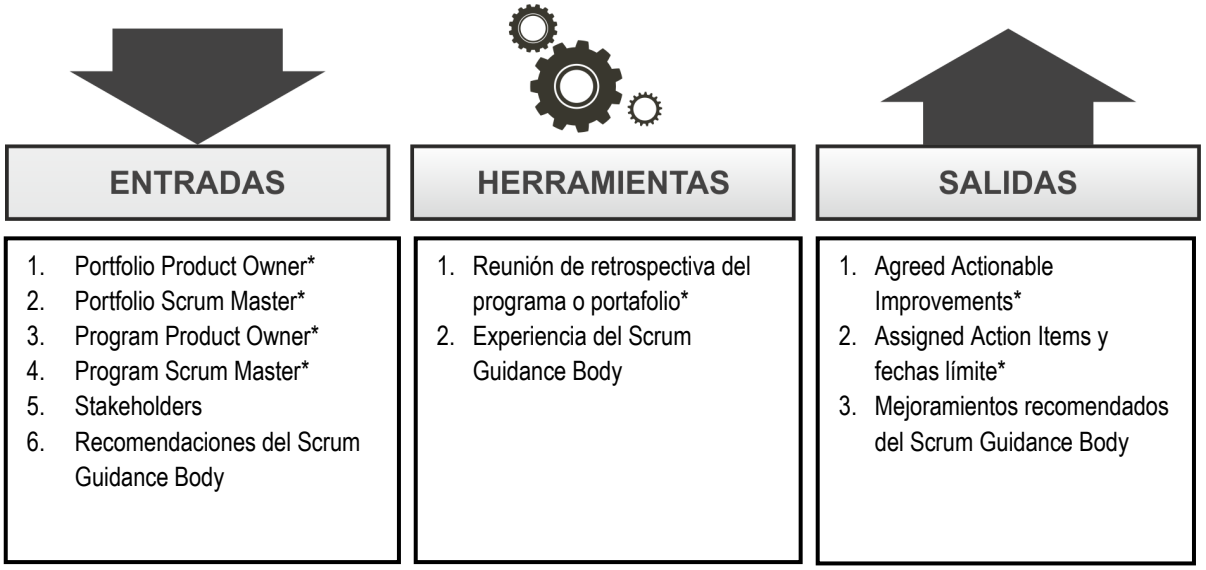


Figura 14-12: Retrospectiva de los lanzamientos del programa o portafolio—Entradas, herramientas y salidas

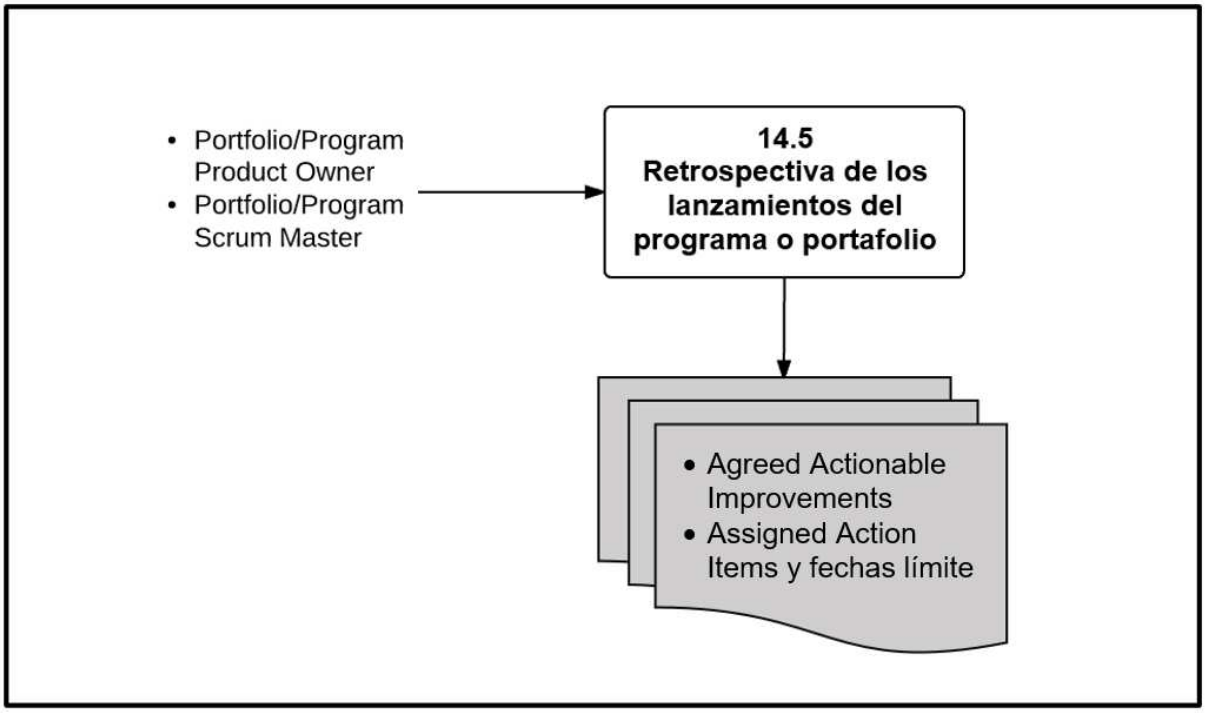


Figura 14-13: Retrospectiva de los lanzamientos del programa o portafolio—Diagrama de flujo de datos

14.5.1 Entradas

14.5.1.1 Portfolio Product Owner*

Descrito en las secciones 3.4.4 y 14.3.1.4.

14.5.1.2 Portfolio Scrum Master*

Descrito en las secciones 3.5.3 y 14.3.1.5.

14.5.1.3 Program Product Owner*

Descrito en las secciones 3.5.2 y 14.3.1.6.

14.5.1.4 Program Scrum Master*

Descrito en las secciones 3.5.2 y 14.3.1.7.

14.5.1.5 Stakeholders

Descrito en la sección 3.3.2.

14.5.1.6 Recomendaciones del Scrum Guidance Body

Descrito en las secciones 8.1.1.11 y 12.2.1.5.

Durante una retrospectiva de los lanzamientos del programa o portafolio, las recomendaciones del Scrum Guidance Body proporcionarán las mejores prácticas relacionadas a los temas pertinentes tales como procedimientos administrativos, auditorías, evaluaciones y criterios de transición del proyecto, de igual forma a las recomendaciones proporcionadas por el Scrum Guidance Body en las retrospectivas al nivel del proyecto (descrito en la sección 12.2.1.5).

14.5.2 Herramientas

14.5.2.1 Reunión de retrospectiva del programa o portafolio

La reunión de retrospectiva del programa o portafolio es similar a la reunión de retrospectiva del proyecto, descrito en la sección 12.2.2.1, pero se lleva a cabo al nivel del programa o portafolio. La principal diferencia es que la frecuencia de las reuniones de retrospectiva del programa o portafolio es menor que las reuniones de retrospectiva del proyecto.

14.5.2.2 Experiencia del Scrum Guidance Body

Descrito en la sección 8.4.2.7.

14.5.3 Salidas

14.5.3.1 Agreed Actionable Improvements*

Descrito en la sección 11.2.3.1.

14.5.3.2 Assigned Action Items y fechas límite*

Descrito en la sección 11.2.3.2.

14.5.3.3 Mejoramientos recomendados del Scrum Guidance Body

Descrito en la sección 13.1.3.6.

Como resultado del proceso de *Retrospectiva de los lanzamientos del programa o portafolio*, se pueden presentar sugerencias o retroalimentación para mejoramientos potenciales del Scrum Guidance Body. Estos mejoramientos recomendados serán analizados y aceptados o rechazados por dicho órgano (véase la sección 14.2: proceso de *Revisar y actualizar el Scrum Guidance Body*). Si Guidance Body acepta dichas sugerencias, estas se incorporan como actualizaciones a la documentación del Scrum Guidance Body.

APÉNDICE A. DESCRIPCIÓN GENERAL DE ÁGIL

A.1 Introducción

El propósito de este apéndice es familiarizar al lector con el concepto de desarrollo ágil y las diversas metodologías ágiles.

En las siguientes secciones se incluyen:

A.2 Descripción general—Esta sección analiza la definición y los factores detrás del surgimiento de Ágil.

A.3 Manifiesto Ágil—En esta sección se presenta el *Manifiesto Ágil*, sus principios, y la *Declaración de Interdependencia* para proporcionar el contexto histórico de Ágil.

A.4 Métodos Ágil—Esta sección ofrece una breve descripción de las metodologías ágiles específicas tales como:

- **Lean Kanban**
- **Extreme Programming (XP)** – *Programación Extrema*
- **Crystal Methods** – *Métodos Crystal*
- **Dynamic Systems Development Methods (DSDM)** – *Métodos de desarrollo de sistemas dinámicos*
- **Feature Driven Development (FDD)** – *Desarrollo basado en funcionalidades*
- **Test Driven Development (TDD)** – *Desarrollo guiado por pruebas*
- **Adaptive Software Development (ASD)** – *Desarrollo adaptativo de software*
- **Agile Unified Process (AUP)** – *Proceso Unificado Ágil*
- **Domain Driven Design (DDD)** – *Diseño guiado por el dominio*

A.2 Descripción general

La palabra “ágil” generalmente hace referencia a la capacidad de moverse o responder con rapidez y facilidad; ser ágil. En cualquier tipo de disciplina de administración, la agilidad es una calidad, y, por lo tanto, es algo bueno que se debe buscar. Específicamente, la gestión ágil de proyectos implica ser adaptativo durante la creación de un producto, servicio u otro resultado.

Es importante entender que, aunque el desarrollo de métodos ágiles es altamente adaptativo, también es necesario considerar la estabilidad en su proceso de adaptación.

A.2.1 El surgimiento de Ágil

Los rápidos cambios en la tecnología, las demandas y expectativas del mercado han creado mayores desafíos en el desarrollo de productos y servicios mediante el uso de modelos tradicionales de gestión de proyectos. Esto abrió el camino para la conceptualización e implementación de métodos y valores ágiles en muchas organizaciones. Los modelos de desarrollo ágil atienden las deficiencias asociadas con los modelos tradicionales de gestión de proyectos para satisfacer las crecientes demandas ambientales y expectativas que enfrentan las organizaciones. Dado a que los modelos tradicionales de gestión de proyectos en general hacen énfasis en una amplia planificación anticipada y se ajustan al plan una vez que se establece, tales modelos no tuvieron éxito al intentar cumplir la realidad de los frecuentes cambios ambientales.

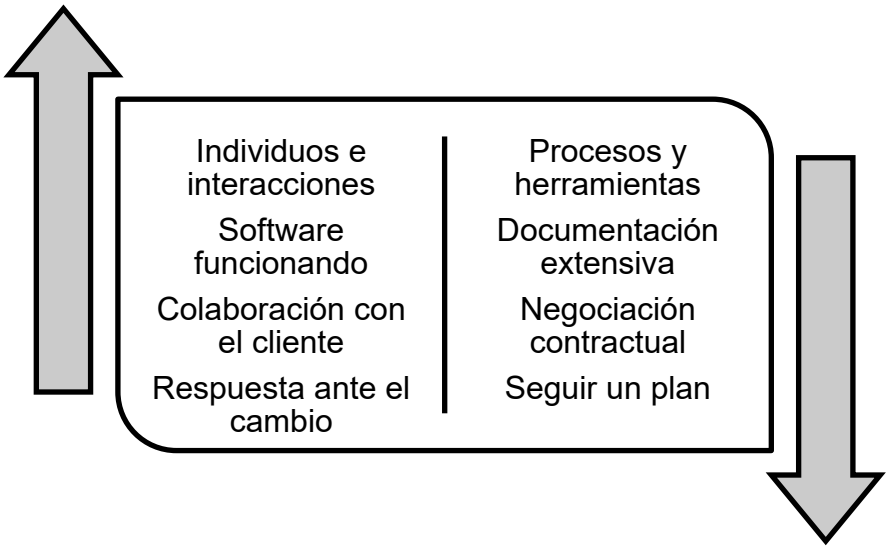
Ágil depende de la planificación adaptiva y del desarrollo y la entrega iterativa. Se enfoca principalmente en el valor de las personas al hacer eficazmente el trabajo. Aunque las metodologías adaptativas e incrementales han existido desde los años cincuenta, únicamente las metodologías que conforman el *Manifiesto Ágil* generalmente se consideran verdaderamente “ágiles”.

A.3 El Manifiesto Ágil

En febrero del 2001, un grupo de 17 gurús de la informática, desarrolladores de software y administradores, se reunieron para discutir los métodos de desarrollo de software ligero. Formaron la Alianza Ágil (*Agile Alliance*) y lo debates de estas reuniones resultaron después en un *Manifiesto para el desarrollo ágil de software* (*Manifesto for Agile Software Development*). El manifiesto fue escrito por Flowler y Highsmith (2001) y suscrito después por todos los participantes a fin de establecer los lineamientos básicos de cualquier método ágil.

El propósito del Manifiesto Ágil se plasmó de la siguiente forma:

Estamos descubriendo formas mejores de desarrollar
software tanto por nuestra propia experiencia como ayudando a terceros.
A través de este trabajo hemos aprendido a valorar:



Esto es, aunque valoramos los elementos de la derecha,
valoramos más los de la izquierda.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

El permiso para su reproducción fue proporcionado por los autores mencionados mediante aviso en:
http://agilemanifesto.org/

Las cuatro contrapartes del Manifiesto Ágil se elaboran de la siguiente forma:

1. **Individuos e interacciones sobre procesos y herramientas**
   Aunque los procesos y las herramientas ayudan a terminar con éxito un proyecto, son las personas quienes asumen, participan e implementan un proyecto y determinan cuáles procesos y herramientas utilizar. Por lo tanto, los actores clave en cualquier proyecto son las personas, por ello, el énfasis debe estar en ellos y en sus interacciones en vez de los complicados procesos y herramientas.

2. **Software funcionando sobre documentación extensiva**
   Aunque la documentación es necesaria y útil para cualquier proyecto, muchos equipos se centran en la recopilación y el registro de descripciones cualitativas y cuantitativas de los entregables, cuando el valor real que se le entrega al cliente es en forma de un software funcional. Por lo tanto, en vez de la documentación detallada, el enfoque ágil está en la entrega de un software de buen funcionamiento en incrementos a lo largo del ciclo de vida del producto.

3. **Colaboración con el cliente sobre negociación contractual**
   Tradicionalmente a los clientes se les ha visto como participantes externos, involucrados principalmente al inicio y al final del ciclo de vida del producto y cuya relación se basaba en contractos y en su cumplimiento. Ágil cree en un enfoque de valor compartido en el cual los clientes se consideran colaboradores. El equipo de desarrollo y el cliente trabajan unidos para evolucionar y desarrollar el producto.

4. **Responder ante el cambio sobre seguir un plan**
   En el mercado actual —donde los requerimientos del cliente, las tecnologías disponibles y los patrones empresariales cambian constantemente—, es fundamental abordar el desarrollo de productos en una forma adaptativa que permita la incorporación de cambio y rápidos ciclos de vida de desarrollo de producto en vez de enfatizar el seguimiento de planes formados probablemente con información obsoleta.

A.3.1 Principios de Manifiesto Ágil

Los 12 principios del Manifiesto Ágil de Fowler y Highsmith (2001) son los siguientes:

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software valioso.

2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

3. Entregamos software funcional frecuentemente, entre dos semanas y dos meses, de preferencia al menor tiempo posible.

4. Los responsables de negocio y los desarrolladores deben trabajar juntos de forma cotidiana durante todo el proyecto.

5. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.

