

Diseño de Sistemas

Clase 1:

Introducción y conceptos de Diseño

Hugo R. Cordero S.



Objetivos

- *Conocer los procesos de Desarrollo de Software y el ciclo de vida*
- *Conocer las principales metodologías de desarrollo de software*
- *Familiarizarse con los conceptos generales del Diseño de Sistemas*
- *Recordar el diseño Orientado a Objetos*



Introducción al Proceso del software

Para preparar una cena

- Personas: Empleados de una empresa de catering
- Producto: La cena que se sirve
- Proyecto: La secuencia de acciones de servir una cena concreta
- Proceso: Las instrucciones sobre cómo se sirve una cena



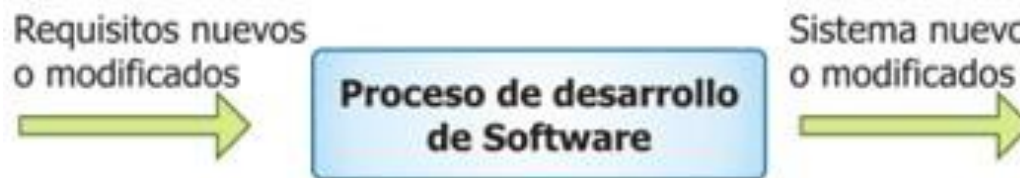
Para el proyecto el curso

- Personas: Estudiantes del curso
- Producto: Aplicación que elijan
- Proyecto: El proyecto del curso para implementar la aplicación
- Proceso: El conjunto de actividades que llevarán a la creación de la aplicación



Proceso del software

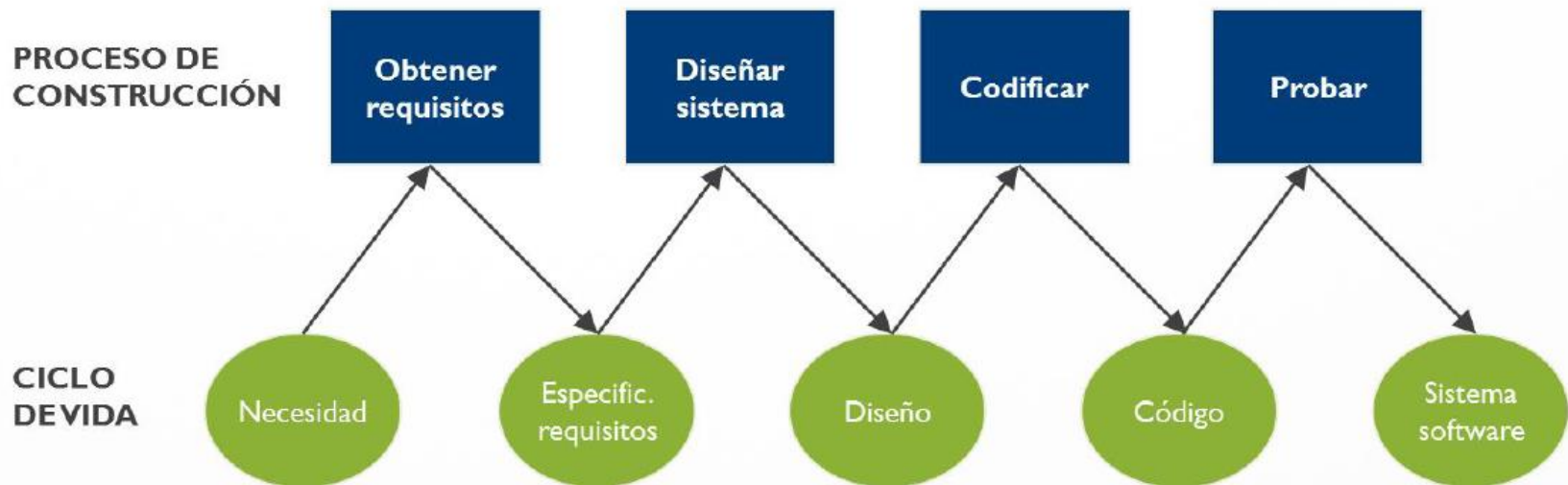
- El *proceso de desarrollo del software* tiene como propósito la producción eficaz y eficiente de un producto de software que reúna los requisitos del cliente.



- Es un proceso intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas.
- El proceso de desarrollo de software no es único. No existe un proceso de software universal que sea efectivo para todos los contextos de proyectos de desarrollo

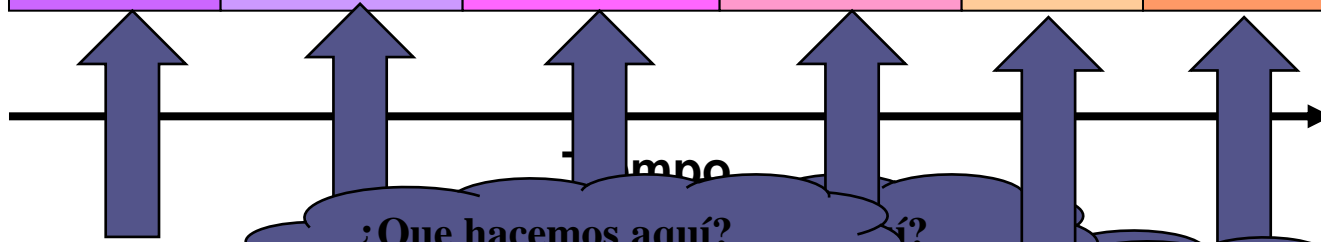
Proceso del software y ciclo de vida

- Proceso: conjunto de actividades que conducen a la creación de un producto software
- Ciclo de vida: conjunto de estados por los cuales pasa el software



