

Módulo 7

Fundamentos de desarrollo Agile

Metodologías Ágiles



Módulo 7

AE 1.1

OBJETIVOS

Comprender la evolución de las metodologías ágiles y su diferencia con modelos tradicionales. Conocer Scrum, sus principios, roles, prácticas y artefactos clave. Entender la importancia de la visión del producto y la gestión de lanzamientos en Scrum.



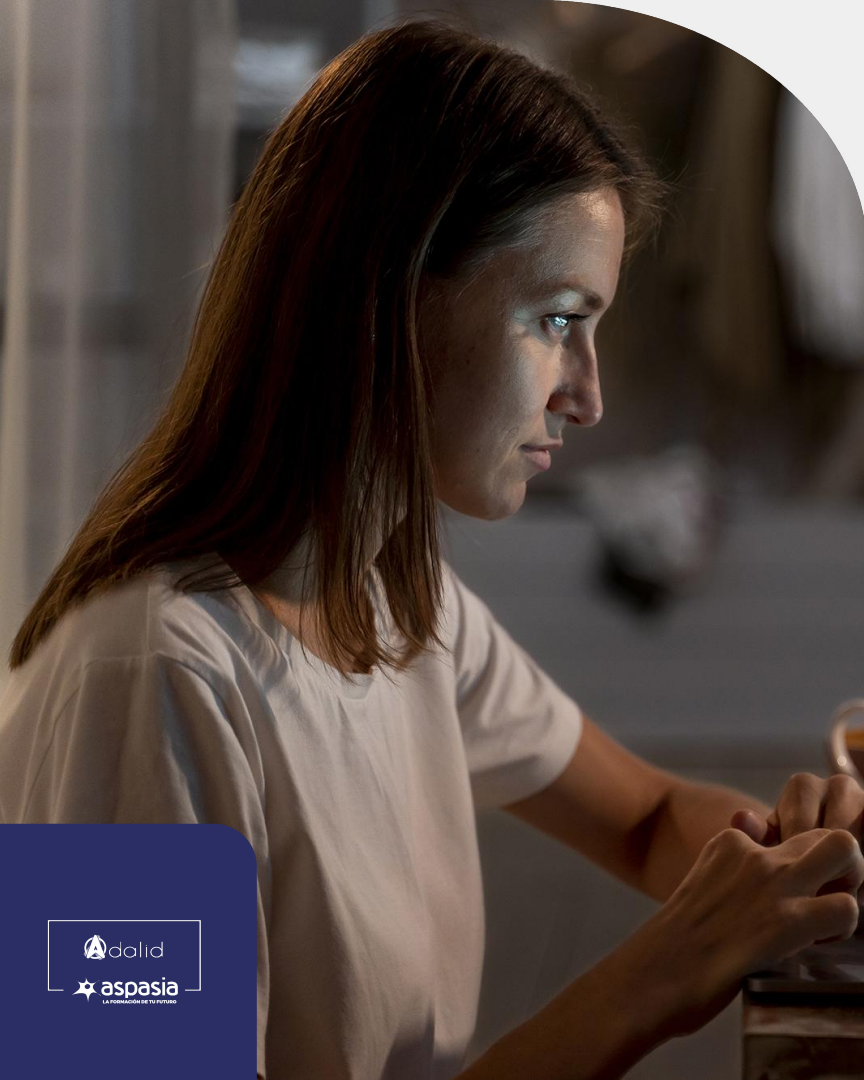
¿QUÉ VAMOS A VER?

- Metodologías Ágiles.
- Orígenes de las metodologías ágiles.
- Diferencias con modelos tradicionales de desarrollo (Waterfall, RUP, Modelo PMBOK).
- Diferencias entre “Agile”, “Agilidad” y “Agilismo”.
- Manifiesto Ágil.
- Valores y Principios del Manifiesto Ágil.
- Abanico de Metodologías Ágiles.
 - DSDM-Atern.
 - Lean Software Development.
 - Extreme Programming (XP).
 - Kanban.



¿QUÉ VAMOS A VER?