6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.

7. El software funcionando es la medida principal de progreso.

8. Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deben ser capaces de mantener un ritmo constante de forma indefinida.

9. La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.

10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

12. A intervalos regulares el equipo reflexiona sobre cómo ser más eficaz para después ajustar y perfeccionar su comportamiento según corresponda.

A.3.2 Declaración de interdependencia

La *Declaración de Interdependencia* de gestión de proyectos ágiles fue redactada a principios del 2005 por un grupo de 15 líderes de proyecto como suplemento del Manifiesto Ágil. Enumera seis valores de gestión necesarios para reforzar la mentalidad del desarrollo ágil, particularmente en la gestión de proyectos complejos o inciertos.

La declaración destaca que los equipos del proyecto, los clientes y demás stakeholders son interdependientes y están conectados, lo cual se debe reconocer a fin de lograr el éxito. Los valores por si mismos son también interdependientes.

Nosotros...

Aumentamos el retorno sobre la inversión *enfocándonos en la aportación continua de valor.*

Entregamos resultados fiables *mediante la participación de clientes en las interacciones frecuentes, así como la propiedad compartida.*

Esperamos la incertidumbre *y la gestionamos a través de iteraciones, con anticipación y adaptación.*

Damos rienda suelta a la creatividad y la innovación *reconociendo que las personas son la fuente última de valor, y creando un ambiente donde pueden ser la diferencia.*

Aumentamos el rendimiento *mediante la orientación del grupo a los resultados y la responsabilidad compartida para la efectividad del equipo.*

Mejoramos la eficacia y fiabilidad *a través de estrategias, procesos y prácticas específicas.*

Anderson. D., Augustine, S., Avery, C., Cockburn, A., Cohn, M., *et al.* 2005

A.4 Métodos Ágiles

En la década de los noventa y a principios del 2000, se originó y obtuvo fuerza una serie de metodologías ágiles. Aunque difieren en una variedad de aspectos, lo que tienen en común se deriva de su apego al *Manifiesto Ágil*.

A continuación, se describen brevemente los siguientes métodos ágiles:

- **Lean Kanban**
- **Extreme Programming (XP)** – *Programación Extrema*
- **Crystal Methods** – *Métodos Crystal*
- **Dynamic Systems Development Methods (DSDM)** – *Métodos de desarrollo de sistemas dinámicos*
- **Feature Driven Development (FDD)** – *Desarrollo basado en funcionalidades*
- **Test Driven Development (TDD)** – *Desarrollo guiado por pruebas*
- **Adaptive Software Development (ASD)** – *Desarrollo adaptativo de software*
- **Agile Unified Process (AUP)** – *Proceso Unificado Ágil*
- **Domain Driven Development (DDD)** – *Desarrollo guiado por el dominio*

A.4.1 Lean Kanban

El concepto de Lean optimiza el sistema de una organización a fin de producir resultados valiosos con base en sus recursos, necesidades y alternativas, a la vez que se reducen las perdidas. Las pérdidas (*waste*) pudieran ser por la fabricación de algo equivocado, la imposibilidad de aprender, o prácticas que impidan el proceso. Debido a que estos factores tienen una naturaleza dinámica, una organización Lean evalúa la totalidad de su sistema y refina constantemente sus procesos. El fundamento de Lean es la reducción de la duración de cada ciclo (cada interacción), lo cual lleva a un aumento en la productividad, reduciendo retrasos y ayudando a detectar errores en las primeras etapas, disminuyendo en consecuencia la cantidad total del trabajo necesario para finalizar una tarea. Los principios del software Lean se han implementado con éxito en el desarrollo de software.

Kanban literalmente significa “cartel” o “letrero”, e implica el uso de ayuda visual para dar seguimiento a la producción. El concepto fue introducido por Taiichi Ohno, considerado como el padre de los Sistemas de Producción Toyota (TPS, por sus siglas en inglés). El uso de ayuda visual es eficaz y se ha convertido en una práctica común. Algunos ejemplos incluyen: tarjetas de tarea, Scrumboards y Burndown Charts. Dichos métodos generaron atención debido a su práctica en Toyota, empresa líder en gestión de procesos. Lean Kanban integra el uso de métodos de visualización según lo prescrito por Kanban aunado a los principios de Lean, creando así un sistema visual de gestión de proceso evolutivo incremental.

A.4.2 Extreme Programming *(Programación Extrema)*

El Extreme Programming (XP), o *Programación Extrema*, creada en la Chrysler Corporation, obtuvo impulso en la década de 1990. La Programación Extrema, o XP, por sus siglas en inglés, evita el aumento radical del costo de software cambiante con el paso del tiempo. Las características claves del XP incluyen el desarrollo incremental, horarios flexibles, códigos de prueba automatizados, comunicación verbal, diseño en evolución constante y la vinculación a corto y largo plazo de todos los involucrados.

La Programación Extrema valora la comunicación, la retroalimentación, la simplicidad y el valor. Los distintos roles en el enfoque de XP incluyen: el cliente, el desarrollador, el tracker y el coach. Prescribe varias prácticas de codificación, de desarrollo y de negocio, así como eventos y artefactos para lograr un desarrollo eficaz y eficiente. La Programación Extrema ha sido adoptada extensamente debido a sus prácticas de ingeniería bien definidas.

A.4.3 Crystal Methods *(Métodos Crystal)*

Las metodologías Crystal para el desarrollo de software fueron introducidas por Alistair Cockburn a principios de la década de 1990. La intención de los métodos Crystal es centrarse en las personas; ser ligeros y fáciles de adaptar. Debido a que las personas son primordiales, el proceso de desarrollo y las herramientas no son fijas, sino que se ajustan a los requerimientos y características específicas del proyecto. Se utiliza el espectro de colores para decidir sobre la variante de un proyecto. Los factores tales como la comodidad, el dinero a discreción, dinero esencial y la vida, juegan un papel importante para determinar el "peso" de la metodología, lo cual se representa en varios colores del espectro. La familia Crystal se divide en: *Crystal Clear* (claro como el cristal), *Crystal Yellow* (cristal amarillo), *Crystal Orange* (cristal naranja), *Crystal Orange Web* (cristal naranja web), *Crystal Red* (cristal rojo), *Crystal Maroon* (cristal marrón), *Crystal Diamond* (cristal diamante) y *Crystal Sapphire* (cristal zafiro).

Todos los métodos Crystal tienen cuatro roles: patrocinador ejecutivo (*executive sponsor*), diseñador líder (*lead designer*), desarrolladores y usuarios experimentados. Los métodos Crystal recomiendan varias estrategias y técnicas para lograr agilidad. Un ciclo de proyecto Crystal consiste de la implementación del acta constitutiva (*chartering*), ciclo de entrega y cierre (*wrap-up*).

A.4.4 Dynamic Systems Development Methods, DSMS
(Métodos de desarrollo de sistemas dinámicos)

El framework del sistema de desarrollo de sistemas dinámicos (DSMS, por sus siglas en inglés) fue publicado inicialmente en 1995 y lo administra el Consorcio DSMS. El DSMS fija la calidad y el esfuerzo en términos de costo y tiempo al principio y ajusta los entregables del proyecto para cumplir con los criterios fijos mediante la priorización de los entregables en las categorías: "Debe tener" (*Must have*), "Debería tener" (*Should have*), "Podría tener" (*Could have*) y "No tendrá" (*Won't have*), con el uso de la técnica de

priorización MoSCoW. El DSMS es un método orientado en sistemas con seis fases distintas: pre-proyecto; viabilidad; fundamentos; exploración e ingeniería; desplazamiento y evaluación de beneficios.

A.4.5 Feature Driven Development, FDD *(Desarrollo basado en funcionalidades)*

El desarrollo basado en funcionalidades (FDD, por sus siglas en inglés) fue introducido por Jeff De Luca en 1997 y opera bajo el principio de concluir un proyecto mediante su fragmentación en pequeñas funciones valoradas por el cliente que puedan presentarse en menos de dos semanas. El FDD tiene dos principios fundamentales: el desarrollo de software es una actividad humana y es una funcionalidad valorada por el cliente.

El FDD define seis roles principales: Project Manager, Chief Architect, Developer Manager, Chief Programmers, Class Owners y Domain Experts, aunados a una serie de funciones complementarias. El proceso FDD es iterativo y consiste en el desarrollo de un modelo general; en crear una lista de características y después planificar, diseñar y crear con base en la característica.

A.4.6 Test Driven Development, TDD *(Desarrollo guiado por pruebas)*

Conocido también como “desarrollo primero por pruebas” (del inglés: *Test-First Development*), el desarrollo guiado por pruebas fue introducido por Kent Beck, uno de los creadores de la Programación Extrema. El desarrollo guiado por pruebas es un método de desarrollo de software que implica redactar primero códigos de prueba automáticos y desarrollar la cantidad mínima de código necesario para avanzar después hacia la siguiente prueba. Todo el proyecto se divide en pequeñas características valoradas por el cliente que deben desarrollarse en el ciclo de desarrollo más breve posible. Las pruebas se redactan con base en los requerimientos y especificaciones del cliente. Las pruebas diseñadas en la etapa anterior se utilizan para diseñar y redactar el código de producción.

El desarrollo guiado por pruebas (TDD, por sus siglas en inglés) se puede clasificar en dos niveles: *Acceptance TDD* (ATDD), lo cual requiere de una prueba distinta de aceptación, y *Developer TDD* (DTDD) que implica la redacción de una sola prueba de desarrollo. El TDD se ha popularizado debido a las numerosas ventajas que ofrece como los resultados confiables, la retroalimentación constante y la reducción de tiempo de depuración (*debugging*).

A.4.7 Adaptive Software Development, ASD *(Desarrollo adaptativo de software)*

El desarrollo adaptativo de software (ASD, por sus siglas en inglés) surgió a partir del rápido trabajo de desarrollo de aplicaciones por parte de Jim Highsmith y Sam Bayer. Los aspectos más destacados del ASD son la constante adaptación de procesos al trabajo con el que se cuenta, el suministro de soluciones a los problemas que surgen en los grandes proyectos, así como el desarrollo iterativo e incremental con prototipos continuos.

Al ser un método de desarrollo impulsado por el riesgo y tolerante al cambio, el ASD indica que un plan no puede aceptar incertidumbres y riesgos, ya que esto sería indicativo de un plan deficiente y fallido. El desarrollo adaptativo de software se basa en características y se guía por metas. La primera fase en este tipo de desarrollo es la especulación (a diferencia de la planificación), seguida de las fases de colaboración y aprendizaje.

A.4.8 Agile Unified Process, AUP *(Proceso Unificado Ágil)*

El Proceso Unificado Ágil (AUP, por sus siglas en inglés) evolucionó del Proceso Unificado Racional de IBM (del inglés: *IBM's Rational Unified Process*). El Proceso Unificado Ágil, desarrollado por Scott Ambler, combina las técnicas ágiles probadas y examinadas por la industria, tales como el desarrollo guiado por pruebas (TDD), la modelación ágil, la gestión ágil de cambios y la refactorización de base de datos, a fin de brindar un producto funcional de la mejor calidad.

El Proceso Unificado Ágil modela sus procesos y técnicas con base en los valores de las herramientas de simplicidad, agilidad, personalización, auto-organización e independencia y se enfoca en actividades de alto valor. Los principios y valores del Proceso Unificado Ágil se ponen en acción en las fases de: Incepción (inicio), Elaboración, Construcción y Transición.

A.4.9 Domain-Driven Design, DDD *(Diseño guiado por el dominio)*

El diseño guiado por el dominio (DDD, por sus siglas en inglés) es un método de desarrollo ágil diseñado para manejar diseños complejos con la implementación vinculada a un modelo evolutivo. Fue conceptualizado en el 2004 por Eric Evans y gira en torno al diseño de un dominio central. La palabra "dominio" se define como un área de actividad en la cual el usuario aplica un programa o funcionalidad. Muchas de estas áreas se procesan en lotes y se diseña un modelo. El modelo consiste en un sistema de abstracciones que se pueden utilizan para diseñar un proyecto general y resolver los problemas relacionadas a los dominios en lote. Los valores centrales del DDD incluyen: el diseño orientado en el dominio, diseño guiado por el modelo, el lenguaje ubicuo y el contexto limitado.

En el DDD se establece un lenguaje ubicuo y se modela el dominio. Después sigue el diseño, el desarrollo y la evaluación. La refinación y refactorización del modelo del dominio se lleva a cabo hasta que sea satisfactorio.

APÉNDICE B. AUTORES Y COLABORADORES DE LA *GUÍA SBOK*™

Este apéndice incluye los nombres de las personas que contribuyeron al desarrollo y producción de la *Guía SBOK™.*

SCRUMstudy™ agradece a todas estas personas por su apoyo continuo y reconoce su colaboración en el desarrollo de la *Guía de SBOK™.*

B.1 Autor principal

Tridibesh Satpathy

B.2 Co-autores y Comité de Expertos en la Materia

Winfried Hackmann
Gaynell Malone
J. Drew Nations
Buddy Peacock
Deepak Ramaswamy
Ruth Kim

B.3 Colaboradores y revisores

James Pruitt
Elizabeth Lynne Warren
Syed Ashraf
Melvin Wofford Jr.
Srinivas Reddy Kandi
Girish Kulkarni
Abdelnaser Dwaikat
Anu Ravi
Morris Feigel
Ian Glenister
Jo Pereira

Saurabh Gupta
Nikhil Bhargava
Simon Robertson
Dusan Kamenov
Prof. Dr. Akram Hassan
Meena Elangovan
Olumide Idowu
Vinod Kumar
Michael Rauch
Joe Schofield

B.4 Colaboradores y revisores

Co-autores y expertos en la materia

R-A Alves
Winfried Hackmann
Quincy D. Jordan
Gaynell Malone
J. Drew Nations
Buddy Peacock
Karen Lyncook
Jaimie M. Rush
Elizabeth Lynne Warren
Ruth Kim
Mehul Doshi
Gaurav Garg
Ajey Grandhem
Sayan Guha
Dusan Kamenov
Vinay Jagannath
Deepak Ramaswamy

Ahmed Touseefullah Siddiqui

Revisores y equipo de edición

Corey T. Bailey
Sohini Banerjee
Vince Belanger
Bobbie Green
Magaline D. Harvey
Ravneet Kaur
Robert Lamb
Mimi LaRaque
Melissa Lauro
Richard Mather
Lachlan McGurk
Madhuresh Kumar Mishra
Neha Mishra
Yogaraj Mudalgi
Jose Nuñez
Obi Nwaojigba
Bryan Lee Perez
James Pruitt
Charles J. Quansah
Frank Quinteros
Nadra Rafee
Tommie L. Sherrill
Barbara Siefken
Sandra A. Strech
Frances Mary Jo Tessler
Chrys Thorsen
Mike Tomaszewski
Ron Villmow

APÉNDICE C. ACTUALIZACIONES A LA TERCERA EDICIÓN

Este apéndice brinda un resumen de las actualizaciones implementadas en la tercera edición de la *Guía SBOK*™ en comparación a la edición anterior.

C.1 Resumen de cambios

El ámbito de las actualizaciones realizadas en la tercera edición de la *Guía SBOK*™ se enfoca primordialmente en las siguientes principales áreas:

- Descripción mejorada y ampliada de los roles y responsabilidades del Scrum Framework, particularmente en su relación a grandes proyectos, programas y portafolios.
- Aclaración y racionalización de los procesos identificados en la fase de Planificación y estimación. Incluye la simplificación de las reuniones en estos procesos.
- Contenido adicional que aborda el escalamiento de Scrum en grandes proyectos y a nivel de la empresa.

