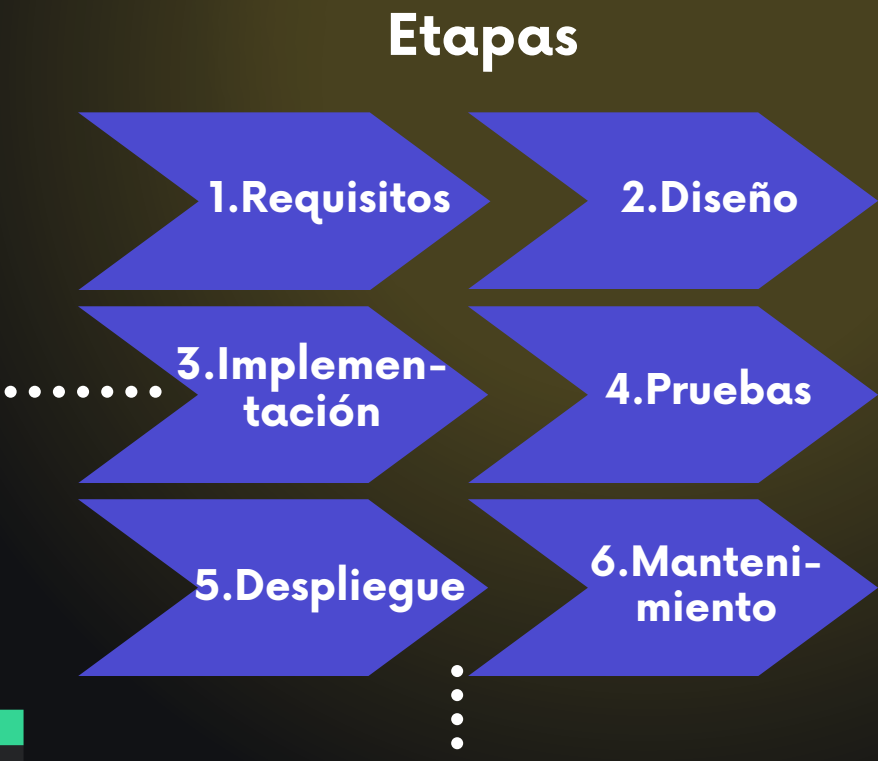


Metodologías tradicionales

Las metodologías tradicionales en el desarrollo de software siguen un enfoque lineal y secuencial, estableciendo todos los requisitos al inicio y avanzando por etapas fijas. No permiten cambios durante el proceso y se centran en la recopilación exhaustiva de requisitos. Estas metodologías han sido reemplazadas por enfoques más ágiles y flexibles que valoran la adaptación y la colaboración. Las metodologías ágiles permiten ajustar el software a medida que se desarrolla, entregando iteraciones rápidas y adaptadas a las necesidades del cliente. Esto ha llevado a una mayor eficiencia y satisfacción en el desarrollo de software.

1.Cascada

La metodología en cascada es un enfoque secuencial y lineal de desarrollo de software. Cada etapa se completa antes de avanzar a la siguiente, lo que implica que los cambios son costosos de implementar una vez que se avanza. Es adecuada para proyectos con requisitos estables, pero puede resultar inflexible frente a cambios.