Proceso de Desarrollo de Software

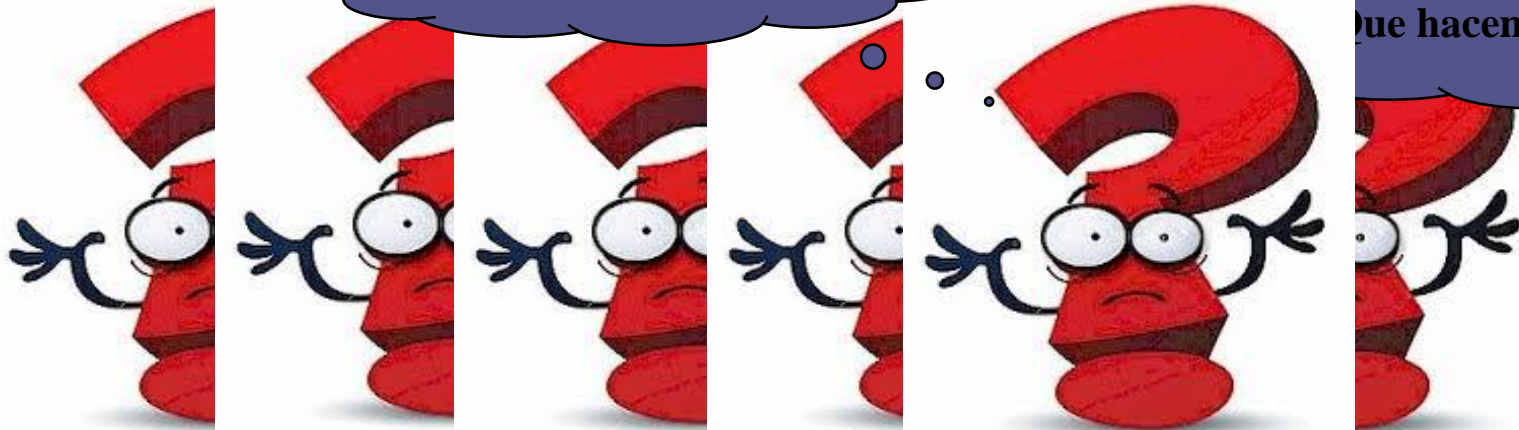
← Fases del Proyecto →



¿Que hacemos aquí?

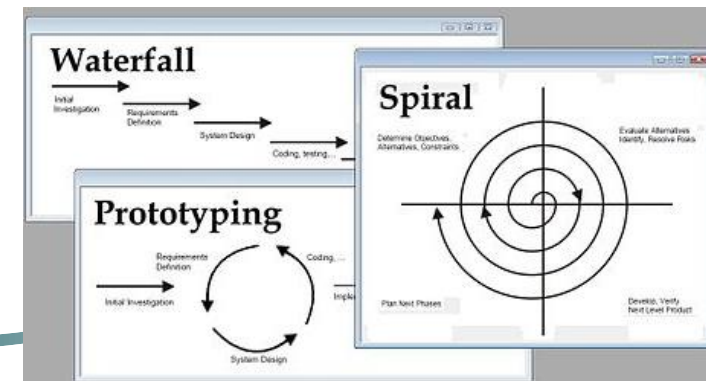
¿?

¿Que hacemos aquí?



Metodologías de Desarrollo Software

- Un modelo de proceso, o paradigma de Ingeniería de Software, o modelo del ciclo de vida de desarrollo de software, es una plantilla o marco que define el proceso a través del cual se crea software
- Dicho de otra forma, los procesos son instancias de un modelo de proceso
- A veces los términos proceso y modelo de proceso se utilizan indistintamente





Metodologías de Desarrollo Software

Modelos Tradicionales

- Modelo en Cascada
- Basado en Prototipos
- Modelo Estructurado
- Desarrollo rápido de aplicaciones
- Modelo Incremental

Modelos Evolutivos

- Modelo evolutivo
- Modelo en Espiral
- Modelo Iterativo
- Desarrollo concurrente



Metodologías de Desarrollo Software

Modelos Especializados

- Desarrollo basado en componentes
- Proceso unificado de Rational (RUP)
- De métodos formales
- Desarrollo orientado a aspectos

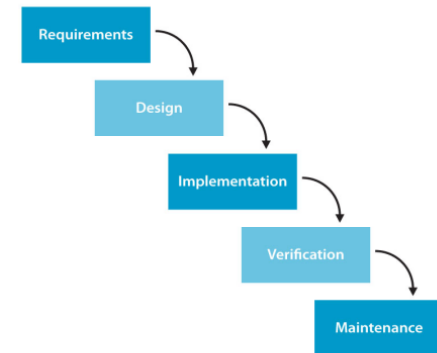
Modelos Ágiles

- Programación extrema (XP)
- Scrum
- Desarrollo de software adaptativo (ASD)
- Otros como: DSDM, FDD y LD

Modelo en Cascada

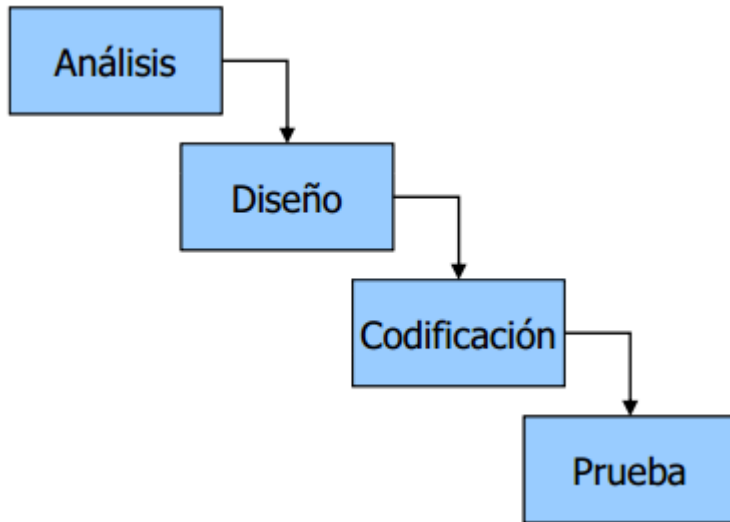
- Este fue el primer modelo de desarrollo de software que se publicó. Se derivó de otros procesos de Ingeniería.
- Toma las actividades fundamentales del proceso como son especificación, desarrollo, validación y evolución
- También conocido como modelo lineal
- Modelo satisfactorio sólo en desarrollos conocidos y estables

[http://www.serena.com/docs/agile/papers/
Managing-The-Development-of-Large-Software-Systems.pdf](http://www.serena.com/docs/agile/papers/Managing-The-Development-of-Large-Software-Systems.pdf)