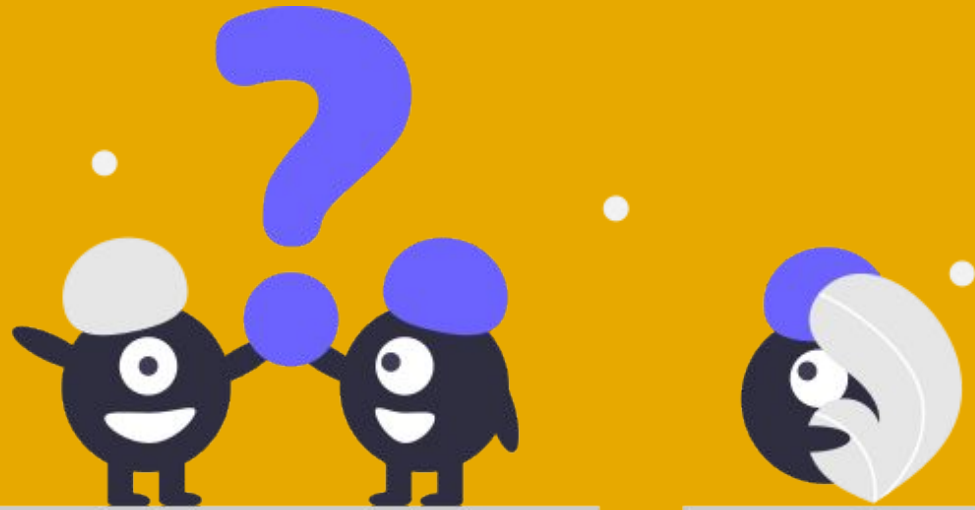
- Fundamentos de Scrum.
- Qué es Scrum.
- Principios y Valores de Scrum.
- Roles en Scrum.
 - ScrumMaster.
 - Scrum Developers.
 - Product Owner.
- Prácticas en Scrum.
 - Sprint Planning.
 - Daily Scrum.
 - Sprint Review.
 - Sprint Retrospective.

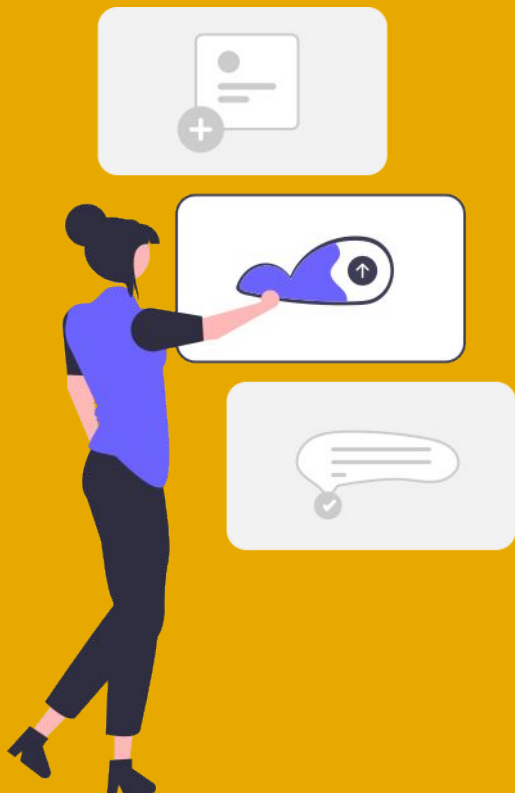


¿QUÉ VAMOS A VER?

- Artefactos en Scrum.
 - Product Backlog.
 - Sprint Backlog.
 - Incremento.
- Visión del Producto en Scrum.
 - Producto y su visión.
 - Cómo estimar.
 - Definición de Alcance.
 - Release Management.
 - Scrum como modelo iterativo y evolutivo.
 - Mínimo Producto Viable (MVP).

¿Sabes que es una Metodologías Ágiles?

