

Ingeniería de Software 6ª Edición

Ian Somerville

Addison Wesley

Resumen Cap. 3 Procesos del software

Modelos del proceso del software

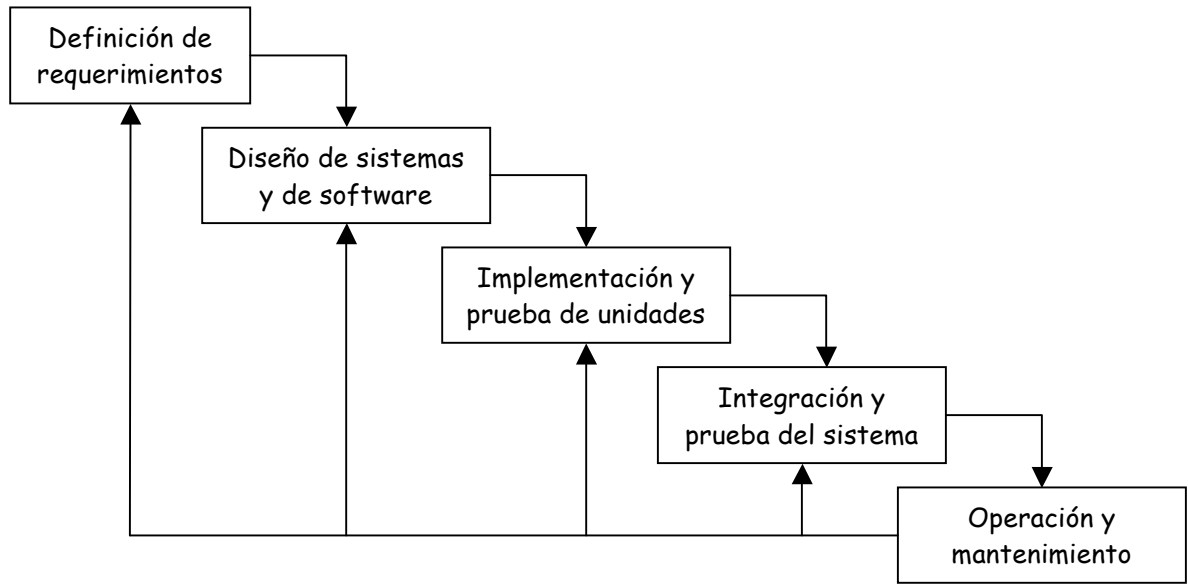
Un modelo del proceso del software es una representación abstracta de un proceso del software. Cada modelo del proceso representa un proceso desde una perspectiva particular por lo que sólo provee información parcial acerca de este proceso.

Estos modelos generales no son descripciones definitivas de los procesos del software. Más bien, son abstracciones útiles que se pueden utilizar para explicar diferentes enfoques para desarrollar software.

Los modelos de proceso que se discuten en este capítulo son:

El modelo de cascada

El primer modelo de proceso de software que se publicó se derivó de otros procesos de ingeniería. Debido a la cascada de una fase a otra, este modelo se conoce como "modelo de cascada" o como ciclo de vida del software. Las principales etapas de este modelo se transforman en actividades fundamentales de desarrollo:

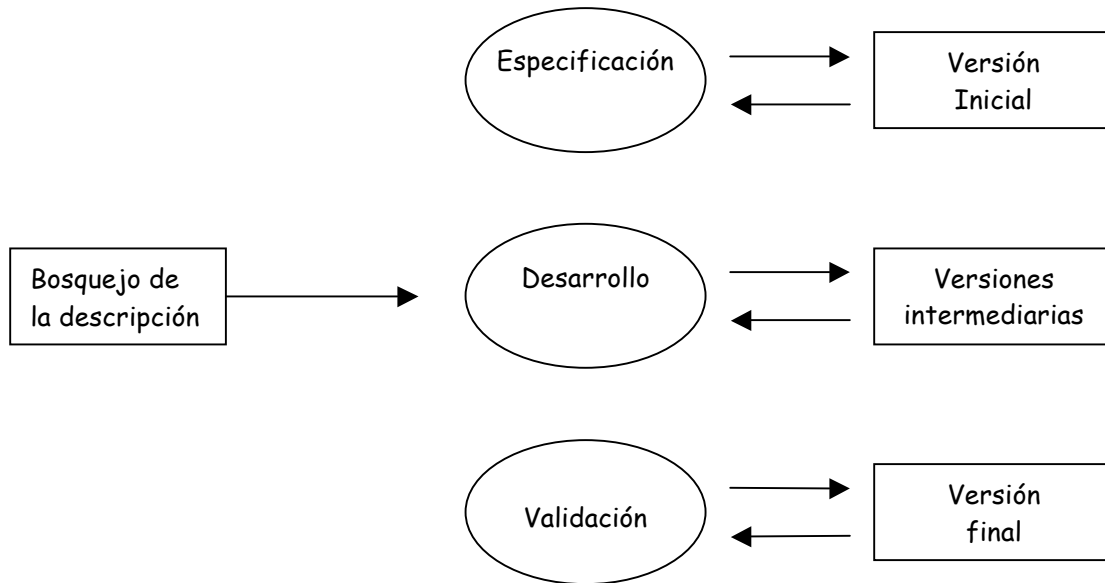


Desarrollo evolutivo

Éste se basa en la idea de desarrollar una implementación inicial, exponiéndola a los comentarios del usuario y refiriéndola a través de las diferentes versiones hasta que se desarrolla un sistema adecuado. Existen dos tipos de desarrollo evolutivo:

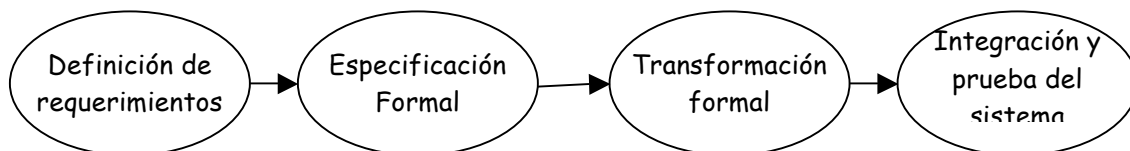
1. Desarrollo exploratorio.- El objetivo del proceso es trabajar con el cliente para explorar sus requerimientos y entregar un sistema final.
2. Prototipos desechables.- el objetivo del proceso de desarrollo evolutivo es comprender los requerimientos del cliente y entonces desarrollar una definición mejorada de los requerimientos del sistema.

Actividades concurrentes



Desarrollo formal de sistemas

El proceso de desarrollo se basa en la transformación matemática formal de una especificación del sistema a un programa ejecutable.

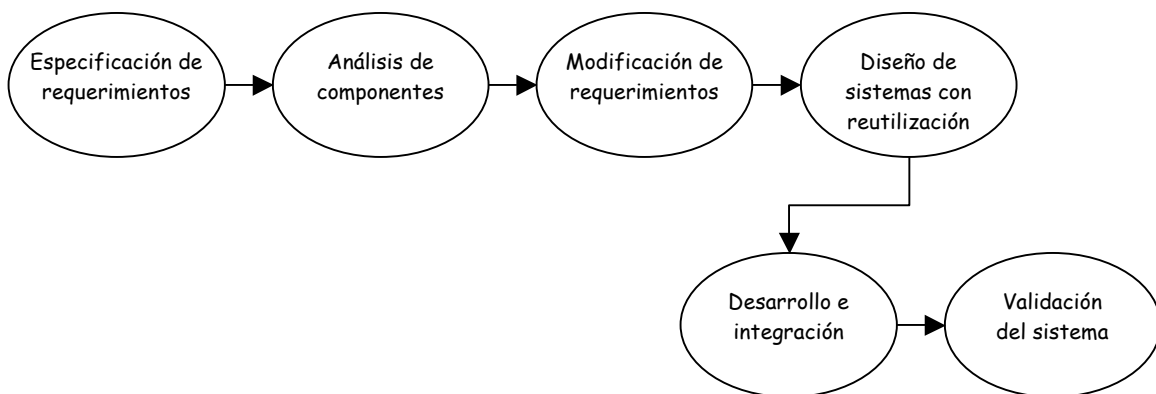


Este enfoque Es adecuado para el desarrollo de sistemas que tienen requerimientos severos de protección, fiabilidad o seguridad. El enfoque formal simplifica la creación de pruebas de protección o seguridad para demostrar a los clientes o a los organismos de certificación que el sistema cumple los requerimientos de protección o seguridad.

Desarrollo orientado a la reutilización

Este enfoque basado en la reutilización se compone de un gran número de componentes de software reutilizable, así como de marcos de trabajo para estos. Algunas veces estos componentes son sistemas por sí mismos que se utilizan para proveer funcionalidad específica, como dar formato al texto, efectuar cálculos numéricos, etc.

Aunque la etapa de especificaciones de requerimientos y la de validación son comparables con otros procesos, las etapas intermedias en el proceso orientado a la reutilización son diferentes.



