

Guía de los
FUNDAMENTOS DE SCRUM
(GUÍA DEL SBOK®)

Cuarta Edición

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

Incluye dos capítulos sobre el escalamiento de Scrum
en grandes proyectos y para la empresa

Guía integral para el desarrollo de proyectos por medio de Scrum



Guía de los

FUNDAMENTOS DE SCRUM

(Guía del SBOK®)

Cuarta Edición

**Incluye dos capítulos sobre el escalamiento de Scrum
en grandes proyectos y para la empresa**

Guía integral para el desarrollo de proyectos por medio de Scrum

© 2022 SCRUMstudy™, una marca de VMEdU, Inc. Todos los derechos reservados.

Información del catálogo de publicación de la Biblioteca del Congreso

Guía de los fundamentos de Scrum (Guía del SBOK®) – Cuarta edición

Título original en inglés: *A Guide to the Scrum Body of Knowledge (SBOK® Guide)* – Fourth edition

Incluye referencia bibliográfica e índice.

ISBN: 978-098992520-4

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; sbok@scrumstudy.com

Sitio web: www.scrumstudy.com

“SBOK”, el logotipo de SCRUMstudy, “SFC” “SDC”, “SMC”, “SAMC”, “SPOC”, “SSMC”, “SSPOC” 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 *Guía de los fundamentos de Scrum (Guía del 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: support@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 de los fundamentos de Scrum (Guía del SBOK®)* proporciona los lineamientos para la aplicación exitosa de Scrum: el desarrollo ágil de productos y el método más popular de entrega de proyectos. Scrum, tal como se define en la *Guía del SBOK®*, es un marco de trabajo 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 del SBOK®* está diseñada para utilizarse como referencia y guía de fundamentos tanto por profesionales con experiencia en Scrum y demás practicantes de desarrollo de productos o servicios, como por personas que no tengan experiencia o conocimiento de Scrum o cualquier otro método de entrega de proyectos. Esta nueva edición de la *Guía del SBOK®* brinda información adicional sobre las mejores prácticas de Scrum, particularmente en las áreas para escalar en Scrum. A medida que aumenta la popularidad y aplicación del marco de trabajo de Scrum y evoluciona a nivel global, nuestra meta es compartir las lecciones aprendidas y las mejores prácticas como parte de la *Guía del SBOK®*.

La *Guía del SBOK®* se basa en el conocimiento y la visión conjunta obtenida de miles de proyectos a través de una variedad de organizaciones e industrias. Esta cuarta edición se suma a la aportación colectiva de expertos en Scrum y en el desarrollo de proyectos. En forma particular, la retroalimentación de la comunidad mundial de Scrum jugó un papel importante en la identificación de mejoras y adiciones a la *Guía del SBOK®*^T. 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 marco de trabajo de la *Guía del 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 del SBOK®* es un marco de trabajo 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 de la *Guía del 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	El objetivo de la <i>Guía del SBOK®</i>	5
1.4	Marco de trabajo de la <i>Guía del SBOK®</i>	6
1.4.1	¿Cómo se utiliza la <i>Guía del SBOK®</i> ?	7
1.4.2	Principios de Scrum.....	8
1.4.3	Aspectos de Scrum.....	10
1.4.4	Procesos de Scrum.....	14
1.4.5	Scrum para grandes proyectos	18
1.4.6	Scrum para la empresa.....	18
1.5	Scrum vs. Gestión tradicional de proyectos	19
2.	PRINCIPIOS.....	21
2.1	Introducción	21
2.2	Guía de roles	22
2.3	Control de proceso empírico.....	22
2.3.1	Transparencia.....	22
2.3.2	Inspección.....	24
2.3.3	Adaptación	25
2.4	Autoorganización.....	27
2.4.1	Beneficios de la autoorganización	28
2.5	Colaboración	29
2.5.1	Beneficios de la colaboración en los proyectos de Scrum.....	30
2.5.2	La importancia de la coubicación en la colaboración.....	31
2.5.3	Colaboración en equipos dispersos	32
2.6	Priorización basada en valor	33
2.7	Time-boxing.....	35
2.7.1	Time-boxes de Scrum.....	35
2.8	Desarrollo iterativo.....	37

2.9	Scrum vs. Gestión tradicional de proyectos	40
3.	ORGANIZACIÓN.....	41
3.1	Introducción	41
3.2	Guía de roles	42
3.3	Roles en un proyecto Scrum	42
3.3.1	Roles principales.....	42
3.3.2	Roles secundarios	43
3.4	Product Owner.....	46
3.4.1	La voz del cliente	48
3.5	Scrum Master	48
3.6	Equipo Scrum	50
3.6.1	Selección de personal.....	51
3.6.2	Tamaño del Equipo Scrum	51
3.7	Scrum en proyectos, programas y portafolios	52
3.7.1	El funcionamiento de Scrum en un proyecto grande	52
3.7.2	Roles principales adicionales en proyectos grandes	53
3.7.3	El funcionamiento de Scrum en una empresa	55
3.7.4	Roles principales adicionales en una empresa.....	56
3.7.5	Ejemplos de proyectos, programas y portafolios	58
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 en Scrum	62
3.10.1	Modelo de dinámica de grupo de Tuckman	62
3.10.2	Gestión de conflictos.....	63
3.10.3	Técnicas de gestión de conflictos	63
3.10.4	Estilos de liderazgo.....	64
3.10.5	Teoría de jerarquía de necesidades de Maslow	66
3.10.6	Teoría X, Teoría Y, Teoría Z.....	67
4.	JUSTIFICACIÓN DEL NEGOCIO	68
4.1	Introducción	68
4.2	Guía de roles	69
4.3	Entrega basada en valor.....	69
4.3.1	Responsabilidades del Product Owner en la justificación del negocio	70

4.3.2	Responsabilidades de otros roles de Scrum en la justificación del negocio	71
4.4	La importancia de la justificación del negocio	72
4.4.1	Factores para determinar la justificación del negocio	72
4.4.2	La justificación del negocio y el ciclo de vida del proyecto	73
4.5	Técnicas de justificación del negocio	74
4.5.1	Estimación del valor del proyecto	75
4.5.2	Planificar según el valor	77
4.5.3	Clasificación de priorización relativa	79
4.5.4	Mapeo de historias	80
4.6	Justificación continua de valor	81
4.6.1	Análisis del valor ganado (EVA)	81
4.6.2	Diagrama de flujo acumulado (CFD)	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	88
5.1	Introducción	88
5.2	Guía de roles	89
5.3	Definición de calidad	89
5.3.1	Calidad y alcance	89
5.3.2	Calidad y valor del negocio	90
5.4	Criterios de aceptación y backlog priorizado del producto	91
5.4.1	Redacción de criterios de aceptación	92
5.4.2	Definición de listo	93
5.4.3	Definición de terminado (o criterios de terminado)	93
5.4.4	Criterios mínimos de terminado	94
5.4.5	Aceptación o rechazo de elementos del backlog priorizado del producto	95
5.5	Gestión de calidad en Scrum	95
5.5.1	Planificación de calidad	96
5.5.2	Garantía de calidad y control de calidad	97
5.5.3	Ciclo de planificar, hacer, verificar y actuar (PDCA)	99
5.6	Resumen de responsabilidades	100

5.7	Scrum vs. Gestión tradicional de proyectos	102
6.	CAMBIO	103
6.1	Introducción	103
6.2	Guía de roles	104
6.3	Información general	104
6.3.1	Solicitudes de cambio aprobadas y no aprobadas	105
6.4	Cómo entender el cambio en Scrum	106
6.4.1	El equilibrio entre la flexibilidad y la estabilidad	106
6.4.2	Cómo incorporar la flexibilidad.....	107
6.5	Integración del cambio.....	112
6.5.1	Cambios a un sprint.....	112
6.6	Cambio en portafolios y programas.....	117
6.6.1	En programas	117
6.6.2	En portafolios.....	117
6.7	Resumen de responsabilidades	119
6.8	Scrum vs. Gestión tradicional de proyectos	120
7.	RIESGO	121
7.1	Introducción	121
7.2	Guía de roles	122
7.3	¿Qué es un riesgo?	122
7.3.1	Diferencia entre riesgos y problemas.....	122
7.3.2	Actitud de riesgo	123
7.4	Procedimiento de gestión de riesgos.....	124
7.4.1	Identificación de riesgos	124
7.4.2	Evaluación de riesgos.....	125
7.4.3	Priorización de riesgos.....	128
7.4.4	Mitigación de riesgos	129
7.4.5	Comunicación de riesgos.....	130
7.5	Minimizar riesgos por medio de Scrum	131
7.6	Riesgos en portafolios y programas	132
7.6.1	En el portafolio.....	132
7.6.2	En los programas.....	132
7.7	Resumen de responsabilidades	134

7.8	Scrum vs. Gestión tradicional de proyectos	135
8.	INICIO	137
8.1	Crear la visión del proyecto	141
8.1.1	Entradas	143
8.1.2	Herramientas	144
8.1.3	Salidas	146
8.2	Identificar al Scrum Master y a los interesados del negocio.....	147
8.2.1	Entradas	149
8.2.2	Herramientas	150
8.2.3	Salidas	151
8.3	Formar el Equipo Scrum.....	153
8.3.1	Entradas	155
8.3.2	Herramientas	156
8.3.3	Salidas	158
8.4	Desarrollar épicas.....	159
8.4.1	Entradas	160
8.4.2	Herramientas	163
8.4.3	Salidas	165
8.5	Crear el backlog priorizado del producto	166
8.5.1	Entradas	167
8.5.2	Herramientas	169
8.5.3	Salidas	172
8.6	Realizar la planificación de la liberación.....	174
8.6.1	Entradas	175
8.6.2	Herramientas	177
8.6.3	Salidas	178
8.7	Diagrama de flujo de la fase de inicio.....	180
9.	PLANIFICACIÓN Y ESTIMACIÓN	181
9.1	Crear historias de usuario.....	185
9.1.1	Entradas	186
9.1.2	Herramientas	188
9.1.3	Salidas	189
9.2	Estimar historias de usuarios.....	191

9.2.1	Entradas	192
9.2.2	Herramientas	193
9.2.3	Salidas	196
9.3	Comprometer historias de usuario.....	196
9.3.1	Entradas	197
9.3.2	Herramientas	198
9.3.3	Salidas	199
9.4	Identificar tareas	201
9.4.1	Entradas	202
9.4.2	Herramientas	203
9.4.3	Salidas	204
9.5	Estimar tareas	206
9.5.1	Entradas	207
9.5.2	Herramientas	208
9.5.3	Salidas	209
9.6	Actualizar el backlog del sprint	210
9.6.1	Entradas	211
9.6.2	Herramientas	212
9.6.3	Salidas	214
9.7	Diagrama de flujo de la fase de planificación y estimación	216
10.	IMPLEMENTACIÓN.....	217
10.1	Crear entregables.....	221
10.1.1	Entradas	222
10.1.2	Herramientas	226
10.1.3	Salidas	227
10.2	Realizar el Daily Standup	229
10.2.1	Entradas	230
10.2.2	Herramientas	231
10.2.3	Salidas	233
10.3	Refinar el backlog priorizado del producto	234
10.3.1	Entradas	235
10.3.2	Herramientas	237
10.3.3	Salidas	238

10.4	Diagrama de flujo de la fase de implementación	239
11.	REVISIÓN Y RETROSPECTIVA	240
11.1	Demostrar y validar el sprint	243
11.1.1	Entradas	244
11.1.2	Herramientas	246
11.1.3	Salidas	247
11.2	Retrospectiva del sprint	248
11.2.1	Entradas	249
11.2.2	Herramientas	250
11.2.3	Salidas	252
11.3	Diagrama de flujo de la fase de revisión y retrospectiva	253
12.	LIBERACIÓN	254
12.1	Enviar entregables	257
12.1.1	Entradas	258
12.1.2	Herramientas	259
12.1.3	Salidas	260
12.2	Retrospectiva de la liberación	261
12.2.1	Entradas	262
12.2.2	Herramientas	263
12.2.3	Salidas	264
12.3	Diagrama de flujo de la fase de liberación	265
13.	ESCALAMIENTO DE SCRUM EN GRANDES PROYECTOS	266
13.1	El impacto de los grandes proyectos en los procesos fundamentales de Scrum	268
13.1.1	Inicio	268
13.1.2	Planificación y estimación	273
13.1.3	Implementación	276
13.1.4	Revisión y retrospectiva	278
13.1.5	Liberación	279
13.2	Entradas y salidas adicionales para los grandes proyectos	280
13.2.1	Organización de Scrum en los grandes proyectos	280
13.2.2	Plan de colaboración los Product Owners*	281
13.2.3	Plan de colaboración de los Scrum Masters y los Equipos Scrum	281
13.2.4	Recursos compartidos*	282

13.2.5	Especialización del equipo.....	282
13.2.6	Ambiente y cronograma de ambiente*	283
13.2.7	Plan de preparación para la liberación*	283
13.3	Herramientas adicionales para los grandes proyectos	283
13.3.1	Plan de comunicación para grandes proyectos	283
13.3.2	Planificación de recursos para grandes proyectos*	284
13.3.3	Identificación del ambiente*	284
13.3.4	Asignación del backlog priorizado del producto*	285
13.3.5	Reunión de Scrum de Scrums (SoS)*	285
13.3.6	Métodos de preparación para la liberación*	286
13.3.7	Sprint de preparación para la liberación	286
13.3.8	Herramienta para un proyecto de Scrum	287
14.	ESCALAMIENTO DE SCRUM PARA LA EMPRESA	288
14.1	El impacto de programas o portafolios en los procesos fundamentales a nivel del proyecto	290
14.1.1	Inicio	290
14.1.2	Planificación y estimación.....	293
14.1.3	Implementación.....	293
14.1.4	Revisión y retrospectiva.....	294
14.1.5	Liberación	295
14.2	Procesos adicionales para escalar Scrum en la empresa (programa o portafolio).....	296
14.3	Crear o actualizar los equipos del programa o portafolio	299
14.3.1	Entradas	300
14.3.2	Herramientas	301
14.3.3	Salidas	301
14.4	Crear o actualizar los componentes del programa o portafolio	303
14.4.1	Entradas	304
14.4.2	Herramientas	305
14.4.3	Salidas	307
14.5	Revisar y actualizar el Scrum Guidance Body.....	309
14.5.1	Entradas	310
14.5.2	Herramientas	310
14.5.3	Salidas	311
14.6	Crear o refinar el backlog priorizado del programa o portafolio.....	312

14.6.1	Entradas	313
14.6.2	Herramientas	315
14.6.3	Salidas	317
14.7	Crear o actualizar las liberaciones del programa o portafolio.....	318
14.7.1	Entradas	319
14.7.2	Herramientas	320
14.7.3	Salidas	321
14.8	Retrospectiva de las liberaciones del programa o portafolio	322
14.8.1	Entradas	322
14.8.2	Herramientas	323
14.8.3	Salidas	324
APÉNDICE A. INFORMACIÓN GENERAL SOBRE LA AGILIDAD		326
APÉNDICE B. AUTORES Y COLABORADORES DE LA <i>GUÍA DEL SBOK®</i>		336
APÉNDICE C. ACTUALIZACIONES EN LA CUARTA EDICIÓN.....		340
REFERENCIAS.....		346
GLOSARIO		348
ÍNDICE TEMÁTICO		386

LISTA DE FIGURAS

Figura 1-1: Flujo de Scrum en un sprint.....	2
Figura 1-2: Marco de trabajo de la <i>Guía del SBOK®</i>	7
Figura 1-3: Principios de Scrum.....	9
Figura 1-4: Organización en Scrum	11
Figura 2-1: Transparencia en Scrum.....	24
Figura 2-2: Inspección en Scrum	25
Figura 2-3: Adaptación en Scrum	26
Figura 2-4: Retos en la gestión tradicional de proyectos	27
Figura 2-5: Objetivos de un equipo autoorganizado	29
Figura 2-6: Beneficios de la colaboración en proyectos de Scrum	31
Figura 2-7: Priorización basada en valor.....	35
Figura 2-8: Duración del Time-Box para las reuniones de Scrum.....	37
Figura 2-9: Scrum vs Cascada tradicional	39
Figura 3-1: Roles de Scrum: Resumen	43
Figura 3-2: Características deseadas de los roles principales de Scrum.....	51
Figura 3-3: Preguntas durante la reunión de Scrum de Scrums	53
Figura 3-4: Scrum para proyectos, programas y portafolios en la organización	58
Figura 3-5: Etapas de Tuckman de desarrollo de grupos	61
Figura 3-6: Teoría de jerarquía de necesidades de Maslow	65
Figura 4-1: Entrega de valor en Scrum vs. Proyectos tradicionales.....	70
Figura 4-2: Jerarquía de responsabilidades en la justificación del negocio	71
Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto	74
Figura 4-4: Mapeo de flujo de valor	79
Figura 4-5: Análisis de Kano	79
Figura 4-6: Mapeo de historias	79
Figura 4-7: Ejemplo de diagrama de flujo acumulado (DFA)	84
Figura 5-1: Diagrama de flujo del incremento del proyecto.....	92
Figura 5-2: Ciclo PDCA en Scrum.....	99
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: Motivos de los interesados del negocio 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.....	114
Figura 6-8: Incorporación de cambios en portafolios y programas	118
Figura 7-1: Ejemplo de árbol de probabilidad	126
Figura 7-2: Ejemplo de un diagrama de Pareto	126

Figura 7-3: Ejemplo de matriz de probabilidad e impacto	127
Figura 7-4: Proceso de priorización de riesgos	129
Figura 7-5: Ejemplo de un Risk Burndown Chart (gráfica de disminución de riesgos)	131
Figura 7-6: Manejo de riesgos en portafolios y programas	131
Figura 8-1: Información general sobre la fase de inicio	139
Figura 8-2: Información general sobre la fase de inicio (fundamentales).....	140
Figura 8-3: Crear la visión del proyecto: entradas, herramientas y salidas.....	141
Figura 8-4: Crear la visión del proyecto: diagrama del flujo de datos	142
Figura 8-5: El proceso de Análisis de brecha.....	145
Figura 8-6: Identificar al Scrum Master y a los interesados del negocio: entradas, herramientas y salidas	147
Figura 8-7: Identificar al Scrum Master y a los interesados del negocio: Diagrama de flujo de datos	148
Figura 8-8: Formar el Equipo Scrum: entradas, herramientas y salidas	153
Figura 8-9: Formar el Equipo Scrum: Diagrama de flujo.....	154
Figura 8-10: Desarrollar épicas: entradas, herramientas y salidas	159
Figura 8-11: Desarrollar épicas: Diagrama de flujo.....	160
Figura 8-12: Crear el backlog priorizado del producto: entradas, herramientas y salidas.....	166
Figura 8-13: Crear el backlog priorizado del producto: Diagrama de flujo de datos	167
Figura 8-14: Realizar la planificación de la liberación: entradas, herramientas y salidas	174
Figura 8-15: Realizar la planificación de la liberación: Diagrama de flujo de datos	175
Figura 8-16: Fase de inicio: Diagrama del flujo.....	180
Figura 9-1: Información general de la fase de planificación y estimación	183
Figura 9-2: Información general de la fase de (fundamentales).....	184
Figura 9-3: Crear historias de usuario: entradas, herramientas y salidas	185
Figura 9-4: Crear historias de usuario: Diagrama de flujo.....	186
Figura 9-5: Estimar historias de usuario: Entradas, herramientas y salidas	191
Figura 9-6: Estimar historias de usuario: Diagrama de flujo	192
Figura 9-7: Comprometer tareas: entradas, herramientas y salidas	196
Figura 9-8: Comprometer historias de usuario: Diagrama de flujo.....	197
Figura 9-9: Scrumboard tradicional.....	197
Figura 9-10: Scrumboard con cuatro secciones.....	197
Figura 9-11: Identificar tareas: entradas, herramientas y salidas.....	201
Figura 9-12: Identificar tareas: Diagrama de flujo	202
Figura 9-13: Scrumboard con tareas identificadas.....	205
Figura 9-14: Estimar tareas: entradas, herramientas y salidas	206
Figura 9-15: Estimar tareas: Diagrama de flujo de datos	207
Figura 9-16: Actualizar el backlog del sprint: entradas, herramientas y salidas.....	210
Figura 9-17: Actualizar el backlog del sprint: Diagrama de flujo de datos.....	211
Figura 9-18: Sprint Burndown Chart	215
Figura 9-19: Sprint Burnup Chart	215
Figura 9-20: Fase de planificación y estimación: diagrama de flujo.....	216
Figura 10-1: Información general de la fase de implementación	219

Figura 10-2: Información general de la fase de implementación (fundamentales).....	220
Figura 10-3: Crear entregables: entradas, herramientas y salidas	221
Figura 10-4: Crear entregables: Diagrama de flujo de datos	222
Figura 10-5: Scrumboard con tareas “Por hacer”, “En proceso” y “Terminada”	223
Figura 10-6: Realizar el Daily Standup: entradas, herramientas y salidas.....	229
Figura 10-7: Realizar el Daily Standup: diagrama de flujo de datos	230
Figura 10-8: Refinar el backlog priorizado del producto: entradas, herramientas y salidas.....	234
Figura 10-9: Refinar el backlog priorizado del producto: diagrama de flujo de datos	235
Figura 10-10: Fase de implementación: Diagrama de flujo.....	239
Figura 11-1: Información general de la fase de revisión y retrospectiva	241
Figura 11-2: Información general de la fase de revisión y retrospectiva (fundamentales)	242
Figura 11-3: Demostrar y validar el sprint: entradas, herramientas y salidas	243
Figura 11-4: Demostrar y validar el sprint: Diagrama de flujo	244
Figura 11-5: Retrospectiva del sprint: entradas, herramientas y salidas.....	244
Figura 11-6: Retrospectiva del sprint: Diagrama de flujo	244
Figura 11-7: Fase de revisión y retrospectiva: Diagrama de flujo	244
Figura 12-1: Información general de la fase de liberación	255
Figura 12-2: Información general de la fase de liberación (fundamentales)	256
Figura 12-3: Enviar entregables: entradas, herramientas y salidas	257
Figura 12-4: Enviar entregables: Diagrama de flujo de datos	258
Figura 12-5: Retrospectiva de la liberación: entradas, herramientas y salidas	261
Figura 12-6: Retrospectiva de la liberación: Diagrama de flujo de datos	262
Figura 12-7: Fase de liberación: Diagrama de flujo	265
Figura 14-1: Información general de proceso de escalamiento de Scrum para la empresa	305
Figura 14-2: Crear o actualizar los equipos del programa o portafolio: entradas, herramientas y salidas.....	307
Figura 14-3: Crear o actualizar los componentes del programa o portafolio: entradas, herramientas y salidas ..	311
Figura 14-4: Reunión de Scrum de Scrums	314
Figura 14-5: Revisar y actualizar el Scrum Guidance Body: entradas, herramientas y salidas	317
Figura 14-6: Crear o refinar el backlog priorizado del programa o portafolio: entradas, herramientas y salidas ..	320
Figura 14-7: Crear o actualizar las liberaciones del programa o portafolio: entradas, herramientas y salidas	326
Figura 14-8: Retrospectiva de las liberaciones del programa o portafolio: entradas, herramientas y salidas.....	330

LISTA DE TABLAS

Tabla 1-1: Resumen de los procesos fundamentales de Scrum.....	14
Tabla 1-2: Resumen de procesos adicionales de Scrum.....	18
Tabla 1-3: Scrum vs. Gestión tradicional de proyectos.....	20
Tabla 3-1: Responsabilidades del Product Owner en los procesos de Scrum.....	47
Tabla 3-2: Responsabilidades del Scrum Master en los procesos de Scrum	48
Tabla 3-3: Responsabilidades del Equipo Scrum en los procesos de Scrum	50
Tabla 3-4: Resumen de las responsabilidades pertinentes a la organización	59
Tabla 4-1: Fórmulas del valor ganado	81
Tabla 4-2: Resumen de las responsabilidades pertinentes a la justificación del negocio	86
Tabla 5-1: Criterios de terminado en cascada	93
Tabla 5-2: Resumen de las responsabilidades pertinentes a la calidad	99
Tabla 6-1: Resumen de las responsabilidades pertinentes al cambio	119
Tabla 7-1: Resumen de las responsabilidades pertinentes a los riesgos	134
Tabla 13-1: El impacto de los grandes proyectos en los procesos fundamentales de Scrum: Fase de inicio	227
Tabla 13-2: El impacto de los grandes proyectos en los procesos fundamentales de Scrum: Fase de planificación y estimación	280
Tabla 13-3: El impacto de los grandes proyectos en los procesos fundamentales de Scrum: Fase de implementación	282
Tabla 13-4: El impacto de los grandes proyectos en los procesos fundamentales de Scrum: Fase de revisión y retrospectiva.....	284
Tabla 13-5: Impacto de los grandes proyectos en los procesos fundamentales de Scrum: Fase de liberación	285
Tabla 14-1: Impacto de un programa o portafolio en los procesos fundamentales de Scrum: Fase de inicio	300
Tabla 14-2: Impacto de un programa o portafolio en los procesos fundamentales de Scrum: Fase de implementación	301
Tabla 14-3: Impacto de un programa o portafolio en los procesos fundamentales de Scrum: Fase de revisión y retrospectiva.....	302
Tabla 14-4: Impacto de un programa o portafolio en los procesos fundamentales de Scrum: Fase de liberación.....	303

1. INTRODUCCIÓN

La *Guía de los fundamentos de Scrum (Guía del 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 marco de trabajo integral que incluye los principios, los aspectos y los procesos de Scrum.

