

Metodologías de trabajo

Módulo 3. Desarrollo ágil de software.

Sesión 6. De requerimientos a User Stories

Requerimientos del sistema

Un requisito o requerimiento es un documento con una especificación de una condición (o capacidad) de un software que necesita un usuario para resolver un problema o lograr un objetivo.

Requerimientos funcionales

Describen la funcionalidad o los servicios que se espera que el producto provea.

- Requerimientos de producto

Especifican el comportamiento del producto. Incluye lo referente a rendimiento del sistema y fiabilidad (tasa de fallos aceptable). (??)

Requerimientos organizacionales

Se derivan de políticas y procedimientos existentes en la organización del cliente y en la desarrollador.

Requerimientos no funcionales

Describen la fiabilidad, respuesta en el tiempo y capacidad de procesamiento que se espera del producto.

De requerimientos a User stories

Es la representación de un requerimiento por parte del cliente o usuario. Describe el comportamiento esperado del producto desde el punto de vista del usuario, y tiene como ventaja que divide una gran cantidad de funcionalidades en partes más pequeñas para facilitar la planificación e implementación. Se centran el valor que se obtiene al usar el producto.

Las historias de usuario son cortas y transmiten la esencia principal y las características más importantes. Deben transmitirse de forma oral. Las escribe el Product Owner y son accesibles para el cliente, el equipo de desarrollo y todos los stakeholders. Por tanto, no deben estar escritas en lenguaje demasiado técnico.

Las User stories deben ser (modelo **Invest**):

- Independientes. Deberían poder completarse en cualquier orden.

- Con las historias de Usuario se genera el Product Backlog, incluyendo únicamente las descripciones de alto nivel del requerimiento.

User story mapping



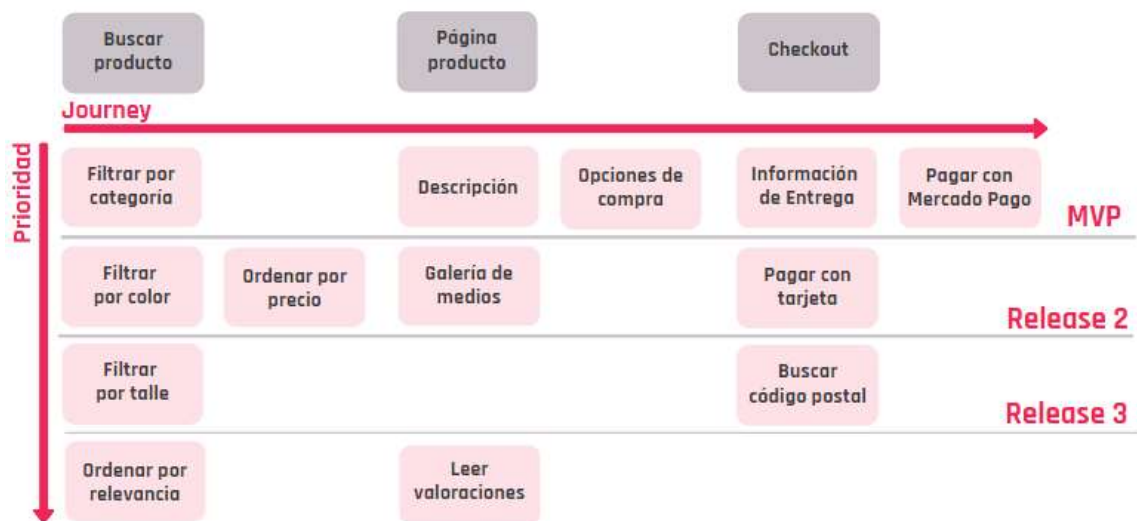
- ## Beneficios del User Story Map:

- Clarificar el scope y features
- Priorizar los componentes por valor de negocio
- Detectar dependencias entre componentes

- Identificar entregas o releases
- Ayudar a delimitar el alcance

Armar el User Story Map:

1. Identificar los procesos del negocio
2. Identificar las funcionalidades
3. Definir y detallar las funcionalidades
4. Identificar MVP y roadmap



User Story Mapping y release plan

¿Qué es un Product Road Map?

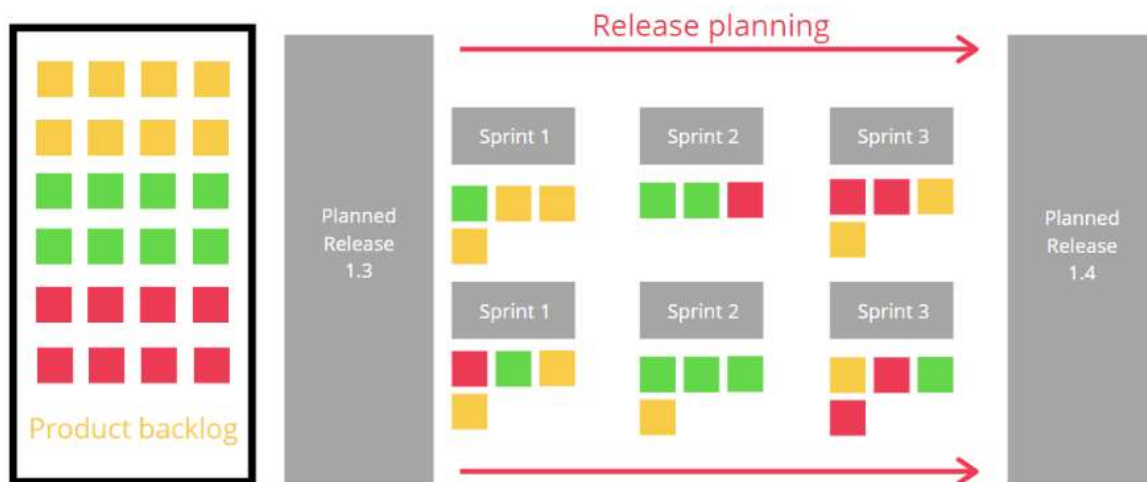
Es un documento estratégico que permite establecer la dirección del desarrollo.