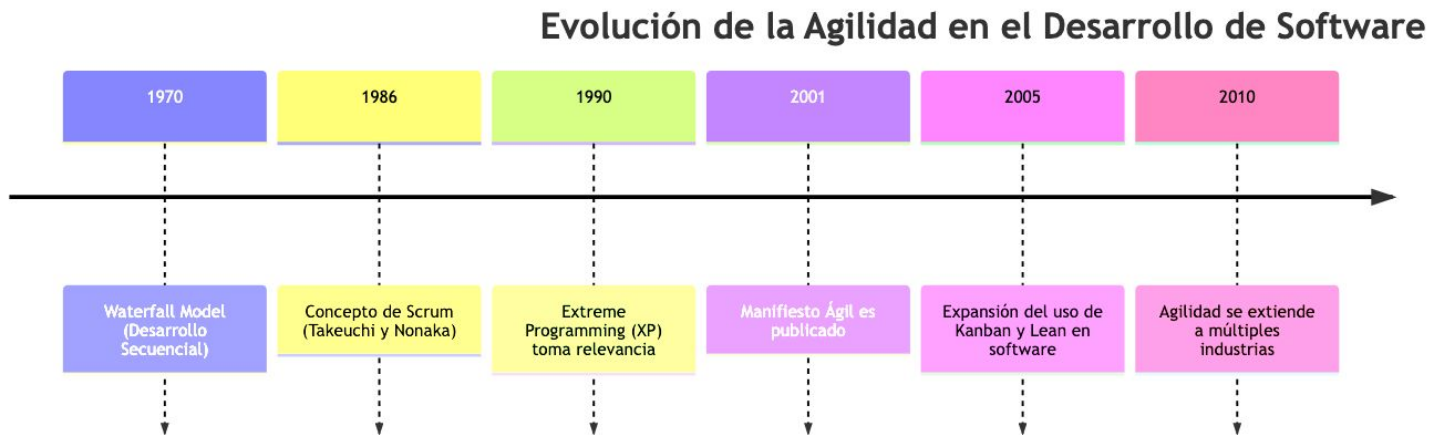




Metodologías Ágiles

Orígenes de las Metodologías Ágiles

Las metodologías ágiles surgieron como una respuesta a los problemas de los **modelos tradicionales de desarrollo de software**, que eran **rígidos, lentos y poco adaptativos** a los cambios.



Orígenes de las Metodologías Ágiles

Ante de implementar las Metodologías Ágiles:

- Los proyectos tardaban demasiado tiempo en entregar valor.
- Se seguía un plan rígido sin adaptarse a cambios.
- Se generaban requisitos extensos y documentación pesada antes de empezar a desarrollar.
- Equipos de desarrollo tenían poca interacción con el cliente.

Diferencias con Modelos Tradicionales de Desarrollo

Comparación entre Modelos Tradicionales y Ágiles

Característica	Waterfall (Cascada)	RUP (Rational Unified Process)	PMBOK (Gestión de Proyectos)	Ágil (Scrum, XP, Kanban, etc.)
Planificación	Completa al inicio	Por fases iterativas	Enfocada en gestión de proyectos	Adaptativa y continua
Flexibilidad	Baja, cambios son costosos	Moderada	Baja, documentación pesada	Alta, adaptable a cambios
Entrega de Valor	Hasta el final del proyecto	Entregas iterativas	Generalmente en fases largas	Entregas continuas y tempranas
Interacción con el Cliente	Solo al inicio y al final	En momentos clave del proyecto	Planificada en la fase de ejecución	Constante y colaborativa
Documentación	Extensa y detallada	Moderada	Completa y estructurada	Mínima y útil

Diferencias con Modelos Tradicionales de Desarrollo

Diferencias entre "Agile", "Agilidad" y "Agilismo"

Concepto	Descripción
Agile	Un conjunto de metodologías de desarrollo basadas en el Manifiesto Ágil (Scrum, Kanban, XP, etc.).
Agilidad	La capacidad de un equipo u organización para responder a cambios rápidamente y de manera efectiva.
Agilismo	La interpretación extrema de Agile, a veces mal aplicada sin estructura ni planificación.

Ejemplo:

- **Agile:** Un equipo implementa Scrum en su empresa.
- **Agilidad:** Un equipo adopta reuniones diarias y entregas iterativas sin seguir una metodología específica.
- **Agilismo:** Una empresa dice que es "ágil" pero no tiene procesos claros ni disciplina.