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

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

El término “producto” en la *Guía del 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 solo seis miembros, hasta proyectos grandes y complejos con varios cientos de integrantes.

Este primer capítulo describe la finalidad y el marco de trabajo de la *Guía del 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 Liberación. 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 o lineamientos establecidos por el Scrum Guidance Body de la organización (SGB). Los capítulos 13 y 14 son nuevas adiciones a la *Guía del SBOK®*, mismas que brindan las directrices para escalar 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 del SBOK®*

1.4 Marco de trabajo de la *Guía del SBOK®*

1.5 Scrum vs. Gestión tradicional de proyectos

1.1 Información general de Scrum

Un proyecto Scrum implica 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. Los proyectos se ven afectados por restricciones 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 marco de trabajo adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto. Aunque el marco de trabajo de Scrum, tal como se define en la *Guía del SBOK®*, se utiliza principalmente para presentar proyectos y crear productos, también se puede utilizar para gestionar el mantenimiento constante de productos y servicios, dar seguimiento a los problemas y gestionar cambios.

Scrum garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo. El marco de trabajo de Scrum, tal como se define en la *Guía del 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 es el uso de equipos multidisciplinarios, autoorganizados 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.

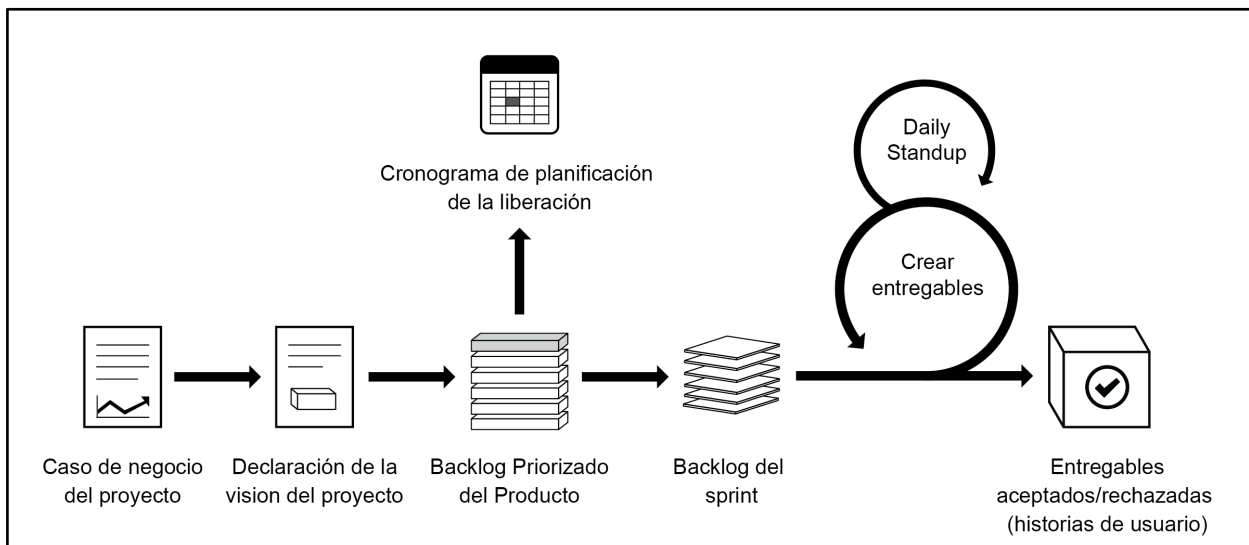


Figura 1-1: Flujo de Scrum en un sprint

El ciclo de Scrum empieza con una reunión de interesados del negocio, durante la cual se crea la visión del proyecto. Después, el Product Owner desarrolla un backlog priorizado del producto que contiene una lista de requerimientos del negocio y del proyecto por orden de importancia en forma de una historia de usuario.

Cada sprint inicia con una reunión de planificación del sprint, donde se ponen a consideración las historias de usuario de mayor prioridad para ser incluidas en el sprint. Un sprint generalmente tiene una duración de una a cuatro semanas en las cuales el Equipo Scrum trabaja en la creación de entregables potencialmente enviables o incrementos del producto. Durante el sprint, se llevan cabo Daily Standups muy breves y concretos, donde los miembros del equipo discuten el avance diario. Hacia el final del sprint, se lleva a cabo una reunión de revisión del sprint en la cual se hace una demostración de los entregables al Product Owner y a los interesados del negocio relevantes. El Product Owner acepta los entregables solo si cumplen con los criterios de aceptación predefinidos. El ciclo del sprint concluye con una reunión de retrospectiva del sprint, donde el equipo analiza la forma de mejorar su trabajo y su desempeño a medida que avanza al siguiente sprint.

1.1.1 Breve historia de Scrum

A mediados de los años 80s, Hirotaka Takeuchi y Ikujiro Nonaka definieron una estrategia flexible e incluyente para el desarrollo de productos, donde el equipo de desarrollo trabaja en unidad para alcanzar un objetivo común. Definieron 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 método en estudios de casos de diversas industrias manufactureras. Takeuchi y Nonaka propusieron que el desarrollo de productos no debe parecer una carrera de relevos secuencial, sino que debe ser parecido 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”.

Desde entonces, varios practicantes, expertos y autores de Scrum siguen refinando la conceptualización de Scrum y su marco de trabajo. Uno de los grandes logros en Scrum fue la creación de la *Guía del SBOK®* en el año 2013. Con el paso del tiempo, la *Guía del SBOK®* ha ido mejorando con aportes y revisiones presentadas por miles de practicantes de Scrum y ágil, incluyendo más de 5,000 instructores en más de 110 países que enseñan Scrum y utilizan el marco de trabajo de la *Guía del SBOK®*. Esta cuarta edición es el producto de mayor refinación, lo cual garantiza que la *Guía del SBOK®* sigue siendo válida y relevante en un mundo cambiante.

La *Guía del SBOK®* se ha convertido en el estándar de la industria para compañías o personas interesadas en implementar las prácticas de Scrum y ágil. En los últimos años, Scrum ha aumentado en popularidad y hoy en día es la opción predilecta para el desarrollo de proyectos de muchas organizaciones a nivel mundial. Con el fin de facilitar su aplicación a nivel mundial, la *Guía del SBOK®* ha sido traducida a varios idiomas, incluyendo español, portugués, francés, italiano, árabe, chino y japonés. Para obtener más información sobre cómo obtener las versiones traducidas, visite el sitio www.scrumstudy.com.

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 el 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 continua de valor:** Los procesos iterativos permiten la entrega continua de valor tan frecuentemente como el cliente lo requiera a través del proceso de *Enviar 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, les permite continuar trabajando de manera indefinida.
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 innecesario conducen a mayores niveles de eficiencia.
9. **Motivación:** Los procesos de *Realizar el 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 coubicación de equipos multidisciplinarios 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 interesados del negocio asegura un marco de trabajo orientado al cliente.
13. **Ambiente de alta confianza:** Los procesos de *Realizar el 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 usuario* permite que los miembros del equipo hagan suyo el proyecto y su trabajo lleve a una mejor calidad.
15. **Alta velocidad:** Un marco de trabajo de colaboración permite a los equipos multidisciplinarios altamente cualificados alcanzar su potencial y una alta velocidad.
16. **Ambiente innovador:** Los procesos de *Retrospectiva del sprint* y *Retrospectiva de la liberación* 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, lo ideal es que los equipos Scrum tengan de seis a diez miembros. Esta práctica pudiera ser la razón de la idea equivocada de que el marco de trabajo de Scrum solo se puede utilizar para proyectos pequeños. Sin embargo, este marco de trabajo puede ampliarse fácilmente para utilizarse de manera eficaz en grandes proyectos, programas y portafolios. En situaciones donde el tamaño del Equipo Scrum es mayor a diez personas, se pueden formar diversos equipos para trabajar en el proyecto. El enfoque lógico de las directrices y los principios de este marco de trabajo pueden utilizarse para gestionar proyectos de cualquier tamaño, que abarcan grandes geografías y organizaciones. Los proyectos grandes pueden tener múltiples equipos Scrum trabajando de manera 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 El objetivo de la *Guía del SBOK®*

Se ha comprobado que Scrum es el marco de trabajo preferente para la entrega de proyectos con el fin de presentar un alto valor de negocio y mejorar el retorno de la inversión. El enfoque de Scrum en la entrega impulsada por el valor ayuda a que los equipos de Scrum presenten resultados durante el proyecto tan pronto como les sea posible. La *Guía del SBOK®* ha sido desarrollada como un medio para orientar a organizaciones y a 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 en varias organizaciones e industrias. En su desarrollo se tomaron en cuenta las aportaciones de muchos expertos en Scrum y profesionales de la gestión de proyectos.

La *Guía del SBOK®* es especialmente útil:

- para los miembros del equipo principal de Scrum, incluyendo a:
 - Product Owners que deseen entender plenamente el marco de trabajo de Scrum y particularmente las inquietudes del cliente o los interesados del negocio 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 al supervisar la aplicación del marco de trabajo de Scrum en proyectos de este tipo.
 - Miembros del Equipo Scrum que deseen comprender mejor los procesos de Scrum y las herramientas asociadas que se pueden utilizar 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 consulta para cualquier persona que interactúe con el equipo principal de Scrum, incluyendo, entre otros, al Portfolio Product Owner, Portfolio Scrum Master, Program Product Owner, Program Scrum Master, Scrum Guidance Body e interesados del negocio (patrocinador, cliente y usuarios).

- como un manual para cualquier persona que no tenga experiencia previa o conocimiento del marco de trabajo de Scrum, pero quiera aprender más sobre el tema.

El contenido de la *Guía del SBOK®* también es útil para las personas que se preparan para hacer 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 Marco de trabajo de la *Guía del SBOK®*

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

1. **Principios:** Se describen en el capítulo 2 y explican la información sobre los seis principios que constituyen el fundamento sobre el que se basa Scrum.
2. **Aspectos:** Se describen en los capítulos del 3 al 7 y describen los cinco aspectos que se consideran importantes en todos los proyectos Scrum.
3. **Procesos:** Se describen en los capítulos del 8 al 12 e incluyen los diecinueve procesos fundamentales de Scrum y sus entradas, herramientas y salidas asociadas. El capítulo 13 aborda las entradas, herramientas y salidas adicionales necesarias para escalar Scrum en grandes proyectos, mientras que el capítulo 14 describe los procesos adicionales necesarios para escalar Scrum para la empresa.

La figura 1-2 ilustra el marco de trabajo de la *Guía del SBOK®* y la interacción entre los principios, aspectos y procesos. Los tres son de igual importancia para entender y aplicar el marco de trabajo de Scrum.

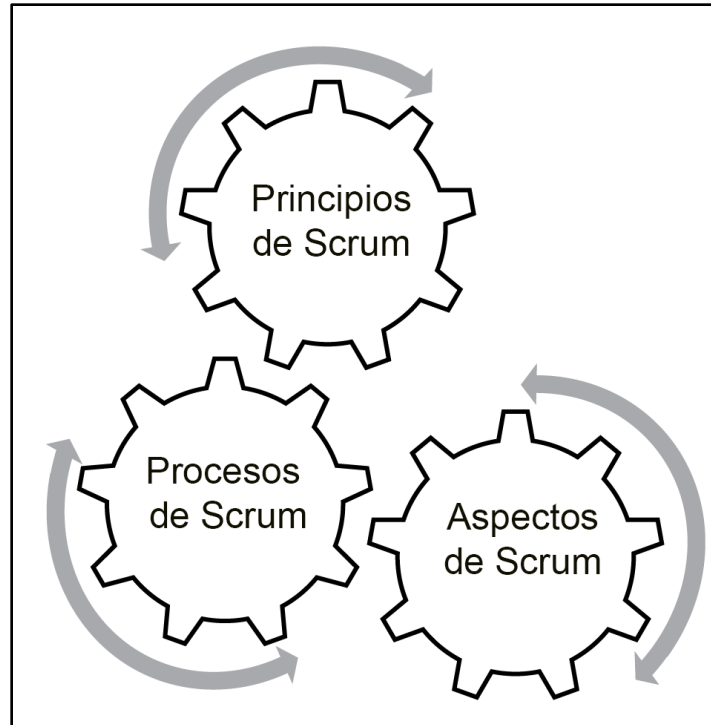


Figura 1-2: Marco de trabajo de la *Guía del SBOK®*

1.4.1 ¿Cómo se utiliza la *Guía del SBOK®*?

La *Guía del SBOK®* puede utilizarse como una referencia y fuente de información 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. A fin de facilitar la mejor aplicación del marco de trabajo de Scrum, la *Guía del 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 se consideran 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 marco de trabajo 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 la *Guía del 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. Las entradas, herramientas y salidas adicionales se aplicarían solamente cuando se escale Scrum en grandes proyectos (capítulo 13) y los procesos adicionales se utilizarían cuando se escale Scrum en las empresas (capítulo 14).

1.4.2 Principios de Scrum

Los principios de Scrum son los lineamientos básicos para la aplicación del marco de trabajo de Scrum y deben implementarse de manera 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. Autoorganización (*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.

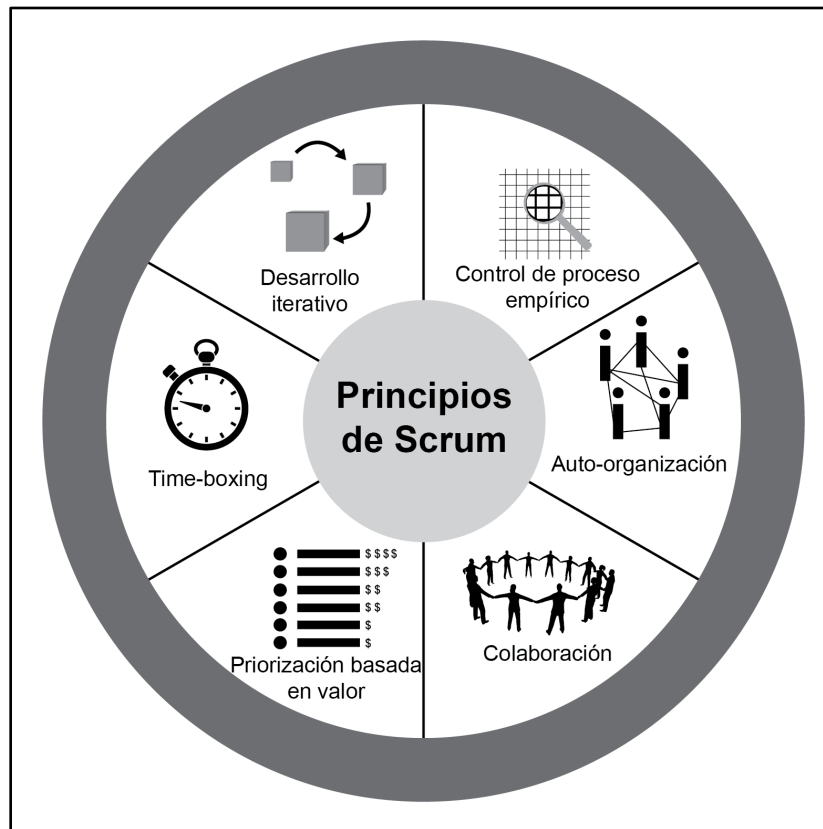


Figura 1-3: Principios de Scrum

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 marco de trabajo 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 del SBOK®*. El mantener los principios intactos y usarlos apropiadamente infunde confianza en el marco de trabajo 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. El control del proceso empírico ayuda a aprender por medio de la experimentación, especialmente cuando el problema no está bien definido o cuando no existen soluciones claras.
2. **Autoorganización:** Este principio se enfoca en los trabajadores de hoy en día, que entregan un valor considerablemente mayor cuando se autoorganizan, 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 entre sí, con el cliente y otros interesados del negocio para ofrecer el mayor valor posible.
4. **Priorización basada en valor:** Este principio resalta 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 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 la manera de 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 principales:** Los roles principales (*core roles*) son aquellos que se requieren de manera obligatoria para crear el producto o servicio del proyecto. Las personas a quienes se les asignan los roles principales 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 a los integrantes del equipo principal de Scrum:

- **Product Owner:** Es la persona responsable de lograr el máximo valor de negocio 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.
- **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.
- **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 secundarios:** Los roles secundarios (*non-core roles*) 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 secundarios deben tenerse en cuenta en cualquier proyecto de Scrum.

Los roles secundarios son:

- **Interesados del negocio:** 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 interesados del negocio. Los interesados del negocio forman parte de un subconjunto de interesados en un proyecto de Scrum. Entre los interesados del proyecto están todas las personas y grupos afectados por el proyecto de Scrum, tanto dentro como fuera de la organización (por ejemplo: todos los roles principales y secundarios, los proveedores, grupos internos, expertos, entre otros).
- **Scrum Guidance Body (SGB):** Es un rol opcional, que generalmente consiste en un conjunto de documentos 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.

- **Proveedores:** Son personas u organizaciones externas, ofrecen productos o servicios que no están dentro de las competencias centrales de la organización del proyecto.

La figura 1-4 ilustra la estructura organizacional Scrum.

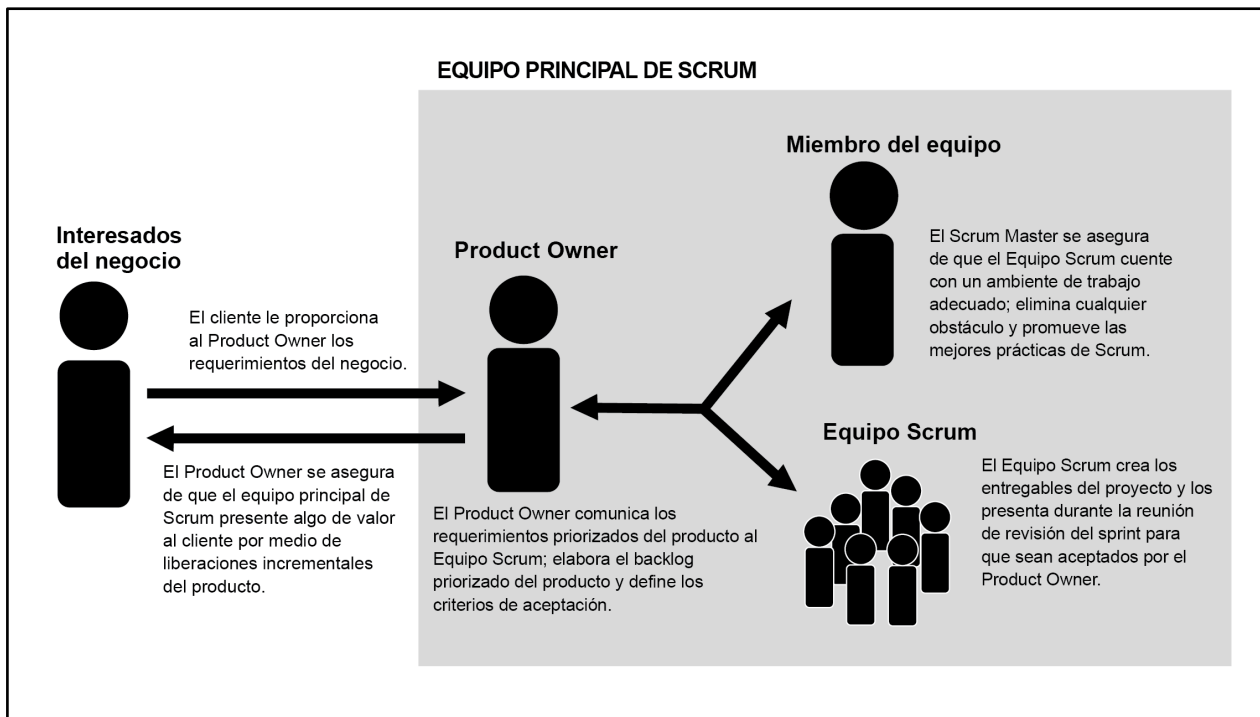


Figura 1-4: Organización en Scrum

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 basada en 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 de su tamaño o complejidad. 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 interesados del negocio.