También se realizaron mejoras generales en todo el texto a fin de asegurar que la información sea correcta, clara y completa. Incluye actualizaciones apropiadas a las tablas y figuras.

C.2 Actualizaciones en la tercera edición por capítulo

Capítulo	Cambios importantes realizados
1	• Claridad y consistencia mejorada. • Referencia adicional sobre dos nuevas certificaciones: SSMC™ y SSPOC™ (sección 1.3). • Actualización a la siguiente designación de certificación: AEC • Procesos de Scrum actualizados (sección 1.4.4) para reflejar los nuevos nombres de los procesos en la fase de Planificación y estimación (véase el capítulo 9). También se agregó el proceso analizado en el capítulo 13 y 14 sobre el Escalamiento de Scrum en grandes proyectos y el Escalamiento de Scrum para la empresa.
2	• Simplificación del lenguaje de las tres preguntas diarias en el proceso de *Realizar Daily Standup* para que sea más genérico a la hora de reunión del día (sección 2.7.1). • Se proporciona una descripción más detallada de la reunión de planificación del sprint (sección 2.7.1).

Capítulo	Cambios importantes realizados
3	• En general, este capítulo fue reestructurado para consolidar las descripciones de los roles y responsabilidades bajo los roles principales de Scrum: Product Owner (sección 3.4), Scrum Master (sección 3.5) y Equipo Scrum (sección 3.6). Incluye definiciones ampliadas, particularmente para los roles relacionados a los grandes proyectos, programas y portafolios. • Actualización del resumen de responsabilidades (sección 3.8) para incluir los roles del Chief Product Owner y Chief Scrum Master.
4	• Actualización del resumen de responsabilidades (sección 4.8) para incluir los roles del Chief Product Owner y Chief Scrum Master.
5	• Descripción mejorada de la definición de terminado (sección 5.4.2) y de los criterios mínimos de terminado (sección 5.4.3). • Actualización del resumen de responsabilidades (sección 4.8) para incluir los roles del Chief Product Owner y Chief Scrum Master.
6	• Actualización del resumen de responsabilidades (sección 6.7) para incluir los roles del Chief Product Owner y Chief Scrum Master.
7	• Actualización del resumen de responsabilidades (sección 7.7) para incluir los roles del Chief Product Owner y Chief Scrum Master.
8	• Traslado de las descripciones del Program Product Owner y Program Scrum Master al capítulo 3 para efectos de consistencia. • Cambios menores para actualizar la terminología y las figuras a fin de que coincidan con las actualizaciones realizadas en otros capítulos.
9	• El proceso de *Aprobar, estimar y comprometer historias de usuario* fue reemplazado por los siguientes dos procesos: *Estimar historias de usuario* (sección 9.2) y *Comprometer historias de usuario* (sección 9.3). Esto se hizo para proporcionar una mejor claridad a las entradas, herramientas y salidas relevantes a las actividades que se llevan a cabo en estos procesos. • Se definió una nueva herramienta llamada “Métodos de estimación” para consolidar varias de las técnicas individuas en la adición anterior (secciones 9.2.2.3 y 9.5.2.3). • Se modificó el nombre del proceso de *Crear tareas* a *Identificar tareas* (sección 9.4) para aclarar que las tareas se definen o se identifican con base en las historias de usuario comprometidas anteriormente. • Las entradas, herramientas y salidas de todos los procesos en la fase de Planificación y estimación fueron evaluadas y ajustadas para efectos de corrección.

Capítulo	Cambios importantes realizados
10	• Simplificación del lenguaje de las tres preguntas diarias en el proceso de *Realizar Daily Standup* para que sea más genérico a la hora de reunión del día (sección 10.2.2.2). • Cambios menores para actualizar la terminología y las figuras para que coincidan con las actualizaciones en otros capítulos.
11	• Eliminación del proceso de *Convocar a un Scrum de Scrums*. Ahora se aborda en el capítulo 13: Escalamiento de Scrum para grandes proyectos. • Cambios menores para actualizar la terminología y las figuras para que coincidan con las actualizaciones en otros capítulos.
12	• Cambios menores para actualizar la terminología y las figuras para que coincidan con las actualizaciones en otros capítulos.
13	• Escalamiento de Scrum para grandes proyectos - Capítulo completamente nuevo agregado como contenido nuevo.
14	• Escalamiento de Scrum para la empresa - Capítulo completamente nuevo agregado como contenido nuevo.

REFERENCIAS

Anderson, D., Augustine, S., Avery, C., Cockburn, A., Cohn, M., DeCarlo, D., Fitzgerald, D., Highsmith, J., Jepsen, O., Lindstrom, L., Little, T., McDonald, K., Pixton, P., Smith, P., and Wysocki, R. (2005) "Declaration of Interdependence," obtenido en septiembre de 2013, http://www.pmdoi.org/.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001) "Manifesto for Agile Software Development," obtenido en septiembre de 2013, http://agilemanifesto.org/.

Fellers, G. (1994) *Why Things Go Wrong: Deming Philosophy In A Dozen Ten-Minute Sessions*. Gretna, LA: Pelican Publishing.

Greenleaf, R. K. (1977) *Servant Leadership: A Journey into the Nature of Legitimate Power and Greatness. Mahwah*, NJ: Paulist Press.

Kano, N., Seraku, N., Takahashi, F., and Tsuji, S. (1984) "Attractive Quality and Must Be Quality." *Quality*, 14 (2): 39–48.

Leffingwell, D. and Widrig, D. (2003) *Managing Software Requirements: A Use Case Approach, 2nd ed*. Boston: Addison-Wesley.

Maslow, A. H. (1943) "A Theory of Human Motivation." *Psychological Review*, 50 (4): 370–396.

McGregor, D. (1960) *The Human Side of Enterprise*. Nueva York: McGraw-Hill.

Patton, J. (2005) "It's All in How You Slice." Better *Software*, enero: 16–40.

Spears, L. C. (2010) "Character and Servant Leadership: Ten Characteristics of Effective, Caring Leaders." *The Journal of Virtues & Leadership*, 1 (1): 25–30.

Takeuchi, H. and Nonaka, I. (1986) "The New New Product Development Game." *Harvard Business Review*, enero–febrero: 137–146.

GLOSARIO

Acta constitutiva del proyecto (*Project Charter*)

Un acta constitutiva del proyecto (conocida en inglés como *Project Charter*) es una declaración oficial de los objetivos y resultados deseados del proyecto. En varias organizaciones, el acta constitutiva del proyecto es el documento que autoriza el proyecto oficial y formalmente, dándole al equipo la autoridad por escrito para iniciar el proyecto.

Actionable Escalations

El Scrum Guidance Body puede determinar que algunas de las políticas de la empresa no permiten a los equipos obtener los máximos beneficios de la aplicación de Scrum. En tal caso, se debe accionar un escalamiento para obtener la autorización en el cambio de política.

Actitud de riesgo

En esencia, la actitud de riesgo de los stakeholders determina cuánto riesgo considera aceptable el stakeholder(s). Es un factor determinante cuando deciden tomar acciones para mitigar posibles riesgos adversos.

Acuerdo de entregables funcionales (*Working Deliverables Agreement*)

Los entregables que cumplen con los criterios de aceptación, reciben el cierre formal del negocio y la aprobación formal por parte del cliente o del patrocinador.

Adaptación

La adaptación sucede cuando el equipo principal de Scrum y el (los) stakeholder(s) aprende(n) por medio de la transparencia e inspección, adaptando después lo aprendido para mejorar el trabajo que realizan.

Agreed Actionable Improvements

Los llamados *Agreed Actionable Improvements* son el principal resultado del proceso de *Retrospectiva del sprint*. Forman parte de la lista de *actionable items* que ha elaborado el equipo para hacer frente a los problemas y mejorar los procesos a fin de mejorar también su desempeño en futuros sprints.

Alcance (*Scope*)

El alcance de un proyecto es la suma total de todos los incrementos del producto y todo el trabajo necesario para desarrollar el producto final.

Ambiente(s)

Es la identificación y documentación de todos los ambientes necesarios para desarrollar y evaluar los entregables del proyecto.

Amenazas

Las amenazas son riesgos que pudieran afectar el proyecto en forma negativa.

Análisis de brecha

El análisis de brecha (conocido en inglés como *Gap Analysis*) es una técnica que se utiliza para comparar el estado actual con el estado deseado y determinar la forma de acortar la brecha entre ambos

Análisis de Kano

El análisis de Kano fue desarrollado en 1984 por Noriaki Kano y consiste en clasificar las características o requisitos en cuatro categorías con base en las preferencias del cliente:

1. Calidad atractiva (*Exciters/Delighters*)
2. Calidad unidimensional (*Satisfiers*)
3. Calidad requerida (*Dissatisfiers*)
4. Calidad indiferente (*Indifferent*)

Análisis de Pareto

Esta técnica de evaluación de riesgos implica la clasificación de riesgos por magnitud. Ayuda al Equipo Scrum a atender los riesgos por orden de impacto probable en un proyecto.

Análisis de Stakeholder

El análisis estándar de stakeholders se utiliza para identificar a los stakeholders al nivel del programa y portafolio. Los detalles adicionales relacionados a los stakeholders del programa o portafolio pueden ser identificados como prototipos (*personas*) en el proceso de *Crear y refinar el backlog del programa o portafolio*.

Análisis del valor ganado (*Earned Value Analysis*)

El análisis del valor ganado (AVG) analiza el verdadero rendimiento del proyecto en comparación al rendimiento planeado en un punto previsto. El análisis del valor ganado mide las variaciones actuales en la agenda del proyecto, así como el costo de funcionamiento y prevé del costo final con base en el rendimiento actual determinado.

Análisis FODA

El FODA (SWOT en inglés) es un enfoque estructurado para la planificación que ayuda a evaluar las Fortalezas, Oportunidades, Debilidades y Amenazas relacionadas con un proyecto. Este tipo de análisis ayuda a identificar tanto los factores internos como los externos que podrían afectar el proyecto.

Apetito de riesgo

El apetito de riesgo es la cantidad de incertidumbre que esto dispuesta a asumir un stakeholder o una organización.

Auto-organización

Scrum cree que los empleados son trabajadores auto motivados y buscan aceptar mayores responsabilidades. Por lo tanto, pueden ofrecer mucho más valor cuando se organizan por cuenta propia.

Árboles de probabilidad

Los eventos potenciales se representan en un diagrama con una rama para cada resultado posible de los acontecimientos. La probabilidad de cada resultado se indica en la rama apropiada, y estos valores se pueden utilizar para calcular el impacto general de la ocurrencia de riesgos en un proyecto.

Assigned Action Items y fechas límite

Una vez que se han elaborado y refinado, los *Assigned Action Items* (elementos de acción asignados) y las fechas límite, el Equipo Scrum puede considerar los puntos de acción para implementar las mejoras. Cada elemento de acción contará con una fecha límite de conclusión.

Aversión al riesgo

La aversión al riesgo es una de las categorías de la función de utilidad. Describe al stakeholder que no está dispuesto a aceptar un riesgo sin importar el beneficio o la oportunidad anticipada.

Backlog Priorizado del Producto

El Backlog Priorizado del Producto es un solo documento de requisitos que define el alcance del proyecto, proporcionando una lista de prioridades de las características del producto o servicio a ser entregado por el proyecto.

Backlog priorizado del programa o del portafolio actualizado

El backlog priorizado del programa o del portafolio puede ser actualizado con nuevas historias de usuario, con nuevas solicitudes de cambio, riesgos identificados, historias de usuario actualizadas o la re-priorización de las historias de usuario existentes.

Benchmarking

Una empresa debe referenciar (*benchmark*) con regularidad sus propias prácticas en comparación a otras industrias a fin de estar al día con la competencia. El denominado *benchmarking* es el proceso de comprar los procesos empresariales de una organización y los parámetros de desempeño con aquellos en las empresas líderes en la misma industria o en otras.

Beneficios del proyecto

Los beneficios del proyecto incluyen todas las mejoras cuantificables de un producto, servicio o resultado que se pudieran obtener durante la conclusión satisfactoria de un proyecto.

Búsqueda de riesgos (*Risk Seeking*)

La búsqueda de riesgos es una de las categorías de la función de utilidad que se refiere a que un stakeholder esté dispuesto a aceptar el riesgo incluso si ofrece un aumento marginal de retorno o beneficio al proyecto.

Calendario del equipo

El calendario del equipo contiene información sobre la disponibilidad de los miembros, incluyendo información relacionada a las vacaciones de los empleados, licencias, ausencias temporales y días festivos.

Calidad

La calidad se define como la capacidad con la que cuenta el producto terminado o los entregables para cumplir con los criterios de aceptación y de alcanzar el valor de negocio que el cliente espera.

Chief Product Owner

En los grandes proyectos, el Chief Product Owner prepara y refina el Backlog Priorizado del Producto para el proyecto. Él o ella coordina el trabajo entre los Product Owners de los equipos Scrum. Los Product Owners, a su vez, se encargan de administrar sus respectivas partes en el Backlog Priorizado del Producto.

Chief Scrum Master

En los grandes proyectos, el Chief Scrum Master es responsable de moderar las reuniones de Scrum de Scrums (SoS) y eliminar los impedimentos que afectan a los varios equipos.

Ciclo PDCA/PDSA

El ciclo de planificar, hacer, verificar y actuar (PDCA, por sus siglas en inglés), conocido también como Ciclo Deming o Shewhart, fue desarrollado por el doctor W. Edwards Deming, considerado como el padre del control de calidad moderno, y por el doctor Walter A. Shewhart. Deming modificó después el nombre de "planificar, hacer, verificar y actuar" a "planificar, hacer, estudiar y actuar" (PDSA, por sus siglas en inglés) ya que consideraba que el término "estudiar" hacía énfasis en el análisis en vez de simplemente inspección, como lo implica el término "verificar". Tanto Scrum como el Ciclo Deming/Shewhart/PDCA son métodos iterativos enfocados en la mejora continua.

Clasificación relativa de priorización (*Relative Prioritization Ranking*)

La clasificación relativa de priorización, conocida en inglés como: *Relative Prioritization Ranking*, es una simple enumeración de historias de usuario por orden de prioridad. Es un método eficaz para determinar las historias deseadas para cada iteración o lanzamiento del producto o servicio.

Cliente

El cliente es un individuo u organización que adquiere el producto servicio u otro resultado de un proyecto. En cualquier organización, dependiendo del proyecto, puede haber clientes internos (dentro de la misma organización) o clientes externos (fuera de la organización).

Clientes meta para el lanzamiento (*Target Customers for Release*)

No todos los lanzamientos estarán dirigidos a todos los stakeholders o usuarios. El(los) stakeholder(s) puede(n) optar por limitar ciertos lanzamientos a un subconjunto de usuarios. El plan de lanzamiento debe especificar cuáles son los clientes en quienes se va a enfocar el lanzamiento.

Colaboración

La colaboración en Scrum se refiere a que el equipo principal de Scrum trabaja e interactúa con los stakeholders para crear y validar los resultados del proyecto a fin de cumplir con los objetivos que se plantean en la visión del proyecto. La colaboración se produce cuando un equipo trabaja en conjunto para contraponer los aportes del otro a fin de producir algo más grande

Comparación por pares (*Paired Comparison*)

La comparación por pares es una técnica donde se prepara una lista de todas las historias de usuario en el Backlog Priorizado del Producto. Después, cada historia de usuario se toma en forma individual y se compara con otras historias en la lista, una a la vez. Cada vez que se comparan dos historias de usuario, se toma una decisión en cuanto a cuál de las dos es más importante. Por medio de este proceso, se puede generar una lista priorizada de historias de usuario.