El Manifiesto Ágil

Creado en 2001 por 17 expertos en desarrollo de software, el Manifiesto Ágil define los valores fundamentales para el desarrollo ágil.

Valores del Manifiesto Ágil:

- Individuos e interacciones sobre procesos y herramientas.
- Software funcionando sobre documentación extensiva.
- Colaboración con el cliente sobre negociación de contratos.
- Respuesta ante el cambio sobre seguir un plan rígido.

El Manifiesto Ágil

Principios del Manifiesto Ágil:

- La prioridad es satisfacer al cliente con entregas rápidas y continuas.
- Bienvenidos los cambios de requisitos en cualquier etapa del proyecto.
- Entregas de software frecuentes y funcionales.
- Colaboración entre equipos de negocio y desarrollo.
- Confianza y apoyo a los equipos motivados.
- Comunicación cara a cara como la mejor forma de transmitir información.
- Software funcionando como medida principal de progreso.
- Desarrollo sostenible sin sobrecargar al equipo.
- Excelencia técnica y buen diseño mejoran la agilidad.
- Simplicidad y eficiencia en el trabajo.
- Equipos autoorganizados generan mejores resultados.
- Reflexión y mejora continua en cada iteración.

El Manifiesto Ágil

Ejemplo en la Práctica:

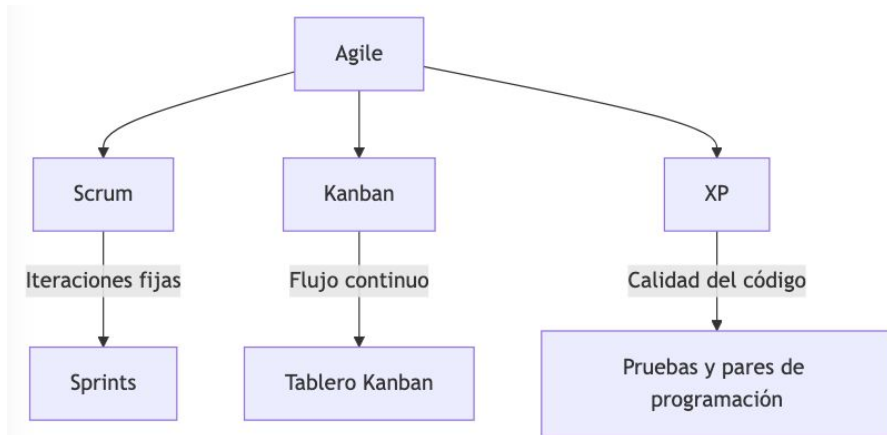
En lugar de planear 12 meses de requisitos, se entrega una versión usable en 2 semanas y se mejora en cada sprint.

Desarrollo ágil = **Mayor Valor en Menos Tiempo**

Abanico de Metodologías Ágiles

Metodología	Descripción
Scrum	Trabajo en iteraciones llamadas sprints (2-4 semanas). Usa roles y reuniones específicas.
Kanban	Flujo de trabajo visual con tarjetas en un tablero. No tiene iteraciones fijas.
Extreme Programming (XP)	Foco en buenas prácticas técnicas como Pair Programming y TDD (Test Driven Development) .
Lean Software Development	Minimiza el desperdicio, maximiza el valor y busca la eficiencia.
DSDM-Atern	Metodología enfocada en proyectos de alta prioridad con entregas rápidas.

Abanico de Metodologías Ágiles



Ejemplo:

- Scrum se usa en desarrollo de productos con iteraciones bien definidas.
- Kanban se usa en equipos de soporte o mantenimiento sin ciclos fijos.
- XP es ideal para proyectos donde la calidad del código es crítica.



SCRUM

¿Qué es Scrum?

Scrum es un marco de trabajo ágil para gestionar proyectos complejos.

Se basa en **iteraciones cortas y entregas incrementales**, promoviendo la colaboración, adaptabilidad y mejora continua.

¿Qué es Scrum?

Características Clave de Scrum:

- Basado en sprints (iteraciones cortas de 1 a 4 semanas).
- Equipos autogestionados y multifuncionales.
- Enfoque en entregar valor continuo.
- Adaptabilidad a cambios y mejoras constantes.

