

Programador de Videojuegos

CLASE 1 : 28/05/2025



Índice de temas de la clase

- Proyecto
- Software
- Gestión de proyecto
- Metodología
- Modelos de ciclos de vida de un producto software

¿Qué es un proyecto?

Proyecto

- Es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único.
- ¿Por qué inicia?
- ¿Por qué finaliza?
- Los proyectos buscan cumplir objetivos mediante la producción de entregables.

Proyecto - ¿Por qué inicia?

- Cumplir requisitos regulatorios, legales o sociales.
- Satisfacer necesidades de los interesados.
- Implementar o cambiar estrategias de negocio o tecnologías.
- Crear, mejorar o reparar productos, procesos o servicios.

Proyecto - ¿Por qué finaliza?

- Se lograron los objetivos.
- Los objetivos no se cumplirán.
- Se agotó el financiamiento.
- Ya no existe la necesidad del proyecto.
- Los recursos humanos o físicos no están disponibles.
- Por conveniencia o causa legal.

Los proyectos buscan cumplir
objetivos mediante la producción de
entregables.

Objetivo

- Meta hacia la que se debe dirigir el trabajo
- Posición estratégica que se quiere lograr
- Fin que se desea alcanzar
- Un resultado a obtener
- Un producto a producir
- Un servicio a brindar

S M A R T

- Specific (Específico)
- Measurable (Medible)
- Attainable (Alcanzable)
- Relevant (Relevante)
- Time-bounded (Con límite de tiempo)

Entregable

- Cualquier producto, resultado o capacidad único y verificable que se produce para completar un proceso, una fase o un proyecto.
- Puede ser tangible o intangible.

¿Qué es el software?

Software

- Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

¿Por qué falla un proyecto?

¿Por qué falla un proyecto?

- Objetivos no claros.
- No se identificaron todos los interesados.
- Planificación pobre o nula.
- Inadecuado control de cambios.
- Inadecuado manejo de riesgos.

Gestión de proyecto

Es la aplicación de:

- Conocimientos
- Habilidades
- Herramientas y
- Técnicas

Beneficios de la gestión de proyecto

- Cumplimiento de los objetivos del proyecto.
- Satisfacción de las expectativas de los interesados.
- Mayor predictibilidad.
- Mayores probabilidades de éxito.
- Respuesta a los riesgos de forma oportuna.
- Optimización del uso de recursos.
- Mayor control sobre cronograma y presupuesto.

Metodología

Metodología

Las metodologías son un conjunto de principios, procesos, herramientas y técnicas que guían la ejecución de un proyecto. Cada metodología tiene un enfoque único, pero todas buscan lo mismo: maximizar la probabilidad de éxito del proyecto.

Beneficios de utilizar metodologías de gestión de proyectos

- Planificación estructurada.
- Mayor control de tiempo y costos.
- Mejora de la calidad.
- Gestión de recursos eficiente.
- Reducción de riesgos.
- Satisfacción del cliente y los interesados (stakeholders).

Utilización de metodologías para la gestión de proyectos

Ciclo de Vida

Aparición, desarrollo y finalización de la funcionalidad de un determinado elemento.

Un ciclo de vida debe:

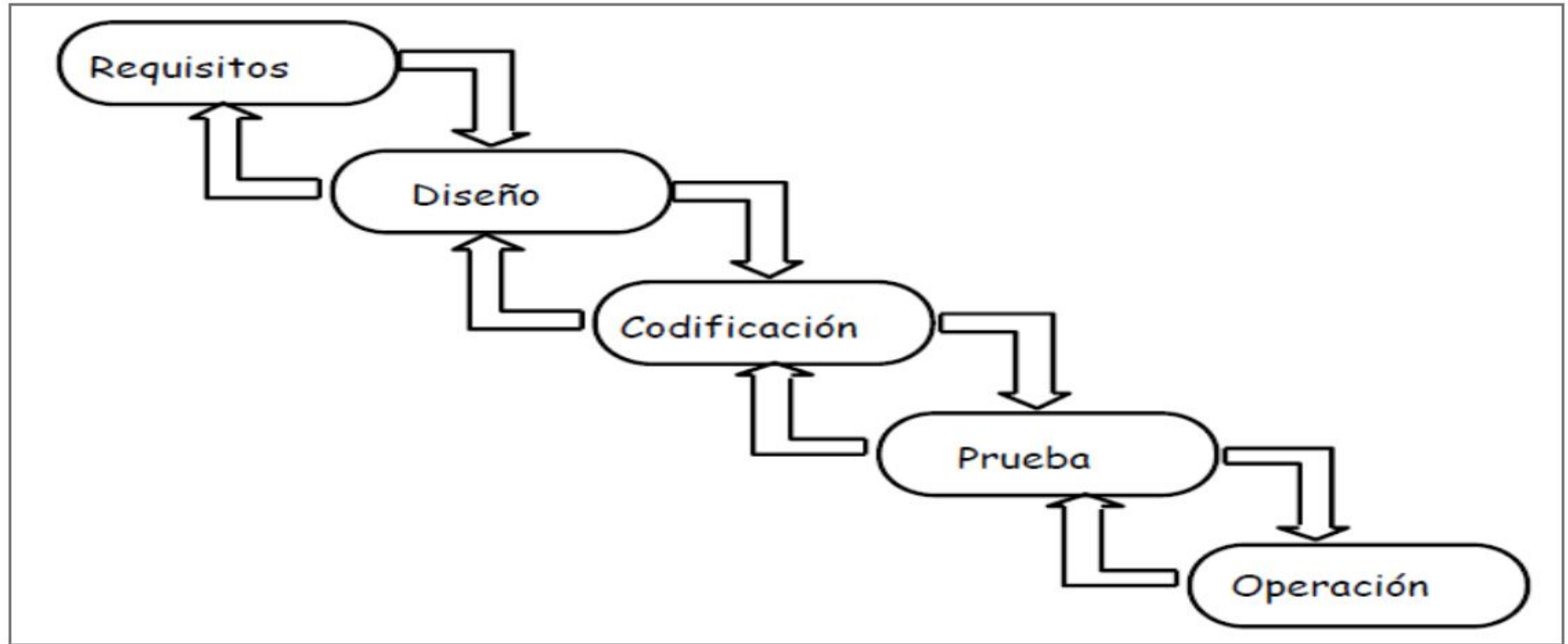
- Determinar el orden de las fases del Proceso Software
- Establecer los criterios de transición para pasar de una fase a la siguiente.

Enfoque de desarrollo CASCADA

Modelo de ciclo de Vida: CASCADA

- El modelo en cascada asume que los requisitos de un sistema pueden ser congelados antes de comenzar el diseño.
- Su uso:
 - Obliga a especificar los requisitos antes de diseñarlo.
 - Obliga a diseñar antes de codificar.
 - Permite al líder del proyecto seguir y controlar los progresos de un modo más exacto.
 - Requiere que el proceso de desarrollo genere una serie de documentos que posteriormente pueden utilizarse para la validación y el mantenimiento del sistema.

Fases del Modelo de ciclo de Vida en CASCADA



Modelo de ciclo de vida en CASCADA: Ventajas y desventajas

Ventajas

- Claridad en cada fase: fácil de entender y administrar.
- Documentación completa en cada etapa.
- Útil para proyectos bien definidos y con requisitos estables.

Desventajas

- Rigidez: no se adapta bien a cambios de requisitos.
- No permite retroceder fácilmente una vez que se avanza a otra fase.
- Difícil de aplicar a proyectos grandes o complejos con alta incertidumbre.

Enfoque de desarrollo PROTOTIPADO

Modelo de ciclo de Vida: PROTOTIPADO

- El modelo por prototipo tiene como objetivo contrarrestar la congelación de requisitos mal comprendidos.
- Se enfoca en construir una versión inicial (prototipo) del software para ayudar a comprender mejor los requisitos del sistema. Este enfoque es útil cuando no se tienen los requisitos completamente claros desde el principio.
- El problema del prototipo es la elección de las funciones que se desean incorporar y cuáles hay que dejar fuera.

Fases del Modelo de ciclo de Vida PROTOTIPADO

1. Análisis preliminar y especificación de requisitos.
2. Diseño, desarrollo e implementación del prototipo.
3. Prueba del prototipo.
4. Refinamiento iterativo del prototipo.
5. Refinamiento de las especificaciones de requisitos.
6. Diseño e implementación del sistema final.

Modelo de ciclo de vida PROTOTIPADO: Ventajas y desventajas

Ventajas

- Permite descubrir y refinar requisitos de forma temprana.
- Involucra al usuario desde el principio.
- Reduce el riesgo de que el producto final no cumpla las expectativas.

Desventajas

- Puede llevar a un desarrollo desorganizado si no se controla adecuadamente.
- Los usuarios pueden pensar que el prototipo es el producto final.
- A veces se invierte mucho tiempo en mejorar el prototipo en lugar del producto real.