Comprometer historias de usuario

En este proceso, el Product Owner aprueba las historias de usuario de un sprint. Después, el Equipo Scrum calcula el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario. Por último, el equipo se compromete a entregar al cliente los requerimientos en forma de historias de usuario aprobadas, estimadas y comprometidas.

Comunicación de riesgo

La comunicación de riesgo implica la comunicación de los resultados de los primeros cuatros pasos de la gestión de riesgos a los stakeholders apropiados y determinar su percepción respecto a eventos inciertos.

Contrato de desarrollo en fases (*Development in Phases Contract*)

Este contrato facilita la disponibilidad de fondos cada mes o cada trimestre después de concluir satisfactoriamente un lanzamiento. Incentiva tanto al cliente como al proveedor y garantiza que el riesgo monetario del cliente esté limitado a un periodo específico, ya que los lanzamientos fracasados no reciben financiamiento.

Contrato de entrega incremental (*Incremental Delivery Contract*)

Este contrato incluye puntos de inspección en intervalos frecuentes. Ayuda a que el cliente o los stakeholders tomen decisiones periódicas sobre el desarrollo a lo largo de un proyecto en cada punto de inspección. El cliente puede aceptar el desarrollo del producto, optar por detener su desarrollo o solicitar modificaciones al mismo.

Contrato de incentivos y sanciones

Este contrato se basa en el acuerdo de que el proveedor será recompensado con un incentivo económico si los productos del proyecto se entregan a tiempo, pero incurrirá en sanciones económicas si la entrega se realiza tarde.

Contrato Joint Venture

Este tipo de contratos se utiliza generalmente cuando dos o más partes se asocian para realizar el trabajo de un proyecto. Ambas partes involucradas en el proyecto lograrán algún retorno sobre la inversión, ya que los ingresos o beneficios que sean generados, se habrán de compartir entre las partes.

Control de calidad

El control de calidad es la ejecución de las actividades de calidad planeadas por el Equipo Scrum en el proceso de creación de entregables que la potencialidad de enviarse. Incluye también el aprendizaje de cada serie de actividades realizadas a fin de lograr una mejora continua.

Control del proceso empírico

Un modelo de control de proceso empírico ayuda a tomar decisiones basadas en la observación y en la experimentación en vez de la planificación inicial detallada. Se base en las tres principales ideas de transparencia, inspección y adaptación.

Contenido del lanzamiento

Consiste en la información esencial sobre los entregables que puede ayudar al equipo de atención al cliente.

Costo de oportunidad

El costo de oportunidad es el valor de la siguiente mejor opción de negocio o proyecto que fue descartado en favor del proyecto seleccionado.

Costos del proyecto

Los costos del proyecto son las inversiones y demás costos de desarrollo en un proyecto.

Co-ubicación (*Collocation*)

Co-ubicación (del inglés: *collocation*) es tener a todos los miembros del equipo principal de Scrum ubicados en el mismo lugar de trabajo, aprovechando las ventajas de una mejor coordinación, resolución de problemas, intercambio de conocimiento y aprendizaje.

Crear el Backlog Priorizado del Producto

En este proceso se elaboran y se refinan las épicas dándoles después prioridad para crear un Backlog Priorizado del Producto para el proyecto. Durante este proceso también se establecen los criterios de terminado.

Crear el Sprint Backlog

En este proceso, el equipo principal de Scrum lleva a cabo reuniones de planificación del sprint donde el grupo crea un Sprint Backlog que contiene todas las tareas a realizar durante el sprint.

Crear entregables

Crear entregables es el proceso en el cual el Equipo Scrum trabaja en las tareas del Sprint Backlog para crear entregables del sprint.

Crear historias de usuario

En este proceso se crean las historias de usuario y los criterios de aceptación de las historias de usuario. Las historias de usuario generalmente las escribe el Product Owner y están diseñadas para asegurar que los requisitos del cliente estén claramente representados y puedan ser plenamente comprendidos por todos los stakeholders.

Crear la visión del proyecto

En este proceso se revisa el caso de negocio del proyecto para crear la declaración de visión del proyecto (*Project Vision Statement*) que servirá de inspiración y brindará un enfoque para la totalidad del proyecto. Durante este proceso se identifica al Product Owner.

Criterios de aceptación de historias de usuario

Cada historia de usuario cuenta con sus respectivos criterios de aceptación. Las historias de usuario son subjetivas de tal forma que los criterios de aceptación brindan la objetividad requerida para que las historias de usuario se consideren terminadas o no terminadas durante la revisión del sprint, dando claridad al equipo sobre lo que se espera de una historia de usuario.

Criterios de estimación

El principal objetivo del uso de los criterios de estimación es mantener tamaños relativos de estimación y minimizar la necesidad de volver a estimar. Los criterios de estimación pueden expresarse de muchas formas. Dos ejemplos comunes son los puntos de historia y el tiempo ideal.

Criterios de selección de miembros

Los stakeholders desarrollan los criterios de selección de miembros para definir quiénes serán los miembros del Scrum Guidance Body, cuáles serán sus respectivos roles y responsabilidades, la cantidad de integrantes, así como la experiencia y las habilidades necesarias.

Criterios de terminado

Los criterios de terminado son un conjunto de reglas que se aplican a todas las historias de usuarios. Es importante contar con una definición clara de terminado, ya que elimina la ambigüedad de los requisitos y ayuda a que el equipo se apegue a las normas obligatorias de calidad. Esta clara definición se utiliza para crear los criterios de terminado, que son un resultado del proceso de *Crear el Backlog Priorizado del Producto*. Una historia de usuario se considera terminada cuando se demuestra al Product Owner y es aprobada por este, quien juzga con base a los criterios de terminado y a los criterios de aceptación de las historias de usuario.

Criterios mínimos de aceptación

Los criterios mínimos de aceptación son declarados por la unidad empresarial. Después se convierten en parte de los criterios de aceptación para cualquier historia de usuario para dicha unidad empresarial. Cualquier funcionalidad definida por la unidad empresarial debe satisfacer dichos criterios mínimos de aceptación si busca ser aceptada por el Product Owner.

Cronograma de planificación del lanzamiento (*Release Planning Schedule*)

Un cronograma de planificación del lanzamiento (*Release Planning Schedule*) es una de las salidas más importantes del proceso de *Realizar la planificación del lanzamiento*. Un cronograma de planificación del lanzamiento indica cuáles entregables serán entregados al cliente, así como los intervalos planificados y fechas para los lanzamientos. Tal vez no exista un lanzamiento programado al final de cada iteración del sprint.

Cuadrícula de probabilidad e impacto (*Probability Impact Grid*)

Es una cuadrícula donde se evalúan los riesgos para establecer la probabilidad de ocurrencia y del impacto potencia en los objetivos del proyecto. Generalmente, se asigna una clasificación numérica tanto para la probabilidad, como para el impacto en forma independiente. Ambos valores se multiplican para obtener una puntuación de la gravedad del riesgo, lo cual se puede utilizar para priorizar los riesgos.

Cuatro preguntas por equipo

Conjunto de preguntas formuladas en cada reunión de Scrum de Scrums (SoS). Cada representante del Equipo Scrum proporciona actualizaciones de su equipo, que generalmente se presentan en forma de respuesta as cuatro preguntas específicas:

1. ¿En qué ha trabajado mi equipo desde la última reunión?
2. ¿Qué va a hacer mi equipo hasta la próxima reunión?
3. ¿Qué esperaban los demás equipos que hiciera el nuestro y que no se ha hecho?
4. ¿Qué planes tiene hacer nuestro equipo que pudieran afectar a otros equipos?

Daily Standup Meeting

El Daily Standup es una breve reunión diaria con un time-box de 15 minutos. Los miembros del equipo se reúnen para informar sobre cómo avanza el proyecto, respondiendo a las siguientes tres preguntas:

1. ¿Qué he hecho desde la última reunión?
2. ¿Qué tengo planeado hacer antes de la siguiente reunión?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Declaración de la visión del proyecto (*Project Vision Statement*)

El resultado clave del proceso de *Crear la visión del proyecto* es una declaración bien estructurada de la declaración de la visión del proyecto (*Project Vision Statement*). Una buena visión del proyecto explica las necesidades empresariales, así como lo que el proyecto busca cumplir en vez de cómo habrá de satisfacer la necesidad.

Dependencias discrecionales

Las dependencias discrecionales son aquellas que se colocan en el flujo de trabajo por decisión propia. Normalmente, el Equipo Scrum determina cuáles son las dependencias discrecionales con base en experiencias anteriores o las mejores prácticas en un campo o dominio específico.

Demostrar y validar el sprint

En este proceso, el Equipo Scrum demuestra los entregables del Sprint al Product Owner y a los stakeholders relevantes durante una reunión de revisión del sprint.

Dependencias externas

Las dependencias externas son aquellas que están relacionadas a las tareas, actividades o productos fuera del alcance del trabajo a realizar por el Equipo Scrum, pero que son necesarias para concluir una tarea del

proyecto o crear un entregable del proyecto. Las dependencias externas generalmente están fuera del alcance del Equipo Scrum.

Dependencias internas

Las dependencias internas son aquellas entre las tareas, productos o actividades que están bajo el control del Equipo Scrum y dentro de un enfoque de trabajo a ser ejecutado por dicho equipo.

Dependencias obligatorias

Dependencias que son inherentes en la naturaleza del trabajo (como una limitación física) o pueden darse debido a obligaciones contractuales o por requerimientos legales.

Despunte basado en riesgo (*Risk-Based Spike*)

Los despuntes basados en riesgo son básicamente experimentos que implican una investigación o hacer un prototipo para entender mejor los riesgos potenciales. En un *spike*, se lleva a cabo un intenso ejercicio de dos a tres días (de preferencia al inicio de un proyecto, antes del proceso de *Desarrollar épica(s)* o de *Crear el Backlog Priorizado del Producto*) para ayudar al equipo a determinar las incertidumbres que pudieran afectar al proyecto.

Desarrollar épica(s)

En este proceso la Declaración de visión del proyecto sirve como base para el desarrollo de épicas. Se pueden llevar a cabo reuniones de grupos de usuarios para hablar sobre las épicas adecuadas.

Desarrollo iterativo

El desarrollo iterativo es la entrega gradual de valor al cliente.

Descomposición

La descomposición (también conocida como “segmentación”) es una herramienta donde las tareas de alto nivel se dividen en niveles inferiores y más detallados. Los miembros del Equipo Scrum segmentan las historias de usuario en tareas. Las historias de usuario del Backlog Priorizado del Producto deben estar suficientemente segmentadas a un nivel que le brinde al Equipo Scrum la información adecuada para crear entregables a partir de las tareas mencionadas en la lista de tareas.

Determinación de dependencia

Una vez que el Equipo Scrum ha seleccionado las historias de usuario para un determinado sprint, deben considerar las dependencias, incluyendo aquellas relacionadas a la disponibilidad de personal, así como

cualquier dependencia técnica. Documentar adecuadamente las dependencias ayuda a los equipos Scrum a determinar el orden relativo en el cual deben ejecutarse las tareas para crear los entregables del sprint. Las dependencias destacan también la relación e interacción entre las tareas dentro del Equipo Scrum que trabaja en un determinado Sprint y con otros equipos Scrum en el proyecto.

Deuda técnica

La deuda técnica, conocida también como deuda de diseño o deuda de código, es el trabajo al que los equipos dan menor prioridad; el trabajo que omiten o que no terminan a medida que trabajan en la creación de los principales entregables asociados al producto del proyecto. La deuda técnica se acumula y se debe saldar a futuro.

Diagrama de flujo acumulativo (DFA)

El diagrama de flujo acumulativo (DFA) es una herramienta útil para la elaboración de informes y para el seguimiento de los resultados del proyecto. Proporciona una representación sencilla y visual del avance del proyecto en un punto de tiempo determinado. Se utiliza generalmente para brindar un estado de mayor nivel de la totalidad del proyecto y no para actualizaciones individuales diarias de sprints.

Dinero de Monopoly

El dinero de Monopoly es una técnica que consiste en darle al cliente dinero del juego Monopoly, o "dinero falso", equivalente a la cantidad del presupuesto del proyecto, solicitando que lo distribuyan entre las historias de usuario que están a consideración. De esta forma, el cliente prioriza con base en lo que está dispuesto a pagar por cada historia de usuario.

Duración del sprint

Con base en las diversas entradas, incluyendo los requerimientos del negocio y el cronograma de planificación del lanzamiento, el Product Owner y el Equipo Scrum deciden la duración del sprint para el proyecto. Una vez determinada, la duración del sprint generalmente permanece igual durante el proyecto.
La duración del Sprint es la duración de los sprints determinados para un proyecto.

Effort Estimated Task List

La llama *Effort Estimated Task List* (lista de tareas del esfuerzo estimado) es una lista de tareas asociadas con las historias de usuario incluidas en un sprint. El esfuerzo estimado se expresa en términos de los criterios de estimación acordados por el equipo. El Equipo Scrum utiliza la *Effort Estimated Task List* durante las reuniones de planificación del sprint a fin de crear el Sprint Backlog y el Sprint Burndown Chart.

Elementos no funcionales propuestos para el Backlog Priorizado del Producto (*Proposed Non-Functional Items for Product Backlog*)

Los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint o en las reuniones de retrospectiva del sprint. Estos elementos deben agregarse al Backlog Priorizado del Producto a medida que se descubren.

Entregables aceptados (*Accepted Deliverables*)

Los entregables que cumplen con los criterios de aceptación de las historias de usuario son aceptados por el Product Owner. Estos se consideran entregables aceptados que pueden presentarse al cliente si así lo desean.

Entregables del sprint (*Sprint Deliverables*)

Los entregables del sprint son los incrementos del producto o los entregables que se terminan al final de cada sprint.

Entregables funcionales (*Working Deliverables*)

Esta salida es el entregable final enviable (del inglés: *shippable deliverable*) para el cual fue sancionado el proyecto.

Entregables potencialmente enviables de los proyectos

Los entregables potencialmente enviables (*Potentially Shippable Deliverables*) de los proyectos son entradas valiosas para la coordinación al nivel del programa o portafolio. Al final de los sprints en proyectos, se completan los entregables o incrementos de producto. Las historias de usuario incluidas en dichos incrementos cumplen con los criterios en la definición de terminado, así como con sus respectivos criterios de aceptación.

Entregables rechazados

Los entregables rechazados son aquellos que no cumplen con los criterios de aceptación definidos. Después de cada reunión de revisión del sprint, se actualiza y se mantiene una lista de entregables rechazados incluyendo los entregables que no fueron aceptados.

Enviar entregables

En este proceso, los entregables aceptados se envían o se pasan a los stakeholders pertinentes. Un acuerdo formal de los entregables funcionales documenta la conclusión satisfactoria del sprint.

Environment Meeting (*Reunión de ambiente*)

Esta reunión se lleva a cabo para identificar el tipo y la cantidad de ambientes necesarios para desarrollar, administrar y evaluar los entregables del proyecto. En esta reunión también se discuten los recursos necesarios para establecer los ambientes requeridos.

Environment Schedule

El *Environment Schedule* es un horario/calendario que indica cómo los equipos Scrum compartirán los ambientes. Proporciona los días y periodos asignados a cada equipo para el uso de cada ambiente.

