

# Etapa 2:

# Planificación de los incrementos

- [7.2 - Etapa 2: Planificación del Sprint](#)

# 7.2 - Etapa 2: Planificación del Sprint

Durante esta etapa del ciclo de vida, se realiza la planificación de cada uno de los Sprints, está compuesta por 3 prácticas, 2 de las cuales (7 y 8) pueden ser ejecutadas bajo el concepto “Planificación en paralelo”

**Nota:** Las 3 prácticas de esta fase son iterativas

## dibujo

ID	Práctica	Rol vs Nivel de Involucramiento
	Etapa 2: planificación	
7	7.2.1 - Escribir las historias de usuario y tareas (7)	Product Owner: Alto - Scrum Master: Medio • Equipo de desarrollo: Alto
8	7.2.2 - Priorizar las Historias de Usuario (8)	Product Owner: Alto - Scrum Master: Medio - Equipo de desarrollo: Alto
9	7.2.3 - Reunión de Planificación del Sprint (9)	Product Owner: Alto - Scrum Master: Alto - Equipo de desarrollo: Alto

## 7.2.1 - Escribir las historias de usuario y tareas (7)

### 7.2.1.1 - ¿Qué son las Historias de Usuario?

Las historias de usuario son una forma más entendible y precisa para la escritura de los requerimientos a desarrollar en el proyecto.

Están compuestas por 3 elementos principalmente:

- **Como:** Describe el Rol de la persona o grupo que solicita (o usaría) la funcionalidad o requerimiento.
- **Quiero:** Describe la necesidad o requerimiento del usuario, por lo general, es una frase corta.
- **Para:** Describe el beneficio esperado por el usuario una vez se desarrolle el requerimiento.

Por lo general, las historias de usuario se complementan de otros aspectos que ayudan al equipo a entenderlas mejor, algunos de estos pueden ser:

- Criterios de aceptación
- Criterios de prueba
- Prototipos

### Desarrollando mejores historias de usuario

Bill Wake inventó el acrónimo INVEST para describir las características de una buena historia de usuario:

- **Independiente:** Las historias pueden completarse en cualquier orden.
- **Negociable:** Los detalles de la historia son co-creados por los desarrolladores y los clientes durante el desarrollo.
- **Valiosa:** La funcionalidad es valiosa para los clientes o los usuarios del producto.
- **Estimable:** Los desarrolladores pueden encontrar una estimación razonable para construir la historia.
- **Pequeña (Small):** Las historias deberían construirse en poco tiempo, generalmente alrededor de "días/persona". Se tiene que poder construir muchas historias en una iteración.
- **Probable (Testeable):** Se debe poder escribir pruebas que verifiquen que el producto de la historia funcione adecuadamente.

## Criterios de aceptación para las historias de usuario

Los criterios de aceptación describen con mayor detalle el trabajo que debe ser completado en cada historia de usuario. En la mayoría de ocasiones, estos criterios describen a nivel técnico el trabajo a desarrollar. Algunas características de estos criterios son:

- Los criterios de aceptación son únicos para cada historia de usuario o tarea a completar
- Estos criterios suelen ser definidos por el Product Owner
- Estos criterios permiten saber si esa historia de usuario es funcional y cumple las necesidades, expectativas y/o requerimientos de los usuarios y/o clientes.

## 7.2.1.2 - Mockups y prototipos iniciales de proyecto

Los Mockups representan el diseño del producto antes de su desarrollo, consideran el flujo que debe seguir el producto, y sirven para mostrar las posibles funcionalidades al cliente y así confirmar que estas entregarán el valor esperado.

Los mockups normalmente son un conjunto de bocetos, diseños, diagramas y/o representaciones; y es especialmente necesario contar con un mockup o prototipo previo al Sprint, pues éstos servirán de guía a los miembros del equipo.

## **dibujo**

### 7.2.1.3 - Planificación en paralelo

Una actividad que puede representar una gran diferencia es la planificación en paralelo. Esta actividad consiste en formar un pequeño equipo de personas que se encargan de realizar las actividades de planificación paralelamente al desarrollo de los elementos realizados. Con esto se logrará que las reuniones de planificación del Sprint sean muchísimo más eficientes. Algunas de las cosas que se puedan hacer en paralelo al desarrollo son:

- Realizar las entrevistas o investigaciones necesarias para escribir las historias de usuario.
- Realizar los prototipos y confirmar el valor con el usuario.