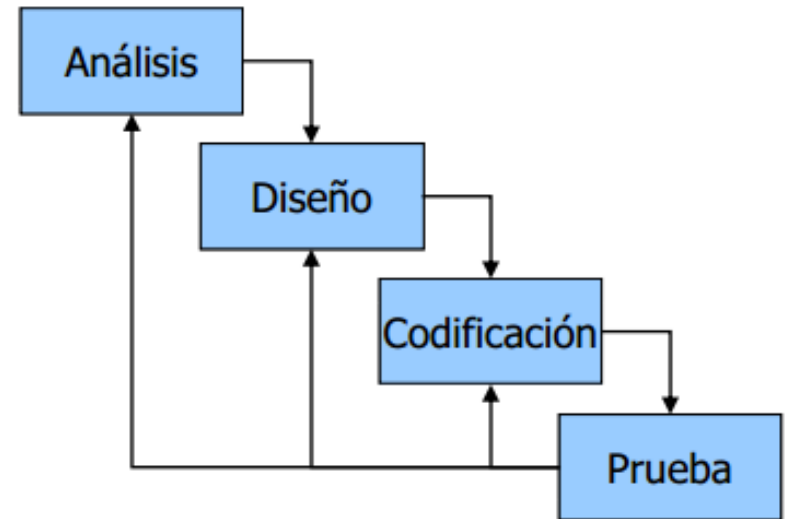


Modelo en Cascada

Clásico



Clásico con retroalimentación





Modelo basado en Prototipos

- Características
 - Es una aplicación que funciona
 - Su finalidad es probar varias suposiciones con respecto a las características requeridas por el sistema
 - Se crean con rapidez
 - Evolucionan a través de un proceso iterativo
 - Tienen un costo bajo de desarrollo
- Tipos
 - Desechables, es una versión rudimentaria que posteriormente será desechada
 - Evolutivos, se convertirá en la versión final del sistema

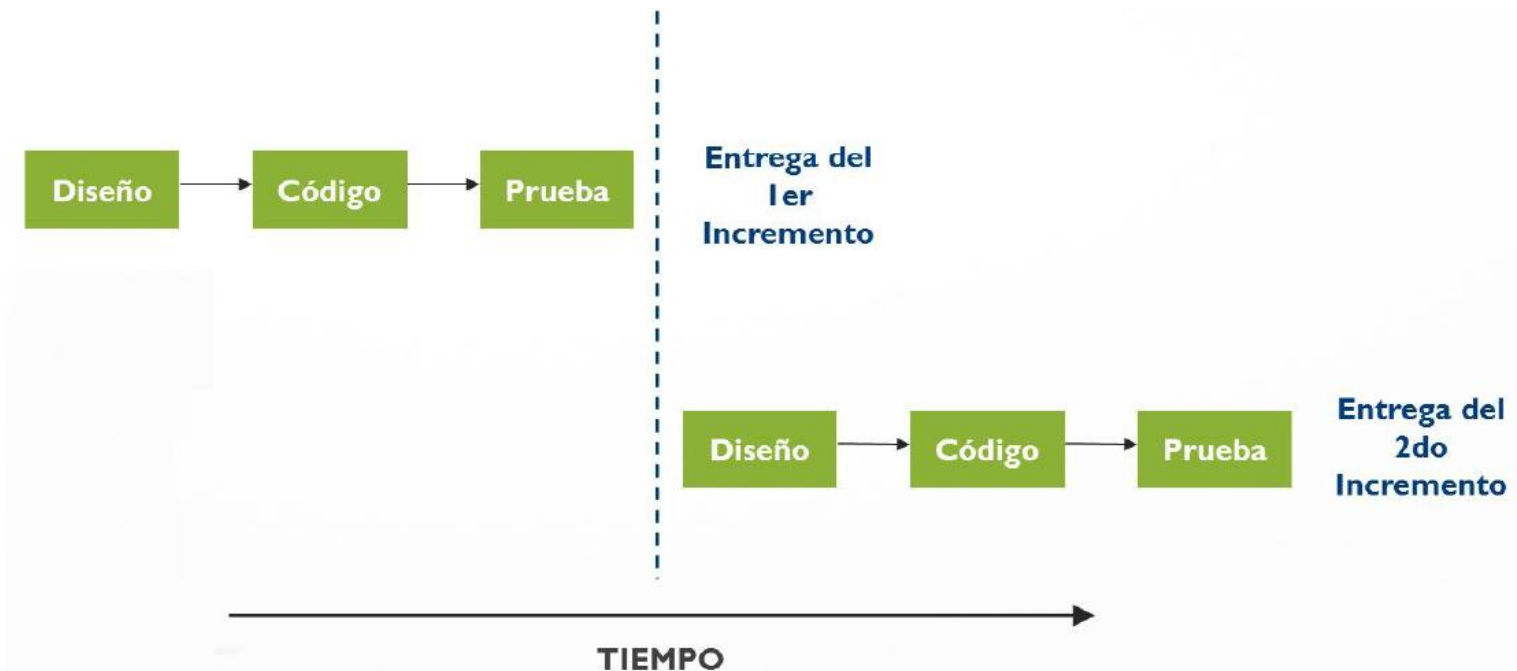
Modelo basado en Prototipos

- Combinación de prototipos desechables y evolutivos para los requisitos poco conocidos
 - Conocido como prototipado operativo

| | DESECHABLE | EVOLUTIVO |
|------------------------|-----------------------------------|---|
| Enfoque de desarrollo | Rápido y sin rigor | Riguroso |
| Qué construir | Sólo las partes problemáticas | Primero las partes bien entendidas. Sobre una base sólida. |
| Directrices del diseño | Optimizar el tiempo de desarrollo | Optimizar la modificabilidad |
| Objetivo último | Desecharlo | Incluirlo en el sistema |