Épica(s)

Las épicas se escriben en las etapas iniciales del proyecto, cuando la mayoría de las historias de usuario son funcionalidades de alto nivel o descripciones de productos que están ampliamente definidas. Las épicas son historias de usuario grandes sin refinar en el Backlog Priorizado del Producto.

Equipo Scrum

El Equipo Scrum es uno de los roles del equipo principal de Scrum. El Equipo Scrum trabaja en la creación de entregables del proyecto y contribuye a la realización del valor del negocio para todos los stakeholders y del proyecto.

Escalas de tiempo del proyecto (*Project Timescales*)

Las escalas de tiempo reflejan la duración de un proyecto y el tiempo durante el cual se obtendrán los beneficios del mismo.

Especialización del equipo

En un proyecto grande tal vez sea necesaria la especialización del equipo. Existen tres dimensiones de especialización de equipo. La primera dimensión es la necesidad de cumplir con una tarea específica.
La segunda dimensión es la necesidad de los integrantes individuales del equipo de contar con habilidades especiales. La tercera dimensión es que pudiera haber limitaciones en la flexibilidad del equipo.

Esquemas simples

Los esquemas simples implican etiquetar elementos como prioridad “1”, “2”, “3” o “alta”, “media” y “baja”, y así sucesivamente. Aunque se trata de un método sencillo y directo, puede llegar a ser problemático, ya que a menudo hay una tendencia en etiquetar todo como prioridad “1” o “alta”.

Estructura de distribución de riesgos (*Risk Breakdown Structure)*

En esta estructura, se agrupan los riesgos con base en sus categorías o modalidades. Por ejemplo, los riesgos se pueden clasificar como financieros, técnicos o relacionados a la seguridad.

Estándares de la industria

Los nuevos estándares de la industria o los cambios a los estándares existentes deben ser implementados a fin de mantener un producto o servicio viable. Por lo tanto, las historias de usuario relacionadas deben ser incluidas en el backlog priorizado del programa y/o del portafolio según corresponda.

Estimación por afinidad (*Affinity Estimation*)

La estimación por afinidad (del inglés: *Affinity Estimation*) es una técnica que se utiliza para estimar rápidamente un gran número de historias de usuarios con el uso de categorías. Dichas categorías pueden ser pequeñas, medianas o grandes, o bien, pueden estar enumeradas utilizando valores de punto de historia (*point story values*) para indicar el tamaño relativo. Algunos de los beneficios clave de este método son que el proceso es muy transparente, visible para todos y fácil de llevar a cabo.

Estimar historias de usuario

En este proceso, el Product Owner aclara las historias de usuario para que el Scrum Master y el Equipo Scrum puedan estimar el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario.

Estudio de mercado

El estudio de mercado es el proceso organizado de investigar, recopilar, cotejar y analizar la información relacionada a las preferencias del cliente respecto a los productos. Generalmente incluye extensos datos sobre tendencias del mercado, segmentación del mercado y procesos de comercialización.

Etapa de desempeño (*Performing*)

Es la etapa final de la formación del equipo cuando este se encuentra más unido y opera en su nivel más alto en términos de rendimiento. Los miembros se han convertido en un equipo eficiente de profesionales que son consistentemente productivos.

Etapa de enfrentamiento (*Storming*)

La etapa de enfrentamiento, conocida en inglés como *Storming Stage*, es la segunda etapa de formación del equipo donde este empieza a tratar de cumplir con el trabajo; sin embargo, puede encontrar conflictos de poder y, con frecuencia, existe un caos o confusión entre los miembros del equipo.

Etapa de formación (*Forming*)

La etapa de formación es la primera etapa en la formación de un equipo. Generalmente se considera una etapa divertida, ya que todo es novedoso y no se han encontrado dificultades con el proyecto.

Etapa de normalización (*Norming*)

La tercera etapa de la formación del equipo es cuando el equipo empieza a madurar, resolver sus diferencias internas, y encontrar soluciones para así trabajar juntos. Se considera un período de ajuste.

Evaluación de riesgo

La evaluación de riesgo es la evaluación y cálculo de los riesgos identificados.

Experiencia en la redacción de historias de usuario

El Product Owner —con base en su interacción con los stakeholders, en su conocimiento del negocio, en la experiencia y en las aportaciones del equipo—, desarrolla las historias de usuario que habrán de formar la Backlog Priorizado del Producto inicial para el proyecto.

Experiencia del equipo

Es la experiencia colectiva de los miembros del Equipo Scrum para entender las historias de usuario y las tareas en el Sprint Backlog a fin de crear los entregables finales. La experiencia del equipo se utiliza para evaluar las entradas necesarias a fin de ejecutar el trabajo planificado del proyecto.

Experiencia del Scrum Guidance Body

La experiencia del Scrum Guidance Body se refiere a las reglas y regulaciones documentadas, directrices de desarrollo o estándares y mejores prácticas.

Explorador—Comprador—Vacacionista—Prisionero (ECVP)

Es un ejercicio que se puede llevar a cabo al inicio de la reunión de retrospectiva del sprint para entender la mentalidad de los participantes y establecer el tono de la reunión. Se les pide a los asistentes que indiquen de forma anónima lo que mejor representa su punto de vista en la reunión.

Fase de implementación

La fase de Implementación incluye los procesos relacionados con la ejecución de las tareas y actividades para crear un producto de un proyecto.

Fase de inicio

Esta fase se compone de los procesos relacionados a la iniciación de un proyecto: Crear la visión del producto, Identificar al Scrum Master y stakeholder(s), Formar el Equipo Scrum, Desarrollar épica(s), Crear el Backlog Priorizado del Producto y Realizar la planificación del lanzamiento.

Fase de planificación y estimación

La fase de Planificación y Estimación consiste en procesos relacionados a la planificación y estimación de tareas, los cuales incluyen: *Crear historias de usuario*, *Estimar historias de usuario*, *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y *Crear el Sprint Backlog*.

Fechas límite de implementación actualizadas para los proyectos

Las fechas límite de implementación en los proyectos pueden ser actualizadas para que se refleje el impacto de historias de usuario nuevas o modificadas que deban cambiar o introducir requerimientos nuevos.

Fichas (*Index Cards*)

Las fichas, conocidas en inglés como *index cards*, se utilizan para dar seguimiento a las historias de usuario durante todo el proyecto. Esto aumenta la visibilidad y la transparencia, facilitando la detención oportuna de cualquier problema que pueda surgir.

Formar el Equipo Scrum

En este proceso, se seleccionan a los miembros del Equipo Scrum. Normalmente, el Product Owner es el responsable principal de la selección de los miembros del equipo, pero a menudo lo hace en colaboración con el Scrum Master.

Función de utilidad (*Utility Function*)

La función de utilidad (del inglés: *Utility Function*) es un modelo utilizado para medir la preferencia del stakeholder por el riesgo o su actitud hacia el riesgo. Esto define el nivel del stakeholder para aceptar riesgos.

Garantía de calidad

La garantía de calidad es el proceso de evaluación y los estándares que rigen la gestión de calidad en un proyecto a fin de garantizar que sigan siendo relevantes. Las actividades relacionadas a la garantía de calidad se llevan a cabo como parte del trabajo.

Gestión de calidad

En Scrum, el control de calidad les permite a los clientes conocer cualquier problema en el proyecto en forma anticipada, ayudándoles a reconocer si el proyecto habrá o no de funcionarles. En Scrum, la gestión de calidad se facilita mediante tres actividades interrelacionadas:

1. Planificación de calidad
2. Control de calidad
3. Garantía de calidad

Gestión de conflictos

Los miembros del equipo utilizan técnicas de gestión de conflictos para atender los conflictos que surjan durante un proyecto Scrum. Las fuentes de conflictos evolucionan principalmente debido a los horarios, prioridades, recursos, informes de jerarquía, cuestiones técnicas, procedimientos, personalidad y costos.

Grandes equipos principales

Los grandes equipos principales se componen por el Chief Product Owner, el Chief Scrum Master, los Scrum Masters, Productos Owners y miembros seleccionados de los equipos Scrum en los grandes proyectos.

Herramientas automatizadas de software

Las herramientas automatizadas de software son aquellas herramientas de software que se utilizan para la planificación, recopilación de información y distribución.

Herramientas de seguimiento del sprint

Las herramientas de seguimiento del sprint se utilizan para rastrear el proceso de un sprint y saber dónde se encuentra el Equipo Scrum en términos de conclusión de tareas el Sprint Backlog. Se puede utilizar una variedad de herramientas para dar seguimiento al trabajo de un sprint, pero una de las más comunes es Scrumboard, conocido también como tablero de tareas o gráfica de proceso.

Historias de usuario

Las historias de usuario se apegan a una estructura específica predefinida y son una forma simple de documentar los requerimientos y funcionalidades que desea el usuario final. Los requerimientos expresados en las historias de usuario son oraciones breves, sencillas y fáciles de entender. El formato estándar predefinido da como resultado en una comunicación mejorada entre los stakeholders, así como en mejores estimaciones por parte del equipo.

Identificación de riesgo

La identificación de riesgo es un paso importante en la gestión de riesgos, que implica el uso de varias técnicas para identificar todos los posibles riesgos.

Identificar el ambiente

En un proyecto grande, es importante identificar el tipo y cantidad de ambientes necesarios, ya que los varios equipos Scrum iniciarán y concluirán sus sprints el mismo día.

Identificar tareas

En este proceso, las historias de usuario comprometidas se desglosan en tareas específicas y se compilan en una lista de tareas. Esto se hace como parte de la reunión de planificación del sprint.

Impedimento

Un impedimento es cualquier obstáculo o barrera que reduce la productividad del Equipo Scrum.

Inspección

La inspección es el monitoreo necesario para seguir el control del proceso empírico para garantizar que los entregables del proyecto se ajusten a los requisitos.

Justificación continua de valor

La justificación continua de valor describe la constante evaluación del valor del negocio para determinar si aún existe justificación o viabilidad para la ejecución del proyecto.

Justificación del negocio

La justificación del negocio demuestra las razones para emprender un proyecto. Responde a la pregunta: "¿Por qué es necesario este proyecto?" La justificación del negocio es lo que impulsa todas las decisiones relacionadas aun proyecto.

Lecciones aprendidas del Equipo Scrum

Se espera que el Equipo Scrum (equipo auto organizado y empoderado) aprenda de los errores cometidos durante el sprint y que estas lecciones aprendidas ayuden a mejorar su desempeño en futuros sprints.

Líder asertivo *(Assertive Leader)*

Los líderes asertivos confrontan los problemas y demuestran confianza para establecer autoridad con respeto.

Líder autocrático

Los líderes autocráticos toman decisiones por su cuenta, dándoles poco o nada de tiempo a los miembros del equipo antes de tomar una decisión. Este estilo de liderazgo se debe utilizar solamente en raras ocasiones.

Líder de apoyo y entrenamiento (*Coaching/Supportive Leader*)

Los líderes de apoyo y entrenamiento (*Coaching and Supportive*) dan instrucciones; después apoyan y monitorean a los miembros del equipo al escuchar, ayudar, alentar y presentar una perspectiva positiva en tiempos de incertidumbre.

Líder delegador

Los líderes delegadores participan en la mayoría de la toma de decisiones; sin embargo, delegan parte de las responsabilidades de planificación a los miembros del equipo, particularmente si son competentes para manejar las tareas asignadas. Este estilo de liderazgo es apropiado en situaciones en las que el líder está en sintonía con los detalles específicos del proyecto y cuando el tiempo es limitado.

Líder directivo

Los líderes directivos instruyen a los miembros del equipo sobre las tareas que se requieren y sobre cómo y cuándo deben llevarse a cabo.

Líder enfocado en las tareas (*Task-Oriented Leader*)

Los líderes que se enfocan en las tareas (*Task-Oriented*) se encargan de que se hagan las tareas con apego a los plazos.

Líder Laissez Faire

Es un estilo de liderazgo en el que el equipo se queda sin supervisión y donde el líder no interfiere con las actividades laborales diarias. Esto a menudo conduce a un estado de anarquía.

Líderes serviciales (*Servant Leaders*)

El liderazgo servicial (*Servant Leadership*) implica escuchar cuidadosamente, tener empatía, comprometerse al servicio, tener visión y compartir el poder y la autoridad con los miembros del equipo. Este estilo de liderazgo logra resultados centrándose en las necesidades del equipo. Asimismo, es la personificación del rol del Scrum Master.

Lista corta de riesgos (*Risk Prompt Lists*)

Conocidas en inglés como *Risk Prompt List*, estas listas se utilizan para estimular el pensamiento respecto a la fuente de donde se pudieran originar los riesgos. Dichas listas para distintas industrias y tipos de proyectos están disponibles al público.

Lista de tareas

Es una lista integral que contiene todas las tareas a las que se ha comprometido el Equipo Scrum en el actual sprint. Contiene descripciones de cada tarea.

Listas de verificación de riesgos (*Risk Checklists*)

Las listas de verificación de riesgos (conocidas en inglés como: *Risk Checklists*) pueden incluir puntos clave a considerarse cuando se identifican los riesgos, riesgos comunes encontrados en un proyecto Scrum, o incluso categorías de riesgo que el equipo debe atender.

Lluvia de ideas (*Brainstorming*)

Son sesiones donde los stakeholders y los miembros del equipo principal de Scrum comparten abiertamente ideas por medio de discusiones y sesiones de intercambio de conocimientos, generalmente dirigidas por un facilitador.

Mapa de flujo de valor (*Value Stream Mapping*)

El mapa de flujo de valor utiliza diagramas de flujo para ilustrar el flujo de información necesaria para completar un proceso. Esta técnica pudiera utilizarse para racionalizar un proceso ayudando a determinar los elementos que no aportan valor.

Mapeo de historias (*Story Mapping*)

El mapeo de historias, conocido en inglés como *Story Mapping*, es una técnica para proporcionar un esquema visual del producto y sus componentes clave. El mapeo de historias, formulado por primera vez por Jeff Patton (2005), se utiliza comúnmente para ilustrar la ruta del producto. Los mapas de historia

representan la secuencia de las iteraciones de desarrollo del producto y trazan las características que serán incluidas en el primer, segundo, tercero y subsecuentes lanzamientos.

Matriz de recursos organizacionales (*Organizational Resource Matrix*)

La matriz de recurso organizacional es una representación jerárquica de una combinación de una estructura organizacional funcional y una estructura organizacional del proyecto. Las organizacionales matrices reúnen a los miembros del equipo para un proyecto de distintos departamentos funcionales, tales como: tecnologías de la información, finanzas, comercialización, ventas y demás departamentos a fin de crear equipos interfuncionales (*cross-functional*).

Matriz de requerimientos de habilidades (*Skills Requirement Matrix*)

La matriz de requerimientos de habilidades, también llamada framework de competencia, se utiliza para evaluar las carencias de habilidades y los requisitos de formación para los miembros del equipo. Una matriz de habilidades traza las habilidades, capacidades y el nivel de interés de los miembros del equipo en el uso de dichas habilidades y capacidades en un proyecto. Al utilizar esta matriz, la organización puede evaluar los vacíos de habilidades en los miembros del equipo e identificar a los empleados que van a necesitar más formación en un área o competencia particular.

Mejor coordinación del equipo

La reunión de Scrum de Scrums facilita la coordinación del trabajo entre los varios equipos Scrum. Es de especial importancia cuando existen tareas que involucran dependencias entre equipos. Esto permite exponer rápidamente las incompatibilidades y discrepancias entre el trabajo y los entregables de los distintos equipos. Este foro brinda también a los equipos la oportunidad de mostrar sus logros y dar su retroalimentación a los demás equipos.