### 7.2.1.4 - Desglosar las Historias de Usuario en tareas

Esta actividad sirve para identificar las posibles dependencias entre las historias de usuario, para ello, se elabora un listado con todas las tareas que se deben llevar a cabo para completar la Historia de Usuario.

Normalmente se distinguen 4 tipos de dependencias entre historias de usuario:

- Dependencias obligatorias.
- Dependencias discrecionales.
- Dependencias externas.
- Dependencias internas. Algunas técnicas que pueden ser usadas para desglosar las Historias de Usuario en tareas son:
  - Los Diagramas de Flujo.
  - Los Diagramas de Gantt.

## 7.2.2 - Priorizar las Historias de Usuario (8)

Esta práctica está relacionada con la planificación en paralelo, debido a que mientras el equipo termina de desarrollar lo establecido para el último Sprint, el Product Owner realiza una priorización de las historias de usuario que pueden considerarse para el siguiente Sprint y que se pueden tomar como base para guiar el rumbo del equipo.

Es importante que esta priorización se realice antes de que se lleve a cabo la Reunión de Planificación del Sprint con el fin de optimizar el tiempo.

Los elementos que se deben considerar para la priorización de las historias de usuario son:

- Valor para el negocio.
- Riesgo.
- Dependencia.

## 7.2.3 - Reunión de Planificación del Sprint (9)

### 7.2.3.1 - Seleccionar el trabajo a desarrollar

El Product Owner expone el objetivo que el Sprint debería lograr y los elementos del Product Backlog que, si se completan en el Sprint, lograrían el objetivo del Sprint, con esto el Equipo de Desarrollo trabaja para proyectar la funcionalidad que se desarrollará durante el Sprint.

La entrada a esta reunión está constituida por el Product Backlog y tomando la priorización de historias de usuario realizada por el Product Owner, incluyendo además:

- El último incremento de producto.
- La capacidad proyectada del Equipo de Desarrollo para el Sprint.
- La velocidad del Equipo de Desarrollo.
- Fallos encontrados al producto o incrementos de producto (registro de errores).

El número de elementos del Product Backlog seleccionados para el Sprint depende únicamente del Equipo de Desarrollo. Solo el Equipo de Desarrollo puede evaluar qué es capaz de lograr durante el Sprint que comienza.

Durante la Planificación del Sprint el Equipo Scrum define el objetivo del Sprint. El objetivo del Sprint debería lograrse durante el Sprint a través de la implementación del Product Backlog y proporciona una guía al Equipo de Desarrollo del por qué se está construyendo el incremento.

## 7.2.3.2 - Estimación del trabajo seleccionado

Es muy importante realizar la estimación del trabajo seleccionado (comúnmente escrito en forma de historias de usuario) para poder hacer planificaciones más precisas.

Para garantizar una estimación más precisa, se deberían considerar criterios como:

- Tamaño.
- Complejidad.
- Duración.
- Cantidad de recursos.
- Riesgos.
- Limitaciones.

Normalmente las técnicas de estimación usadas por Scrum están basadas en el juicio de expertos, sin embargo existen otras técnicas que se pueden utilizar y combinar:

### **Planning Poker**

El póker de planificación consiste en un conjunto de tarjetas numéricas que sirven para realizar la estimación de tareas o historias de usuario.

- El póker puede ir numerado de 1 a 10 o utilizando la sucesión de Fibonacci.
- El uso del póker promueve una mayor interacción y una mejor comunicación entre los participantes, además de hacer más dinámica la reunión de planificación del Sprint.

### **Fist of Five**

Es una técnica que permite lograr el consenso en los Equipos Scrum. Se hace usando los dedos de la mano, donde el número de dedos que se utiliza para la votación indica el nivel de acuerdo y el deseo para el debate:

- Un dedo: no estoy de acuerdo con la conclusión del grupo y tienen grandes preocupaciones.