Modelos derivados del uso de prototipos

- **Maqueta:** Aporta al usuario ejemplo visual de entradas y salidas. La diferencia con el anterior es que en los prototipos desechables se utilizan datos reales, mientras que las maquetas son formatos encadenados de entrada y salida con datos simples estáticos.
- **Prototipo desechable:** Se usa para ayudar al cliente a identificar los requisitos de un nuevo sistema. En el prototipo se implantan sólo aquellos aspectos del sistema que se entienden mal o son desconocidos. El usuario, mediante el uso del prototipo, descubrirá esos aspectos o requisitos no captados. Todos los elementos del prototipo serán posteriormente desechados.
- **Prototipo evolutivo:** Es un modelo de trabajo del sistema propuesto, fácilmente modificable y ampliable, que aporta a los usuarios una representación física de las partes claves del sistema antes de la implantación. Una vez definidos todos los requisitos, el prototipo evolucionará hacia el sistema final. En los prototipos evolutivos, se implantan aquellos requisitos y necesidades que son claramente entendidos, utilizando diseño y análisis en detalle así como datos reales.

Enfoque de desarrollo INCREMENTAL

Modelo de ciclo de Vida: INCREMENTAL

- El modelo de ciclo de vida incremental es un modelo donde el producto se construye y entrega por partes (incrementos), y cada incremento añade funcionalidad al sistema ya existente. Es una combinación del modelo en cascada con la entrega por fases.

Fases del Modelo de ciclo de Vida INCREMENTAL

1. Recolección de Requisitos Globales
2. Diseño Global del Sistema
3. División en módulos o incrementos
4. Desarrollo del primer incremento
5. Entrega del incremento al usuario
6. Recopilación de retroalimentación
7. Desarrollo de incrementos adicionales
8. Mantenimiento final

Modelo de ciclo de vida INCREMENTAL: Ventajas y desventajas

Ventajas

- Se entrega software funcional desde las primeras etapas.
- Más flexible que el modelo en cascada.
- Fácil de detectar errores tempranos.
- Permite retroalimentación continua del cliente.

Desventajas

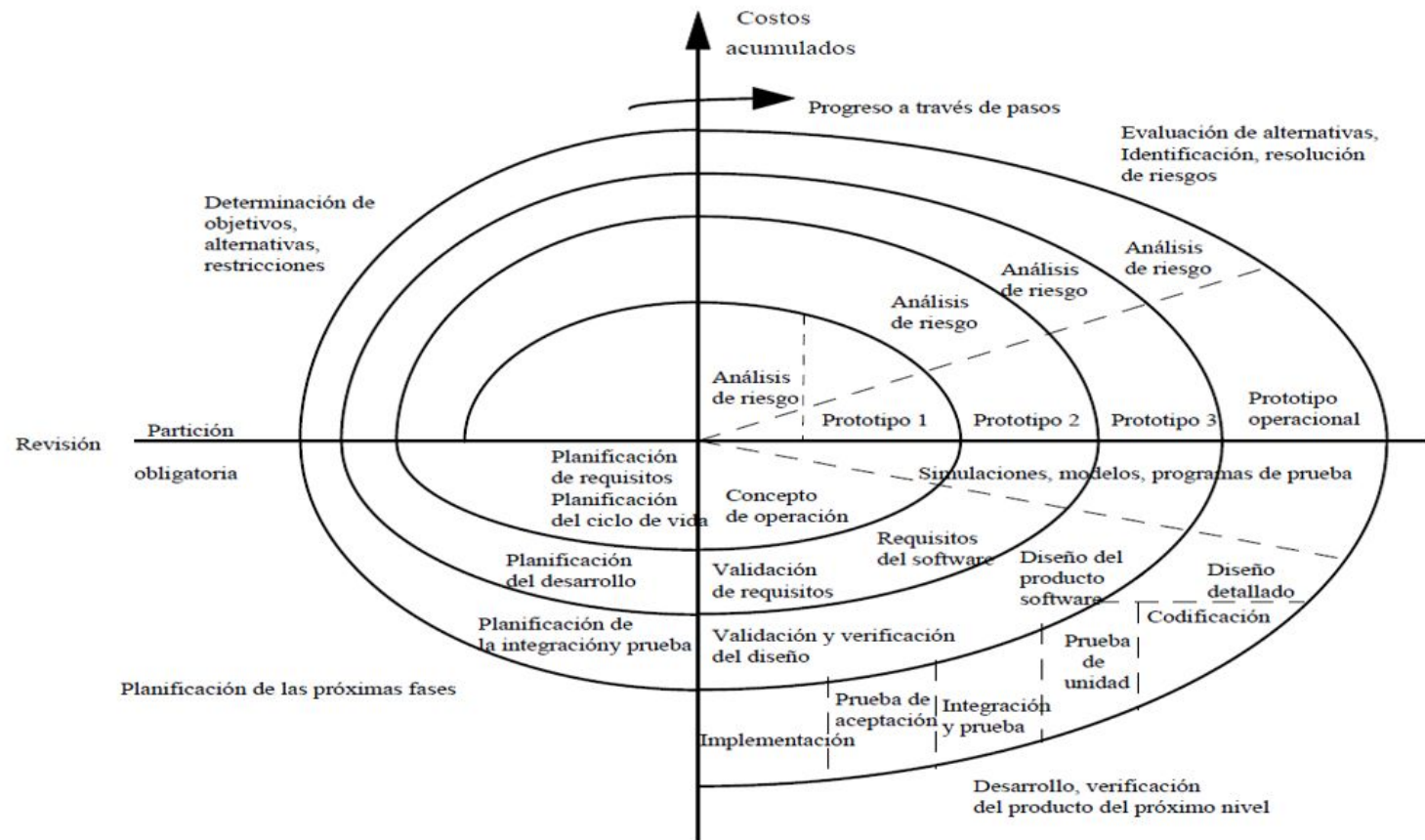
- Requiere una buena planificación inicial para dividir bien los incrementos.
- Puede haber problemas de integración entre módulos si no se controla bien.
- El diseño puede volverse complejo si se hacen muchos cambios con cada incremento.