Mejora continua

La mejora continua es un enfoque en Scrum donde el equipo aprende de las experiencias y de la participación de los stakeholders para mantener constantemente actualizado el Backlog Priorizado del Producto con cualquier cambio en los requisitos.

Método de los 100 puntos *(100-Point Method)*

El método de los 100 puntos fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes.

Métodos de desplazamiento organizacional

Los mecanismos de desplazamiento de cada organización tienden a ser diferentes basados en su respectiva industria, en los usuarios meta y el posicionamiento. Dependiendo del producto a entregarse, el desplazamiento puede ser remoto o puede incluir el envío físico o la transición de un artículo.

Métodos de preparación de lanzamiento (*Release Preparation Methods*)

Los métodos de preparación de lanzamiento son métodos que se utilizan para ejecutar las tareas identificadas en el Plan de preparación de lanzamiento (*Release Readiness Plan*) a fin de tener los entregables listos para ser enviados/lanzados.

Métodos de priorización del lanzamiento (*Release Prioritization Methods*)

Los métodos de priorización del lanzamiento se utilizan para desarrollar un plan del lanzamiento. Estos métodos son específicos a la industria y organización, y generalmente son determinados por la alta gerencia de la organización.

Miembros del Scrum Guidance Body

Los miembros del Scrum Guidance Body (SGB) pueden ser expertos de Scrum, Scrum Masters selectos, Product Owners y miembros del equipo (en todos los niveles). Sin embargo, debe haber un límite en la cantidad de miembros que puede tener el SGB a fin de asegurar que permanezca relevante y no tenga una naturaleza prescriptiva.

Misión de la empresa

La misión de la empresa ofrece un framework para formular las estrategias de la empresa y orienta la toma de decisiones generales en la compañía.

Mitigación de riesgo

La mitigación de riesgo es un paso importante en la gestión de riesgos que implica el desarrollo de una estrategia adecuada para hacer frente a un riesgo.

Membresía actualizada del Scrum Guidance Body

Como resultado de una evaluación en la membresía del Scrum Guidance Body, se pueden incluir a nuevos miembros en dicho órgano y se pueden remover miembros existentes.

Necesidades del negocio

Las necesidades del negocio son aquellos resultados del negocio que se espera que cumpla el proyecto, tal como se documenta en la declaración de visión del proyecto.

Neutral al riesgo

Neutral al riesgo es una de las categorías de la función de utilidad que describe al stakeholder que ni tiene aversión al riesgo, ni busca riesgos; cualquier decisión que no se ve afectada por el nivel de incertidumbre de los resultados. Cuando dos posibles escenarios llevan el mismo nivel de beneficio, el stakeholder neutral al riesgo no se preocupará si uno de dichos casos es más riesgoso que el otro.

Notas del lanzamiento (*Release Notes*)

Las notas del lanzamiento deben incluir criterios de envío externos o relacionados al mercado para que el producto sea enviado.

Número de historias

El término número de historias se refiere a la cantidad de historias de usuario que se entregan como parte de un solo sprint. Se puede expresar en términos de un simple conteo o de un conteo ponderado.

Oportunidades

A los riesgos con posibilidades de causar un impacto positivo en el proyecto se les conoce como oportunidades.

Patrocinador (*Sponsor*)

El patrocinador (del inglés: *sponsor*) es la persona o la organización que provee recursos y apoyo para el proyecto. El patrocinador es también el stakeholder a quien todos le deben rendir cuentas al final.

Patrones de diseño

Los patrones de diseño proporcionan una manera formal de registrar una resolución a un problema de diseño en un campo específico de especialización. Dichos patrones registran tanto el proceso que se utiliza como la resolución, misma que puede reutilizarse después para mejorar la tomar de decisiones y la productividad.

Plan de pilotaje (*Piloting Plan*)

El plan pilotaje (conocido en inglés como: *Piloting Plan*) se puede utilizar para trazar un desplazamiento piloto a detalle. El alcance y los objetivos del desplazamiento, la base de usuarios seleccionada para la implementación, un cronograma del desplazamiento, los planes de transición, la preparación requerida del usuario, los criterios de evaluación para el desplazamiento y otros elementos clave relacionados al desplazamiento, se identifican en el plan de pilotaje y se comparten con los stakeholders.

Planificar para el valor (*Planning for Value*)

La planificación para el valor es la justificación y confirmación del valor del proyecto. La responsabilidad de determinar cómo se crea valor recae en los stakeholders (patrocinadores, clientes y/o usuarios), mientras que el Equipo Scrum se concentra en lo que se habrá de desarrollar.

Planning Poker

El Planning Poker, conocido también como *Estimation Poker*, es un derivado de la técnica Wideband Delphi. Esta técnica de estimación implementa el consenso para estimar los tamaños relativos de las historias de usuario o el trabajo necesario para desarrollarlos.

Plan de colaboración

La colaboración es un elemento de suma importancia en Scrum. El plan de colaboración delinea la forma en la que participan y colaboran entre sí las personas encargadas de la toma de decisiones, los stakeholders y los miembros del equipo.

Plan de comunicación

En muchos proyectos existe un plan de comunicación. El plan especifica los registros que se deben crear y mantener durante todo el proyecto. Se utiliza una variedad de métodos para transmitir a los stakeholders información importante sobre el proyecto. El plan de comunicación define los métodos e indica quién es responsable de las distintas actividades de comunicación.

Plan de colaboración de equipos Scrum

El Plan de colaboración de equipos Scrum define la forma en la que los varios equipos Scrum colaboran entre sí a fin de proveer el más alto valor en el menor tiempo posible.

Plan de colaboración de Product Owners

El Plan de colaboración de Product Owners debe definir la forma en la que los múltiples Product Owners colaboran con el Chief Product Owner.

Plan de formación de equipo (*Team Building Plan*)

Debido a que el Equipo Scrum es interfuncional, cada integrante debe participar activamente en todos los aspectos del proyecto. El Scrum Master debe identificar los problemas con los miembros del equipo y hacer frente a estos a fin de mantener un equipo eficaz.

Planificación de calidad

La planificación de calidad es la identificación y definición del proyecto requerido de un sprint y del proyecto, así como los criterios de aceptación, cualquier método de desarrollo a seguir y las responsabilidades clave de los miembros del Equipo Scrum en relación a la calidad.

Políticas de la empresa

Las políticas de la empresa son una serie de principios, reglas y directrices formuladas o adoptadas por una organización. Cambiar las políticas de una empresa afectaría las historias de usuario creadas, ya que fueron creadas en apego a las políticas existentes.

Portafolio

Un portafolio es un grupo de programas relacionados con la finalidad de entregar resultados de negocio como se define en la declaración de la visión del portafolio (*Portfolio Vision Statement*). El Backlog Priorizado del Portafolio incorpora el Backlog Priorizado del Producto de todos los programas en el portafolio

Portfolio Product Owner

El Portfolio Product Owner define los objetivos y prioridades estratégicas del portafolio.

Portfolio Scrum Master

El Portfolio Scrum Master resuelve problemas, elimina impedimentos, facilita y dirige las reuniones del portafolio.

Presupuesto del proyecto

El presupuesto del proyecto es un documento financiero que incluye el costo del personal, materiales y otros gastos relacionados en un proyecto. Por lo general, dicho presupuesto es autorizado por el(los) patrocinador(es), o *sponsors* para asegurar que haya suficientes fondos disponibles.

Priorización

La priorización puede definirse como la determinación del orden de las cosas y la separación de lo que se hará ahora de lo que se puede hacer después.

Priorización basada en el valor para el cliente

La priorización basada en el valor para el cliente le da importancia primordial al cliente y se esfuerza primero en implementar las historias de usuario con más alto valor. Dichas historias de usuario de alto valor se identifican y se colocan en la parte superior del Backlog Priorizado del Producto.

Priorización MoSCoW

El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: "Debe tener" (*Must have*), "Debería tener" (*Should have*), "Podría tener" (*Could have*) y "No tendrá" (*Won't have*). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de "debería tener", siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de "gustaría que tuviera" siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.

Priorización de riesgos

La priorización de riesgos es un paso importante en la gestión de riesgos que implica la priorización de estos y que habrán de incluirse en una acción específica en el Backlog Priorizado del Producto.

Problemas (*Issues*)

Los problemas (en inglés: *issues*) generalmente son certezas bien definidas que actualmente se están sucediendo en el proyecto, por lo que no hay necesidad de realizar una evaluación de probabilidad como se hace con un riesgo.

Problemas resueltos

En las reuniones de Scrum de Scrums, los miembros del Equipo Scrum tienen la oportunidad de discutir con transparencia los problemas que tienen un impacto en el proyecto. Esta discusión y resolución oportuna de los problemas en las reuniones de Scrum de Scrums mejora en buena medida la coordinación entre los distintos equipos de Scrum y reduce también la necesidad de rediseñar o volver a realizar el trabajo.

Proceso de estimación de tareas

En este proceso, el equipo principal de Scrum, en un taller de estimación de tareas, calcula el esfuerzo necesario para lograr cada tarea en la lista. El resultado de este proceso es la llamada *Effort Estimated Task List*.

Proceso de Identificar al Scrum Master y stakeholder(s)

En este proceso se identifican el Scrum Master y a los stakeholders mediante el uso de criterios de selección específicos.

Product Owner

El Product Owner es la persona responsable de maximizar el valor del negocio en el proyecto. Es la persona responsable de articular los requerimientos del cliente y mantener la justificación del negocio del proyecto.

Producto

La palabra "producto" en la *Guía SBOK™* puede referirse a un producto, servicio o cualquier otro entregable que brinde valor al cliente.

Program Product Backlog actualizado

Program Product Backlog que se somete a un refinamiento constante a fin de incorporar cambios y nuevos requerimientos.

Program Product Owner

El Program Product Owner define los objetivos estratégicos y las prioridades del programa.

Program Scrum Master

El Program Scrum Master resuelve problemas, elimina impedimentos, facilita y lleva a cabo reuniones del programa.

Programa

Un programa es un grupo de proyectos relacionados con la finalidad de entregar resultados de negocio definidos en la declaración de la visión del programa (*Program Vision Statement*). El Backlog Priorizado del Programa incorpora al Backlog Priorizado del Producto de todos los proyectos del programa.

Prototipos (*Personas*)

Los prototipos, conocidos en inglés como *personas*, son personajes ficticios altamente detallados que representan a la mayoría de los usuarios y demás stakeholders que pudieran no utilizar directamente el producto final. Los prototipos se crean para identificar las necesidades de los usuarios.

Proyecto

Un proyecto es un emprendimiento colaborativo para crear nuevos productos o servicios, o para obtener resultados como los que se definen en la declaración de la visión del proyecto (*Project Vision Statement*). Los proyectos por lo general se ven afectados por limitaciones de tiempo, costo, alcance, calidad, personal y la capacidad de la organización.

Puño de cinco (*Fist of Five*)

El puño de cinco, o *Fist of Five*, es un mecanismo sencillo y rápido que se puede utilizar como práctica de estimación, así como técnica general de formación de consenso colectivo. Tras el debate inicial sobre sobre la estimación de un elemento, se les pide a los miembros del Equipo Scrum que voten en una escala de 1 a 5 utilizando sus dedos.

Rango de estimación

Las estimaciones de los proyectos deben presentarse en rangos. Las cifras exactas pueden dar la impresión de ser altamente precisas cuando en realidad tal vez no lo sean. De hecho, las estimaciones, según su definición, se entiende que no son precisamente exactas. Los rangos de estimación deben basarse en el nivel de confianza que el equipo tenga en cada cálculo.

Razonamiento del proyecto

El razonamiento del proyecto incluye todos los factores que este requiere, ya sean positivos, negativos, elegidos o no (por ejemplo: capacidad inadecuada para cumplir con la demanda actual y la demanda prevista, la disminución en la satisfacción del cliente, baja utilidad, requerimientos legales, etc.).

Realizar Daily Standup

Es un proceso en donde se realiza diariamente una reunión muy enfocada con un time-box asignado. A esta reunión se le conoce como Daily Standup Meeting, y es un foro que le permite a los miembros del Equipo Scrum ponerse al día sobre sus progresos y sobre cualquier impedimento que pudieran estar enfrentando.

Realizar planificación del lanzamiento

En este proceso, el equipo principal de Scrum revisa las historias de usuario de alto nivel en el Backlog Priorizado del Producto para desarrollar un programa de planificación del lanzamiento, que esencialmente es un programa de implementación por fases que se puede compartir con el (los) stakeholder(s). Durante este proceso, también se determina la duración de los sprints.

Recursos compartidos

Los recursos compartidos pueden ser el personal, el ambiente y el equipamiento que necesitan todos o algunos equipos Scrum que trabajan en un proyecto. En un proyecto grande, los recursos compartidos pueden estar limitados; son recursos que necesitan todos o parte de los equipos Scrum al mismo tiempo.

Refactorización (*Refactoring*)

La refactorización es una herramienta específica para proyectos de software. El objetivo de esta técnica es mejorar el mantenimiento del código existente y hacerlo más sencillo, más conciso y más flexible. Refactorizar significa mejorar el diseño del código actual sin cambiar el comportamiento del mismo. Implica lo siguiente:

- Eliminación de código repetitivo y redundante
- Separar los métodos y las funciones en rutinas más pequeñas
- Definir claramente las variables y los nombres de los métodos
- Simplificar el diseño del código
- Hacer que el código sea más fácil de entender y de modificar

Refinar el Backlog Priorizado del Producto

Proceso en el cual se refina y se actualiza constantemente el Backlog Priorizado del Producto.

Regulaciones

Las regulaciones incluyen cualquier normativa federal, estatal o local a la que se deba acatar el programa o portafolio. En ocasiones tal vez sea necesario actualizar las recomendaciones del Scrum Guidance Body para reflejar nuevas regulaciones.

Representantes del Equipo Scrum

Representante nombrado por el equipo para que lo represente en las reuniones de Scrum de Scrums, basado en quien puede desempeñar mejor el rol dependiendo de los actuales problemas y circunstancias.

Requerimientos del negocio

La suma de todos los conocimientos adquiridos mediante las diversas herramientas tales como las entrevistas a los clientes o usuarios o los cuestionarios, las sesiones JAD, el análisis de brecha, el análisis FODA y otras reuniones, ayudan a desarrollar una mejor perspectiva sobre los requerimientos y ayuda en la creación del Backlog Priorizado del Producto.

Resultados de la evaluación/benchmarking

Los resultados de la evaluación/benchmarking ayudan a establecer un estándar mínimo al momento de crear un producto o servicio y conllevan a cambios en los criterios de terminado. También pueden ayudar a establecer un estándar mínimo al crear un producto o servicio y llevar a criterios de terminado modificados. En ocasiones pueden también proporcionar el impulso para que un Program Product Owner o Portfolio Product Owner desarrollen nuevas historias de usuario para implementar las mejores prácticas.

Retorno sobre la inversión (RSI)

El retorno sobre la inversión (RSI), al utilizarse para la justificación de un proyecto, evalúa los ingresos netos esperados que se buscan obtener a partir de un proyecto. Se calcula deduciendo los costos esperados o la inversión en un proyecto de su ingreso previsto; después se divide (la utilidad neta) por los costos previstos a fin de obtener la tasa de retorno.