Cuándo usar Scrum:

- Desarrollo de software y productos digitales.
- Proyectos con requisitos cambiantes.
- Equipos que buscan mejorar la colaboración y productividad.

Principios y Valores de Scrum

3 Principios Clave:

- **Transparencia:** Todos pueden ver el progreso y los problemas.
- **Inspección:** Se revisa constantemente el avance y la calidad.
- **Adaptación:** Se ajustan estrategias según sea necesario.

Los 5 Valores de Scrum:

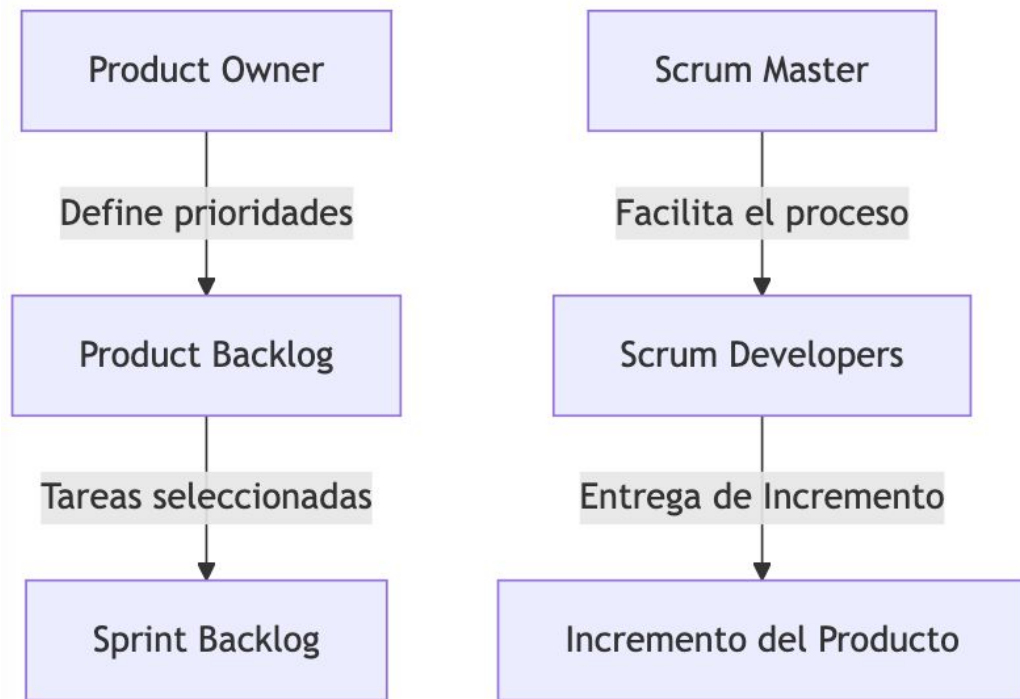
- **Compromiso:** El equipo se compromete a alcanzar los objetivos del Sprint.
- **Coraje:** Se fomenta la toma de decisiones y enfrentar desafíos.
- **Enfoque:** Se prioriza lo más importante en cada Sprint.
- **Respeto:** Todos los miembros del equipo colaboran y se respetan.
- **Apertura:** Se aceptan opiniones, retroalimentación y mejoras.

Roles en Scrum

Scrum se basa en 3 roles principales:

Rol	Responsabilidades
Scrum Master	Facilita el proceso, elimina impedimentos, educa sobre Scrum.
Product Owner	Define prioridades, gestiona el backlog y representa al cliente.
Scrum Developers	Desarrollan el producto, realizan estimaciones y colaboran en la entrega.