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 interesados del negocio para mantener constantemente actualizado el backlog priorizado del producto con cualquier cambio en los requisitos. El backlog priorizado del producto nunca solo se finaliza hasta el cierre o conclusión del proyecto. Cualquier cambio en los requisitos debe reflejar los cambios en el entorno del negocio, 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 enviabiles, se les denomina “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 interesados del negocio (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 marco de trabajo 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 los interesados del negocio (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); y que es muy difícil, si no imposible, que los interesados del negocio 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 riesgos 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. Los procesos de Scrum generalmente no son secuenciales, sino iterativos y pudieran superponerse unos con otros. 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 del SBOK®* tal como se muestra en la tabla 1-1.

Capítulo	Fase	Procesos fundamentales de Scrum
8	Inicio	<ol style="list-style-type: none"> 1. Crear la visión del proyecto 2. Identificar al Scrum Master y a los interesados del negocio 3. Formar Equipos Scrum 4. Desarrollar épicas 5. Crear el backlog priorizado del producto 6. Realizar la planificación de la liberación
9	Planificación y estimación	<ol style="list-style-type: none"> 7. Crear historias de usuario 8. Estimar historias de usuario 9. Comprometer historias de usuario 10. Identificar tareas 11. Estimar tareas 12. Actualizar el backlog del sprint
10	Implementación	<ol style="list-style-type: none"> 13. Crear entregables 14. Realizar el Daily Standup 15. Refinar el backlog priorizado del producto
11	Revisión y retrospectiva	<ol style="list-style-type: none"> 16. Demostrar y validar el sprint 17. Retrospectiva del sprint
12	Liberación	<ol style="list-style-type: none"> 18. Enviar entregables 19. Retrospectiva de la liberación

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 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 gran escala que requieren de una coordinación entre múltiples equipos, existen entradas, herramientas y salidas adicionales de Scrum que se definen en el capítulo 13: Escalar Scrum en grandes proyectos.

Existen también procesos específicos definidos cuando se implementa Scrum al nivel de negocio, lo cual se aborda en el capítulo 14: Escalar Scrum para la empresa.

1.4.4.1 Fase de inicio

Los procesos relevantes en la fase de inicio son los siguientes:

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 a los interesados del negocio:* En este proceso se identifica al Scrum Master y a los interesados del negocio 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 épicas:* 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 en el proyecto. A este punto también se establecen los criterios de terminado.
6. *Realizar la planificación de la liberación:* 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 de la liberación, que es esencialmente un programa de implementación por fases que se puede compartir con los interesados del negocio del proyecto. En este proceso también se determina la duración del sprint.

1.4.4.2 Fase de planificación y estimación

Los procesos relevantes en la fase de estimación y liberación son los siguientes:

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 interesados del negocio. 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 explica 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 lista de tareas con esfuerzo estimado (*Effort Estimated Task List*).
12. *Actualizar el backlog del sprint:* En este proceso, el equipo principal de Scrum elabora un backlog del sprint que contiene todas las tareas a ser completadas en un sprint como parte de la reunión de planificación del sprint.

1.4.4.3 Fase de implementación

Los procesos relevantes en la fase de implementación son los siguientes:

13. *Crear entregables:* En este proceso, el Equipo Scrum trabaja en las tareas en el backlog del sprint 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 actualizarse en una lista de impedimentos.
14. *Realizar el 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. *Refinar el 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 Fase de revisión y retrospectiva

Los procesos relevantes en la fase de revisión y retrospectiva son los siguientes:

16. *Demostrar y validar el sprint:* En este proceso, el Equipo Scrum muestra los entregables del sprint al Product Owner y a los interesados del negocio relevantes en una reunión de revisión del sprint. El objetivo de esta reunión es asegurar que el Product Owner apruebe y acepte las historias de usuario del 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 de lecciones aprendidas que pueden aplicarse a futuros sprints. Frecuentemente, como resultado de esta discusión, puede haber mejoras aceptadas o recomendaciones actualizadas por parte del Scrum Guidance Body.

1.4.4.5 Fase de liberación

Los procesos relevantes en la fase de liberación son los siguientes:

18. *Enviar entregables:* En este proceso, los entregables aceptados se entregan o se envían a los interesados del negocio relevantes. La conclusión satisfactoria del sprint se documenta en un acuerdo de entregables funcionales.
19. *Retrospectiva de la liberación:* En este proceso, mismo que concluye el proyecto, los interesados del negocio y miembros del equipo principal de Scrum se reúnen para reflexionar sobre el proyecto e identificar, documentar e internalizar las lecciones aprendidas. A menudo, estas lecciones llevan a la documentación de mejoras accionables acordadas, que se implementarán en futuros proyectos.

1.4.4.6 Reuniones o ceremonias de Scrum

Las reuniones de Scrum tienen una función importante en la implementación eficaz del marco de trabajo de Scrum y son el medio principal para la implementación de los principios de Scrum. Las reuniones importantes de Scrum y los procesos respectivos en las que se llevan a cabo estas reuniones se describen en la tabla 1-2:

Reunión de Scrum	Proceso de Scrum
Reunión de visión del proyecto	<ul style="list-style-type: none"> • Crear la visión del proyecto
Reuniones con grupos de usuarios	<ul style="list-style-type: none"> • Desarrollar épicas • Crear historias de usuario
Reuniones de grupos de enfoque	<ul style="list-style-type: none"> • Desarrollar épicas • Crear historias de usuario
Sesiones o reuniones de planificación de la liberación	<ul style="list-style-type: none"> • Realizar la planificación de la liberación
Reuniones de revisión del backlog priorizado del producto	<ul style="list-style-type: none"> • Estimar historias de usuario • Refinar el backlog priorizado del producto
Reuniones de planificación del sprint	<ul style="list-style-type: none"> • Estimar historias de usuario • Comprometer historias de usuario • Identificar tareas • Estimar tareas • Actualizar el backlog del sprint
Daily Standup	<ul style="list-style-type: none"> • Realizar el Daily Standup
Reunión de revisión del sprint	<ul style="list-style-type: none"> • Demostrar y validar el sprint
Reunión de retrospectiva del sprint	<ul style="list-style-type: none"> • Retrospectiva del sprint
Reunión de retrospectiva de la liberación	<ul style="list-style-type: none"> • Retrospectiva de la liberación

Tabla 1-2: Reuniones y procesos de Scrum

1.4.5 Scrum para grandes proyectos

Al trabajar en grandes proyectos donde se requiere el trabajo de múltiples (cuatro o más) equipos de Scrum con varios Product Owner y múltiples Scrum Masters, los capítulos del 8 al 12 siguen siendo válidos, aunque tal vez sean necesarias consideraciones adicionales y actualizaciones a las entradas, herramientas y salidas. Esto puede incluir necesidades adicionales de coordinación y sincronización. El impacto de los procesos fundamentales de Scrum al escalar Scrum en grandes proyectos se describe en el capítulo 13.

La definición de lo que constituye un proyecto grande generalmente depende de la organización o de la complejidad de los proyectos emprendidos. Un criterio clave para saber si un proyecto se considera pequeño en vez de grande es si requiere de múltiples Scrum Masters o múltiples Product Owners. Si el proyecto requiere solo un Scrum Master y un Product Owner, generalmente estos pueden manejar cualquier actividad adicional de comunicación y sincronización que requiera el proyecto.

1.4.6 Scrum para la empresa

Al trabajar con Scrum al nivel de una empresa (un programa o un portafolio), tal vez sea necesaria la participación de cientos de equipos con miles de personas a cargo de múltiples proyectos dentro de los programas o portafolios de la compañía. El uso de Scrum a nivel de un programa o portafolio tendrá ciertos efectos en los proyectos subyacentes. En general, los proyectos de Scrum se pueden ejecutar con los procesos fundamentales de Scrum descritos en los capítulos del 8 al 12 si se trata de proyectos pequeños, pero se deben tomar en cuenta las consideraciones plasmadas en el capítulo 13 con relación a los grandes proyectos (que cuentan con múltiples Product Owners o Scrum Masters).

Algunos de los retos que surgen a nivel del programa o del portafolio son similares a las que se presentan en los grandes proyectos de Scrum. Los principales retos en los grandes proyectos son la sincronización entre los equipos y la colaboración general. Esto también representa un reto al aplicar Scrum al nivel del programa o del portafolio. Sin embargo, los más grandes retos en un programa o en un portafolio pueden presentarse en el aspecto de negocio, dado que pudieran contraponerse las prioridades del negocio en los distintos proyectos y estar en conflicto con los objetivos generales del programa o del portafolio. Estas metas y prioridades deben estar en armonía.

Al implementar Scrum al nivel de la empresa, no solo se deben aplicar entradas, herramientas y salidas tal como en un proyecto grande, también hay procesos adicionales específicos que son necesarios para atender la priorización adicional, la armonización y las actividades de coordinación. Estas consideraciones adicionales se abordan en el capítulo 14.

1.5 Scrum vs. Gestión tradicional de proyectos

La tabla 1-3 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	Solo 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	Autoorganizada	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	Cumplimiento del plan
Retorno de la inversión	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 1-3: Scrum vs. Gestión tradicional de proyectos

2. PRINCIPIOS

2.1 Introducción

Los principios de Scrum son la base en la que se funda el marco de trabajo de Scrum. 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 modificarse 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 marco de trabajo presentado en *Guía de los fundamentos de Scrum (Guía del SBOK®)*. Al mantener los principios intactos y usarlos apropiadamente se infunde confianza en el usuario del marco de trabajo de Scrum respecto al cumplimiento de los objetivos del proyecto. Los principios se consideran los lineamientos básicos para la aplicación del marco de trabajo de Scrum.

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

- Portafolios, programas, o proyectos de *cualquier* industria;
- Productos, servicios, o cualquier otro resultado que se entregue a los interesados el negocio;
- Proyectos de cualquier tamaño y complejidad.

El término “producto” en la *Guía del 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 solo seis miembros, hasta proyectos grandes y complejos con hasta varios cientos de miembros.

Este capítulo se divide en las siguientes secciones:

2.2 Guía de roles: Esta sección describe la sección o subsección más relevante para cada uno de los roles principales 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 Autoorganización: 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 autoorganizan. Esto se traduce en un mejor sentido de compromiso del equipo y responsabilidad compartida; esto, a su vez, genera un ambiente de trabajo innovador y creativo más adecuado 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 interesados del negocio 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 marco de trabajo de Scrum para entregar el máximo valor de negocio 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, la reunión de revisión del sprint y la reunión de retrospectiva 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 interesados del negocio para que el marco de trabajo de Scrum sea exitoso en cualquier organización.

2.3 Control de 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 ayuda al aprendizaje por medio de la experimentación cuando el problema no está bien definido o cuando no hay soluciones claras. 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 todos puedan observar las facetas de cualquier proceso de Scrum. 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 que pueden ver todos los interesados del negocio 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 de la liberación que se puede usar para coordinar el trabajo de múltiples equipos Scrum y demás interesados del negocio.
- Una clara visibilidad sobre el progreso del equipo a través del uso de Scrumboard, Burndown Chart y otros radiadores de información

- Reuniones de planificación del sprint donde el Equipo Scrum estima el esfuerzo necesario para entregar las historias de usuario de mayor prioridad y se compromete a trabajar en un conjunto de historias de usuario que deberán terminarse en el sprint.
- Daily Standups que se llevan a cabo durante el proceso de *Realizar el Daily 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 interesados del negocio.
- Reuniones de retrospectiva del sprint que se llevan a cabo después de las reuniones de revisión del sprint el último día del sprint, donde el Equipo Scrum dialoga sobre las oportunidades para mejorar en futuros sprints.

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

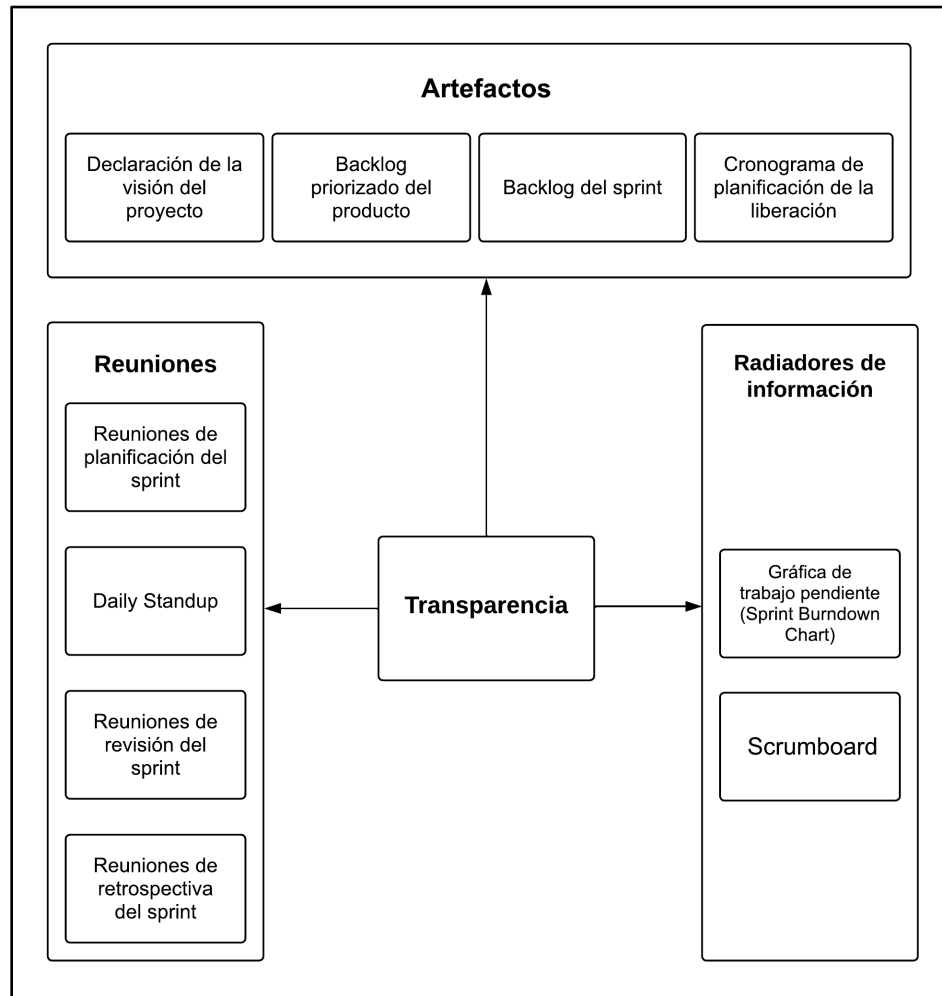


Figura 2-1: Transparencia en Scrum

2.3.2 Inspección

La inspección en Scrum se representa por medio de lo siguiente:

- El uso de un Scrumboard en común y otros radiadores de información que muestran el progreso del Equipo Scrum en completar las tareas del sprint actual.
- Obtener la retroalimentación del cliente y otros interesados del negocio durante los procesos de *Desarrollar épicas*, *Crear el backlog priorizado del producto* y *Realizar la planificación de la liberación*.
- 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.

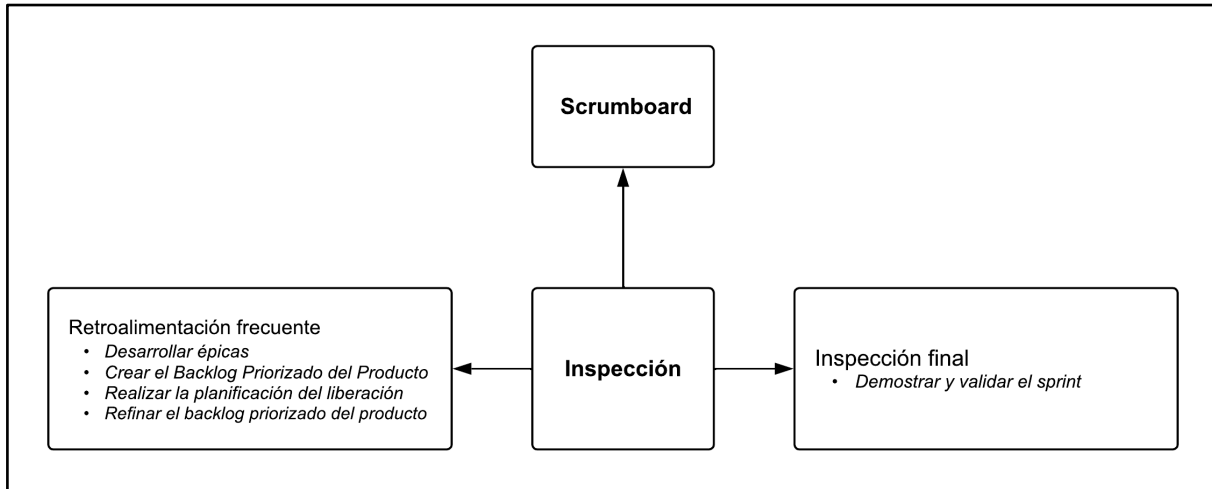


Figura 2-2: Inspección en Scrum

2.3.3 Adaptación

La adaptación se da cuando el equipo principal de Scrum y los interesados del negocio aprenden mediante la transparencia y la inspección, y después se adaptan al hacer mejoras en el trabajo que llevan a cabo. Algunos ejemplos de oportunidades para la adaptación en el marco de trabajo de Scrum son:

- En el Daily Standup, 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 a quienes tienen relativamente menos experiencia en el conocimiento del proyecto o de la tecnología.
- Se lleva a cabo la identificación del riesgo y se repite a lo largo del proyecto. Los riesgos que se identifican se convierten en entradas para varios procesos de Scrum, incluyendo: *Crear el backlog priorizado del producto*, *Refinar el backlog priorizado del producto* y *Demostrar y validar el sprint*.
- Las mejoras pueden resultar en solicitudes de cambios que se discuten y aprueban durante los procesos de *Desarrollar épicas*, *Crear el backlog priorizado del producto* y *Refinar el 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 *Refinar el 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 de la liberación, 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.

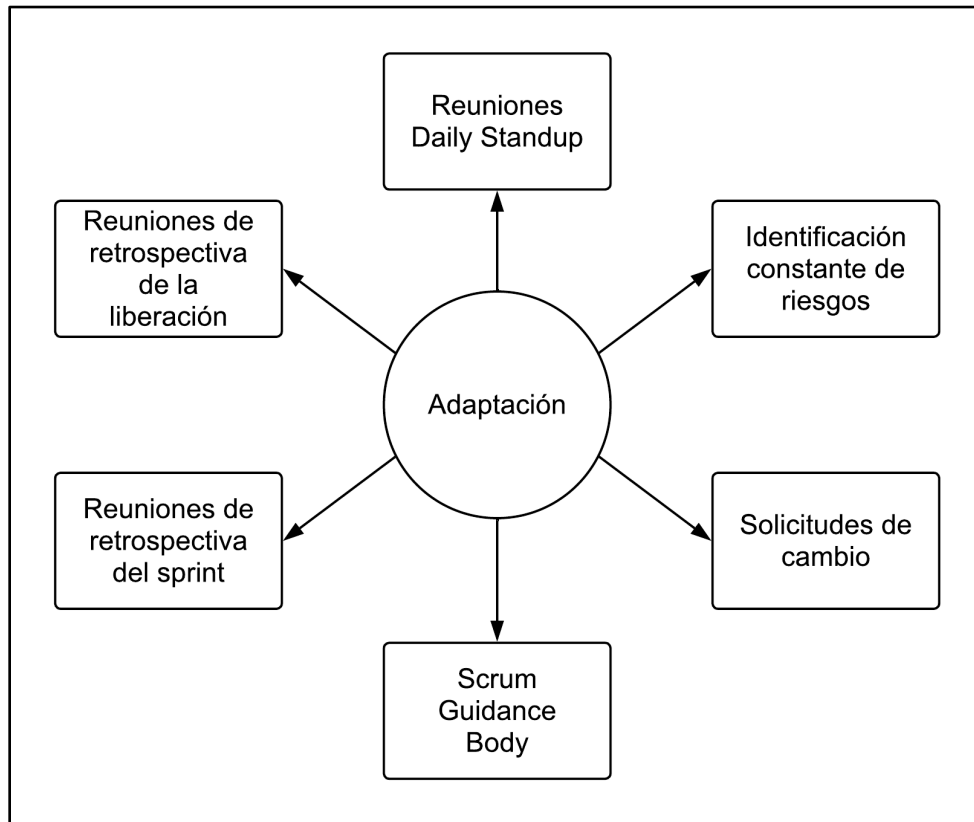


Figura 2-3: Adaptación en Scrum

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 a lo producido por el equipo y esto puede no ser conocido hasta muy tarde en el desarrollo del proyecto.