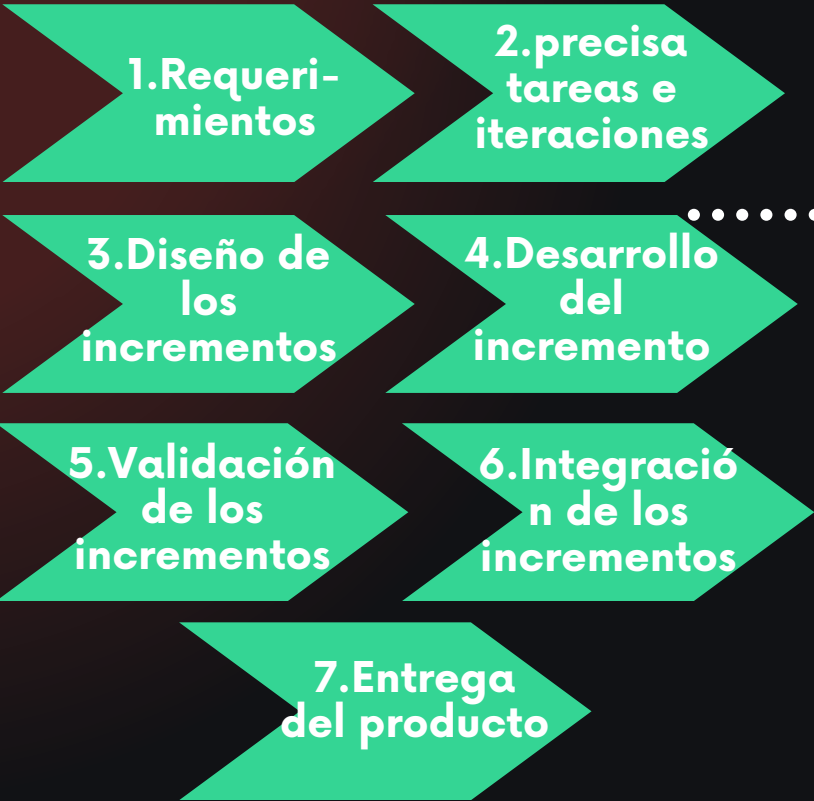


2.Incremental

El modelo de desarrollo incremental consiste en la elaboración progresiva del producto final, agregando funcionalidades en cada etapa para obtener resultados más rápidos. Permite utilizar el software antes de que esté completamente terminado y ofrece mayor flexibilidad. El proyecto se divide en incrementos que entregan funcionalidades según la prioridad asignada a los requisitos. En resumen, este enfoque facilita un desarrollo ágil y flexible al entregar incrementos funcionales en orden de prioridad.

Ventajas	Desventajas
<ul style="list-style-type: none">Estructura claraDocumentación exhaustivaControl de calidad	<ul style="list-style-type: none">Poca flexibilidadRiesgo de retrasos y erroresComunicación limitada

Etapas

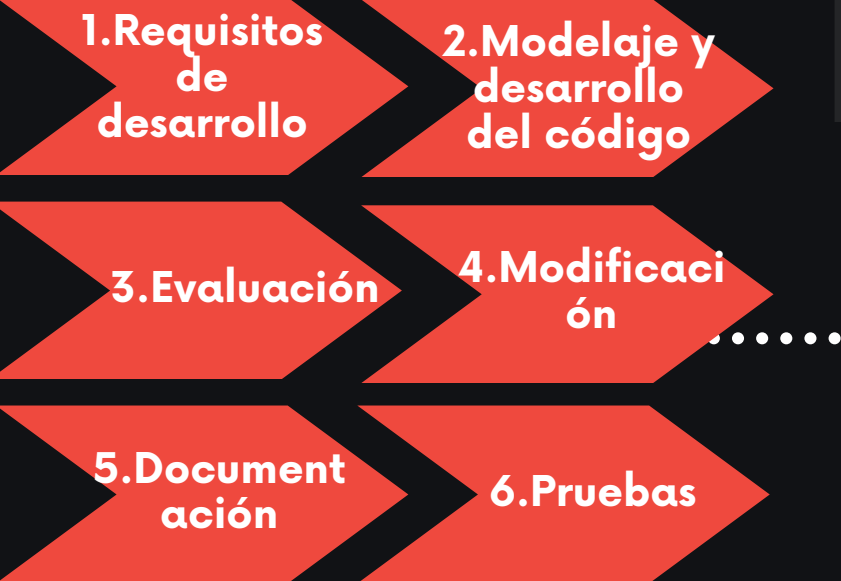


Ventajas	Desventajas
<ul style="list-style-type: none">Entrega temprana de funcionalidadAdaptabilidadMayor participación del clienteMayor control de riesgos	<ul style="list-style-type: none">Mayor complejidad de gestiónRequiere una buena arquitecturaMayor esfuerzo de documentaciónRiesgo de integración

3.Prototipos

La metodología de desarrollo de software prototipo, se fundamenta en la elaboración de un prototipo que se construye rápidamente, para que el cliente o usuario lo pruebe y proporcione su feedback, lo que permite detectar y arreglar lo que está mal, e introducir requerimientos que puedan presentarse, este modelo se fundamenta en el método de ensayo y error para entender las especificidades del producto.