Principios y Valores de Scrum



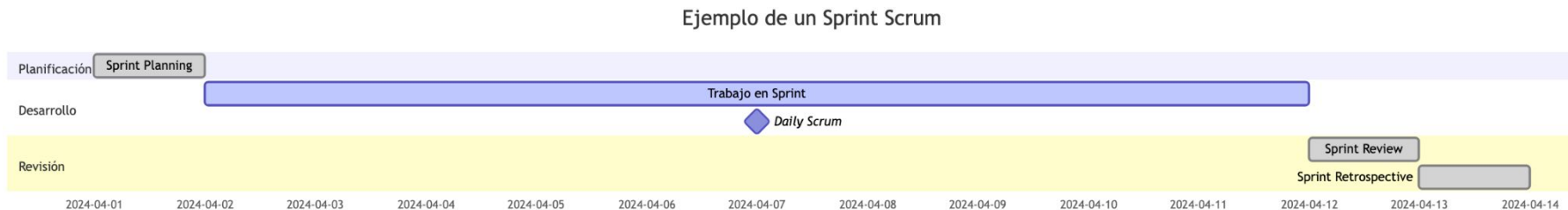
Prácticas en Scrum

Las 4 Ceremonias Clave:

Ceremonia	Descripción
Sprint Planning	Se define qué trabajo se realizará en el Sprint.
Daily Scrum	Reunión diaria para revisar avances y bloqueos.
Sprint Review	Presentación del trabajo terminado al cliente.
Sprint Retrospective	Reflexión del equipo sobre lo que se puede mejorar.

Prácticas en Scrum

Ejemplo de un Sprint de 2 semanas:



Beneficios de las Ceremonias Scrum:

- Mantienen al equipo alineado.
- Detectan problemas a tiempo.
- Fomentan la mejora continua.

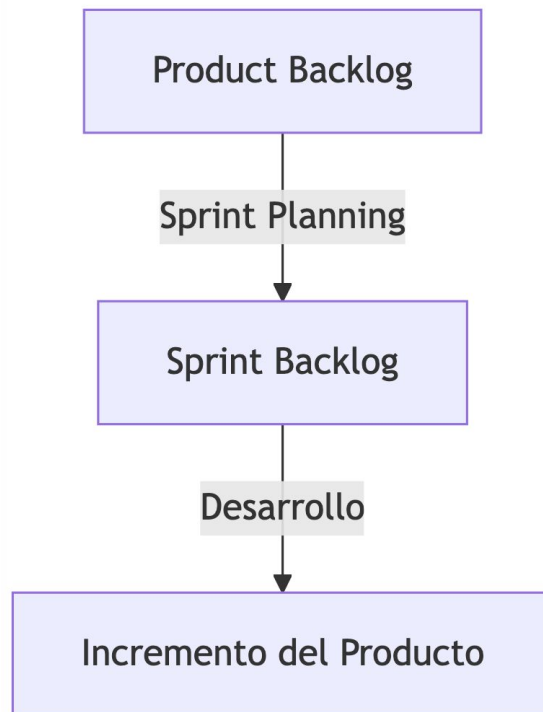
Artefactos en Scrum

Scrum usa 3 artefactos principales:

Artefacto	Descripción
Product Backlog	Lista priorizada de requerimientos y tareas.
Sprint Backlog	Conjunto de tareas seleccionadas para el Sprint actual.
Incremento	Resultado del Sprint: producto funcional y entregable.

Artefactos en Scrum

El objetivo es que cada Sprint genere un **incremento funcional del producto**.



Visión del Producto en Scrum

¿Qué es la Visión del Producto?

Es la dirección estratégica que guía el desarrollo del producto.

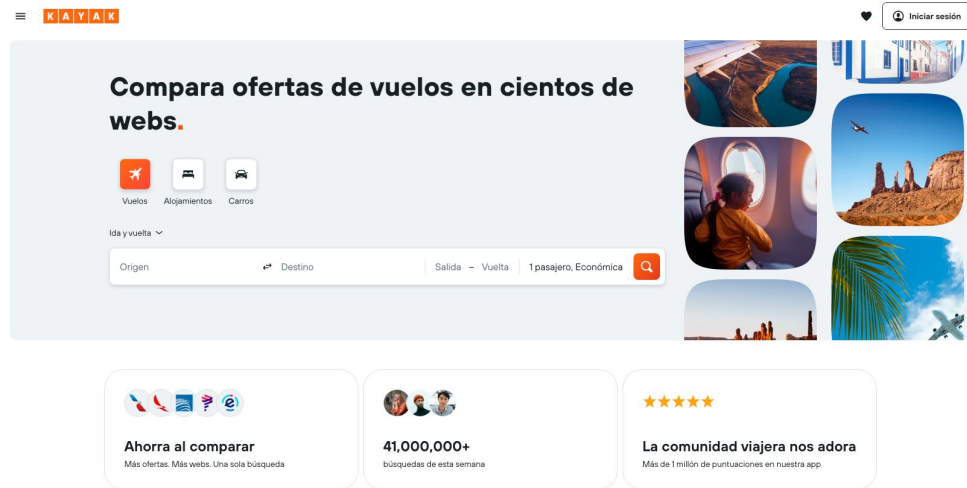
¿Cómo se define la visión?