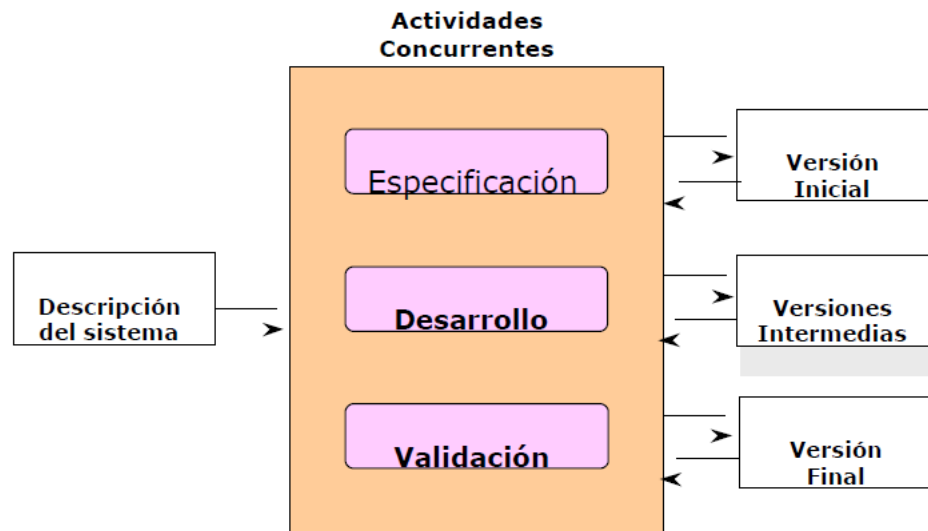
Modelo Incremental

- Modelo incremental según la ISO/IEC TR 15271:1998
- Se tiene requisitos claros



Modelo Evolutivo

- La idea detrás de este modelo es el desarrollo de una implantación del sistema inicial, exponerla a los comentarios del usuario, refinarla en N versiones hasta que se desarrolle el sistema adecuado.
- Gestiona la naturaleza evolutiva del software





Proceso Unificado

- Más conocido por su especificación más detallada dada por Rational, como: Proceso Unificado de Rational (RUP)
- Definido por Rational Software Corporation:
 - Evolución del proceso Objectory (Orientado a Objetos)
 - Utilización de UML como lenguaje de modelado
 - Creado en 1996, posteriormente comprado por IBM en el 2003
- Mantiene un conjunto de métodos adaptables al contexto y necesidades de cada organización
- Es una implementación del modelo en espiral



Proceso Unificado

Características

- Conducido por casos de uso
 - Los casos de uso se implementan para asegurar que toda la funcionalidad se realice en el sistema y verificar y probar el mismo
- Centrado en la arquitectura
 - La arquitectura se describe mediante diferentes vistas del sistema. Es importante establecer una arquitectura básica pronto, realizar prototipos, evaluarla y finalmente refinarla durante el curso del proyecto
- Iterativo e incremental
 - Resulta práctico dividir los grandes proyectos en mini proyectos, cada uno de los cuales es una iteración que resulta en un incremento

Proceso Unificado

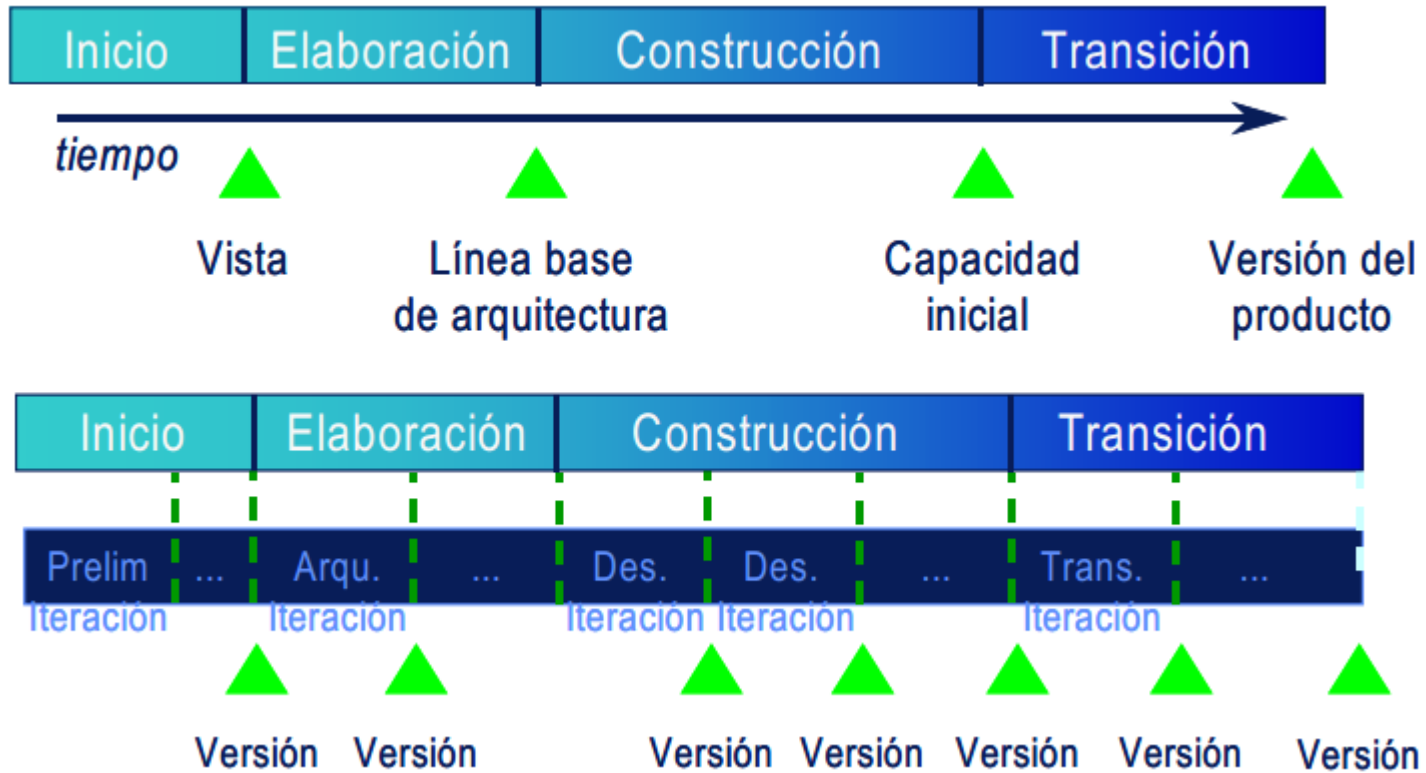


Fases

- Inicio: se define el alcance del proyecto y se desarrollan los casos de negocio
- Elaboración: se planifica el proyecto, se especifican en detalle la mayoría de los casos de uso y se diseña la arquitectura del sistema
- Construcción: se construye el producto
- Transición: el producto se convierte en versión beta. Se corrigen problemas y se incorporan mejoras sugeridas en la revisión