Enfoque de desarrollo ESPIRAL

Modelo de ciclo de Vida: ESPIRAL

- Es un modelo dirigido por el riesgo.
- Para cada ciclo habrá cuatro actividades:
 - Determinar objetivos
 - Análisis del riesgo
 - Desarrollar y probar
 - Planificación
- Medición de avance en cada ciclo:
 - La dimensión radial mide el costo.
 - La dimensión angular mide el grado de avance del proyecto.

Fases del Modelo de ciclo de Vida en ESPIRAL



Modelo de ciclo de vida ESPIRAL: Ventajas y desventajas

Ventajas

- Excelente para proyectos grandes y complejos.
- Permite identificar y reducir riesgos en etapas tempranas.
- Se adapta fácilmente a cambios en los requisitos.
- Mayor calidad y sostenibilidad del software desarrollado.

Desventajas

- Puede ser costoso y complejo de gestionar.
- Requiere experiencia para una buena evaluación de riesgos.
- Difícil de aplicar en proyectos pequeños.

Enfoque de desarrollo ÁGIL

Modelo de ciclo de Vida: ÁGIL

- Es un enfoque iterativo e incremental para el desarrollo de software, centrado en la colaboración con el cliente, la adaptación al cambio, y la entrega continua de valor. Se basa en los principios del Manifiesto Ágil y es ampliamente utilizado en entornos dinámicos y cambiantes.
- El desarrollo ágil de software, no es más que una metodología de gestión de proyectos adaptativa, que permite llevar a cabo, proyectos de desarrollo de software, adaptándose a los cambios y evolucionando en forma conjunta con el software.

Fases del Modelo de ciclo de Vida ÁGIL

Aunque Ágil no sigue fases rígidas como otros modelos, típicamente se compone de estas etapas repetidas en iteraciones o sprints (de 1 a 4 semanas):

1. Inicio del Proyecto
2. Planificación del Sprint
3. Diseño y Desarrollo
4. Pruebas Continuas
5. Entrega del Incremento
6. Revisión y Retroalimentación (Sprint Review)
7. Retrospectiva
8. Repetición del Ciclo

Modelo de ciclo de vida ÁGIL: Ventajas y desventajas

Ventajas

- Alta adaptabilidad a cambios de requisitos.
- Entrega continua de valor al cliente.
- Fomenta la comunicación y colaboración constante.
- Reduce el riesgo de desviarse de lo que el cliente realmente necesita.

Desventajas

- Requiere compromiso constante del cliente.
- Puede ser difícil de manejar en equipos grandes sin experiencia.
- Menos énfasis en documentación formal.

Modelo de ciclo de vida INCREMENTAL \neq Modelo de ciclo de vida ÁGIL

- Aunque ambos entregan el producto por partes (incrementalmente), no son lo mismo. La diferencia principal entre el ciclo de vida incremental y el ciclo de vida ágil radica en la filosofía de trabajo, el nivel de flexibilidad y la forma de manejar los cambios y la colaboración.
- El modelo incremental es más estructurado y predecible. Es ideal para proyectos con requisitos relativamente estables pero que se pueden dividir en partes.
- El modelo ágil es más adaptativo y centrado en el usuario, ideal para proyectos donde los requisitos cambian con frecuencia y se necesita entregar valor rápido