



SCRUM

Metodologías Agiles

1º DAW
MAURICIO LORCA MINGUEZ

Contenido

1. METODOLOGÍAS ÁLGILES	2
2. SCRUM	2
3. LAS HISTORIAS DE USUARIO EN SCRUM:	3
4. LOS TRES ROLES DE SCRUM:.....	4
5. EVENTOS DE SCRUM	5
6. ARTEFACTOS:.....	6
7. ESQUEMA SCRUM.....	7



1.METODOLOGÍAS ÁGILES

En 2001 expertos proponen métodos alternativos para flexibilizar el desarrollo de los proyectos dando lugar al manifiesto de metodología ágil, la gestión ágil es sinónimo de adaptabilidad.

Este manifiesto tiene doce principios asociados a cuatro pilares:

PILARES:

1. Las personas son más importantes que los procesos.
2. El software funcionando es mas importante que la documentación.
3. La colaboración es mas importante de lo que figura en contratos
4. La respuesta al cambio por encima de seguir un plan.

LOS DOCE PRINCIPIOS:

1. Satisfacción del cliente con entregas tempranas del software.
2. Bienvenidos los cambios en los requisitos o requerimientos del software.
3. Liberación frecuente del software funcional.
4. Trabajo conjunto de desarrolladores y responsables del negocio.
5. La motivación del equipo(Los equipos se auto-organizan).
6. Conversación directa entre todas las personas implicadas en el proyecto (conversaciones personales).
7. El software funcionando es la medida del progreso.
8. El proceso ágil promueve desarrollo sostenible.
9. Simplicidad.
10. Equipos auto organizados.
11. Reflexión continua (reuniones sobre el método de trabajo).
12. Atención continúa a la excelencia técnica.

2.SCRUM

Es una técnica de la metodología ágil concreta, es un marco de trabajo con un conjunto de buenas prácticas para la gestión de proyectos basado en entregas parciales del producto final, comenzando por las funcionalidades más importantes para el cliente.

Scrum es por tanto un marco de trabajo de adaptación interactiva e incremental, rápido, flexible y eficaz. Es ligero y fácil de entender, pero extremadamente difícil de dominar.

Esta metodología se utiliza en proyectos que necesitan flexibilidad y rapidez de ejecución, **Scrum** se utiliza en otros ámbitos que no es la programación.



Las fases de desarrollo se solapan en lugar de ser secuenciales.

El proyecto se desgana en pequeños proyectos y se establecen al cliente en entregas regulares y parciales.

Hay tres tipos de requisitos que deben ser implementados por el equipo de desarrollo:

- **Negocio:** son de alto nivel y dan valor a los objetivos de la organización.
- **Usuario:** describen las tareas que realizarán los usuarios.
- **Sistema:** funcionales y no funcionales:
 - **Funcionales:** es lo que hace el sistema (son las funciones del sistema bien descritas).
 - **No funcionales:** que sea extensible, fiables, eficiente, la usabilidad, hardware, plataformas, sistemas operativos, ergonomía...

Un requisito en las metodologías ágiles se enfoca en dar respuesta a las necesidades del usuario. La obtención de los requisitos en metodología ágil se hace dando una aproximación inicial de los requisitos de alto nivel para ir profundizando en cada iteración, incluyendo los cambios necesarios.

Los motivos por los que los requisitos pueden cambiar son:

- Modificaciones solicitadas por el cliente en cada iteración a medida que conoce el producto.
- Riesgos detectados.
- Cambios que surjan por motivos legales, económicos, tecnológicos o de competencia.
- Nuevos requisitos.

3.LAS HISTORIAS DE USUARIO EN SCRUM:

En **Scrum** los requisitos se recolectan a través de las historias de usuario. Una historia de usuario es la representación por escrito de un requisito funcional pero escrito a la manera de una petición del usuario.

Cuando una historia de usuario es muy grande se llama épica, y tienen que estar sujeta a un criterio de aceptación.

Las **historias de usuario** se componen de tres partes:

- El como que representa el “quien”
- El quiero que representa al “qué”
- El para que representa “el objetivo”



Ejemplo de historia de usuario:

Como vendedor (quien) quiero registrar los productos vendidos (que) para poder generar un pedido (objetivo).