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

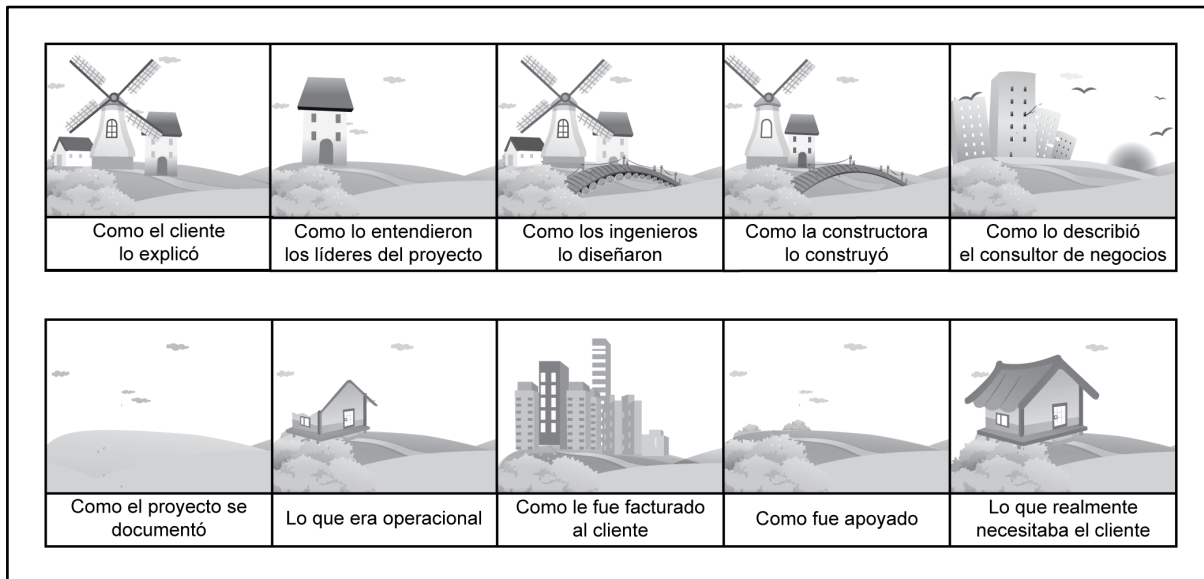


Figura 2-4: Retos en la gestión tradicional de proyectos

2.4 Autoorganización

Scrum sostiene que los empleados son automotivados y que buscan aceptar mayores responsabilidades. Por tanto, ofrecen mucho más valor cuando se autoorganizan.

El estilo de liderazgo preferente en Scrum es el “liderazgo colaborativo”, el cual enfatiza el logro de resultados, centrándose en las necesidades del Equipo Scrum. Véase la sección 3.10.3, donde se describen varios estilos de liderazgo y de gestión.

La autoorganización no significa que a los miembros del equipo se les permita actuar como quieran. Solo 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 interesados del negocio para perfeccionar los requisitos a medida que avanzan a través de los procesos de *Desarrollar épicas* 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 autoorganizado participa en la distribución y estimación de tareas durante los procesos de *Identificar de tareas* y *Estimar de tareas*. Durante estos procesos, cada miembro del equipo tiene la responsabilidad de determinar qué tipo de trabajo hará. El Equipo Scrum también le ayuda al Product Owner a identificar riesgos y dependencias. 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 enviabiles (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 ver que su arduo trabajo es aceptado por parte del cliente y los interesados del negocio.

2.4.1 Beneficios de la autoorganización

La autoorganización, como un principio fundamental 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 selección del mejor y más sencillo método para cumplir con requisitos establecidos.

Los principales objetivos de los equipos autoorganizados 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 *Actualizar el backlog del sprint*.
- Identificar tareas en forma independiente durante el proceso de *Identificación de tareas*
- Aplicar y aprovechar la experiencia de ser un equipo interdisciplinario al trabajar en las tareas durante el proceso de *Crear entregables*
- Entregar resultados tangibles que sean aceptados por el cliente y otros interesados del negocio durante el proceso de *Demostrar y validar el sprint*
- Resolver problemas individuales analizándolos durante el Daily Standup
- 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 autoorganizado.

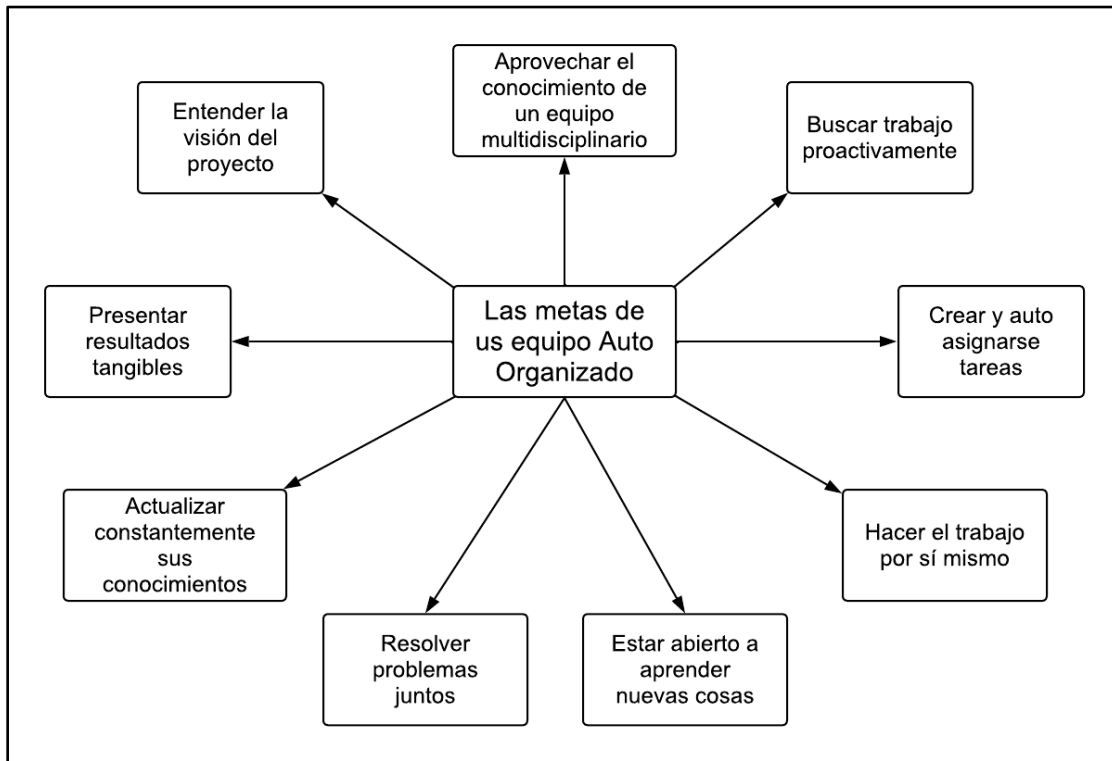


Figura 2-5: Objetivos de un equipo autoorganizado

2.5 Colaboración

La colaboración en Scrum se refiere a que el equipo principal de Scrum trabaja e interactúa con los interesados del negocio 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. Para lograr una colaboración total, es importante establecer confianza entre todos los miembros del equipo y entre el equipo y los interesados del negocio.

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

- **Conciencia:** Las personas que trabajan juntas deben de conocer el trabajo de los demás.
- **Articulación:** Los colaboradores deben distribuir el trabajo en unidades; dividir las unidades entre los miembros del equipo y después reintegrarlo cuando el trabajo esté hecho.
- **Apropiación:** 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 de Scrum

El *Manifiesto Ágil* (Fowler y Highsmith, 2001) hace énfasis en la “colaboración con el cliente sobre negociación contractual”. Por lo tanto, el marco de trabajo de Scrum adopta un enfoque donde los miembros del equipo principal de Scrum (Product Owner, Scrum Master y Equipo Scrum) colaboran entre sí y con los interesados del negocio 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:

- **Se reducen las solicitudes de cambio**

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 épicas* y *Crear el backlog priorizado del producto*, el Product Owner colabora con los interesados del negocio para crear la visión del proyecto, las épicas 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 interesados del negocio a través de un backlog priorizado del producto transparente para crear los entregables del proyecto. Los procesos de *Realizar Daily Standup*, *Refinar el 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.

- **Los riesgos se identifican y se mitigan con eficiencia**

Los riesgos se identifican y se atienden con eficiencia. Por ejemplo, los riesgos del proyecto se identifican y evalúan en los procesos de *Desarrollar épicas*, *Crear entregables* y *Realizar el 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 (tales como mitigación de riesgos) para atender riesgos de alta prioridad.

- **Aumenta la eficiencia**

Se logra el verdadero potencial del equipo. Por ejemplo, el proceso de *Realizar el 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 trabajan juntos para terminarla y cumplir con los objetivos acordados para completar el sprint.

- **Se incorpora la mejora continua**

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 siguiente sprint.

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

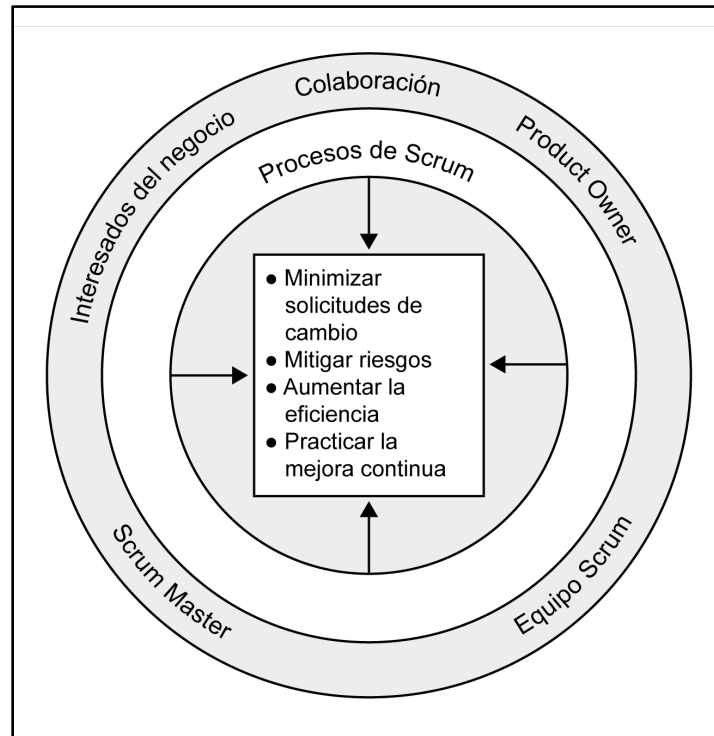


Figura 2-6: Beneficios de la colaboración en proyectos de Scrum

2.5.2 La importancia de la coubicación en la colaboración

En muchas de las prácticas de Scrum se requiere una comunicación de banda ancha. Por tanto, es preferible que los miembros del equipo estén coubicados. La coubicación 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 coubicación son los siguientes:

- Las preguntas se contestan rápidamente.
- Los problemas se solucionan en el momento.
- Hay menos fricción entre las interacciones.
- La confianza se gana con mucha más rapidez.

2.5.3 Colaboración en equipos dispersos

Aunque es preferible contar con equipos coubicados, en ocasiones el Equipo Scrum puede estar disperso. Varios equipos pueden estar trabajando desde distintas ubicaciones (incluso en distintos países) o incluso desde casa. Aun cuando los equipos están coubicados, deben estar siempre preparados para tener la flexibilidad de trabajar desde casa o a distancia debido a circunstancias atenuantes que pudieran afectar la capacidad del equipo para trabajar en un entorno coubicado. En tales situaciones, pudiera ser necesario asegurar que Scrum funcione con equipos dispersos.

2.5.3.1 Herramienta para un proyecto de Scrum

Se recomienda que las compañías busquen herramientas para un proyecto Scrum a fin de asegurar el trabajo disperso y garantizar que los equipos pueden trabajar productivamente, incluso cuando todos los miembros del equipo estén coubicados en el mismo centro de trabajo. Las herramientas deben tener la capacidad para:

- Definir bien todos los roles de Scrum y contar con la función de mensajería o colaboración a fin de que todos los miembros del equipo puedan interactuar entre sí;
- Crear y hacer uso de artefactos importantes de Scrum tales como el backlog priorizado del producto, el backlog del sprint, el Scrumboard, etc.
- Hacer que el flujo de trabajo avance por todos los procesos de Scrum que forman parte de las fases de inicio, planificación, implementación y liberación.
- Escalar hacia niveles de la organización o de la empresa (en caso de que Scrum se implemente dentro de grandes organizaciones o empresas).
- Programar reuniones de Scrum tales como las reuniones de planificación de la liberación, el Daily Standup, las reuniones de planificación del sprint, las reuniones de retrospectiva del sprint, las reuniones de revisión del backlog priorizado del producto, etc. Sin embargo, la reunión en sí se puede llevar a cabo por medio de una herramienta para videoconferencias.
- Ayudar a que los miembros del Equipo Scrum se comuniquen fácilmente entre sí en línea, ya sea uno a uno por medio de grupos dispersos o grupos de discusión (debido a que los equipos eficazmente distribuidos se comunican entre sí constantemente). Sin embargo, a diferencia de los equipos coubicados, los miembros de los equipos dispersos deben entender que otros integrantes pudieran no estar disponibles al mismo tiempo en los medios de comunicación instantánea.
- Anotar las lecciones aprendidas (de las retrospectivas, etc.) con informes apropiados que se generan en el momento.
- Incorporar la automatización a fin de que cualquier plantilla o guía del Scrum Guidance Body esté a disposición de todos los equipos en la organización (por ejemplo: la definición de listo o la definición de terminado).
- Ayudar a que el SGB evalúe el comportamiento relacionado a Scrum (como el número máximo de integrantes, la duración del sprint, etc.).

- Ofrecer la capacidad para clonar proyectos similares, épicas e historias de usuario. Esto permitirá que los integrantes del Equipo Scrum dediquen menos tiempo en documentación innecesaria y puedan aprender de experiencias de trabajo previamente realizado. Esto resulta especialmente útil cuando los Equipo Scrum utilizan procesos similares de implementación para crear categorías idénticas de productos. Por ejemplo: compañías de publicidad que desarrollan anuncios impresos para distintos clientes; empresas constructoras que elaboran planos para carreteras similares, etc.

Los beneficios de utilizar una herramienta para un proyecto de Scrum en un equipo disperso son:

- Facilita el trabajo de los integrantes del equipo en cualquier momento y en cualquier lugar.
- Automatiza los informes, los diagramas, los calendarios, los flujos de trabajo, etc.
- Aplicación de lineamientos similares en la organización mediante la automatización de las recomendaciones del SGB.
- Aumenta la eficiencia debido a la disminución de documentación innecesaria y repetitiva al hacer clonaciones de proyectos similares (por ejemplo: clonación de épicas e historias de usuario).
- Se trabaja con un equipo más diverso (en ocasiones se trabaja desde distintos países) que ofrece su perspectiva regional y su experiencia.
- Menos retos logísticos en comparación a la necesidad de asegurar que todas las personas trabajen desde una sola ubicación. Se ahorran tiempos y costos en los gastos de viajes, centros de trabajo costosos, etc.

En los equipos dispersos es importante prestar especial atención a los principios de Scrum a fin de asegurar de que se implementen en los equipos dispersos. Debe haber énfasis en la capacidad de trabajar en forma colaborativa y transparente en un ambiente de confianza.

2.6 Priorización basada en valor

El marco de trabajo de Scrum se guía por la finalidad de ofrecer el máximo valor de negocio en un mínimo periodo 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 director del proyecto, 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 establecer fechas límite para las tareas delegadas y el uso de matrices de priorización.

Sin embargo, Scrum utiliza la priorización basada en valor como uno de los principios básicos que impulsa la estructura y funcionalidad de todo el marco de trabajo de Scrum; ayuda a que los proyectos se beneficien mediante la capacidad de adaptación y el desarrollo iterativo del producto o servicio. Y lo 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 recibe los requerimientos del negocio del cliente, estos se escriben en forma de épicas e historias de usuario (un formato específico para anotar los requisitos). El Product Owner trabaja con el cliente y otros interesados del negocio para establecer cuáles requisitos del negocio ofrecen el mayor valor de negocio. 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. El Product Owner debe entender lo que el cliente quiere y valora a fin de ordenar las historias de usuario en una lista de mayor a menor prioridad. A esta lista se le llama backlog priorizado del producto y debe incluir todos los requisitos del proyecto. Para priorizar un backlog es necesario establecer la importancia de cada historia de usuario. Los requisitos de alto valor se identifican y trasladan al principio 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 el backlog priorizado del producto* y *Refinar el 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. Estos riesgos se deben tener en cuenta al priorizar las historias de usuario con enfoque basado en el valor (véase el capítulo sobre 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 interesados del negocio 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.

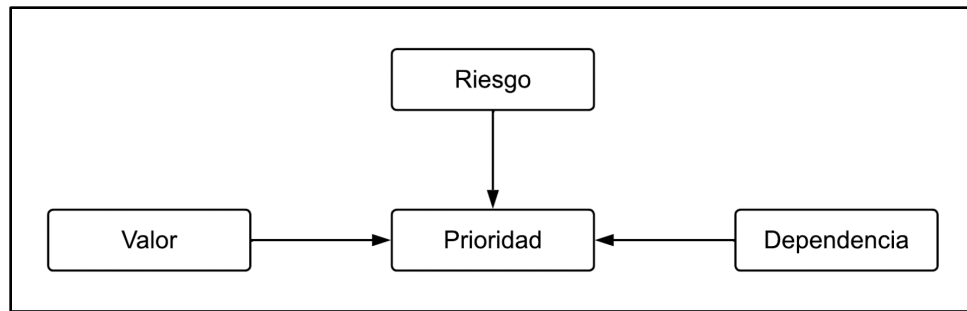


Figura 2-7: Priorización basada en valor

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 los límites de tiempo, Scrum introduce un concepto de *Time-boxing* (asignación de un bloque de tiempo), que propone fijar una cierta cantidad de tiempo a cada proceso y actividad en un proyecto de Scrum. Esto garantiza que los miembros del Equipo Scrum no dediquen mucho o poco tiempo a un trabajo determinado, y que no desperdicien su tiempo y energía en un trabajo que no entienden claramente.

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

- Eficiencia en el proceso de desarrollo
- Menos gastos generales
- Alta velocidad para los equipos
- Equipos más enfocados
- Equipos mejor preparados

El Time-boxing puede utilizarse en muchos procesos de Scrum, por ejemplo, en el proceso de *Realizar el 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 (*gold-plating*).

El Time-boxing es una práctica muy importante en Scrum y debe aplicarse con cuidado. Un 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 cuatro 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, y para brindar mayor flexibilidad al cambio, la duración del sprint debe ser lo más corta posible.

El sprint debe ser también lo suficientemente extenso para que el equipo pueda crear un producto liberable que pueda ser revisado y aprobado por el Product Owner.

- **Reunión de planificación del sprint:** Esta reunión se lleva a cabo antes de cada sprint como parte de los procesos de *Comprometer historias de usuario*, *Identificar tareas*, *Estimar tareas* y *Actualizar el backlog del sprint*. Tiene un time-box de dos horas por cada semana de duración del sprint. Por ejemplo, en un sprint de un mes (cuatro semanas), el time-box de la reunión de planificación del sprint debe ser de ocho horas.

La reunión de planificación del sprint tiene dos objetivos:

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 tareas con esfuerzo estimado se incluyen en el backlog priorizado del producto al que se le dará seguimiento.
- **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 actualmente?

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

- **Reunión de revisión del sprint:** La reunión de revisión del sprint tiene un time-box de una hora por cada semana de duración del sprint. Por ejemplo, en un sprint de cuatro semanas, el time-box de la reunión de revisión del sprint debe ser de cuatro horas. Durante esta reunión, 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:** La reunión de retrospectiva del sprint tiene un time-box de una hora por cada semana de duración del sprint. Por ejemplo, en un sprint de cuatro semanas, el time-box de la reunión de retrospectiva del sprint debe ser de cuatro horas 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 con 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.

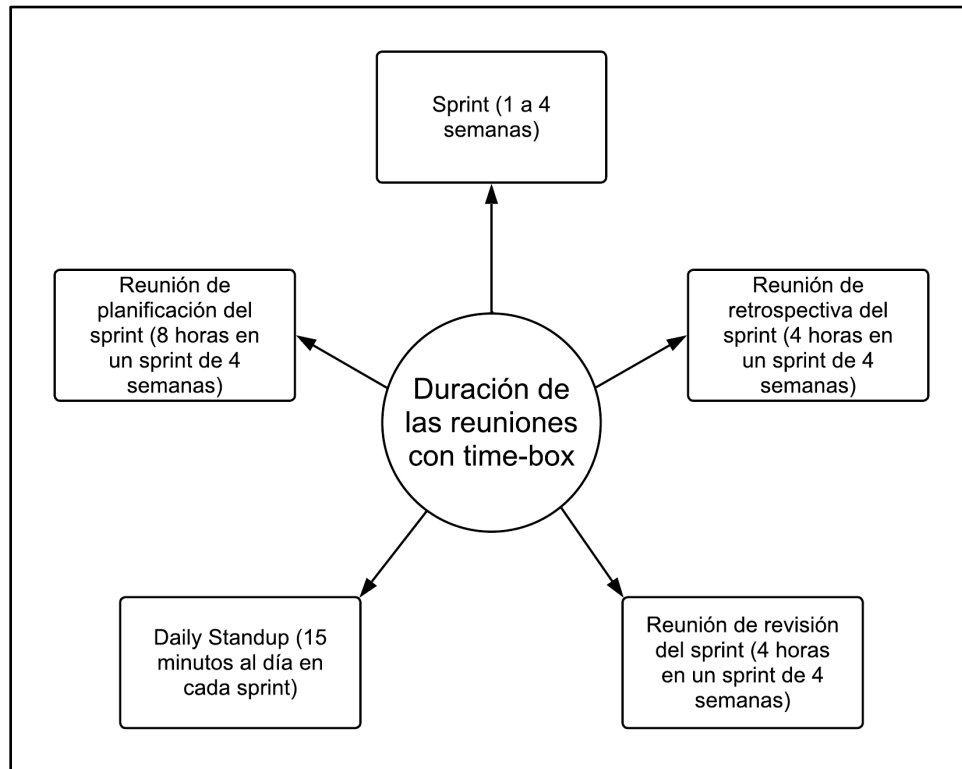


Figura 2-8: Duración del Time-Box para las reuniones de Scrum

2.8 Desarrollo iterativo

El marco de trabajo de Scrum está guiado por el objetivo de ofrecer el máximo valor de negocio 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, es posible que el cliente no pueda definir requisitos muy concretos o no esté seguro de cómo se verá 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 épicas. Las épicas generalmente son muy grandes como para que los equipos las completen en un solo 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 historias de usuario* 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 de la inversión, 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 con qué valor serán entregados en cada sprint.

Posteriormente, los procesos de *Identificar tareas*, *Estimar tareas* y *Actualizar el backlog del sprint* producen el backlog del sprint, 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 autoorganiza, teniendo como objetivo el crear entregables del sprint a partir de las historias de usuario que están en el backlog del sprint. En grandes proyectos, varios equipos multidisciplinarios 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*. El beneficio del desarrollo iterativo es la posibilidad de corregir el rumbo, ya que todas las personas involucradas entienden mejor lo que se debe desarrollar como parte del proyecto e incorporan este aprendizaje en forma iterativa. De esta forma, el tiempo y el esfuerzo necesarios para llegar al punto final se reducen considerablemente y el equipo produce entregables más adecuados el ambiente de negocio final.

