

Ciclos de Vida del Software

Una metodología para el desarrollo de software es un modo de forma sistemática de realizar, administrar y gestionar un proyecto para llevarlo a cabo con altas posibilidades de éxito. ¹

Hay tres principales diferencias entre los ciclos de vida:

- El alcance del ciclo de vida, que depende de hasta donde se desea llegar con el proyecto.
- La cualidad y cantidad de etapas
- La estructura y la sucesión de etapas.

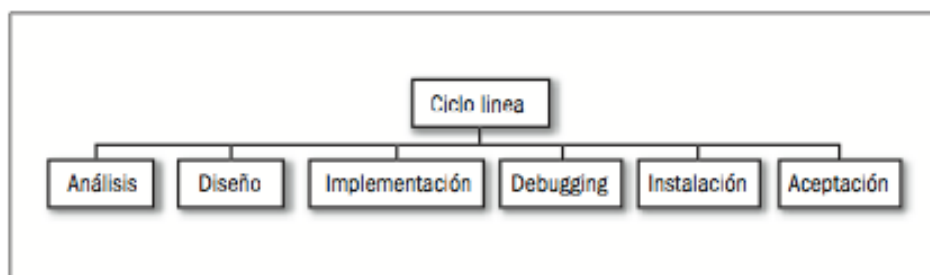
Existen muchos ciclos de vida, aquí se desarrollarán (explicarán) algunos.

1. Ciclo de vida Lineal

Este ciclo de vida describe como descomponer la actividad global del proyecto en etapas separadas, siendo realizadas de forma lineal, es decir, cada etapa se realiza una sola vez y una después de otra.

El riesgo de esta etapa es, que pueden aparecer imprevistos en alguna etapa, que correspondía realizarse en una etapa anterior (que ya pasó).

El ciclo de vida iría así:



2. Desarrollo en cascada

Esta metodología, similar a la lineal, básicamente consiste de las siguientes fases:

- Análisis de requerimientos

Universidad del Valle de Guatemala

Facultad De Ingeniería

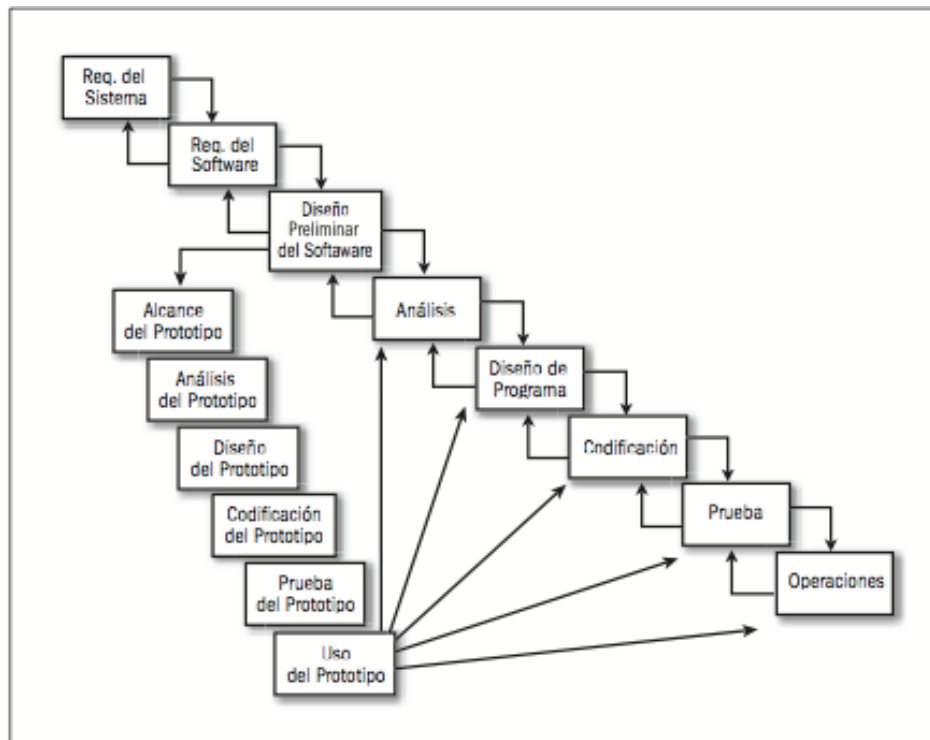
Análisis de Sistemas - CC3003

27 de Enero, 2009.