Iteración de procesos

Conforme cambian los requerimientos del sistema, es necesario apoyar la iteración de procesos en las partes repetidas del proceso. El trabajo de diseño e implementación del sistema debe rehacerse para implementar los cambios en los requerimientos. En esta sección se discuten dos modelos, estos son:

- El desarrollo incremental.- en el que especificación el diseño y la implementación del software se dividen en una serie de incrementos los cuales se desarrollan uno a uno.
- El desarrollo en espiral en el que el desarrollo gira en espiral hacia fuera empezando con un esbozo inicial y terminando con el desarrollo final del sistema. (definición de objetivos, evaluación y reducción de riesgos, desarrollo y validación, planeación).

Especificación del software

La especificación del software, es para establecer que servicios se requieren del sistema y las restricciones de operación y desarrollo del mismo. Esta actividad a menudo se llama ingeniería de requerimientos. Ésta es una etapa particularmente crítica en el proceso del software ya que los errores en esta etapa conducen a problemas posteriores en el diseño e implementación del sistema.

Existen cuatro fases principales en el proceso de ingeniería de requerimientos:

1. Estudio de factibilidad
2. Obtención y análisis de requerimientos
3. Especificación de requerimientos
4. Validación de requerimientos

Diseño e implementación del software

La etapa de implementación de desarrollo de software es el proceso de convertir una especificación del sistema en un sistema ejecutable. Un diseño de software es una descripción de la estructura del software que se va a implementar, los datos que son parte del sistema, las interfaces entre los componentes del sistema y los algoritmos utilizados.

Las actividades del proceso de diseño son:

- Diseño arquitectónico.- Los subsistemas conforman el sistema y su relación se identifica y documenta.
- Especificación abstracta.- Para cada subsistema se produce una especificación abstracta de sus servicios y las restricciones bajo las cuales opera.

- Diseño de la interfaz.- Para cada subsistema se diseña y documenta su interacción con otros subsistemas. Esta especificación de la interfaz debe ser sin ambigüedad ya que permite que el subsistema se utilice sin conocimiento de la forma en que opera.
- Diseño de componentes.- Se asignan servicios a los diferentes componentes y se diseñan sus interfaces.
- Diseño de la estructura de datos.- Se diseña en detalle y especifica la estructura de datos a utilizarse en la implementación del sistema.
- Diseño e algoritmos.- Se diseña en detalle y especifican los algoritmos utilizados para proveer los servicios.

Validación del software

Se utiliza para mostrar que el sistema está acorde a su especificación y que cumple las expectativas del usuario que lo comprará. Incluye procesos de comprobación, como las inspecciones y revisiones, en cada etapa del proceso del software desde la definición de requerimientos hasta el desarrollo del programa. Sin embargo, la mayoría de los casos de validación aparecen después de la implementación, cuando la operación del sistema se pone a prueba.

Las etapas en el proceso de prueba son:

1. Prueba de unidades.- Se prueban los componentes individuales para asegurarse de que operen correctamente.
2. Prueba de módulos.- Un módulo encapsula componentes relacionados, de tal forma que se puede probar sin los otros módulos del sistema.
3. Prueba de subsistemas.- Esta fase comprende la colección de pruebas de los módulos que integran el subsistema.
4. Prueba del sistema.- Los subsistemas se integran para construir el sistema. Este proceso comprende encontrar errores que son el resultado de interacciones no previstas entre los subsistemas y su interfaz.

Evolución del software

La flexibilidad de los sistemas es una de las principales razones por lo que más y más software se incorporan a los sistemas grandes y complejos. Una vez que se decide manufacturar hardware es muy caro hacer cambios en su diseño. Sin embargo, para el software los cambios se pueden hacer en cualquier momento durante o después del desarrollo del sistema. Estos Cambios pueden ser muy caros, pero aún así son mucho más baratos que los correspondientes de los sistemas de hardware.

Ayuda automatizada al proceso

CASE (Ingeniería de Software Asistida por Computadora) es el nombre que se le da al software que se utiliza para ayudar a las actividades del proceso del software como la ingeniería de requerimientos, el diseño, el desarrollo de programas y las pruebas.

La tecnología Case proporciona ayuda al proceso del software automatizando algunas de sus actividades, así como proporcionando información acerca del software en desarrollo.

Las mejoras para la utilización de CASE están limitadas por dos factores:

1. Esencialmente, la ingeniería de software es una actividad de diseño que se basa en la creatividad.
2. En muchas organizaciones, la ingeniería de software es un actividad de equipo y los ingenieros invierten mucho tiempo interactuando con los otros miembros del equipo.

Clasificación del CASE

La clasificación ayuda a comprender tipos de herramientas CASE y su importancia en la ayuda a las actividades del proceso del software. en esta sección se discuten tres herramientas:

- Una perspectiva funcional
- Una perspectiva de proceso
- Una perspectiva de integración