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.

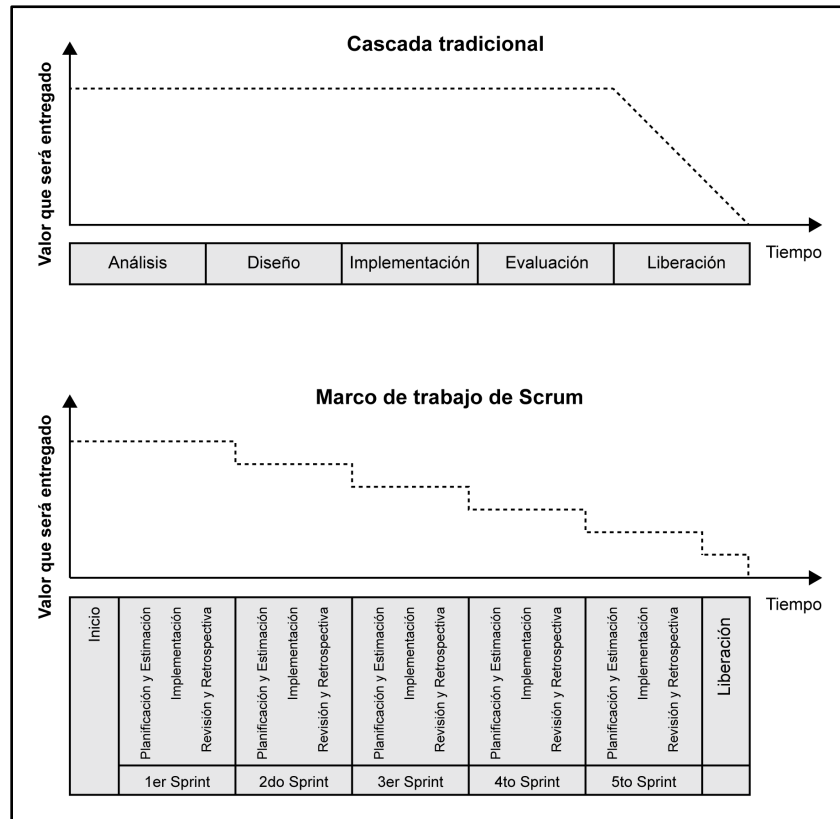


Figura 2-9: Scrum vs Cascada tradicional

2.9 Scrum vs. Gestión tradicional de proyectos

El enfoque de la gestión tradicional de proyectos está en llevar a cabo la planificación detallada del proyecto por adelantado haciendo énfasis en fijar el alcance, el costo y el tiempo gestionar esos parámetros. En ocasiones, la gestión tradicional de proyectos puede llevar a situaciones en las que el cliente no esté satisfecho a pesar del éxito del proyecto.

El marco de trabajo 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 que intentar mapear completamente y planificar en un entorno cambiante no resulta suficiente. Por lo tanto, Scrum fomenta 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 se puedan entregar.

Para entregar la mayor cantidad de valor en el menor tiempo posible, Scrum promueve la priorización y el Timeboxing en vez de fijar el alcance, el costo y el cronograma de un proyecto. Una característica importante de Scrum es la autoorganización, la 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 principales y los roles no centrales, y cómo formar equipos Scrum de alto rendimiento.

La organización, tal como se define en *Guía de los fundamentos de Scrum (Guía del SBOK®)*, se aplica a lo siguiente:

- Portafolios, programas o proyectos de cualquier sector;
- Productos, servicios o cualquier otro resultado que se entregue a los interesados del negocio;
- Proyectos de cualquier tamaño y complejidad.

En la *Guía del 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 solo seis miembros por equipo, hasta proyectos grandes y complejos con cientos de miembros.

Este capítulo está dividido en las siguientes secciones:

3.2 Guía de roles: Esta sección identifica las secciones o subsecciones importantes para un Product Owner, un Scrum Master y un Equipo Scrum.

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

3.4 Product Owner: Esta sección presenta las principales responsabilidades del Product Owner con relación a un proyecto de 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 énfasis en las principales responsabilidades del Equipo Scrum en el contexto de un proyecto de Scrum, programa o portafolio.

3.7 Scrum en proyectos, programas y portafolios: Esta sección se centra en cómo se puede adaptar y utilizar el marco de trabajo de Scrum 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 de negocio y con la gestión de programas.

3.8 Responsabilidades: Esta sección describe 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 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 enfocarse principalmente en las secciones 3.3, 3.6 y 3.8.

3.3 Roles en un proyecto Scrum

Es muy importante entender los roles y las responsabilidades de Scrum a fin de garantizar la implementación exitosa de los proyectos Scrum.

Los roles de Scrum se dividen en dos categorías:

1. **Roles principales:** Los roles principales (*core roles*) 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 secundarios:** Los roles secundarios (*non-core roles*) 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 secundarios también deben tenerse en cuenta en cualquier proyecto de Scrum.

3.3.1 Roles principales

En Scrum hay tres roles principales que, en última instancia, tienen la responsabilidad de cumplir con los objetivos del proyecto. Los roles principales 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 demás.

1. **Product Owner:** El Product Owner es la persona responsable de maximizar el valor del negocio en 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.
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 y 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 director del proyecto en un modelo tradicional de cascada en la gestión de proyectos, en el que el director trabaja como gerente o líder del mismo. El Scrum Master solo 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.

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 principales del Equipo Scrum.

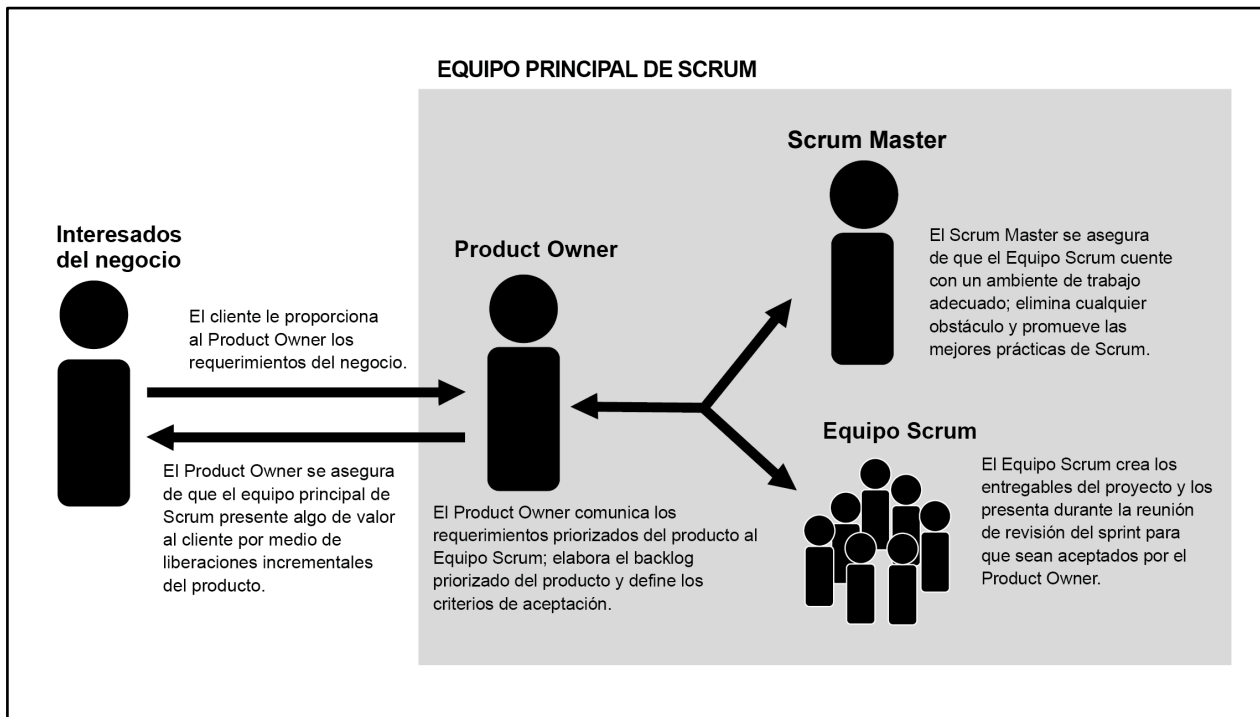


Figura 3-1: Roles de Scrum: Resumen

3.3.2 Roles secundarios

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

Los roles secundarios pueden ser los siguientes:

1. **Interesados del negocio:** Interesados del negocio (*business stakeholders*) 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 información y facilitar la creación del producto del proyecto, servicio, o cualquier otro resultado. Los interesados del negocio influyen en el proyecto a lo

largo del desarrollo del mismo. Los interesados del negocio también pueden desempeñar un rol en los procesos importantes de Scrum tales como *Desarrollar épicas*, *Crear el backlog priorizado del producto*, *Realizar la planificación de la liberación* y *Retrospectiva del sprint* y otros procesos importantes en Scrum. En Scrum se necesita el apoyo total de los interesados del negocio en el proyecto.

El Product Owner tiene la responsabilidad de mantener involucrados a los interesados del negocio. Para conservar el apoyo de los interesados y mantenerlos involucrados se recomienda lo siguiente:

- Asegurar la participación y la colaboración eficaz de los interesados del negocio;
- Evaluar constantemente el impacto del negocio;
- Mantener una comunicación constante con los interesados del negocio;
- Gestionar las expectativas de los interesados del negocio.

En ocasiones, una misma persona u organización tendrán múltiples roles de interesados del negocio. Por ejemplo: el patrocinador y el cliente pudieran ser la misma persona. A continuación, se definen los roles de los interesados del negocio relevantes en un proyecto de Scrum:

- **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:** El patrocinador es la persona o la organización que provee recursos y apoyo para el proyecto. El patrocinador es también el interesado del negocio, a quien todos le deben rendir cuentas al final.

Los patrocinadores buscan entender el balance final relacionado al producto o servicio y generalmente les preocupa más el resultado y no las tareas individuales. Es importante que los patrocinadores que financian el proyecto entiendan claramente lo siguiente:

- Los beneficios de la implementación de Scrum;
 - Las fechas límites y los costos estimados de los proyectos de Scrum;
 - Los riesgos generales de los proyectos de Scrum y los pasos para mitigarlos;
 - Las fechas esperadas de liberación y entregables finales.
2. **Servicios de apoyo:** Los servicios de apoyo son grupos internos y externos que apoyan o se vean afectados por el proyecto de Scrum. Por ejemplo: capacitación, logística, publicidad, finanzas, infraestructura, etc.
 3. **Proveedores:** Los proveedores son personas u organizaciones externas que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.
 4. **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 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 interesados del negocio 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 interesados del negocio, 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 interesados del negocio, 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	<ul style="list-style-type: none"> Define la visión del proyecto Ayuda a crear el acta de constitución del proyecto y su presupuesto
8.2 Identificar al Scrum Master y a los interesados del negocio	<ul style="list-style-type: none"> Ayuda a finalizar la elección del Scrum Master para el proyecto Identifica a los interesados del negocio
8.3 Formar el Equipo Scrum	<ul style="list-style-type: none"> 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 los Scrum Masters
8.4 Desarrollar épicas	<ul style="list-style-type: none"> Crea épicas y personajes
8.5 Crear el backlog priorizado del producto	<ul style="list-style-type: none"> Prioriza los elementos en el backlog priorizado del producto Define los criterios de terminado y cumple la definición de listo
8.6 Realizar la planificación de la liberación	<ul style="list-style-type: none"> Elabora el cronograma de planificación de la liberación Ayuda a determinar la duración del sprint
9.1 Crear historias de usuario	<ul style="list-style-type: none"> Responsable de crear historias de usuario Define los criterios de aceptación de cada historia de usuario
9.2 Estimar historias de usuario	<ul style="list-style-type: none"> Explica las historias de usuario
9.3 Comprometer historias de usuario	<ul style="list-style-type: none"> Trabaja con el Equipo Scrum para que comprometa a las historias de usuario
9.4 Identificar tareas	<ul style="list-style-type: none"> Explica las historias de usuario al Equipo Scrum mientras elabora la lista de tareas
9.5 Estimar tareas	<ul style="list-style-type: none"> Brinda orientación y aclaraciones al Equipo Scrum en la estimación de los esfuerzos para las tareas
9.6 Actualizar el backlog del sprint	<ul style="list-style-type: none"> Explica los requerimientos al Equipo Scrum mientras el equipo elabora el backlog del sprint.
10.1 Crear entregables	<ul style="list-style-type: none"> Aclara los requerimientos del negocio al Equipo Scrum.
10.3 Refinar el backlog priorizado del producto	<ul style="list-style-type: none"> Refina el backlog priorizado del producto.
11.1 Demostrar y validar el sprint	<ul style="list-style-type: none"> Acepta o rechaza los entregables Proporciona la retroalimentación necesaria al Scrum Master y a los equipos Scrum Actualiza el plan de liberación y el backlog priorizado del producto
12.1 Enviar entregables	<ul style="list-style-type: none"> Ayuda a enviar las liberaciones del producto y se coordina con el cliente
12.2 Retrospectiva de la liberación	<ul style="list-style-type: none"> 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 son:

- Establecer 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 a los usuarios 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 de la inversión en general.
- Evaluar la viabilidad y garantizar la entrega del producto o servicio.

El Product Owner no siempre representa a un negocio o cliente externo. Por ejemplo, en un proyecto de informática, los requisitos no funcionales, tales como mejorar el desempeño, la escalabilidad, la capacidad de prueba, la confiabilidad, la seguridad de la información y el cumplimiento, también pueden ser propiedad de grupos de tecnología al interior de la compañía. En tales casos, el Product Owner pudiera ser un arquitecto técnico, el líder técnico, etc.

3.4.1 La voz del cliente

Como representante del cliente y de otros interesados del negocio, al Product Owner se le conoce como “la voz del cliente”, ya que 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.

3.5 Scrum Master

El Scrum Master es el “líder de apoyo” del Equipo Scrum que modera y facilita las interacciones del equipo como su *coach* y motivador. Este rol es responsable de asegurar 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 a los interesados del negocio	<ul style="list-style-type: none"> • Ayuda a identificar a los interesados del negocio en el proyecto
8.3 Formar el Equipo Scrum	<ul style="list-style-type: none"> • Facilita la selección del Equipo Scrum • Facilita la creación del plan de colaboración y el plan de desarrollo del equipo • Garantiza que los recursos de respaldo estén disponibles para el funcionamiento del proyecto sin problemas
8.4 Desarrollar épicas	<ul style="list-style-type: none"> • Facilita la creación de épicas y personajes
8.5 Crear el backlog priorizado del producto	<ul style="list-style-type: none"> • Ayuda al Product Owner a crear backlog priorizado del producto; a definir de los criterios de terminado y a cumplir la definición de listo.
8.6 Realizar la planificación de la liberación	<ul style="list-style-type: none"> • Coordina la creación del cronograma de planificación de la liberación • Ayuda al Product Owner y al Equipo Scrum a establecer la duración del sprint
9.1 Crear historias de usuario	<ul style="list-style-type: none"> • Facilita la creación de historias de usuario y sus criterios de aceptación
9.2 Estimar historias de usuario	<ul style="list-style-type: none"> • Facilita las reuniones del Equipo Scrum para estimar historias de usuario
9.3 Comprometer historias de usuario	<ul style="list-style-type: none"> • Facilita las reuniones del Equipo Scrum para comprometer historias de usuario
9.4 Identificar tareas	<ul style="list-style-type: none"> • Ayuda al Equipo Scrum a crear la lista de tareas para el siguiente sprint
9.5 Estimar tareas	<ul style="list-style-type: none"> • Ayuda al Equipo Scrum a estimar el esfuerzo necesario para completar las tareas acordadas para el sprint
9.6 Actualizar el backlog del sprint	<ul style="list-style-type: none"> • Ayuda al Equipo Scrum a desarrollar el backlog del sprint y el Sprint Burndown Chart
10.1 Crear entregables	<ul style="list-style-type: none"> • Ayuda al Equipo Scrum a crear los entregables acordados para el sprint • Ayuda a actualizar el Scrumboard y la lista de impedimentos
10.2 Realizar el Daily Standup	<ul style="list-style-type: none"> • Se asegura de que el Scrumboard y la lista de impedimentos estén actualizados
10.3 Refinar el backlog priorizado del producto	<ul style="list-style-type: none"> • Organiza las reuniones de revisión del backlog priorizado del producto
11.1 Demostrar y validar sprints	<ul style="list-style-type: none"> • Facilita la presentación de los entregables completados por el Equipo Scrum para la aprobación del Product Owner
11.2 Retrospectiva del sprint	<ul style="list-style-type: none"> • Garantiza que exista un ambiente ideal para el Equipo Scrum del proyecto en los sucesivos sprints
12.2 Retrospectiva de la liberación	<ul style="list-style-type: none"> • 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

3.6 Equipo Scrum

Al Equipo Scrum en ocasiones se le conoce como equipo de desarrollo, ya que es responsable del desarrollo del producto, servicio o de cualquier otro resultado. Consiste en un grupo de personas autoorganizadas que trabajan en las historias de usuario en el backlog del sprint 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	<ul style="list-style-type: none"> Proporciona aportes para la creación del plan de colaboración y del plan de desarrollo del equipo.
8.4 Desarrollar épicas	<ul style="list-style-type: none"> Asegura una comprensión clara de la épicas y personajes.
8.5 Crear el backlog priorizado del producto	<ul style="list-style-type: none"> Entiende las historias de usuario en el backlog priorizado del producto.
8.6 Realizar la planificación de la liberación	<ul style="list-style-type: none"> Acuerda 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	<ul style="list-style-type: none"> Proporciona información al Product Owner en la creación de historias de usuario
9.2 Estimar historias de usuario	<ul style="list-style-type: none"> Estima las historias de los usuarios aprobadas por el Product Owner
9.3 Comprometer historias de usuario	<ul style="list-style-type: none"> Compromete las historias de usuario a realizarse en un sprint
9.4 Identificar tareas	<ul style="list-style-type: none"> Desarrolla una lista de tareas con base en las historias de usuario y dependencias acordadas
9.5 Estimar tareas	<ul style="list-style-type: none"> Estima el esfuerzo de las tareas identificadas y, de ser necesario, actualiza la lista de tareas.
9.6 Actualizar el backlog del sprint	<ul style="list-style-type: none"> Define las historias de usuario y las tareas que serán incluidas en el backlog del sprint y a las que se les da seguimiento en el Sprint Burndown Chart
10.1 Crear entregables	<ul style="list-style-type: none"> Elabora los entregables Identifica riesgos y ejecuta acciones de mitigación de riesgos Identifica los impedimentos a los que se les dará seguimiento en la lista de impedimentos
10.2 Realizar el Daily Standup	<ul style="list-style-type: none"> Actualiza el Scrumboard durante cada sprint Discute los problemas que enfrenta cada miembro y busca soluciones para motivar al equipo
10.3 Refinar el backlog priorizado del producto	<ul style="list-style-type: none"> Participa en las reuniones de revisión del backlog priorizado del producto
11.1 Demostrar y validar sprints	<ul style="list-style-type: none"> Muestra los entregables terminados al Product Owner para su aprobación
11.2 Retrospectiva del sprint	<ul style="list-style-type: none"> Identifica oportunidades de mejora en el sprint actual y decide si está de acuerdo sobre las posibles mejoras viables para el próximo sprint
12.2 Retrospectiva de la liberación	<ul style="list-style-type: none"> Participa en la reunión de retrospectiva de la liberación

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

3.6.1 Selección de personal

La figura 3-2 enumera las características deseables para los roles principales de Scrum.

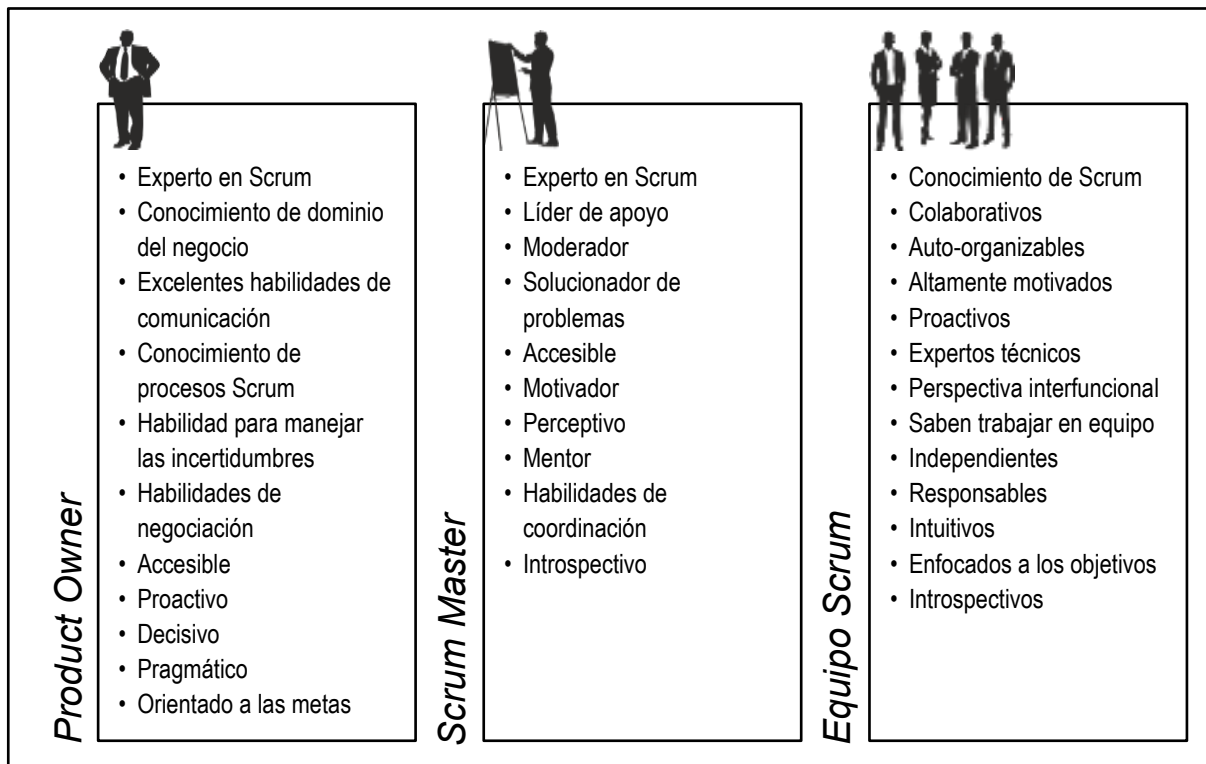


Figura 3-2: Características deseadas de los roles principales de Scrum

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. La meta es contar con suficientes personas en un equipo para hacer el trabajo, pero que siga siendo suficientemente pequeño para lograr una eficaz comunicación y colaboración dentro del equipo. Los equipos más pequeños pudieran enfrentar desventajas. Una posible desventaja 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 El funcionamiento de Scrum en un proyecto grande

Los procesos fundamentales de Scrum descritos en los capítulos del 8 al 12 son válidos para los proyectos de Scrum que cuentan con un Product Owner, un Scrum Master y un Equipo Scrum. Estos generalmente se consideran proyectos de Scrum pequeños.