Retrospect Sprint Log(s)

Los registros de la retrospectiva del sprint, o Retrospect Sprint Log(s), son registros de las opiniones, debates y elementos accionables planteados en la reunión de retrospectiva del sprint. El Scrum Master puede facilitar la creación de dicho registro con la aportación de los miembros del equipo principal de Scrum.

Retrospectiva del proyecto

En este proceso, mismo que concluye el proyecto, los stakeholders y miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones aprendidas. A menudo, estas lecciones llevan a la documentación de Agreed Actionable Improvements, que se implementarán en futuros proyectos.

Retrospectiva del sprint

En este proceso el Scrum Master y el Equipo Scrum se reúnen para analizar las lecciones aprendidas a lo largo del Sprint. Dichas lecciones se documentan y se pueden aplicar a futuros sprints.

Reunión de plan del ambiente (*Environment Plan Meeting*)

Para definir un programa/calendario de cómo los equipos Scrum compartirán ambientes, se utiliza una reunión de plan de ambiente o *Environment Plan Meeting*.

Reunión de planificación de tareas

En una reunión de planificación de tareas, el Equipo Scrum se reúne para planear el trabajo que se hará en el sprint y revisar las historias de usuario asignadas en la parte superior del Backlog Priorizado del Producto. Para ayudar a garantizar que el grupo no se salga del tema, la reunión debe de tener un time-box con una duración estándar limitada a dos horas por semana de duración del sprint.

Reunión de planificación del sprint

Esta reunión se lleva a cabo antes del sprint, como parte de los procesos de *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y *Crear el Sprint Backlog*. Se asigna a un time-box de ocho horas durante un sprint de un mes de duración. La reunión de planificación del sprint se divide en dos partes: Definición del objetivo e Identificación y estimación de tareas.

Reunión de retrospectiva del programa o portafolio

La reunión de retrospectiva del programa o portafolio es similar a la reunión de retrospectiva del proyecto, pero se lleva a cabo al nivel del programa o portafolio. La principal diferencia es que la frecuencia de las reuniones de retrospectiva del programa o portafolio es menor que las reuniones de retrospectiva del proyecto.

Reunión de retrospectiva del proyecto

La reunión de retrospectiva del proyecto es una reunión para determinar las formas en las que la colaboración y eficacia del equipo puede mejorarse en futuros proyectos. También se analizan las oportunidades positivas, negativas y potenciales para mejorar. Esta reunión no tiene un time-box asignado y se puede llevar a cabo en forma presencial o en formato virtual.

Reunión de retrospectiva del sprint

Esta reunión tiene un time-box de cuatro horas en un sprint de un mes, y se lleva a cabo como parte del proceso *Retrospectiva del sprint*. Durante esta reunión, el Equipo Scrum se reúne para revisar y reflexionar sobre el sprint anterior en relación a los procesos que se siguieron, las herramientas empleadas, la colaboración y los mecanismos de comunicación, así como otros aspectos de interés para el proyecto.

Reunión de revisión del Backlog Priorizado del Producto

Una reunión de revisión del Backlog Priorizado del Producto (*Prioritized Product Backlog Grooming Session*) es una reunión formal que ayuda al Equipo Scrum a revisar y a obtener un consenso sobre dicho backlog.

Reunión de revisión del sprint

La reunión de revisión del sprint tiene un time-box de cuatro horas en un sprint de un mes o puede escalarse según la duración del sprint. Durante la reunión de revisión del sprint, el Equipo Scrum presente los entregables del actual sprint al Product Owner quien puede aceptarlos o rechazarlos.

Reunión de riesgos

Los riesgos de pueden priorizar con facilidad por el Product Owner convocando a una reunión del equipo principal de Scrum con la opción de invitar a los stakeholders relevantes a dicha reunión.

Reunión de Scrum de Scrums

Una reunión de Scrum de Scrums es un elemento importante en el escalamiento de Scrum para grandes proyectos. En la reunión típicamente hay un representante de cada Equipo Scrum, por lo general el Scrum Master, aunque también es común que cualquiera en el Equipo Scrum asista a la reunión en caso de ser necesario. Esta reunión generalmente la organiza el Chief Scrum Master y la intención es enfocarse en áreas de coordinación e integración entre los distintos equipos Scrum.

Reunión de visión del proyecto

Una reunión de visión del proyecto es una reunión con el (los) stakeholder(s) del programa, el Program Product Owner, el Program Scrum Master y el Chief Product Owner. Ayuda a identificar el contexto empresarial, los requerimientos del negocio y las expectativas de los stakeholders a fin de desarrollar una eficaz declaración de la visión del proyecto.

Reuniones de grupo de enfoque

En las reuniones del grupo de enfoque las personas se reúnen en una sesión guiada para ofrecer sus opiniones, percepciones o valoraciones sobre un producto, servicio o resultado deseado. Los miembros del grupo de enfoque tienen la libertad de hacerse preguntas de obtener aclaraciones sobre temas o conceptos específicos. Mediante el cuestionamiento, la crítica constructiva y la retroalimentación, los grupos de enfoque conducen a un producto de mejor calidad y con ello contribuyen a la satisfacción de las expectativas de los usuarios.

Reuniones de grupo de usuarios

Las reuniones del grupo de usuarios incluyen a stakeholders relevantes (principalmente usuarios o clientes del producto). Estos brindan al equipo principal de Scrum información de primera mano sobre las expectativas del usuario. Esto ayuda a la formulación de los criterios de aceptación para el producto y proporciona información valiosa para el desarrollo de épicas.

Reuniones de revisión del Backlog Priorizado del Producto

Una reunión de revisión del Backlog Priorizado del Producto (conocida también como sesión de refinamiento del Backlog Priorizado del Producto) es una reunión formal durante el proceso de *Refinar el Backlog Priorizado del Producto* que ayuda al Equipo Scrum a revisar y obtener un consenso sobre dicho backlog.

Reuniones del Scrum Guidance Body

El Scrum Guidance Body se reúne con frecuencia para debatir sobre la posible necesidad de actualizar las recomendaciones de dicho órgano (por ejemplo: recomendaciones mejoradas de las retrospectivas y demás procesos, actualización de regulaciones, etc.). La frecuencia de estas reuniones la decide el Scrum Guidance Body con base en las necesidades específicas de la empresa.

Riesgo

El riesgo se define como un evento incierto o una serie de eventos que pueden afectar los objetivos de un proyecto pudieran contribuir a su éxito o fracaso.

Riesgos del programa y el portafolio

Riesgos relacionados a un portafolio o a un programa que también habrán de afectar los proyectos que forman parte del respectivo portafolio o programa.

Riesgos mitigados

Los riesgos mitigados son aquellos que se tratan o mitigan con éxito por el Equipo Scrum durante el proyecto.

Risk Burndown Chart

Conocida en inglés como *Risk Burndown Chart*, es una gráfica que muestra la gravedad cumulativa del riesgo a lo largo del tiempo. Las probabilidades de los diversos riesgos se trazan una sobre otra para mostrar el riesgo acumulativo en el eje vertical. La identificación y evaluación inicial de los riesgos en el proyecto, así como la creación de el Risk Burndown Chart se realizan al inicio.

Ritmo sostenible (*Sustainable Pace*)

El ritmo sostenible es el ritmo al cual el equipo puede trabajar y sostener cómodamente. Esto se traduce en una mayor satisfacción del empleado, en estabilidad y una mayor precisión en la estimación; todo ello conlleva, en última instancia, a un aumento en la satisfacción del cliente.

Rol no central

Los roles no centrales son los que no son necesariamente obligatorios para el proyecto Scrum, y estos pueden incluir a miembros de los equipos que estén interesados en el proyecto. No tienen ningún rol formal en el equipo del proyecto, y pueden interactuar con el equipo, pero pueden no ser responsables del éxito del proyecto. Los roles no centrales deben tenerse en cuenta en cualquier proyecto de Scrum.

Roles centrales

Los roles centrales son aquellos que se requieren obligadamente para crear el producto del proyecto, están comprometidos con el proyecto, y por último son los responsables del éxito de cada sprint del proyecto y del proyecto en su totalidad.

Sesiones de planificación del lanzamiento (*Release Planning Sessions*)

El objetivo principal de una sesión de planificación de lanzamiento es hacer que el Equipo Scrum cuente con una visión general de los lanzamientos y del calendario de entrega del producto que están desarrollando para que puedan alinearse con las expectativas del Product Owner y los stakeholders relevantes.

Sesiones JAD (*Joint Application Design*)

La sesión Joint Application Design (JAD), o sesión de diseño de aplicación conjunta, es una técnica de recopilación de requisitos. Es taller impartido y altamente estructurado que acelera el proceso de *Crear la visión del proyecto*, ya que permite al(los) stakeholder(s) y a otras personas que toman decisiones llegar a un consenso sobre el alcance, los objetivos y otras especificaciones del proyecto.

Scrumboard

El Scrumboard es una herramienta utilizada por el Equipo Scrum para planificar y dar seguimiento al proceso durante cada sprint. El tablero contiene cuatro columnas para indicar el progreso de las tareas estimadas para el sprint: una columna “por hacer” para las tareas que aún no se inician; una columna “en progreso” para las tareas que ya iniciaron, pero no se han concluido; una columna “en prueba” para las tareas concluidas pero que están en proceso de evaluación y una columna de “terminado” para las tareas que se han concluido y evaluado satisfactoriamente.

Scrum Guidance Body

El Scrum Guidance Body (SGB, por sus siglas en inglés), es un rol opcional que generalmente consiste en un conjunto de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de los objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros

parámetros claves de la organización. El SGB guía el trabajo llevado a cabo por el Product Owner, el Scrum Master y el Equipo Scrum.

Scrum Master

El Scrum Master es uno de los roles en el equipo principal de Scrum. Él o ella facilita la creación de entregables del proyecto, gestiona riesgos, cambios e impedimentos durante el proceso de llevar a cabo el Daily Standup, la retrospectiva del sprint y demás procesos de Scrum.

Solicitud de cambio

Las peticiones para realizar cambios generalmente se presentan como solicitudes de cambio. Dichas solicitudes permanecen como no aprobadas has que se autorizan formalmente.

Solicitudes de cambio aprobadas

Las solicitudes de cambios aprobadas son los cambios que han sido aprobados para incluirse en el Backlog Priorizado del Producto. En ocasiones, las solicitudes de cambio aprobadas pueden presentarlas los administradores del programa o portafolio, y serían entradas que se añadirán a la lista de cambios autorizados del proyecto para su implementación en futuros sprints.

Solicitudes de cambio no aprobadas

Las solicitudes para hacer cambios se presentan por lo general como solicitudes de cambio o *Change Requests*. Las solicitudes de cambio permanecen como no aprobadas hasta que se obtiene una aprobación formal.

Speed Boat

El Speed Boat es una técnica que se puede utilizar para llevar a cabo la reunión de retrospectiva del sprint. Los miembros juegan el rol de la tripulación en una lancha. La lancha debe llegar a una isla: simbólicamente la visión del proyecto. Los asistentes utilizan notas adhesivas para llevar un registro de motores y anclas. Los “motores” ayudan a llegar a la isla, mientras que las “anclas” son cosas que están obstaculizando la llegada. Este ejercicio tiene un time-box de unos cuantos minutos.

Sprint

Un sprint es una iteración con un time-box de una a seis semanas de duración durante el cual el Equipo Scrum crea y trabaja en los entregables del sprint.

Sprint Backlog

El Sprint Backlog es una lista de tareas a ser ejecutadas por el Equipo Scrum en el próximo sprint.

Sprint Burndown Chart

El Sprint Burndown Chart es una gráfica que muestra la cantidad de trabajo pendiente en el actual sprint.

Sprint de preparación del lanzamiento (*Release Readiness Sprint*)

Si hay necesidad de llevar a cabo tareas específicas para preparar un lanzamiento y confirmar que se hayan cumplido los requerimientos mínimos para el lanzamiento, dichas tareas se llevan a cabo en un sprint de preparación del lanzamiento (*Release Readiness Sprint*). De ser necesario, dicho sprint solo se hace una vez por cada lanzamiento como último sprint previo al lanzamiento.

Stakeholder(s)

Stakeholder es un término colectivo que incluye a clientes, usuarios y patrocinadores que interactúan frecuentemente con el Product Owner, con el Scrum Master y con el Equipo Scrum para brindar opiniones y facilitar la creación del producto del proyecto, servicio u otros resultados.

Taller de estimación de tareas

El taller de estimación de tareas permite al Equipo Scrum estimar el esfuerzo requerido para terminar una tarea o serie de tareas y estimar el trabajo necesario del personal para realizar las tareas dentro de un determinado sprint.

Talleres de historias de usuario

Los talleres de historias de usuarios se llevan a cabo como parte del proceso de *Desarrollar épica(s)*. El Scrum Master facilita estas sesiones, donde el equipo principal Scrum interviene y, en ocasiones, se recomienda incluir a otro stakeholders.

Tamaño relativo/Puntos de historia (*Relative Sizing/Story Points*)

Además de utilizarse para la estimación de costos, los puntos de historia (*Story Points*), también pueden utilizarse para calcular el tamaño total de una historia de usuario o de una característica. Este método asigna un punto de historia con base en una evaluación general del tamaño de la historia de usuario tomando en cuenta el riesgo, la cantidad requerida de esfuerzos y el nivel de complejidad.

Tasa interna de retorno (TIR)

La tasa interna de retorno (TIR), es una tasa de descuento sobre una inversión en la cual el valor actual de los flujos de efectivo se iguala al valor actual del flujo de salidas de efectivo a fin de evaluar la tasa de retorno del proyecto. Al hacer un comparativo de los proyectos, generalmente resulta mejor aquel que cuenta con una tasa interna de retorno más elevada.

Técnica Wideband Delphi

El Wideband Delphi es una técnica de estimación basada en grupo para determinar la cantidad de trabajo necesario y el tiempo que tardará en completarse. Los individuos en el equipo proporcionan estimaciones anónimas para cada artículo y las estimaciones iniciales se trazan en una gráfica. Posteriormente, el equipo analiza los factores que influyeron en sus estimaciones y proceden a una segunda ronda de estimación. Este proceso se repite hasta que las estimaciones de los individuos quedan cerca una de la otra y se puede llegar a un consenso para la estimación final.

Teoría X

En la Teoría X, los líderes asumen que los empleados están inherentemente desmotivados y evitarán el trabajo si es posible, lo cual justifica un estilo autoritario de gestión.

Teoría Y

En la Teoría Y, los líderes asumen que los empleados son auto motivados y buscan aceptar una mayor responsabilidad. La Teoría Y implica un estilo más participativo de gestión.

Time-boxing

El Time-boxing es la fijación de breves periodos de periodos para hacer el trabajo. Si el trabajo realizado queda incompleto al final del time-box, este se asigna a un nuevo time-box. El asignar bloques de tiempo fijo proporciona la estructura necesaria para los proyectos Scrum, los cuales tienen un elemento de incertidumbre, son dinámicos por naturaleza y propensos a cambios frecuentes.

Tolerancia de riesgo

La tolerancia de riesgo indica el grado, cantidad o volumen de riesgo que resistirán los stakeholders.

Transparencia

La transparencia permite que todas las facetas de cualquier proceso de Scrum sean observadas por cualquiera. Compartir toda la información conduce a un ambiente de alta confianza.