Etapas



Ventajas	Desventajas
<ul style="list-style-type: none">Retroalimentación tempranaMayor comprensión de requisitosReducción de riesgos	<ul style="list-style-type: none">Costos y tiempoComplejidad de gestiónLimitaciones en la escalabilidad

Metodologías ágiles

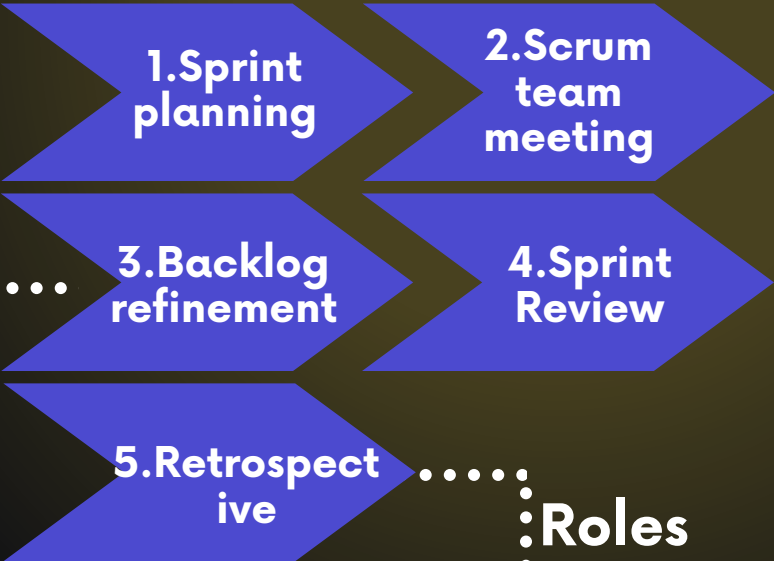
Las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.

En esencia, las empresas que apuestan por esta metodología consiguen gestionar sus proyectos de forma flexible, autónoma y eficaz reduciendo los costes e incrementando su productividad. Con este Master en Metodologías ágiles 100% online aprenderás los aspectos básicos para gestionar las metodologías ágiles dentro de tu empresa.

1. Scrum

Scrum es una metodología ágil de desarrollo de software basada en sprints cortos de 2 a 4 semanas. Los roles clave son el Product Owner, el Scrum Master y el Equipo de Desarrollo. Los artefactos principales son el Product Backlog, el Sprint Backlog y el Incremento. Los eventos incluyen el Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective. Scrum se basa en la transparencia, inspección y adaptación, promoviendo la colaboración y la entrega incremental de valor. Requiere el compromiso de todos los miembros del equipo para tener éxito.

Etapas



Roles



Ventajas

- Flexibilidad
- Mayor satisfacción del cliente
- Colaboración y comunicación mejoradas
- Transparencia
- Mejora continua