Al trabajar en grandes proyectos que requieren del esfuerzo de cuatro o más equipos de Scrum con múltiples Product Owners y múltiples Scrum Masters, los procesos fundamentales de Scrum definidos en los capítulos del 8 al 12 siguen siendo válidos, aunque tal vez sean necesarias consideraciones adicionales y actualizaciones a las entradas, herramientas y salidas. Esto puede incluir necesidades adicionales de coordinación y sincronización. El impacto de los procesos fundamentales de Scrum al escalar Scrum en grandes proyectos se describe en el capítulo 13.

La definición de lo que constituye un proyecto grande generalmente depende de la organización o de la complejidad de los proyectos emprendidos. Un criterio clave para saber si un proyecto se considera pequeño en vez de grande es si requiere de múltiples Scrum Masters o múltiples Product Owners. Si el proyecto requiere solo un Scrum Master y un Product Owner, generalmente estos pueden manejar cualquier actividad adicional de comunicación y sincronización que requiera el proyecto.

Algunos de los motivos tras la necesidad de utilizar entradas, herramientas y salidas adicionales para grandes proyectos son:

Product Owners

- La necesidad de colaborar entre Product Owners al trabajar con interesados del negocio; al refinar el backlog priorizado del producto y al trabajar con múltiples Equipo Scrum.

También es importante destacar que, al escalar Scrum en los grandes proyectos, tal vez sean necesarios servicios adicionales de apoyo, tales como arquitectos, directores de producto, cumplimiento, seguridad de la información, gobernabilidad, etc.

Scrum Masters

- La necesidad de colaboración entre Scrum Masters al atender impedimentos y sincronizar el trabajo de múltiples Equipo Scrum.

Equipos Scrum

- Mayor interacción y dependencias entre los equipos Scrum, ya que la complejidad aumenta en los grandes proyectos.
- La necesidad de gestionar conflictos, resolver problemas y establecer prioridades entre 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.

- La necesidad de establecer un ambiente o área de trabajo 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 la liberación en un proyecto grande.

3.7.2 Roles principales adicionales en proyectos grandes

Al escalar Scrum en grandes proyectos, se necesitan los siguientes roles principales:

3.7.2.1 Chief Product Owner

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 o 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 del 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 que compiten, presentados por los Product Owners con base en su interacción los interesados del negocio. 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. El Chief Product Owner también interactúa con el Program Product Owner para asegurar que los grandes proyectos coincidan con las metas y objetivos del programa. Los Chief Product Owners deben consultar las secciones de la *Guía del 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.

3.7.2.2 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 13.3.5). No existe una jerarquía entre los Scrum Masters: todos son compañeros. El Chief Scrum Master solo trabaja en un nivel de múltiples equipos, mientras que los Scrum Masters trabajan al nivel de un solo equipo.

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

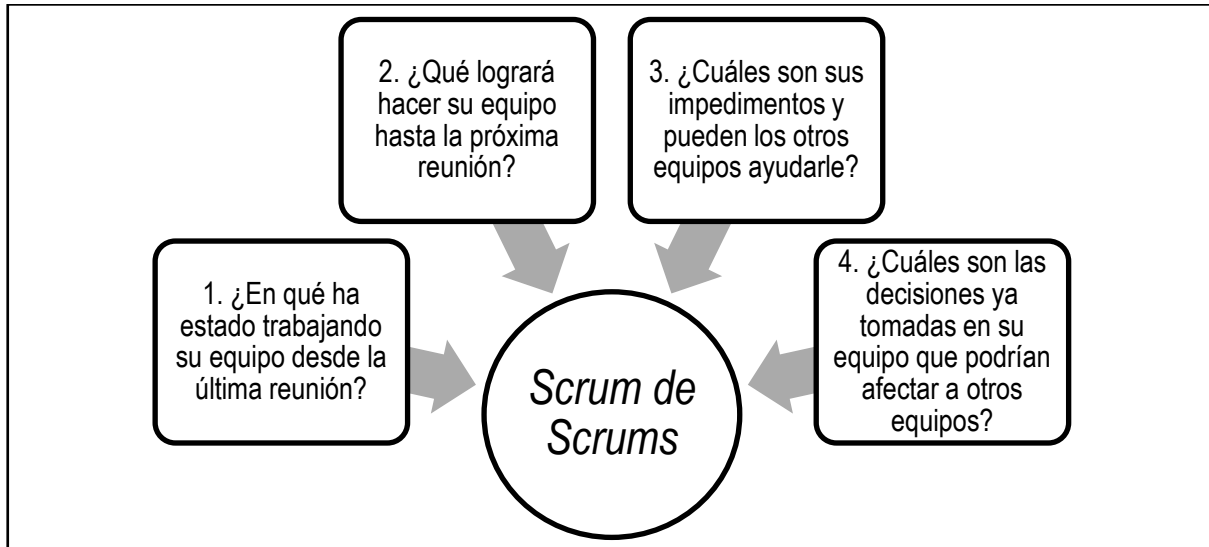


Figura 3-3: Preguntas durante la reunión de Scrum de Scrums

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 propicio en el proyecto para los equipos Scrum, el Chief Scrum Master también colabora 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. 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 que un proyecto grande coincida con las metas y objetivos del programa.

Los Chief Scrum Masters deben consultar las secciones de la *Guía del 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.

3.7.3 El funcionamiento de Scrum en una empresa

Para aplicar las prácticas de Scrum en proyectos que forman parte de un modelo empresarial es necesario entender la configuración y la forma de gobierno de la empresa. Esto generalmente se puede manejar por medio de programas y portafolios.

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. 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 o proyectos relacionados con la finalidad de entregar resultados de negocio tal como se define en la declaración de la visión del portafolio. El backlog priorizado del portafolio incorpora el backlog priorizado del producto de todos los programas en el portafolio. El backlog priorizado del producto de proyectos independientes también forma parte del portafolio.

Los problemas y asuntos que se enfrentan al utilizar Scrum al interior de un programa o portafolio involucran principalmente la coordinación entre muchos equipos. Esto puede llevar al fracaso si no se maneja cuidadosamente. Las herramientas utilizadas para la comunicación deben extrapolarse a fin de que coincidan con los requerimientos de los muchos equipos involucrados en un programa o portafolio. Cada Equipo de Scrum debe abordar no solo la comunicación interna, sino también la comunicación externa con otros equipos y con los interesados del negocio relevantes para el programa o portafolio.

Al aplicar Scrum en la gestión de proyectos en el contexto de un programa o portafolio, es altamente recomendado que se cumplan los principios generales de Scrum descritos en esta publicación. Se entiende que, con el fin de adaptar las interdependencias y actividades generales del programa o portafolio, tal vez sea necesario hacer pequeños ajustes a las herramientas y a la estructura organizacional. Si existe un Scrum Guidance Body, este puede ser responsable de analizar la organización en sus distintos niveles a fin de entender y definir la aplicación apropiada de Scrum y de actuar como órgano de consulta para quienes trabajan en un proyecto, programa o portafolio.

Los portafolios y programas cuentan con equipos independientes con distintos objetivos. Los equipos de gestión de programas buscan entregar capacidades y alcanzar ciertas metas que contribuyan al cumplimiento de objetivos específicos del programa. En contraste, el equipo del portafolio debe equilibrar los objetivos de varios programas a fin de lograr las metas estratégicas de la organización en su conjunto.

Es importante destacar que, al escalar Scrum en una empresa, tal vez sean necesarios servicios adicionales de apoyo, tales como arquitectos, cumplimiento, seguridad de la información, gobernabilidad, etc.

El capítulo 14 incluye información detallada sobre el escalamiento de Scrum en las empresas.

3.7.4 Roles principales adicionales en una empresa

3.7.4.1 Program Product Owner

El rol del Program Product Owner es similar al rol del Product Owner, a diferencia de que busca cumplir las necesidades del programa o de la unidad de negocios en vez de las necesidades de un solo Equipo Scrum.

El Program Product Owner define los objetivos estratégicos y las prioridades del programa. Es responsable de maximizar el valor de negocio en un programa articulando claramente de los requisitos del cliente y manteniendo la justificación del negocio en el programa. El Program Product Owner también gestiona el Backlog del Producto del Programa. Es responsable de impulsar la creación y refinamiento de los entregables al nivel del programa y requiere la coordinación entre los proyectos subyacentes en el programa. También es responsable de coordinarse con otros programas con dependencias o planes de liberación compartidos. El Program Product Owner también se coordina con el Portfolio Product Owner para garantizar que el programa esté en sintonía con el portafolio.

El Program Product Owner también se coordina con el Portfolio Product Owner que corresponda a fin de garantizar que el programa esté en sintonía con el portafolio correspondiente. El Program Product Owner interactúa con el Portfolio Product Owner para asegurar que el programa coincida con los objetivos del portafolio. También participa en el nombramiento de Product Owners en proyectos individuales y se asegura de que la visión, los objetivos, los resultados y las liberaciones de proyectos individuales coincidan con los del programa.

Los Program Product Owners deben consultar las secciones de la *Guía del 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.

3.7.4.2 Portfolio Product Owner

El rol del Portfolio Product Owner es similar al rol del Product Owner y al rol del Program Product Owner, a diferencia de que busca cumplir las necesidades del portafolio o de la unidad de negocios en vez de las necesidades de un solo Equipo Scrum o las necesidades de un programa.

El Portfolio Product Owner toma las decisiones a nivel de portafolio. Tiene una mejor perspectiva para decidir cómo organizar la empresa para cumplir con la visión. El Portfolio Product Owner es responsable de la creación y refinación del Backlog del Producto del Portafolio.

Los Portfolio Product Owners deben consultar las secciones de la *Guía del 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.

3.7.4.3 Program Scrum Master

El rol del Program Scrum Master es similar al rol del Scrum Master, a diferencia de que busca cumplir las necesidades del programa o de la unidad de negocios en vez de las necesidades de un solo Equipo Scrum. El Program Scrum Master es un facilitador que se asegura de que todos los equipos del proyecto cuenten con un ambiente adecuado 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 en todo el programa. Es un facilitador; resuelve problemas y elimina impedimentos al nivel del programa. También es responsable de la coordinación entre todos los proyectos en el programa y de coordinarse con otros programas con planes de liberación o dependencias compartidas.

El Program Scrum Master interactúa con el Portfolio Scrum Master para asegurar que programa coincida 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 las liberaciones de proyectos individuales en el programa coincidan con los del programa.

Los Program Scrum Masters deben consultar las secciones de la *Guía del 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.

3.7.4.4 Portfolio Scrum Master

El rol del Portfolio Scrum Master es similar al rol del Scrum Master, a diferencia de que busca cumplir las necesidades del portafolio o de la unidad de negocios en vez de las necesidades de un solo Equipo Scrum.

Los Portfolio Scrum Masters deben consultar las secciones de la *Guía del 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.

3.7.5 Ejemplos de proyectos, programas y portafolios

Ejemplos de proyectos, programas y portafolios en diferentes industrias y sectores:

Ejemplo 1: Empresa constructora

- Proyecto: Construcción de una casa
- Programa: Construcción de un complejo habitacional
- 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 informática

- Proyecto: Desarrollo del módulo del carrito de compras
- Programa: Desarrollo de un sitio web de comercio electrónico completamente funcional
- Portafolio: Todos los sitios web desarrollados por la empresa hasta ahora

La figura 3-4 ilustra la forma en la que Scrum se puede utilizar para portafolios, programas o proyectos en la organización.

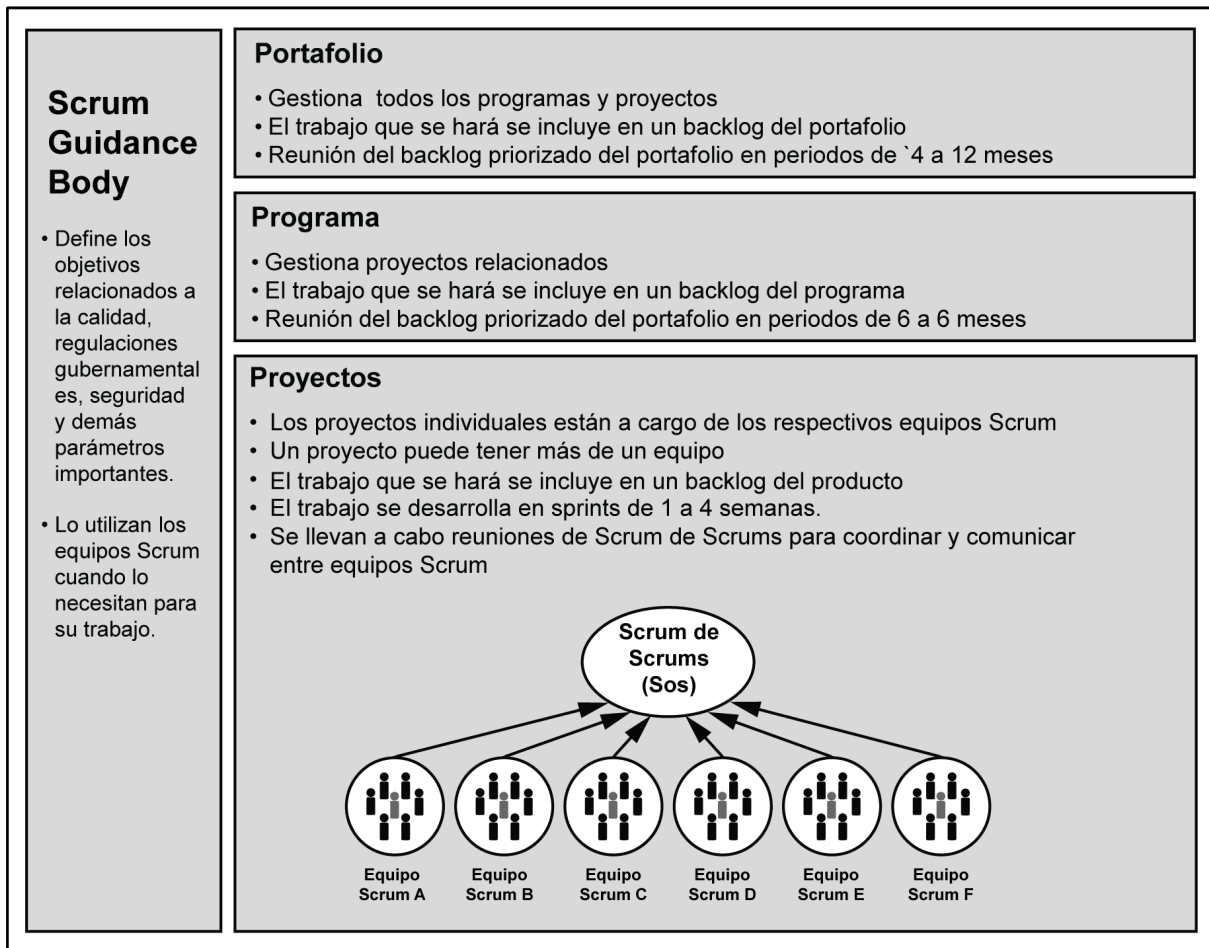


Figura 3-4: Scrum para proyectos, programas y portafolios en la organización

3.8 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	<ul style="list-style-type: none"> • 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 • Acuerda con el Product Owner la duración del sprint
Product Owner/ Chief Product Owner	<ul style="list-style-type: none"> • Elabora los requerimientos generales iniciales del proyecto y arranca el proyecto • Ayuda a nombrar a personas apropiadas en los roles de Scrum Master y Equipo Scrum • Ayuda a asegurar los recursos económicos iniciales y constantes para el proyecto • Establece 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 • Acuerda con el Equipo Scrum la duración del sprint
Scrum Master/ Chief Scrum Master	<ul style="list-style-type: none"> • 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 de la liberación y programa otras reuniones
Program Product Owner	<ul style="list-style-type: none"> • Define los objetivos estratégicos y las prioridades de los programas
Program Scrum Master	<ul style="list-style-type: none"> • Resuelve problemas y coordina las reuniones de los programas
Portfolio Product Owner	<ul style="list-style-type: none"> • Define los objetivos estratégicos y las prioridades de los portafolios
Portfolio Scrum Master	<ul style="list-style-type: none"> • Resuelve problemas y coordina las reuniones de los portafolios
Interesados del negocio	<ul style="list-style-type: none"> • Término colectivo que incluye a clientes, usuarios y patrocinadores • 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	<ul style="list-style-type: none"> • Establece los lineamientos generales y las métricas para desarrollar la descripción de roles para los integrantes del Equipo Scrum • Actúa 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 del proyecto delega autoridad al director del proyecto, 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 director del proyecto, 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 autoorganización y la automotivación, 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 los interesados del negocio relevantes para ajustar los requisitos a medida que avanzan a través de los procesos de *Desarrollar épicas*, *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 en Scrum

3.10.1 Modelo de dinámica de grupo de Tuckman

El enfoque y el marco de trabajo 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). 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. El Scrum Master debe de saber en qué etapa se encuentra el equipo y ayudarlo a lograr un mejor desempeño.

Las cuatro etapas del modelo son las siguientes:

1. **Formación:** Generalmente es una etapa divertida, ya que todo es nuevo y el equipo aún no ha encontrado ninguna dificultad con el proyecto.
2. **Enfrentamiento:** 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:** Es cuando el equipo empieza a madurar, a resolver sus diferencias internas, y a encontrar soluciones para trabajar juntos. Se considera un período de ajuste.
4. **Desempeño:** 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 consistentemente productivos.

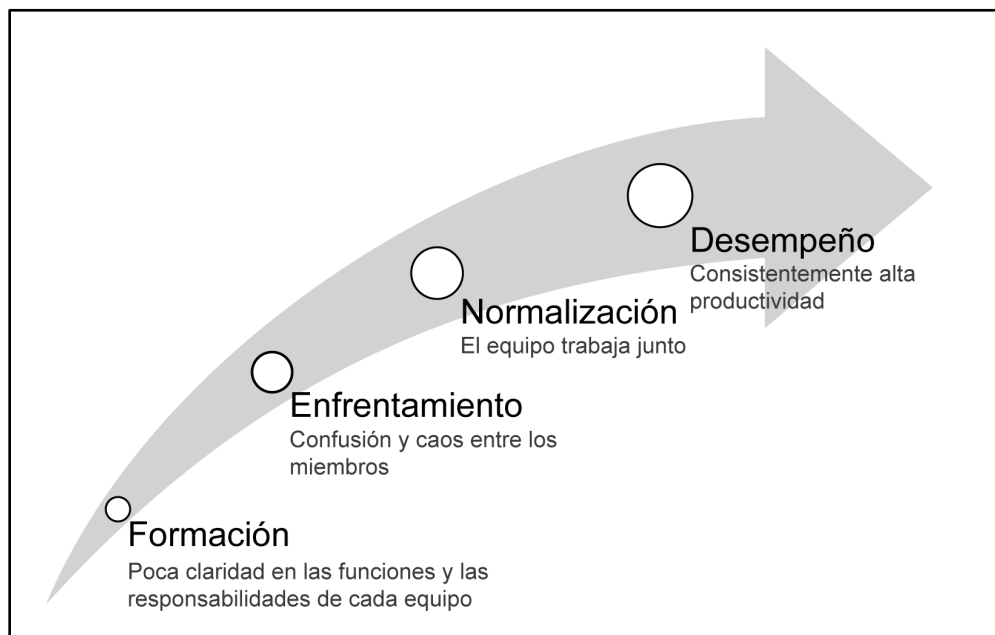


Figura 3-5: Etapas de Tuckman de desarrollo de grupos

3.10.2 Gestión de conflictos

Las organizaciones que aplican el marco de trabajo de Scrum fomentan un ambiente abierto y el 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 personas ajenas al 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. Los orígenes de los 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

Hay generalmente cuatro métodos para gestionar conflictos en una organización que aplica los 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. 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. Esta situación puede darse si hay miembros en el equipo (inclusive los administradores) que utilizan un estilo autoritario o directivo para dar órdenes 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 solo 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 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. Este enfoque no es recomendable cuando se trabaja en proyectos Scrum, ya que los equipos Scrum son por naturaleza autoorganizados 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 de apoyo:** Los líderes saben escuchar, tienen empatía, compromiso y conocimiento, y comparten el poder y la autoridad con los miembros del equipo. Los líderes de apoyo son auxiliares que logran resultados centrándose en las necesidades del equipo. Este estilo es la personificación del rol del Scrum Master.
- **Líder que delega:** Los líderes que delegan están involucrados en la mayoría de la toma de decisiones; sin embargo, delegan parte de las responsabilidades de planificación y decisiones 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 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 entrenamiento:** Los líderes de entrenamiento dan instrucciones; después apoyan y monitorean a los miembros del equipo escuchando, ayudando, alentando y presentando una perspectiva positiva en tiempos de incertidumbre.
- **Líder enfocado en las tareas:** Los líderes que se enfocan en las tareas 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 de apoyo

El estilo de liderazgo preferido en los proyectos Scrum es el liderazgo de apoyo. Larry Spears identifica diez rasgos que debe tener cada líder eficaz:

1. **Escucha:** Se busca que los líderes 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 buenos líderes 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. **Resiliencia:** La motivación y la capacidad de recuperarse a sí mismo y la relación con los demás es un fuerte rasgo de los líderes. Reconocen y se dan la oportunidad de ayudar a sus colegas que están pasando por dolor emocional.
4. **Conciencia:** Ser consciente, especialmente ser autoconsciente, es un rasgo de los líderes. 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 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 practican la persuasión.
6. **Conceptualización:** Una habilidad especial de los buenos líderes 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. **Previsión:** Su mente intuitiva les permite a los líderes utilizar y aplicar las lecciones del pasado y la realidad actual para prever el resultado de situaciones y decisiones actuales.
8. **Administración:** La administración exige un compromiso de servir a los demás. Los líderes 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 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 recursos para el desarrollo personal y profesional, alentando a los trabajadores a participar en la toma de decisiones).
10. **Desarrollo de una comunidad:** Los líderes 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 de apoyo 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.

