

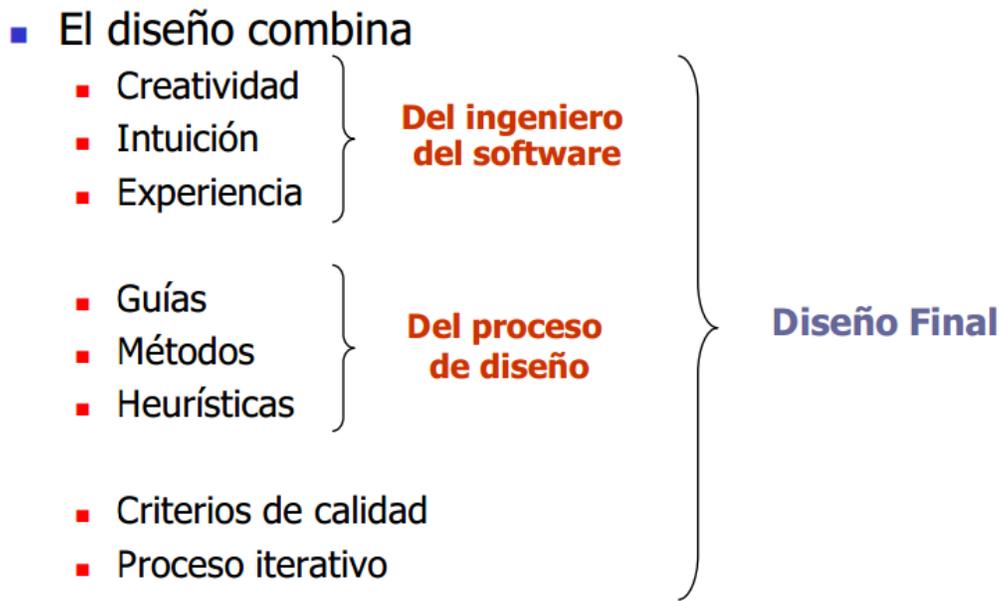
1.1 El diseño del software e Ingeniería del software

Concepto de diseño.- Proceso de aplicar distintas técnicas y principios con el propósito de definir un dispositivo, proceso o sistema con los suficientes detalles como para permitir su realización física [Taylor, 1959]

Proceso común en la actividad humana

- Intuitivamente es el proceso que se trata de formular y evaluar una solución para un problema dado
- En el caso del diseño de un sistema software será la búsqueda de soluciones que se ajusten a los requisitos del usuario
- Actividad necesaria para conseguir un software bien acabado

Diseño como actividad creativa



Definición de diseño de software

□ *Es el proceso de definición de la arquitectura software: componentes módulos, interfaces, procedimientos de prueba y datos de un sistema que se crean para satisfacer unos requisitos especificados [AECC, 1986]*

□ *En un sentido, el diseño es la representación de un objeto que está siendo creado. Un diseño es una información de base que describe aspectos de este objeto, y el proceso de diseño puede ser visto como una elaboración sucesiva de representaciones, tales como añadir más información, puntos de retorno y explorar alternativas [Webster, 1988]*

□ *Es la práctica de tomar una especificación del comportamiento observable externamente y añadir los detalles necesarios para la implementación actual del sistema computacional, incluyendo detalles sobre la interacción de los usuarios, la gestión de tareas y la gestión de datos [Coad y Yourdon, 1991]*

□ *Es un proceso de invención y selección de programas que cumplan los objetivos de un sistema software. La entrada incluye el entendimiento de los requisitos, las restricciones de entorno y los criterios de diseño. La salida del proceso de diseño está compuesta de una arquitectura de diseño que muestra como las piezas están interrelacionadas, de especificaciones de cualquier pieza nueva y de las definiciones de cualquier dato nuevo [Stevens, 1991]*

□ *El diseño de software es el proceso de definir la arquitectura, componentes, interfaces y otras características de un sistema o componente; el resultado de ese proceso IEEE-Std. 610.12 [IEEE, 1999]*

□ *El diseño del 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, algunas veces, los algoritmos utilizados [Sommerville, 2005]*

Evolución del diseño del software

El diseño de software disciplina que evoluciona

□ **Primeros años de la década de los 70s**

- Programación modular [Dennis, 1973]
- Refinamiento descendente [Wirth, 1971]
- Evolución hacia la programación estructurada [Dahl et al., 1972]

□ **Mediados de los 70s**

- Transformaciones de los flujos de datos [Stevens et al., 1974]
- Transformaciones de la estructura de datos [Warnier, 1974], [Jackson, 1975]

□ **Finales de los 80s, década de los 90s**

- Diseño Orientado a Objeto (DOO) [Wirfs-Brock et al., 1990], [Gamma et al., 1995], [Buschmann et al., 1996]
-

Las diferentes tendencias en diseño han dado lugar a métodos de diseño

Ingeniería del software.- La ingeniería de software es una disciplina formada por un conjunto de métodos, herramientas y técnicas que se utilizan en el desarrollo de los programas informáticos (software).