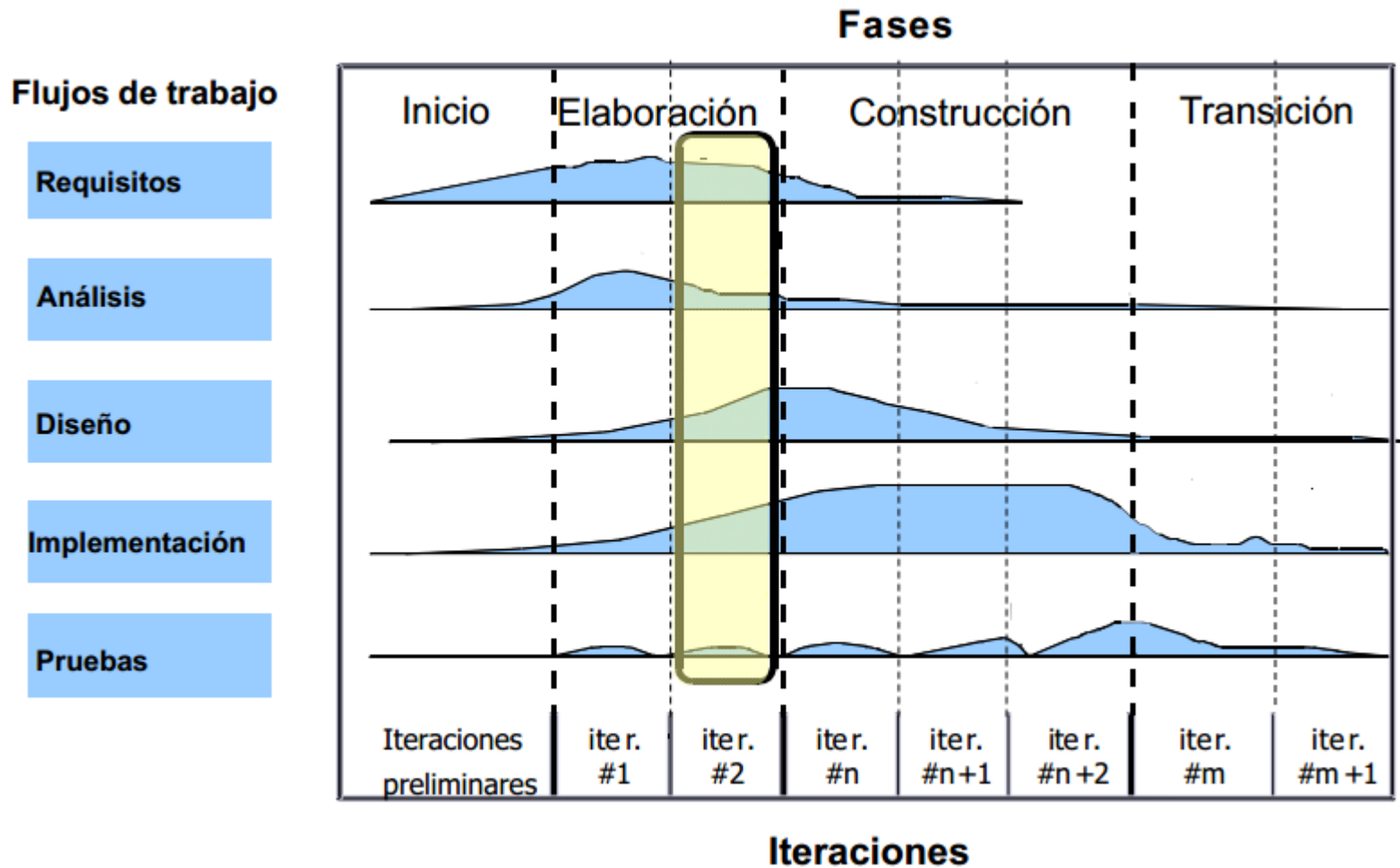
Proceso Unificado

- Cada ciclo concluye con una versión del producto para los clientes



Proceso Unificado

- Las iteraciones discurren a largo de los flujos de trabajo



Metodología ágil: SCRUM

- Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años.
- Está especialmente indicada para proyectos con un rápido cambio de requisitos.
- La palabra scrum procede del rugby, donde designa al acto de preparar el avance del equipo en unidad pasando la pelota a uno y otro jugador