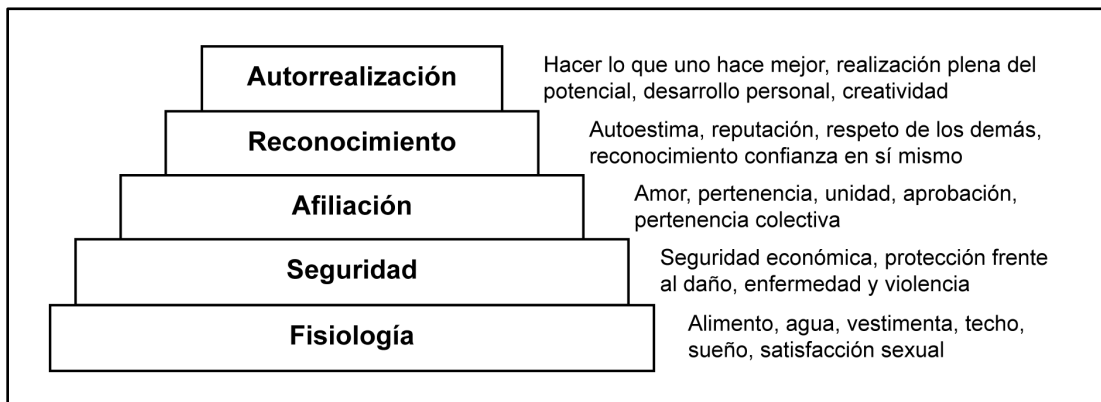


Figura 3-6: Teoría de jerarquía de necesidades de Maslow

Para lograr el éxito, un Equipo Scrum necesita miembros principales y secundarios que hayan alcanzado los niveles de reconocimiento y autorrealización. El concepto de equipos autoorganizados, lo cual es un principio clave en Scrum, requiere que los miembros del equipo sean automotivados, 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 con relación a 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 ayudarlo 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, Teoría Y, Teoría Z

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.

Abraham H. Maslow (1960) propuso la Teoría Z y William Ouchi (1980) presentó otra versión de la Teoría Z, que abunda sobre la Teoría X y la Teoría Y:

- **Teoría Z:** En la versión de Maslow, los líderes de la Teoría Z suponen que los empleados pueden ser motivados aprovechando su deseo de trascender sin ignorar sus motivaciones relacionadas a la jerarquía de necesidades. En la versión de Ouchi, los líderes de la Teoría Z pueden ser motivados promoviendo la estabilidad mediante la seguridad laboral, alta moral y la satisfacción dentro y fuera del trabajo.

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 la Teoría Z, a fin de que puedan ver a las personas como importantes activos y trabajen para desarrollar las habilidades del equipo y empoderar a sus integrantes, expresando su aprecio por el trabajo realizado para alcanzar los objetivos del proyecto.

4. JUSTIFICACIÓN DEL NEGOCIO

4.1 Introducción

El objetivo 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 viable antes de iniciar cualquier proyecto. Esto ayudará a las personas claves que toman decisiones a entender la necesidad de negocio 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 de negocio de la alta gerencia y de los interesados del negocio.

La *justificación del negocio*, según se define en la *Guía de los fundamentos de Scrum (Guía del SBOK®)*, aplica a lo siguiente:

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

El término “producto” en la *Guía del SBOK®* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria, desde proyectos o equipos pequeños con tan solo seis miembros, hasta proyectos grandes y complejos con cientos de miembros.

Este capítulo se divide en las siguientes secciones:

4.2 Guía de roles: Esta sección ofrece información sobre las secciones relevantes para cada uno de los roles principales 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: Esta sección describe 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: Esta sección describe 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 de negocio del marco de trabajo de 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

Un proyecto es un emprendimiento colaborativo para crear nuevos productos o servicios, o para obtener resultados tal como los que se definen en la declaración de la visión del proyecto. Los proyectos por lo general se ven afectados por restricciones 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 debe ser el principal enfoque. El ofrecer valor es algo que está arraigado en el marco de trabajo de Scrum. Scrum facilita la entrega anticipada de valor en el proyecto y lo sigue haciendo a lo largo de su del ciclo de vida.

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 interesados del negocio.

A fin de ofrecer una entrega basada en valor (*Value-driven Delivery*), 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 interesados del negocio 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 marco de trabajo de Scrum sea muy atractivo para los interesados 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 solo 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.

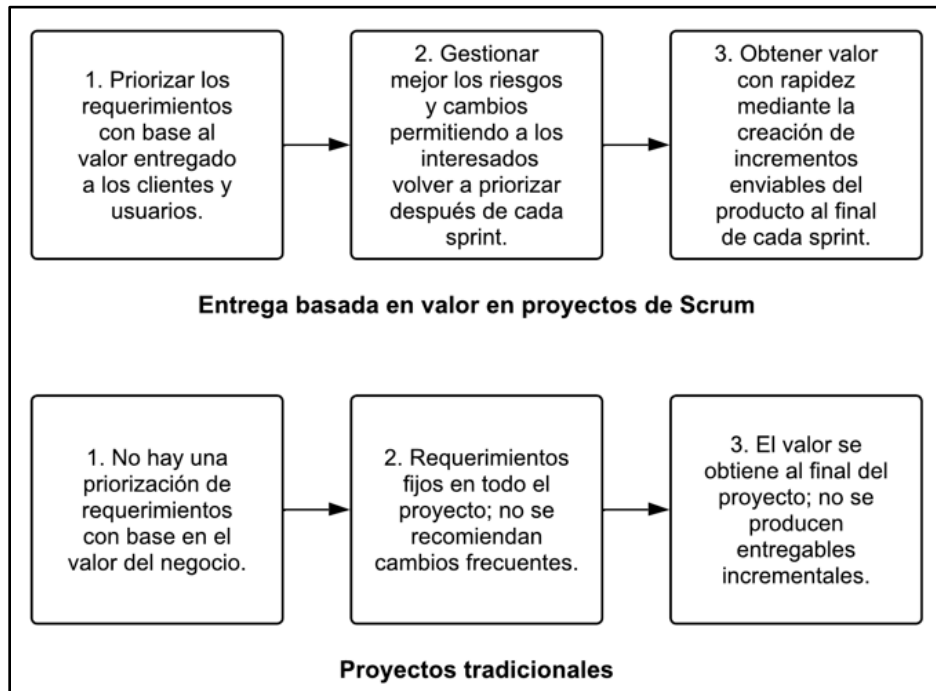


Figura 4-1: Entrega de valor en Scrum vs. Proyectos tradicionales

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 o del patrocinador. Los lineamientos para la evaluación y medición del valor de negocio típicamente pueden ser establecidos por un Scrum Guidance Body.

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

Portfolio Product Owner	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • Entrega el valor de los proyectos • Crea la justificación del negocio para los proyectos • Le confirma el logro de beneficios a los interesados del negocio

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 de Scrum que también contribuyen considerablemente de la siguiente forma:

1. El **patrocinador** 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 liberación 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 lineamientos y recomendaciones pueden después remitirse los equipos centrales de Scrum y a los interesados del negocio.
4. El **Scrum Master** facilita la creación de entregables del proyecto; está a cargo de la gestión de riesgos, de cambios e impedimentos durante los procesos de *Realizar el 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 interesados del negocio 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 interesados del negocio y en el proyecto. El Equipo Scrum también participa en los procesos de *Desarrollar épicas*, *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 La 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 a un 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 interesados del negocio 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 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. Plazos del proyecto

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

7. Costos del proyecto

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

4.4.2 La 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 su ciclo de vida. 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 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 el 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.

3. 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 de la liberación*.

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

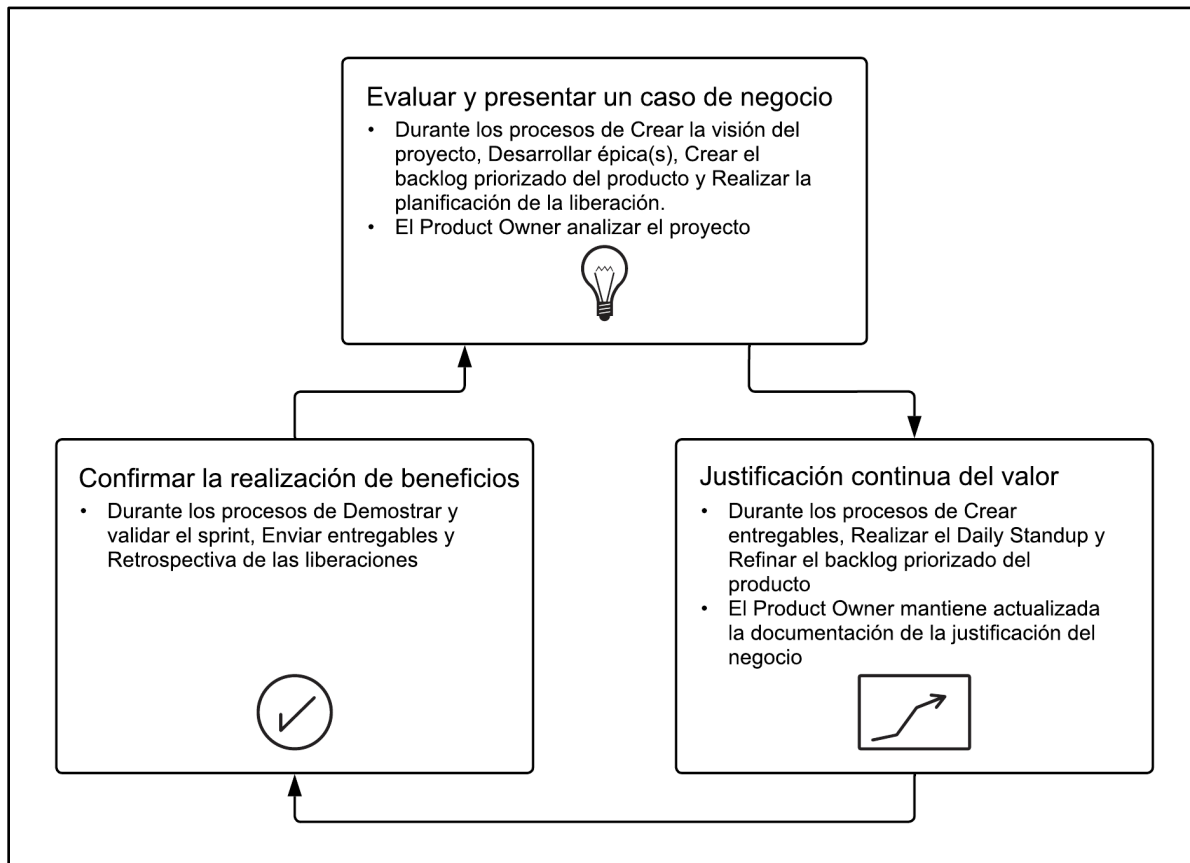


Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto

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 algunas 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 se obtendrá por medio de los proyectos del negocio puede calcularse con diversos métodos, tales como el retorno de la inversión (ROI), el valor presente neto (NPV) y la tasa interna de retorno (IRR).

1. Retorno de la inversión (ROI)

Al utilizar el retorno de la inversión (ROI) para la justificación del proyecto, se evalúan 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 ROI.

Fórmula del retorno de la inversión:

$$\text{ROI} = (\text{Ingresos del proyecto} - \text{Costo del proyecto}) / \text{Costo del proyecto}$$

Ejemplo: El ROI 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:

$$\text{RSI} = (\$300,000 - \$125,000) / \$125,000 = 1.4$$

Por lo tanto, el ROI 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 ROI. Esto ayuda en la evaluación de la justificación continua de valor.

2. Valor presente neto (NPV)

El valor presente neto (NPV) 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 NPV 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 NPV como criterio de selección?

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

Solución: El proyecto A, ya que su NPV 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 NPV (debido a que es el valor actual y no el valor futuro que se considera en el cálculo).

3. Tasa interna de retorno (IRR)

La tasa interna de retorno (IRR), 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 NPV, la IRR es un concepto importante que hay que conocer.

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

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

Solución: El proyecto A, ya que su IRR 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 IRR (tal como en el NPV, es el valor actual y no el valor futuro el que se utiliza para determinar la IRR).

4.5.2 Planificar según 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 interesados del negocio (patrocinadores, clientes o usuarios), mientras que el Equipo Scrum se concentra en lo que se habrá de desarrollar. Algunas de las herramientas comunes recomendadas por un SGB pudieran ser las siguientes:

1. Mapeo de flujo de valor

El mapeo de flujo de valor (*Value Stream Mapping*) utiliza diagramas de flujo para ilustrar el flujo de los pasos necesarios para completar un proceso. Esta técnica pudiera utilizarse para racionalizar un proceso ayudando a identificar y a eliminar elementos que no agregan valor, así como para aumentar la eficiencia. El mapeo de flujo de valor también se puede utilizar para mejorar los procesos de Scrum, por ejemplo, para mejorar la velocidad del sprint

La figura 4-4 muestra cómo la identificación de los tiempos de proceso y los tiempos de espera pueden ayudar a mejorar el sistema mediante la disminución de tiempos de espera y a mejorar la eficiencia de los procesos.

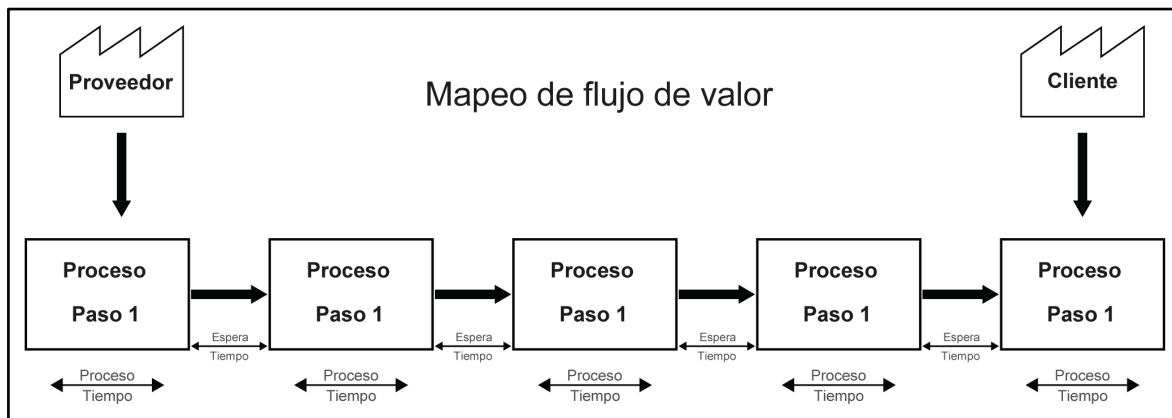


Figura 4-4: Mapeo del flujo de valor

2. Priorización basada en el valor para el cliente

La priorización basada en el valor para el cliente (*Customer Value-based Prioritization*) 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.

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

- **Esquemas simples:** Los esquemas simples implican etiquetar elementos como prioridad “1”, “2”, “3” o “alta”, “media” y “baja”, etc. 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.
- **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.
- **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.
- **Método de los 100 puntos:** El método de los 100 puntos fue desarrollado por Dean Leffingwell y Don Widrig (2003). Implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes.
- **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 4-5 muestra una ilustración del Análisis de Kano. La presencia o ausencia de atributos en un producto o servicio puede afectar la satisfacción del cliente. Esta información puede ayudar a definir las prioridades relativas de las funcionalidades que ofrecen el mayor valor al cliente.

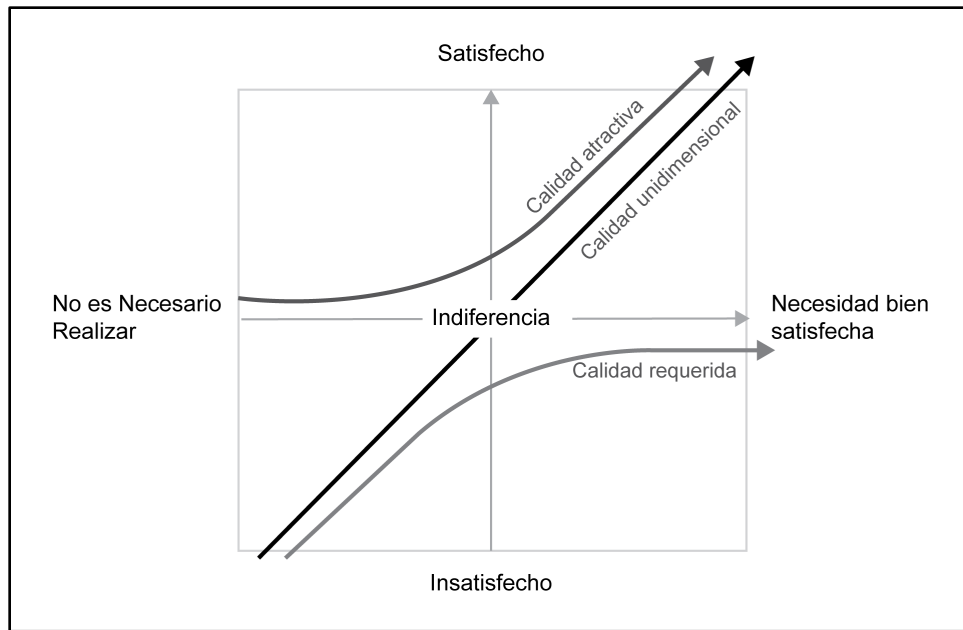


Figura 4-5: Análisis de Kano

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 de priorización relativa

La clasificación de priorización relativa (*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 liberación 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 de la inversión. Normalmente, estas historias de usuario se clasifican como alta prioridad.

4.5.4 Mapeo de historias

El mapeo de historias (*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 mapeos 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 liberaciones. Al implementar Scrum, se entiende perfectamente que esto es solo una imagen del momento y se espera que sean revisados o modificados con frecuencia.

El mapeo de historia de la figura 4-6 muestra cómo el Equipo Scrum planea las distintas liberaciones y les asigna una mayor prioridad a las liberaciones a corto plazo. Se espera que el equipo entienda mejor las historias de usuario en la siguiente liberación, y entre más separadas estén las liberaciones, más probabilidades hay de que cambien las futuras liberaciones.

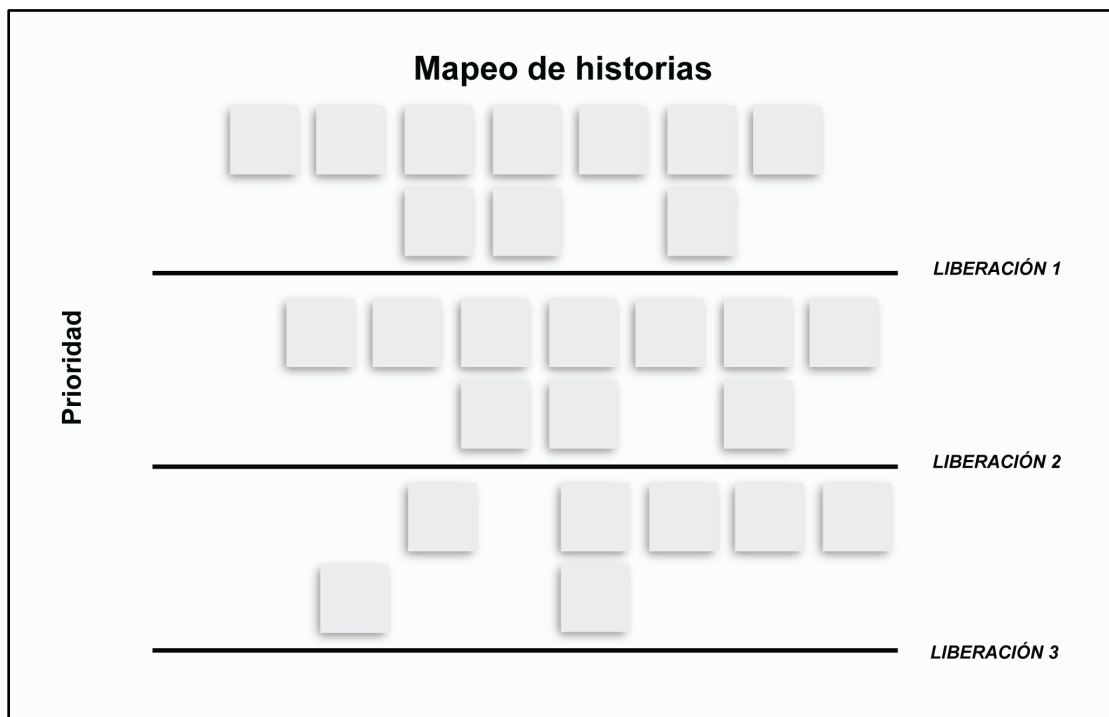


Figura 4-6: Mapeo de historias

4.6 Justificación continua de valor

El valor del negocio debe ser evaluado con frecuencia a fin de establecer si el proyecto aún tiene justificación o viabilidad. La evaluación frecuente de la inversión en el proyecto, con relación al valor del negocio que se está creando, sirve para calificar la viabilidad de un proyecto. Los requisitos esperados del proyecto pueden cambiar con frecuencia, lo que puede afectar la inversión del proyecto y 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 del usuario son ambiguos y los cambios son frecuentes, Scrum ofrece considerables ventajas sobre otros modelos de desarrollo.

Un requisito importante en los proyectos Scrum es monitorear los índices de entrega. 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 interesados del negocio.

4.6.1 Análisis del valor ganado (EVA)

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 (*Earned Value Analysis*) se utiliza para tales efectos.

El análisis del valor ganado (EVA) 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 preciso. El EVA 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 el cronograma del proyecto, así como el costo de funcionamiento y prevé el costo final con base en el rendimiento actual determinado. El EVA generalmente se lleva a cabo al final de cada sprint una vez que se han concluido las historias de usuario y el backlog del sprint.