Esta disciplina trasciende la actividad de programación, que es la actividad principal a la hora de crear un software. El ingeniero de software se encarga de toda la gestión del proyecto para que éste se pueda desarrollar en un plazo determinado y con el presupuesto previsto.

La ingeniería de software, por lo tanto, incluye el análisis previo de la situación, el diseño del proyecto, el desarrollo del software, las pruebas necesarias para confirmar su correcto funcionamiento y la implementación del sistema.

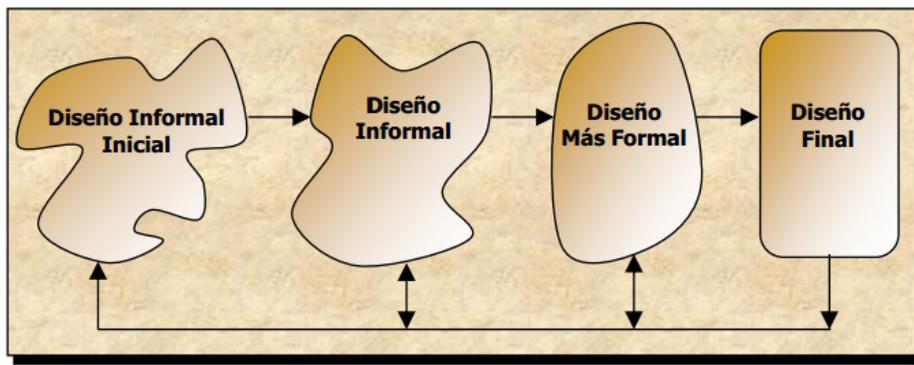
Cabe destacar que el proceso de desarrollo de software implica lo que se conoce como ciclo de vida del software, que está formado por cuatro etapas: concepción, elaboración, construcción y transición.

1.2 El Proceso de diseño del software

El diseño es un proceso de resolución de problemas cuyo objetivo es encontrar y describir una forma

- Para implementar los requisitos funcionales del sistema
- Respetando las restricciones impuestas por los requisitos no funcionales
 - Incluyendo las presupuestarias
- Ajustándose a los principios generales de calidad

El proceso de diseño es, por tanto, un proceso iterativo, mediante el cual se va a realizar una traducción de los requisitos en una representación del software



Diseño como toma de decisiones

- El diseñador se enfrenta a una colección de problemas de diseño
- Cada problema tiene normalmente varias soluciones alternativas
 - Opciones de diseño
- El diseñador toma una decisión de diseño para resolver cada problema
 - Este proceso implica elegir la mejor opción entre las alternativas
- Para tomar las decisiones de diseño el ingeniero software utiliza el conocimiento que tiene de:
 - Los requisitos
 - El diseño realizado hasta el momento
 - La tecnología disponible

- *Los principios de diseño y de las "buenas prácticas"*
- *Lo que ha funcionado bien en situaciones anteriores*

Objetivos de la fase de diseño

- Descomponer el sistema en subsistemas*
 - *Identificar la arquitectura software*
- Determinar las relaciones entre componentes*
 - *Identificar las dependencias entre componentes y determinar los mecanismos de comunicación entre componentes*
- Especificar las interfaces entre los componentes*
 - *Interfaces bien definidas para facilitar la prueba y comunicación entre los componentes*
- Describir la funcionalidad de los componentes*

Actividades del proceso de diseño (i)

- Según Roger S. Pressman (1992)*
- Diseño preliminar*
 - *También llamado diseño de sistema, diseño arquitectónico o diseño de alto nivel*
 - *Identificar los módulos en los que puede dividirse atendiendo a motivos de conveniencia de implementación*
- Diseño detallado*
 - *Se centra en la lógica interna de dichos módulos*
 - *Se ocupa del refinamiento de la representación arquitectónica que lleva a una estructura de datos detallada y a las representaciones algorítmicas del software*

Hay una vertiente técnica y una vertiente de gestión en el diseño

□ Diseño arquitectónico

□ Define la relación entre los elementos estructurales principales del software, los patrones de diseño que se pueden utilizar para lograr los requisitos que se han definido para el sistema, y las restricciones que afectan a la manera en que se pueden aplicar los patrones de diseño arquitectónicos [Shaw y Garlan, 1996]