- Identificando las necesidades del usuario.
- Estableciendo los objetivos a largo plazo.
- Alineando el equipo con los objetivos de negocio.

Visión del Producto en Scrum

Ejemplo de una Visión de Producto:

"Crear una plataforma de reservas de vuelos intuitiva, rápida y confiable para viajeros frecuentes."



Visión del Producto en Scrum

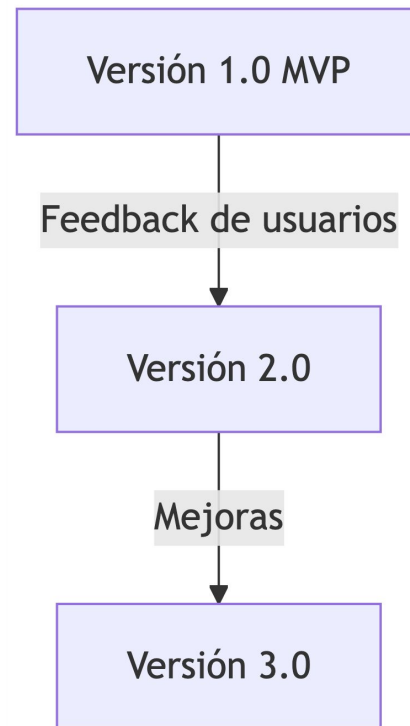
Factores Claves en la Visión:

- **Cómo estimar:** Se usa puntos de historia en lugar de horas.
- **Definición de alcance:** Se ajusta en función del feedback continuo.
- **Gestión de releases:** Se entregan versiones funcionales frecuentemente.
- **Scrum como modelo iterativo:** Cada Sprint mejora el producto.

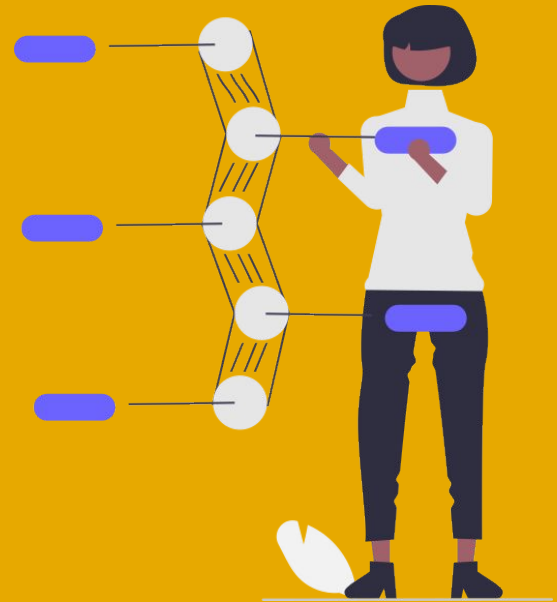
Visión del Producto en Scrum

EJEMPLO DE MVP (PRODUCTO MÍNIMO VIABLE)

Un MVP ayuda a **validar la idea** sin gastar demasiados recursos.



Resumen de lo aprendido



Resumen de lo aprendido

- Las metodologías ágiles nacen como respuesta a modelos rígidos, priorizando flexibilidad y valor.
- Scrum es un marco ágil basado en roles, eventos y artefactos para gestionar proyectos eficientemente.
- La visión del producto en Scrum guía el desarrollo iterativo y la entrega de valor continuo.
- El MVP y la gestión de releases permiten adaptar el producto según necesidades del usuario.

GRACIAS POR TU ATENCIÓN

Nos vemos en la próxima clase