La tabla 4-1 es un resumen de 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	
Variación del cronograma	SV	$EV - PV$
Variación del costo	CV	$EV - AC$
Índice de desempeño del programa	SPI	EV / PV
Índice de desempeño de costo	CPI	EV / AC
Porcentaje completado	% Completado	$(EV / BAC) \times 100$
Estimado al completar 1. Las suposiciones de estimación no válidas 2. Las diferencias actuales son atípicas 3. Las variaciones actuales son típicas	EAC	1. $AC + ETC$ 2. $AC + BAC - EV$ 3. BAC / CPI
Estimado para completar	ETC	$EAC - AC$
Variancia al completar	VAC	$BAC - EAC$

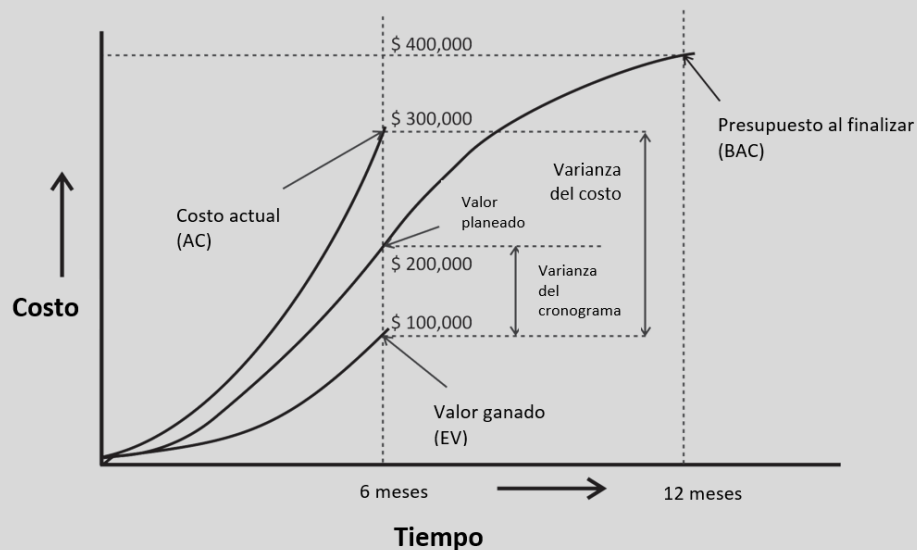
Tabla 4-1: Fórmulas del valor ganado

Ejemplo: Hay un proyecto para desarrollar un sitio web con 4,000 páginas. Suponemos que cada página web tarda igual en completarse, y que cada página representa 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 su límite de tiempo para 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:

- Variación del cronograma (SV) = $EV - PV = \$100,000 - \$200,000 = -\$100,000$
- Variación 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 = $EV / BAC \times 100 = \$100,000 / \$400,000 \times 100 = 25 \%$
 - Por lo tanto, hasta el momento se ha completado el 25 % del trabajo en el proyecto.

4.6.2 Diagrama de flujo acumulado (CFD)

El diagrama de flujo acumulado (CFD, por sus siglas en inglés) 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 4-7 muestra un ejemplo de un CFD en 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 serán 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 ajustado al riesgo, 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 o mejorar el desempeño del proyecto.

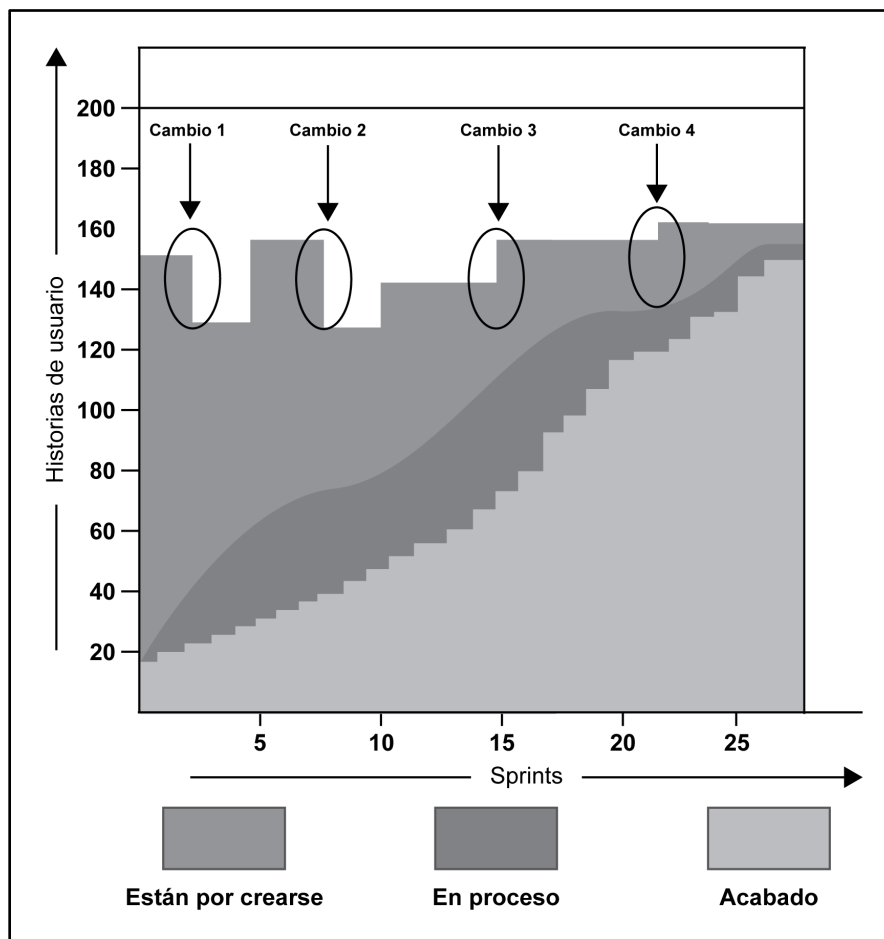


Figura 4-7: Ejemplo de diagrama de flujo acumulado (CFD)

4.7 Confirmar la realización de beneficios

A lo largo del 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

4

La demostración de los prototipos a los clientes y simular su funcionalidad, son técnicas comúnmente utilizadas para confirmar el valor generado.

Generalmente, después de usar las características o después de haberlas visto, los clientes pueden determinar con mayor claridad si son aptas y adecuadas para sus necesidades. Pueden percatarse 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	<ul style="list-style-type: none"> 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 en los proyectos
Product Owner/ Chief Product Owner	<ul style="list-style-type: none"> 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 interesados del negocio
Scrum Master/ Chief Scrum Master	<ul style="list-style-type: none"> Garantiza que los resultados esperados del proyecto sean comunicados al Equipo Scrum y que los entiendan Lleva a cabo la justificación continua de valor para los proyectos
Program Product Owner	<ul style="list-style-type: none"> Garantiza la entrega de valor en los programas Crea la justificación del negocio en 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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> Garantiza la entrega de valor en los portafolios Crea la justificación del negocio en 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	<ul style="list-style-type: none"> Garantiza que se logren los resultados deseados del portafolio Lleva a cabo la justificación continua de valor para los portafolios
Interesados del negocio	<ul style="list-style-type: none"> 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 liberación y del proyecto.
Scrum Guidance Body	<ul style="list-style-type: none"> Establece las directrices generales y las métricas 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 director del proyecto. 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 planificación extensa de largo plazo antes de su ejecución. 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 de la inversión (ROI). Además, la entrega basada en valor (sección 4.3) es un beneficio clave del marco de trabajo de Scrum y brinda una mejor priorización y una más rápida entrega de valor para el 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 de los fundamentos de Scrum (Guía del SBOK®)*, es aplicable a los siguientes:

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

El término “producto” en la *Guía del SBOK®* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan solo seis miembros, hasta proyectos grandes y complejos con cientos de miembros.

Este capítulo se divide en las siguientes secciones:

5.2 Guía de roles: Esta sección presenta información sobre las secciones relevantes para cada uno de los roles principales de Scrum: Product Owner, Scrum Master y el Equipo Scrum.

5.3 Definición de calidad: Esta sección presenta 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 el 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 formas 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 interesados del negocio para mantener constantemente actualizado el backlog priorizado del producto con cualquier cambio en los requerimientos. Dicho backlog se finaliza únicamente hasta el cierre o la conclusión del proyecto. Cualquier cambio en los requisitos refleja los cambios en el entorno de negocio 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. El diálogo constante entre el equipo principal de Scrum y los interesados del negocio (incluyendo 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 constantemente el backlog priorizado del producto. El Product Owner se asegura de que cualquier historia de usuario que debe desarrollar el Equipo Scrum en un sprint sea refinada antes de iniciar el 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. A lo largo del proyecto, el Product Owner, el cliente y el Equipo Scrum analizan y cambian la lista de funcionalidades del producto a fin de 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 de negocio. Para determinar el valor de negocio 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 de negocio 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 interesados del negocio 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 entonces 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 para ser desarrollados 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.

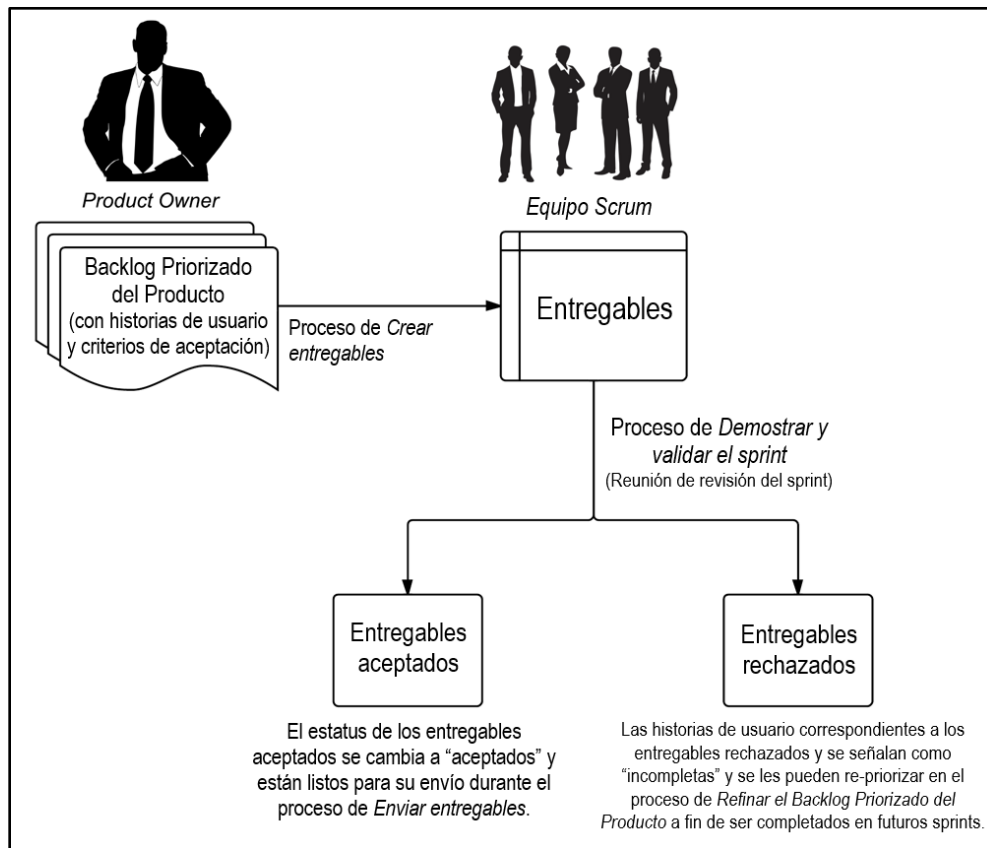


Figura 5-1: Diagrama de flujo del incremento del proyecto

5.4.1 Redacción de criterios de aceptación

Cada historia de usuario tiene sus propios criterios de aceptación y no sustituyen la lista de requerimientos. A continuación, se incluye un ejemplo de cómo se escriben los criterios de aceptación.

Ejemplo:

Personaje: Janine es una profesional 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 en línea a través de múltiples dispositivos y regularmente hace compras en portales de comercio electrónico. **Historia de usuario:** "Como compradora en línea, quiero la posibilidad de guardar y ver mi orden desde cualquiera de mis dispositivos para poder completar el pedido cuando mejor me convenga".

Criterios de aceptación:

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

Es importante que el Product Owner entienda que las historias de usuario que no cumplan con todos los criterios de aceptación no pueden aceptarse como terminadas, aun cuando cumplan con la mayoría. Los proyectos Scrum operan en sprints con un límite de tiempo (time-box) y con un backlog del sprint asignado a cada sprint. Generalmente, la última parte del trabajo pudiera ser la 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” solo tiene dos opciones. Una historia de usuario únicamente puede clasificarse como terminada o como no terminada.

5.4.2 Definición de listo

La definición de listo (*Definition of Ready*) es una serie de reglas o criterios aplicables a cada historia de usuario en el backlog priorizado del producto. Las historias de usuario deben satisfacer la definición de listo antes de ser estimadas e incluidas en un sprint. El Product Owner tiene la responsabilidad de definir adecuadamente la definición de listo en la historia de usuario, ya que, de lo contrario, será imposible obtener estimaciones confiables y el Equipo Scrum no podrá trabajar en los requerimientos de la historia.

De preferencia, la definición de listo debe ser definida por el Scrum Guidance Body. Sin embargo, el Product Owner tal vez deba agregar o actualizar criterios específicos que ya existan un proyecto o una organización. El Equipo Scrum también puede agregar o actualizar la definición de listo. El Equipo Scrum se comprometerá a desarrollar aquellas historias de usuario que satisfagan los criterios de la definición de listo. La revisión de los elementos del backlog del producto en comparación los criterios de la definición de listo es una actividad constante en el proceso de *Refinar el backlog priorizado del producto*.

Algunos de los criterios en la definición de listo pudieran ser los siguientes:

- Las historias de usuario se escriben con suficiente detalle a fin de que los Equipos Scrum las puedan entender y puedan ser utilizadas en las estimaciones;
- Todas las historias de usuario cuentan con criterios de aceptación bien definidos;
- Incluyen cualquier documentación relacionada que pudiera explicar mejor la historia de usuario;
- Las historias de usuario se deben desglosar para que sean suficientemente pequeñas y puedan ser desarrolladas en un solo sprint.

5.4.3 Definición de terminado (o criterios de terminado)

Hay una diferencia clave entre los “criterios de terminado” (*Done Criteria*) y los “criterios de aceptación” (*Acceptance Criteria*). Mientras que los criterios de aceptación aplican solo a 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 interesados del negocio o representantes del negocio.

Al igual que con los criterios de aceptación, se deben de cumplir todas las condiciones de los criterios de terminado para que la historia de usuario se considere terminada. El Equipo Scrum debe de utilizar una lista de verificación de los criterios de terminado en general para garantizar que una tarea está terminada y de que el resultado cumpla con la definición de terminado (DoD). 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.

Generalmente, el Scrum Guidance Body establece y documenta la definición de terminado (o los criterios de terminado). Sin embargo, tal vez existan criterios de terminado específicos en un proyecto u organización que deban ser agregados o actualizados. El Equipo Scrum también puede agregar o actualizar los criterios de terminado.

Los registros y datos necesarios para cumplir con los requisitos de documentación del proyecto se pueden generar a medida que el equipo procede avanza en los sprints y liberaciones. Incluir actividades tales como reuniones de revisión y 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.

5.4.4 Criterios mínimos de terminado

Una unidad de negocio de más alto nivel puede anunciar criterios mínimos de terminado obligatorios, los cuales después formarán parte de los criterios de aceptación de cualquier historia de usuario en esa unidad de negocio. Cualquier funcionalidad definida por la unidad de negocio debe satisfacer dichos criterios mínimos de aceptación si 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 tabla 5-2). Por lo tanto, 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.

Portfolio Product Owner	<ul style="list-style-type: none"> • Establece los criterios mínimos de terminado de todo el portafolio. • Revisa los entregables del portafolio.
Program Product Owner	<ul style="list-style-type: none"> • Establece los criterios mínimos de terminado de todo el programa, los cuales incluyen los criterios de terminado del portafolio. • Revisa los entregables del programa
Chief Product Owner o Product Owner	<ul style="list-style-type: none"> • Establece los criterios mínimos de terminado del proyecto, los cuales incluyen los criterios de terminado del programa. • Revisa los entregables del proyecto

Tabla 5-1: 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 sea necesario.

5.4.5 Aceptación o rechazo de elementos del backlog priorizado del producto

Hacia el final de cualquier iteración, los interesados del negocio y la unidad de negocio 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 interesados del negocio, únicamente el Product Owner tiene la autoridad para rechazar o aceptar 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 importancia y debe entenderse claramente por el equipo. Es responsabilidad del Scrum Master asegurarse 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 interesado 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 entenderse 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 con la orientación del Scrum Guidance Body (ya sea a través de documentos, 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 de Scrum.

Cabe señalar que los interesados del negocio 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, la gestión 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 es quien invierte 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 sobre el negocio y de procesos entre los interesados del negocio y los 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 liberación del producto en las etapas finales su liberación. Asimismo, el costo de los cambios es muy elevado en tales circunstancias, ya que los problemas surgen en etapas posteriores del proyecto. El marco de trabajo de Scrum evita los problemas relacionados a la deuda técnica, garantizando que se definan los entregables terminados

con criterios de aceptación como parte del backlog del sprint 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 esto 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 lograr el trabajo que debe realizarse. Uno de los preceptos más importantes de Scrum y de otras prácticas ágiles como DevOps es mantener un ritmo sostenible.

5

5.5.2 Garantía de calidad y control de calidad

La calidad es necesaria no solo en los productos, sino también en los procesos. La garantía de calidad es la evaluación de procesos y 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 siga 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 una liberación de un sprint. Se puede realizar una comparación de la cantidad de problemas que se encontraron con 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.

El control de calidad es la ejecución de las actividades de calidad planeadas por el Equipo Scrum en el proceso de creación de entregables con la potencialidad de enviarse. Incluye también el aprendizaje de cada serie de actividades realizadas a fin de lograr una mejora continua. Dentro del equipo multidisciplinario, 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 para la mejora continua y contribuyen al mejoramiento del constante control de calidad.

En ocasiones, el Scrum Guidance Body puede definir los procesos y documentos que pueden remitirse a los equipos al momento de hacer sus proyectos a fin de asegurarse 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 (PDCA)

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:

Los lineamientos de la gerencia definen la calidad. Cuando la gerencia puede ofrecer un ambiente propicio 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), 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 5-2 ilustra las etapas del ciclo PDCA y su correlación con los diversos procesos de Scrum.

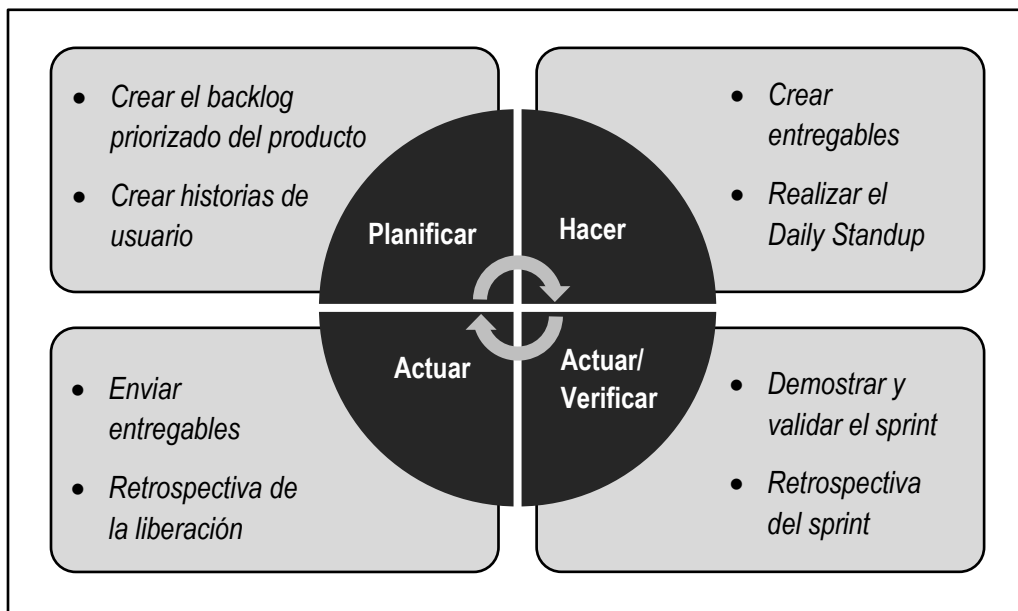


Figura 5-2: Ciclo PDCA en Scrum

5.6 Resumen de responsabilidades

Rol	Responsabilidades
Equipo Scrum	<ul style="list-style-type: none"> Desarrolla y da mantenimiento 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 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. Cumple con los criterios de la definición de terminado de cada entregable
Product Owner/ Chief Product Owner	<ul style="list-style-type: none"> Presenta los requerimientos del negocio en 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 en 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 <i>Demostrar y validar el sprint</i>.
Scrum Master/ Chief Scrum Master	<ul style="list-style-type: none"> Fomenta la mentalidad de “primero el equipo” cuando se trata 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	<ul style="list-style-type: none"> Establece criterios mínimos de aceptación para todo el programa. Revisa los entregables del programa.
Program Scrum Master	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> Establece criterios mínimos de aceptación para todo el portafolio. Revisa los entregables del portafolio.
Portfolio Scrum Master	<ul style="list-style-type: none"> Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.

Interesados del negocio	<ul style="list-style-type: none">• Revisan y opinan sobre los entregables del producto.• Trabajan en colaboración con el Product Owner y con el Equipo Scrum
Scrum Guidance Body	<ul style="list-style-type: none">• Proporciona la definición de terminado.• Proporciona un marco de trabajo 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 5-2: 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 con 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 de negocio 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 director del proyecto define dichas expectativas en términos cuantificables y obtiene la aprobación 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 solo a través de un sistema formal de gestión de cambio, en el cual se estima el impacto de los cambios y el director del proyecto consigue la aprobación de todos los interesados del negocio relevantes.

Sin embargo, en Scrum, el Product Owner colabora con el Equipo Scrum y define los criterios de aceptación de las historias de usuario relacionadas al 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 cuatro semanas).

Una de las principales ventajas de Scrum es el énfasis en la creación de entregables potencialmente liberables 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.