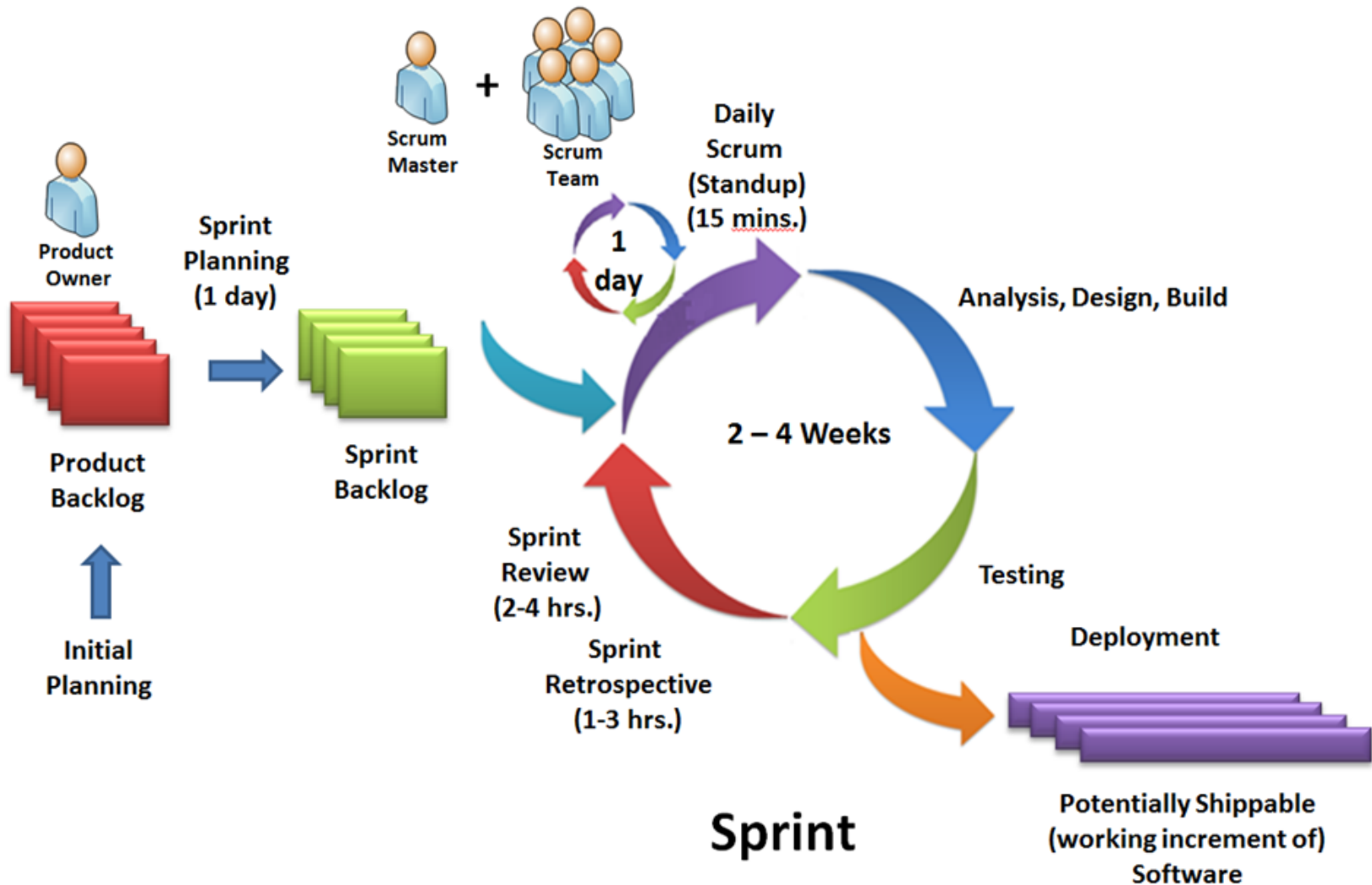


Metodología ágil: SCRUM

- Metodología ágil que define un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental.
- Sus principales características se pueden resumir en dos:
 - Se basa en iteraciones, denominadas sprints, con una duración de no más de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente.
 - La segunda característica importante son las reuniones a lo largo del proyecto. Una reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.



Metodología ágil: SCRUM



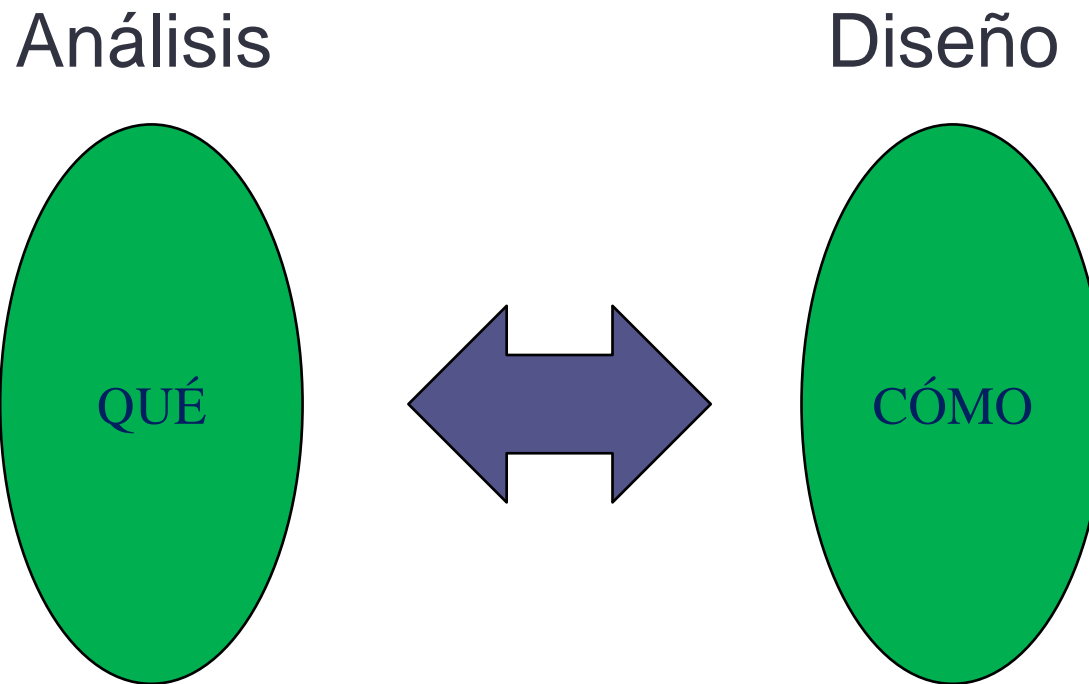


Metodología ágil: SCRUM

- Las metodologías ágiles ofrecen una solución casi a medida para una gran cantidad de proyectos
- Las metodologías ágiles permiten a los pequeños grupos de desarrollo concentrarse en la tarea de construir software fomentando prácticas de fácil adopción y en un entorno ordenado que permiten que los proyectos finalicen exitosamente
- A pesar de las críticas que sufren, son usadas por muchas grandes empresas y se han utilizando en grandes sistemas, lo que hace prever que estas metodologías han llegado para quedarse