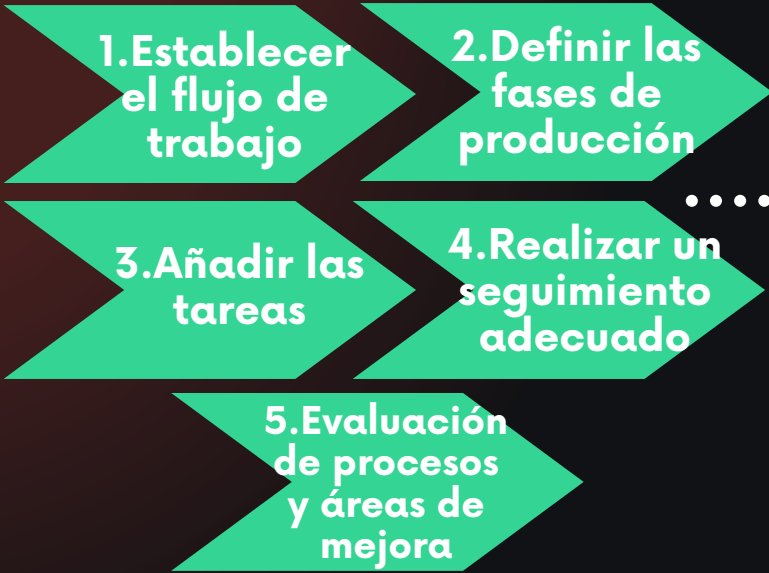
Desventajas

- Requiere un equipo autónomo
- No es adecuado para proyectos complejos
- Requiere una buena gestión del tiempo
- Enfoque en resultados a corto plazo

2. Kanban

Kanban es una metodología basada en la mejora continua y la visualización del flujo de trabajo. Se implementa a través de tableros Kanban, que permiten a los equipos gestionar y visualizar las tareas y la carga de trabajo. Cada columna del tablero representa una etapa del trabajo y las tareas avanzan de una columna a otra hasta completarse. Es una metodología flexible que ayuda a equilibrar el trabajo y la disponibilidad del equipo, fomentando la transparencia y la eficiencia en el proceso de desarrollo.

Etapas



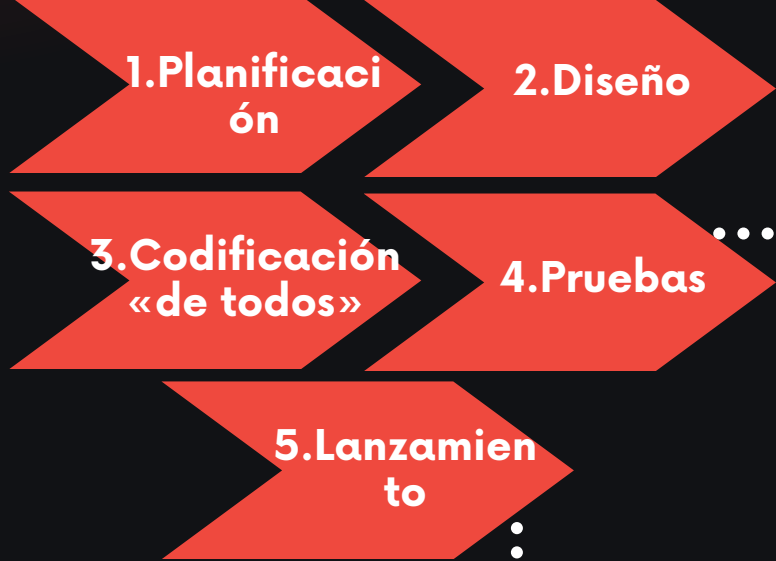
Ventajas

- Flexibilidad
- Visibilidad del flujo de trabajo
- Mejora continua
- Enfoque en la entrega de valor

Desventajas

- Falta de estructura inicial
- Complejidad en proyectos grandes
- Dependencia de la comunicación efectiva
- Riesgo de falta de priorización

Etapas



3. Extreme Programming

Extreme Programming (XP) es una metodología ágil que se enfoca en responder rápidamente a los cambios y garantizar entregas de alta calidad en menos tiempo. Con un mercado dinámico y lleno de incertidumbres, XP permite realizar pruebas y ajustes constantes en proyectos en constante cambio. Esta metodología busca optimizar los procesos y generar más valor para el cliente.

Ventajas

- Flexibilidad
- Entrega rápida y continua
- Comunicación y colaboración
- Calidad del software

Desventajas

- Requiere una comunicación efectiva
- Depende de la disponibilidad del cliente
- No es adecuado para proyectos grandes y complejos
- Requiere un equipo bien capacitado