

Ciclo de Vida del Desarrollo de Software: Modelos Tradicionales vs. Modelos Ágiles

Objetivos de la Clase

- Comprender qué es el ciclo de vida del software.
 - Identificar las etapas comunes del desarrollo de software.
 - Diferenciar entre modelos tradicionales (cascada, espiral, V) y modelos ágiles (Scrum, XP).
 - Analizar ventajas y desventajas de cada enfoque.
-

¿Qué es el Ciclo de Vida del Software?

El ciclo de vida del software es el conjunto de fases o etapas que atraviesa un sistema desde que se concibe hasta que deja de utilizarse.

Etapas comunes:

1. Requisitos
 2. Análisis
 3. Diseño
 4. Implementación
 5. Pruebas
 6. Mantenimiento
-

Modelos Tradicionales

1. Modelo en Cascada

- Secuencial y rígido.

- Cada fase comienza cuando termina la anterior.

Ventajas:

- Claridad en la documentación.
- Buen control del proceso.

Desventajas:

- Dificil adaptación a cambios.
 - Retraso en pruebas y entrega de valor.
-

2. Modelo en V

- Variante del cascada, hace énfasis en la **verificación y validación**.
 - Cada fase de desarrollo tiene su fase de prueba correspondiente.
-

3. Modelo Espiral

- Enfocado en **proyectos grandes y gestión de riesgos**.
 - Iterativo, con énfasis en análisis de riesgos en cada vuelta de la espiral.
-

Modelos Ágiles

¿Qué son los métodos ágiles?

- Enfoque flexible e iterativo.
 - Entrega de valor frecuente.
 - Adaptación constante al cambio.
-

1. Scrum

- Se trabaja por iteraciones llamadas **sprints** (1 a 4 semanas).
 - Roles: Product Owner, Scrum Master, Equipo.
 - Eventos: Daily Stand-up, Sprint Review, Sprint Retrospective.
-

2. Extreme Programming (XP)

- Priorización del código funcional y buenas prácticas de desarrollo.
 - Énfasis en:
 - **Programación en pareja**
 - **Integración continua**
 - **Refactorización**
 - **Pruebas automáticas**
-



Comparativa General

Característica	Tradicional	Ágil
Planificación	Completa al inicio	Iterativa y continua
Cambios	Costosos	Bienvenidos
Entrega	Al final	Frecuente e incremental
Documentación	Extensa	Solo la necesaria
Interacción	Cliente poco presente	Cliente siempre presente

¿Cuándo usar cada uno?

Modelo Tradicional:

- Requisitos estables.
- Proyectos críticos o regulados.
- Equipos grandes con roles definidos.


Modelo Ágil:

- Requisitos cambiantes.
 - Entornos dinámicos.
 - Equipos colaborativos y multidisciplinarios.
-

Actividad en Clase

Divide al curso en 2 grupos:

- Uno defiende un modelo tradicional.
- Otro defiende Scrum o XP.

 Cada grupo debe responder:

- ¿Por qué su modelo es más efectivo?
 - ¿Qué desafíos enfrenta y cómo los resuelve?
-

Cierre y Conclusiones

- No existe un modelo “perfecto”, todo depende del contexto.
- Lo importante es **adaptar el proceso** a las necesidades reales del proyecto, del cliente y del equipo.