- Dos dedos: no estoy de acuerdo con la conclusión del grupo y me gustaría hablar de algunos problemas menores.
- Tres dedos: no estoy seguro y me gustaría asumir la conclusión de consenso del grupo.
- Cuatro dedos: Estoy de acuerdo con la conclusión del grupo, pero me gustaría discutir algunos problemas menores.
- Cinco dedos: Estoy totalmente de acuerdo con la conclusión del grupo.

## 7.2.3.3 - Compromiso del equipo

Una de las actividades clave durante la Planificación de un Sprint es la apropiación de las Historias de Usuario y tareas por parte del equipo de desarrollo, esto se logra cuando los miembros del equipo se auto-asignan el trabajo a desarrollar, a su vez que adquieren el compromiso público de terminar dicho trabajo.

Es sumamente importante garantizar que existe equilibrio en la cantidad de trabajo seleccionado por cada uno de los integrantes del equipo.

Suele suceder que en equipos Scrum donde se encuentran combinados miembros expertos y miembros novatos existirá un “desequilibrio” en etapas tempranas del proyecto, pues los miembros expertos contarán con más capacidad de trabajo mientras los miembros novatos se nivelan en su curva de aprendizaje.

## 7.2.3.4 - ¿Cómo se conseguirá “terminar” el trabajo seleccionado?

Una vez que se ha establecido el objetivo y se han seleccionado los elementos del Product Backlog para el Sprint, el Equipo de Desarrollo decide cómo construirá esta funcionalidad para formar un Incremento de producto “Terminado” durante el Sprint. Los elementos del Product Backlog seleccionados para este Sprint, junto con el plan para terminarlos, recibe el nombre de Sprint Backlog.

Durante la Planificación del Sprint se debe asegurar que se planifica el suficiente trabajo como para que el Equipo de Desarrollo pueda hacer una proyección de lo que cree que puede completar en el Sprint que comienza.

Para el final de esta reunión, el trabajo planificado por el Equipo de Desarrollo para los primeros días del Sprint es descompuesto en unidades de un día o menos. El Equipo de Desarrollo se autoorganiza para asumir el trabajo del Sprint Backlog, tanto durante la Planificación del Sprint como a lo largo del Sprint.

El Product Owner puede ayudar a clarificar los elementos del Product Backlog seleccionados y hacer concesiones. Si el Equipo de Desarrollo determina que tiene demasiado trabajo o que no tiene suficiente trabajo, podría renegociar los elementos del Product Backlog seleccionados por el

Product Owner. El Equipo de Desarrollo podría también invitar a otras personas a que asistan para proporcionar asesoría técnica o relacionada con el dominio.

### 7.2.3.4.1 - Objetivo del Sprint

El objetivo del Sprint es una meta que se establece para cada Sprint durante la Reunión de Planificación del Sprint. Dicha meta describe el alcance de cada Sprint en alineación con el Product Backlog, a su vez proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento.

Definir el objetivo del Sprint brinda al Equipo de Desarrollo cierta flexibilidad con respecto a la funcionalidad que ha de ser construida en el Sprint.

- Los elementos seleccionados del Product Backlog ofrecen una orientación sobre lo que puede ser el objetivo del Sprint.
- El objetivo del Sprint garantiza que el Equipo de Desarrollo trabaje en conjunto y no en iniciativas separadas.
- Por lo general el objetivo del Sprint suele ser una frase corta que explica de forma simple el incremento esperado al final del Sprint. A medida que el Equipo de Desarrollo trabaja mantiene el objetivo del Sprint en mente. Si el trabajo resulta ser diferente de lo esperado, los miembros del equipo escalan la situación con el Scrum Master para contactar al Product Owner y se pueda negociar el alcance del Sprint.

### 7.2.3.4.2 - Spikes

Un Spike sirve para incluir en un sprint tareas que NO implican desarrollar una historia de usuario y por tanto NO aportan directamente al incremento de producto que se está desarrollando.

Algunos ejemplos de Spikes son:

- Capacitación del Equipo.
- Documentación del proyecto y/o producto (ejemplo: Documentar el código fuente).
- Despliegues / Implementaciones.
- Adopción de nuevas herramientas.