Criterios de aceptación de este ejemplo:

- Para registrar los productos vendidos puedo usar el código o el nombre del producto.
- El sistema debe permitir registrar productos vendidos en fechas distintas a la fecha del sistema.
- Al generar el pedido el sistema me muestra el código del pedido.

La historia de usuario debe ser detallada, completa y sin ambigüedades.

Las historias de usuario se dividen en tareas (sucede durante el Sprint Planning).

Una tarea es un requisito en lenguaje de usuario, pero de forma más técnica detallando quienes van a participar y como se va a trabajar.

Cada tarea lo hace el Development Team para completar un ítem del Product backlog.

4. LOS TRES ROLES DE SCRUM:

- **PRODUCT OWNER:** Puede ser el propio cliente o alguien del equipo que traslade las ideas de los clientes en productos factibles, comprende las necesidades del usuario.

Responsabilidades:

- Determina los requisitos generales y las actividades iniciales del proyecto.
 - Representar a los usuarios del producto.
 - Buscar y asegurar los recursos financieros que requiera el proyecto.
 - Analizar la viabilidad del producto.
 - Garantizar que se entregue el producto.
 - Establecer los criterios de aceptación de las historias de usuario.
 - Es quien aprueba o deniega el producto entregado.
- **DEVELOPMENT TEAM:** es el equipo de desarrollo, se recomienda que no sean equipos ni grandes ni muy pequeños (pero no menos de tres personas).
 - **SCRUM MASTER:** es el líder del equipo, y trabaja como facilitador de todo lo que necesite el equipo de desarrollo, pero todos se encuentran al



mismo nivel. Se encarga de aplicar la metodología **Scrum** y de eliminar los obstáculos que surjan. Está al servicio del equipo Scrum y el Product owner.

- **Características:**

- Tiene que conocer bien las reglas del negocio
- Tienen que tener habilidades comunicativas.
- Tiene que conocer **Scrum**.
- Tiene que tener habilidades de negociación.
- Tiene que ser proactivo, decisivo y accesible.

- **Responsabilidades del Scrum Master con respecto del Product Owner son:**

- Asegurar que los objetivos y el alcance del proyecto sea entendido por el equipo.
- Gestionar el Product Backlog.
- Aplicar la agilidad a la hora de aplicar la metodologías ágiles.
- Asegurar que los procedimientos sean ágiles.

- **Responsabilidades del Scrum Master con respecto al Development Team:**

- Guía al equipo para que sea auto-organizado y multifuncional.
- El Scrum Master se asegura de que el Scrum Board este actualizado.
- El Scrum Master se asegura de que las entregas que haga el Development Team sean de calidad.
- Desarrolla la lista de tareas para cada historia de usuario aprobada.
- Calcula el esfuerzo para las tareas.
- Identificar y mitigar los riesgos.

5.EVENTOS DE SCRUM

- **SPRINT PLANING**: reunión de todo el equipo Scrum donde se definen que tareas realizar en el Sprint. Las tareas se seleccionan del Product Backlog. En esta reunión el Scrum Master estima el esfuerzo de cada historia de usuario.
- **DAILY SCRUM**: reunión diaria de 15min entre el Development Team y el Scrum Master, para exponer los avances y dificultades encontrados durante el sprint. Las dificultades encontradas son resueltos por el Scrum Master y el Product Owner.



- **SPRINT REVIEW**: reunión al final del sprint donde el Product Owner presenta lo desarrollado en el sprint al cliente y el Development Team muestra su funcionamiento.

El cliente valida y define nuevas tareas que el Product Owner añade al Product Backlog (las añade mediante historias de usuario)

Si el cliente tiene cambios que proponer deberían de ser mínimos.

- **SPRINT RETROSPECTIVE**: reunión del equipo donde se evalúa el sprint y se proponen mejoras para el siguiente sprint

6. ARTEFACTOS:

Son los elementos que resultan de aplicar el método **Scrum**:

- **PRODUCT BACKLOG**: lista de requisitos del cliente que son las tareas del proyecto, esta lista es ordenada de mayor a menor prioridad por el Product Owner.

De las necesidades del usuario nacen las historias de usuario, el conjunto de historias de usuario del proyecto se llama Product Backlog.

- **SPRINT BACKLOG**: es la lista de historias de usuario seleccionadas para un sprint.
- **INCREMENTO**: es la suma de todas las tareas completadas en un sprint, depende de la complejidad puede llevar de una a cuatro semanas.



7.ESQUEMA SCRUM