□ Diseño de datos

□ Transforma el modelo del dominio de información creado en el análisis en las estructuras de datos necesarias para la implementación del software [Pressman, 2006]

□ Influencia de la estructura de datos en la estructura del programa y en la complejidad de los procedimientos

□ Ocultación de la información y Abstracción

□ Datos bien diseñados conducen a

- Mejor estructura del programa***
- Modularidad efectiva***
- Reducción de la complejidad procedimental***

Principios de Wasserman (1996) para el diseño de datos

□ Los principios sistemáticos del análisis aplicados a la función y al comportamiento también deben aplicarse a los datos

□ Deben identificarse todas las estructuras de datos y las operaciones que se han de realizar sobre cada una de ellas

□ Debe establecerse y usarse un diccionario de datos para definir el diseño de los datos y del programa

□ Deben posponerse las decisiones de datos de bajo nivel hasta el diseño detallado

□ La representación de una estructura de datos sólo debe ser conocida por los módulos que hagan uso directo de los datos contenidos en la estructura

Se debe desarrollar una biblioteca de estructuras de datos útiles y de las operaciones que se les pueden aplicar

El diseño del software y el lenguaje de programación deben soportar la especificación y la realización de tipos abstractos de datos

Diseño a nivel de componentes (diseño procedimental)

Transforma los elementos estructurales de la arquitectura del software en una descripción procedimental de los componentes del software

Diseño de algoritmos

Diseño de la interfaz

Diseño de interfaces hombre-máquina para facilitar al usuario la utilización del sistema

Propósito

- *Recoger de los usuarios la información del sistema y ponerla a disposición de otros usuarios*

- La interfaz cubre las entradas y las salidas

Se debe tener en cuenta la psicología del usuario

- *Sobrecarga de la información*
- *Complejidad de la tarea*
- *Grado de control del sistema permitido al usuario*

Ergonomía

- *Estudio de datos biológicos y tecnológicos aplicados a problemas de mutua adaptación entre el hombre y la máquina [RAE, 2001]*

Por su parte D. E. Webster (1988) y L. A. Belady (1990)

Hablan de:

- Diseño de flujo ascendente (Upstream design)*
 - *Es adaptable y abstracto y tiende a corresponderse con las fases de análisis de requisitos y de diseño preliminar*
- Diseño de flujo descendente (Downstream design)*
 - *Se refiere a los módulos, codificación y documentación, correspondiéndose con las fases de diseño detallado e implementación*

1.3.- Principios del diseño del software

Los principios básicos de diseño hacen posible que el ingeniero del software navegue por el proceso de diseño [Pressman, 2002]

- En el proceso de diseño no deberá utilizarse "orejeras"*
- El diseño deberá poderse rastrear hasta el modelo de análisis*
- El diseño no deberá inventar nada que ya esté inventado*
- El diseño deberá minimizar la distancia intelectual entre el software y el problema, como si de misma vida real se tratara*
- El diseño deberá presentar uniformidad e integración*
- El diseño deberá estructurarse para admitir cambios*
- El diseño deberá estructurarse para degradarse poco a poco, incluso cuando se enfrenta con datos, sucesos o condiciones de operación aberrantes*
- El diseño no es escribir código y escribir código no es diseñar*
- El diseño deberá evaluarse en función de la calidad mientras que se va creando, no después de terminarlo*
- El diseño deberá revisarse para minimizar los errores conceptuales (semánticos)*

1.4 Conceptos del diseño

Los conceptos fundamentales del diseño del software Proporcionan el marco de trabajo necesario para conseguir que se haga correctamente, favorecen la gestión de la complejidad de los sistemas software y la consecución de los factores de calidad que estos sistemas han de exhibir.

Los conceptos del diseño a tratar son [Pressman, 2002]

(Alumnos: Desarrollar los conceptos)

- Abstracción*
- Refinamiento sucesivo (descomposición)*
- Ocultación de la información*
- Modularidad*
- Arquitectura del software*
- Jerarquía de control*
- División estructural*
- Estructura de datos*
- Procedimiento de software*

1.5.- Diseño modular efectivo

Se logra desarrollando módulos con una función claramente definida y evitando una excesiva interacción con otros módulos

Suma de la modularidad y los conceptos de abstracción y ocultación de la información [Presman, 2002]

Ventajas

- Módulos más fáciles de desarrollar
- Módulos más fáciles de mantener y probar
- Facilidad para su reutilización

La independencia se mide mediante dos criterios cualitativos

- Cohesión
- Acoplamiento