- Indica la visión y la dirección del producto.
- Ayuda a comprender cómo podría evolucionar el producto.

¿Qué es un release plan?

Un release plan es un conjunto de historias de usuario (normalmente épicas) agrupadas por versiones del producto —o releases— que se ponen a disposición de los usuarios incrementando el valor para estos respecto de la anterior.

El product roadmap comunica la descripción general de alto nivel de la estrategia de un producto, mientras que un release plan es un documento táctico diseñado para capturar y rastrear las funciones planificadas para un lanzamiento próximo.

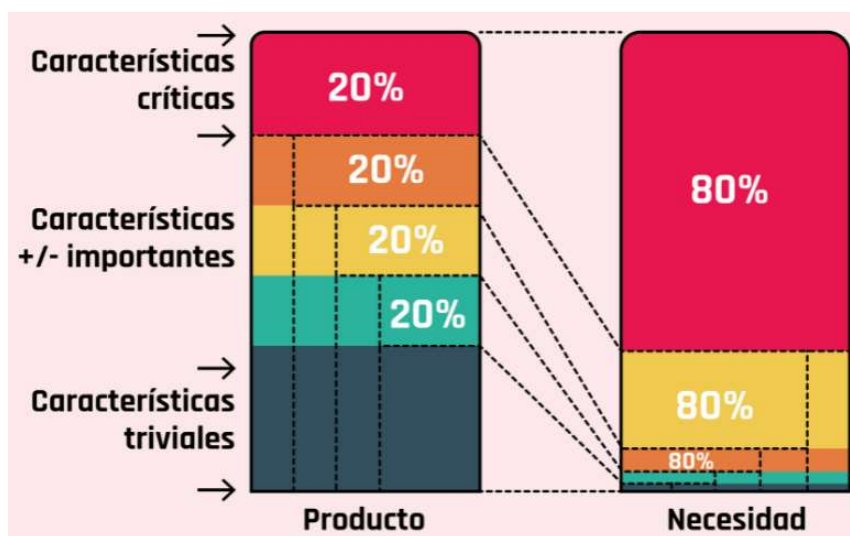


Priorización del backlog

El backlog es una lista priorizada y posiblemente estimada de las user stories del proyecto. Es básicamente un listado de ítems (PBIs) o características del producto a construir, mantenido y priorizado por el product owner.

Principio de Pareto

El 20% del esfuerzo produce el 80% de los resultados.



Técnicas de priorización de User Stories

- MoSCoW
- Puntos de valor de negocio
- Matriz de priorización

Refinamiento de historias

Acotar el alcance, eliminar ambigüedades, reducir el riesgo y simplificar las historias de usuario para entregar la mayor cantidad de valor, de forma independiente, en el menor tiempo posible.

Cartas de Slicing

Slicing o Splitting es una práctica que permite reducir la complejidad y esfuerzo requerido de las Historias de Usuario, y que se aplica a todos los ítems sobre los cuales trabaja el equipo.

Sincrónico

¿Cuál es el problema de la documentación?

- No maneja bien los cambios.
- El software se construye en base a la especificación en lugar de en base a lo que el cliente quiere.
- Toman supuestos falsos.
- Lleva mucho tiempo.

¿Qué prácticas proponen soluciones a este problema?

Prácticas ágiles de desarrollo: Scrum & Peer Programming

- Generar la documentación necesaria y justo a tiempo
- Refactorizar el código constantemente, manteniendo siempre la calidad alta
- Para el ingreso de nuevos desarrolladores, la documentación a consultar está entre líneas
- Incentivar comunicación cara a cara para evacuar dudas.

User Stories:

- Es una "promesa de una conversación"
- Tiene un sentido de negocio (no técnicas)
- Abarcan todos los componentes del sistema.

Beneficios:

- Permiten planificar e implementar más ágilmente.
- Permiten alinear el negocio con el equipo de desarrollo.
- Permiten ampliar el abanico de soluciones.

Épica: Una historia de usuario enorme.

Ejercitación:

Actividad 1

User stories

Definir al menos tres stories según la siguiente visión de producto:

"Construir un sitio web que permita a los clientes personalizar zapatos y comprar pares de zapatos únicos."

10 minutos de trabajo en equipo.
5 minutos de puesta en común.



Técnicas de priorización

- Bussines Value Board: Puntuar por valor de negocio y esfuerzo
- MoSCoW
- Matriz de priorización

User Story mapping

...

Responder

- ¿Cuántas épicas y user stories me quedaron?
- ¿Cuántas user stories por versiones tengo?
- ¿Cuántas incluye mi MVP?
- ¿Cuántas user stories puedo hacer por semana?
- ¿Cuánto va a tardar el primer release?