Tres preguntas diarias

Las tres preguntas diarias se utilizan en los Daily Standups, organizados por el Scrum Master, donde cada miembro del Equipo Scrum brinda información en forma de respuesta a tres preguntas específicas:

1. ¿Qué he hecho desde la última reunión?
2. ¿Qué tengo planeado hacer antes de la siguiente reunión?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Umbral de riesgo (*Risk Threshold*)

El umbral de riesgo es el nivel al cual es riesgo aceptable para la organización del stakeholder. Un riesgo caerá por encima o por debajo del umbral de riesgo. Si está por debajo, es más probable que el stakeholder o la organización acepte el riesgo.

Usuario

Los usuarios son los individuos o la organización que utiliza directamente el producto del proyecto, el servicio u otros resultados. Al igual que los clientes, para cualquier organización, puede haber tanto usuarios internos y como externos. En algunos casos, los clientes y los usuarios pueden ser los mismos.

Valor monetario esperado (*Expected Monetary Value*)

El valor monetario del riesgo está basado en su valor monetario esperado (EMV, por sus siglas en inglés). Dicho valor se calcula multiplicando el impacto monetario por la probabilidad de riesgo, según la aproximación del cliente.

Valor presente neto (VPN)

El valor presente neto (VPN) es un método que se utiliza para determinar el valor neto actual de un futuro beneficio económico, dada una inflación o tasa de interés prevista.

Velocidad del sprint

La velocidad del sprint es la velocidad en la que el equipo puede completar el trabajo en un sprint. Por lo general se expresa en las mismas unidades que se utilizan para la estimación, normalmente puntos de historia o tiempo ideal.

Vendedor

Los vendedores son individuos externos u organizaciones que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.

Visión de la empresa

Entender la visión de la empresa ayuda a que el proyecto mantenga su enfoque en los objetivos de la organización y el futuro probable de la empresa. El Product Owner puede guiarse por la visión de la compañía para crear la declaración de la visión del proyecto.

Voz del cliente (*Voice of the Customer*)

La voz del cliente puede describirse como los requerimientos explícitos e implícitos del cliente, los cuales deben entenderes antes del diseño de un producto o servicio. El Product Owner representa la voz del cliente.

War Room

Salón de guerra, o *War Room* en inglés, es el término que comúnmente se utiliza para describir la ubicación donde trabajan todos los miembros del Equipo Scrum. Normalmente está diseñado de tal manera que los miembros del equipo puedan moverse libremente, trabajar y comunicarse fácilmente, ya que se encuentran ubicados en proximidades inmediatas.

ÍNDICE TEMÁTICO

1

100-point method, 178

A

Acta constitutiva del proyecto, 152
Acuerdo de entregables funcionales, 275
Adaptabilidad, 4
Adaptación, 26
Alta velocidad, 5
Ambiente de alta confianza, 4
Ambiente innovador, 5
Análisis de brecha, 150
Apetito de riesgo, 125
Appropriation, 31
Aprobación, estimación y asignación de historias de usuario, 18, 190
Aprobación, estimación y asignación de historias de usuarios
  Entradas, 202
  Herramientas, 202
  Salidas, 205
Aprobadas de cambio, 105
Articulation, 31
Asesoramiento de expertos RH, 157
Asignación de un bloque de tiempo, 10, 24
Aspectos de Scrum, 11
Autocrático, 65
Auto-organización, 10, 23
Awareness, 31

B

Brainstorming, 127

C

Calidad, 14
Calidad y valor del negocio, 91
Cambio, 14
Capacitación y costos de capacitación, 157
Cartera, 57
Caso de negocio del proyecto, 147
Centrado en el cliente, 4
Clientes meta para el lanzamiento, 186
Colaboración, 10, 23
Colocated Teams, 33
Confirmar la realización de beneficios, 85
Contratos aplicables, 168
Control del proceso empírico, 10, 23, 24
Costo actual, 81
Costo de los recursos, 158
Creación de entregables
  Entradas, 233
  Herramientas, 235
  Salidas, 236
Creación de entregables, 19, 228, 231
Creación de historias de usuario
  Herramientas, 196
  Salidas, 198
Creación de historias de usuario, 18, 190, 193
  Entradas, 195
Creación de la lista de pendientes del sprint
  Entradas, 217
  Herramientas, 218
  Salidas, 219
Creación de la lista de pendientes del sprint, 19, 215
Creación de la lista de pendientes del Sprint, 190
Creación de la lista priorizada de pendientes del producto
  Entradas, 176
  outputs, 179
Creación de la lista priorizada de pendientes del producto, 18, 142, 174
Creación de la visión del proyecto
  Entradas, 147
  Salidas, 151
Creación de la visión del proyecto, 18, 142, 145
Creación de la visión del proyecto, 149
Creación de la visión del proyecto, 153
Creación de tareas

Herramientas, 208
Salidas, 209
Creación de tareas, 19, 190, 206
Create Prioritized Product Backlog
Herramientas, 177
Create Project Vision, 327
Criterios de aceptación de historias de usuario, 198
Criterios de estimación, 212
Criterios de selección, 157
Criterios de terminado, 180
Cronograma de planificación del lanzamiento, 185
Cronograma de planificación del lanzamiento actualizado, 248
Cuestionarios, 171

D

Data Flow Diagram
Initiate phase, 187
Demostración y validación del sprint
Entradas, 257
Herramientas, 258
Salidas, 259
Demostración y validación del sprint, 19, 252, 255
Demostraciones, 85
Dependencias, 180
Dependencias discrecionales, 213
Dependencias externas, 213
Dependencias internas, 213
Dependencias obligatorias, 213
Desarrollo de épica(s), 18, 142, 165
Entradas, 167
Herramientas, 170
Salidas, 172
Desarrollo iterativo, 24, 39, 108
Desarrollo Iterativo, 10
Desempeño, 62, 377
Disponibilidad y compromiso de las personas, 156
Duración del sprint, 114, 185

E

El equipo principal de Scrum, 109
Enfrentamiento, 62
Entrega basada en el valor, 69
Entrega basada en valor, 70
Entrega continúa de valor, 4
Entregables aceptados, 259
Entregables del sprint, 236
Entregables efectivos, 4
Entregables rechazados, 246
Entrevistas al usuario o cliente, 171
Envío de entregables
Entradas, 273
Herramientas, 274
Salidas, 275
Envío de entregables, 20, 268, 271
Épica (s), 172
Equilibrio, 106
Equipo principal Scrum, 167
Equipo Scrum, 54
Equipo Scrum motivado, 243
Escalabilidad de Scrum, 5
Escalas de tiempo del proyecto, 74
Estilos de liderazgo, 65
Estimación de tareas
Herramientas, 212
Salidas, 213
Estimación de tareas, 19, 190, 210
Estimaciones, 180
Estimado al completar, 81
Estimado para completar, 82
Evaluación de riesgos, 127
Experiencia del cuerpo de asesoramiento de Scrum, 171, 236, 258, 264, 279
Experiencia en la redacción de historias de usuario, 196
Expert Advice from HR, 162

F

Fase de implementación—Diagrama de flujo de datos, 250
Fase de lanzamiento—Diagrama de flujo de datos, 282
Fase de planificación y estimación—Diagrama de flujo de datos, 225
Fase de revisión y retrospectiva—Diagrama de flujo de datos, 266
Formación, 62
Formación de equipos Scrum, 18

Formación de un equipo Scrum
Entradas, 161
Herramientas, 162
outputs, 163
Formación de un equipo Scrum, 142
Formación de un equipo Scrum., 159

G

Ganar-Ganar, 63
Ganar-Perder, 64
Gestión de conflictos, 63

H

Historias de usuario, 202

I

Identificación de riesgos, 126
Identificación del Scrum Master y el(los) socio(s), 18, 142
Identificar al Scrum Master y al socio(s), 153
Entradas, 155
Herramientas, 157
Salidas, 158
Identified Scrum Team, 163
Implementación, 19
Índice de desempeño del programa, 81
Información de proyectos previos, 169
Inicio, 18, 141
Inspección, 26
Integración continua, 112
Integración del cambio, 112

J

Justificación del negocio, 13

L

Lanzamiento, 20
Lanzamientos del producto, 275
Lecciones aprendidas del equipo Scrum, 265
Leyes y regulaciones, 168
Líder asertivo, 65
Líder de apoyo y entrenamiento, 65
Líder delegador, 65
Líder directivo, 65
Líder *Laissez Faire*, 65
Líder servicial, 65
Lista priorizada de pendientes del producto, 179
Lista priorizada de pendientes del producto actualizada, 246
Listas de verificación de riesgos, 127

M

Mantenimiento de la lista priorizada de pendientes del producto
Entradas, 246
Herramientas, 247
Salidas, 248
Mantenimiento de la lista priorizada de pendientes del producto, 19, 228, 244
Mantenimiento de la participación de los socios, 59
Matriz de recursos organizacionales, 156
Matriz de requerimientos de habilidades, 156, 384
Mejora continua, 4
Mejoras accionables aceptadas, 264
Métodos de priorización del lanzamiento, 185
Modelo de dinámica de grupo de Tuckman, 62
Motivación, 4

N

Normalización, 62

O

Organización, 11

P

Paired comparison, 177
Patrocinador, 47
People Costs, 162
Perder-Ganar, 64
Perder-Perder, 64
Plan de colaboración, 164
Plan de desarrollo del equipo, 164
Plan de pilotaje, 274
Planificar para el valor, 78
Porcentaje completado, 81
Presupuesto al finalizar, 81

Presupuesto del proyecto, 152
Priorización basada en el valor, 24
Priorización basada en el valor que se entrega al cliente, 111
Priorización basada en el valor que se entrega cliente, 78
Priorización basada en valor, 10, 33
priorizada de pendientes del producto refinada, 186
Procedimiento de gestión de riesgos, 126
Proceso de aprobación de cambios, 105
Proceso de desarrollo eficiente, 4
Procesos de Scrum, 16
producto, 1
Programa, 57
Propietario del producto del programa, 147
Propietario del producto identificado, 151
Prototipos, 172
Proyecto, 57
Proyecto de prueba, 148
Prueba de concepto, 148

R

Realizar el plan de lanzamiento
Entradas, 183
Herramientas, 184
Salidas, 185
Realizar el plan de lanzamiento, 142, 181
Realizar la planificación del lanzamiento, 18
Realizar reunión diaria de pie
Entradas, 240
Herramientas, 241
Salidas, 242
Realizar reunión diaria de pie, 19, 228, 238
Recomendaciones de cuerpo de asesoramiento de Scrum, 196
Recomendaciones del cuerpo de asesoramiento de Scrum, 149, 169, 177, 184, 202, 234, 247, 262, 274, 278
Relative prioritization ranking, 80
Requerida, 79, 364
Requerimientos de las personas, 155
Requerimientos de negocio, 176
Requisitos de recursos, 162
Resolución de problemas de forma más rápida, 4
Resource Costs, 163
Responsabilidad colectiva, 4
Retroalimentación continua, 4
Retrospectiva de sprint, 252
Retrospectiva de Sprint, 20
Retrospectiva del proyecto
Herramientas, 279
inputs, 278
Salidas, 280
Retrospectiva del proyecto, 20, 268, 276
Retrospectiva del sprint
Entradas, 262
Herramientas, 262
Salidas, 264
Retrospectiva del sprint, 260
Reunión de la visión del proyecto, 149, 151
Reunión de planificación del sprint, 37
Reunión de retrospectiva del sprint, 37
Reunión de revisión del sprint, 37, 258
Reunión de riesgos, 128
Reunión diaria, 36
Reunión diaria de pie, 241
Reuniones de estimación de tareas, 212
Reuniones de planificación de tareas, 208
Reuniones de planificación del sprint, 218
Reuniones del grupo de usuarios, 170
Revisión y retrospectiva, 19
Riesgo, 15
Riesgo e incertidumbre, 180
Riesgos del programa y cartera, 168
Riesgos identificados, 173
Ritmo sostenible, 4
Ritmo Sostenible, 98
Roles centrales, 11
Roles no centrales, 12

S

Scrum Developer Certified (SDC™), 6
Scrum Guidance Body Expertise, 179
Scrum Master, 51
Scrum Master Certified (SMC™), 6
Scrum Master del programa, 147
Scrum Master identificado, 158
Scrum Product Owner Certified (SPOC™), 6

Scrum Team Selection, 162
Scrum vs. Gestión de proyectos tradicional, 22
Selección de personal, 56
Sesiones de diseño de aplicación conjunta, 150
Sesiones de planificación del lanzamiento, 184
Socio(s) del programa, 147
Socio(s) identificado(s), 158
Software, 235
Solicitudes de cambio, 167
Solicitudes de cambio no aprobadas, 167
Sprint, 36
Substitutos, 164

T

Tablero de Scrum, 233
Talleres de historias de usuario, 170
Técnicas de comunicación, 248
Teoría X, 68
Teoría Y, 68
Transparencia, 4, 24

U

User Story Estimation Methods, 179
Usuarios, 47

V

Valor, 179
Valor ganado, 81
Valor planeado, 81
Value Stream Mapping, 78
Variancia al completar, 82
Varianza del costo, 81
Vendedores, 47
Videoconferencia, 242
Visión de la compañía, 148

La guía esencial para la entrega exitosa de proyectos con el uso de Scrum.

El *SBOK™* fue desarrollado como medio para crear una guía necesaria para organizaciones y profesionistas que buscan implementar Scrum, así como para aquellos que ya lo hacen y que buscan hacer las mejoras necesarias en sus procesos existentes. Está basada en la experiencia obtenida de miles de proyectos en una variedad de industrias y organizaciones. En su desarrollo se tomó en cuenta la aportación de muchos expertos en Scrum y profesionales en la entrega de proyectos. El enfoque de Scrum en la entrega guiada por el valor, ayuda a los equipos Scrum a presentar resultados tan temprano como sea posible en el proyecto, mejorando así el retorno sobre la inversión en empresas que implementan Scrum como su framework preferencial para la entrega de proyectos. Asimismo, la gestión de cambios se facilita mediante el uso de breves ciclos iterativos de desarrollo de productos y la frecuente interacción entre los clientes y los equipos Scrum.

La *Guía SBOK™* puede ser utilizada como referencia y guía de conocimiento tanto por practicantes de Scrum con experiencia u otros practicantes en el desarrollo de productos o servicios, así como individuos sin conocimiento o experiencia previa en Scrum u otro método para la entrega de proyectos. El primer capítulo describe el propósito y el framework de la *Guía SBOK™*; ofrece una introducción de los conceptos clave de Scrum y un resumen de sus principios, aspectos y procesos. Los capítulos del 3 al 7 abundan sobre los aspectos de Scrum que deben abordarse a lo largo de cualquier proyecto: organización, justificación del negocio, calidad, cambio y riesgo. Los capítulos del 8 al 12 cubren los 19 procesos fundamentales que se llevan a cabo en un proyecto Scrum. Dichos procesos forman parte de las cinco fases de Scrum: Inicio, Planificación y estimación, Implementación, Revisión y retrospectiva y Lanzamiento. En dichos capítulos se describen los detalles sobre las entradas y salidas relacionadas a cada proceso, así como las varias herramientas que se pueden utilizar en cada una.

Esta tercera edición de la *Guía SBOK™* se suma al conocimiento colectivo del framework de Scrum con un contenido ampliado relacionado al escalamiento de Scrum en grandes proyectos, así como el escalamiento de Scrum para la empresa, lo cual se aborda en los capítulos 13 y 14, respectivamente.

Aunque la *Guía SBOK™* es un libro integral de consulta sobre Scrum, sus contenidos están organizados para una consulta fácil y para disfrutar de su lectura, independientemente del conocimiento previo que tenga el lector sobre el tema.

SCRUMstudy
Targeting success

SCRUMstudy es una marca de VMEdu, Inc.