Carlos Eduardo López Carney - Carné #08107

- Diseño del sistema
- Diseño del programa
- Codificación
- Pruebas
- Implementación
- Mantenimiento

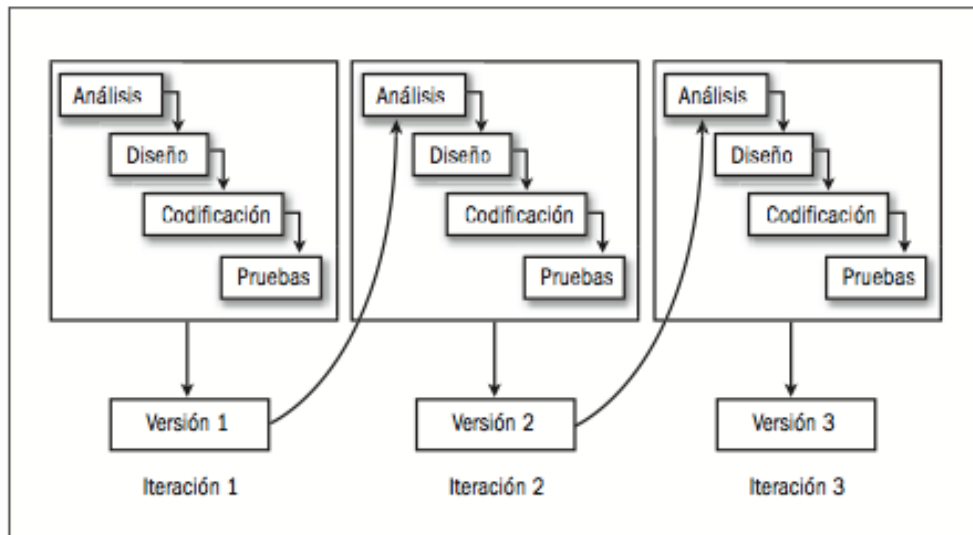
Una desventaja es que, en la vida real, es raro que un proyecto siga una secuencia lineal, lo cual puede hacer que no se realice bien.



3. Ciclo de vida iterativo

Este ciclo de vida, se deriva del ciclo de vida en cascada puro, visto en clase. Busca reducir el riesgo que surge entre las necesidades del usuario y el producto final por malos entendidos durante la etapa de solicitud de requerimientos del usuario.

Al final de cada iteración, se le entrega al cliente una versión mejorada o con mayores funcionalidades del producto. El cliente lo recibe y evalúa el producto, para corregirlo o aceptar que está bien. Esto se repite hasta llegar a la meta, que es, realizar un producto que satisfaga al cliente.



4. Ciclo de vida orientado a objetos. ²

Esta técnica fue presentada en la década de los 90. Al igual que en la filosofía del paradigma de la programación orientada a objetos, en esta metodología, cada funcionalidad o requerimiento solicitado por el usuario, es considerado un objeto.

Los objetos están representados por un conjunto de propiedades y por el comportamiento que tendrán (atributos y métodos).

La principal característica de este es la abstracción de los requerimientos del usuario, por lo que es mucho más flexible que los restantes, ya que permite analizar y desarrollar características esenciales de un objeto, dejando otras preocupaciones menos relevantes, atrás.

5. Ciclo de vida Espiral ³

Este toma las ventajas del modelo de desarrollo en cascada y el de prototipos añadiendo el concepto de análisis de riesgo.

Definiendo cuatro actividades.

- **Planificación:** se recolectan los requisitos iniciales o nuevos requisitos a añadir en cada iteración.
- **Análisis de riesgo:** se decide si se es capaz de desarrollar el sistema basado en los requisitos.
- **Ingeniería:** se desarrolla un prototipo.
- **Evaluación del cliente:** el cliente comenta el prototipo. Si no está satisfecho, se añaden los nuevos requisitos para la siguiente iteración.

Es más usado en proyectos grandes.

Conclusiones y caso del sistema de una Biblioteca

No hay ninguna metodología “ideal” para todos los proyectos, lo que si existe es alguna que se acople más a la cultura de la empresa, capacidad de la empresa o complejidad del sistema.

Para el sistema de la Biblioteca, usaría el ciclo de vida orientado a objetos. La razón es que, hay que construir diferentes módulos e.g. módulo de clientes, módulo de libros, conector a la base de datos. Cada módulo podría modelarse como un objeto, por ejemplo:

Módulo de clientes

Atributos: Conector a la base de datos.

Métodos: Cobrar mensualidad, ingresar cliente.

Módulo de libros

Atributos: Conector a la base de datos.

Métodos: Prestar Libro, Recibir libro de vuelta, ingresar libro.

Conector a base de datos

Atributos: Host, Puerto

Métodos: Crear tabla, insertar en tabla, remover en tabla.

Bibliografía

¹ Implementación y Debugging, Capítulo 1, Ciclos de Vida del Software, http://www.cepeu.edu.py/LIBROS_ELECTRONICOS_3/lpcu097%20-%2001.pdf

² Implementación y Debugging, Capítulo 1, Ciclos de Vida del Software, http://www.cepeu.edu.py/LIBROS_ELECTRONICOS_3/lpcu097%20-%2001.pdf

³ B. Boehm, “A Spiral Model of Software Development and Enhancement”, IEEE, Mayo 1988.