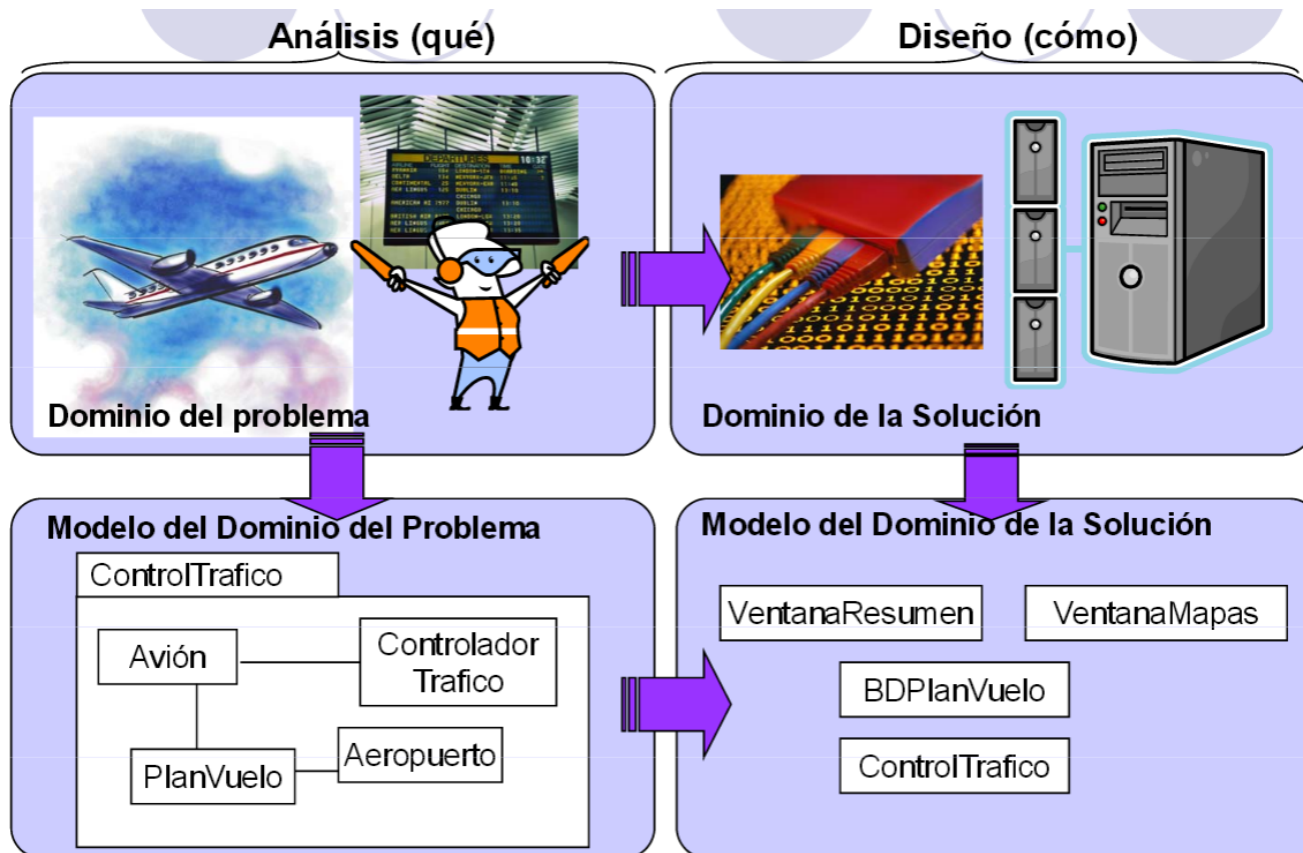
Introducción al Diseño

- El Diseño



Introducción al Diseño

- El Diseño





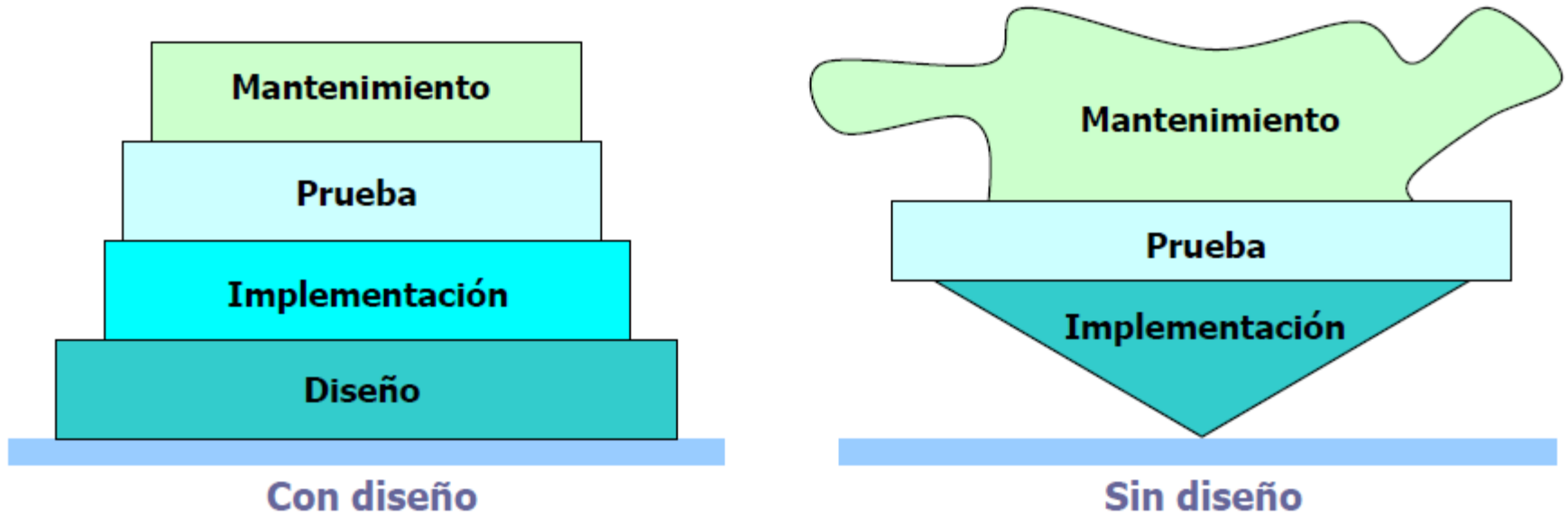
Conceptos generales del Diseño

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

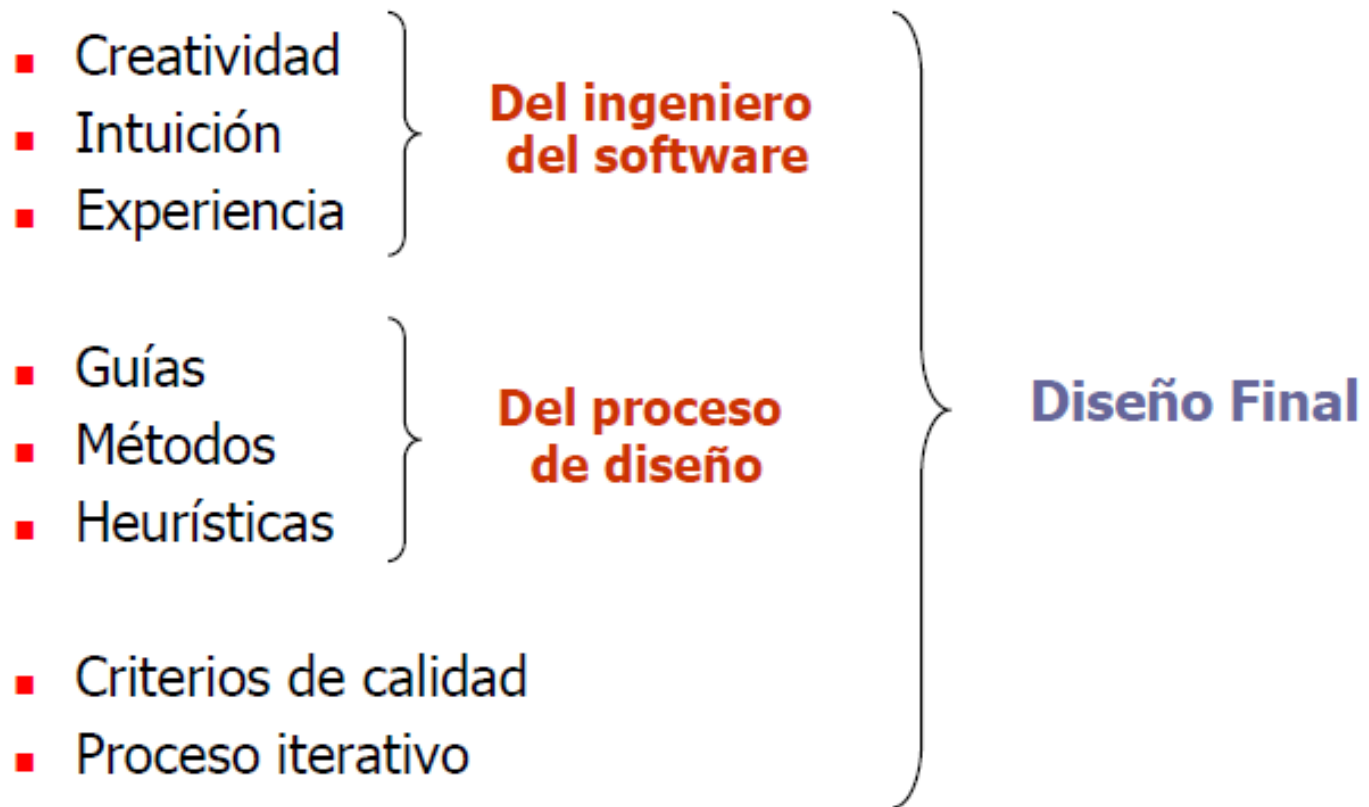
Importancia del Diseño

Importancia del diseño en el ciclo de vida:



Conceptos generales del Diseño

El diseño como actividad creativa, combina:

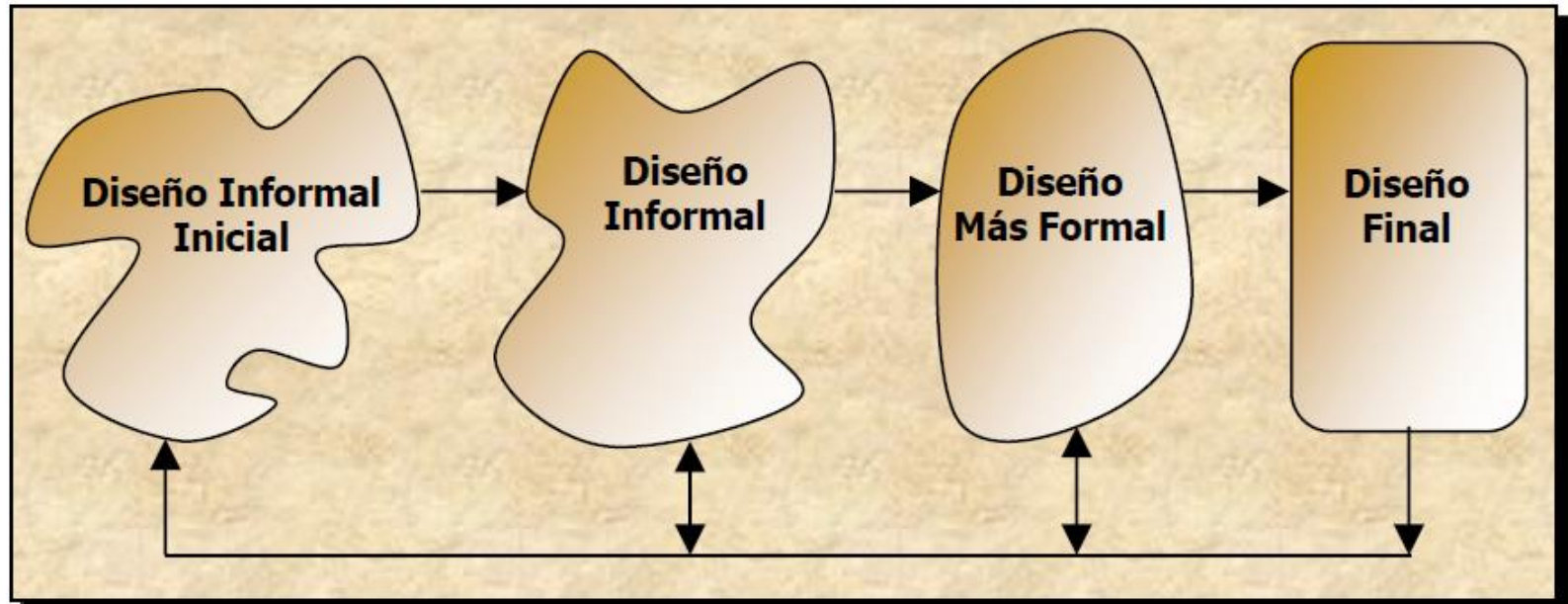




Diseño de Sistemas

- El diseño sirve para para implementar los requerimientos funcionales del sistema.
- Debe respetar las restricciones impuestas por los requerimientos no funcionales (incluyendo restricciones de presupuesto)
- Se ajusta a los principios generales de calidad
- Por tanto, un proceso iterativo, mediante el cual se va a realizar una traducción de los requisitos en una representación del software

Proceso del Diseño de Sistemas





Diseño de Sistemas

El diseñador se enfrenta a una colección de problemas de diseño

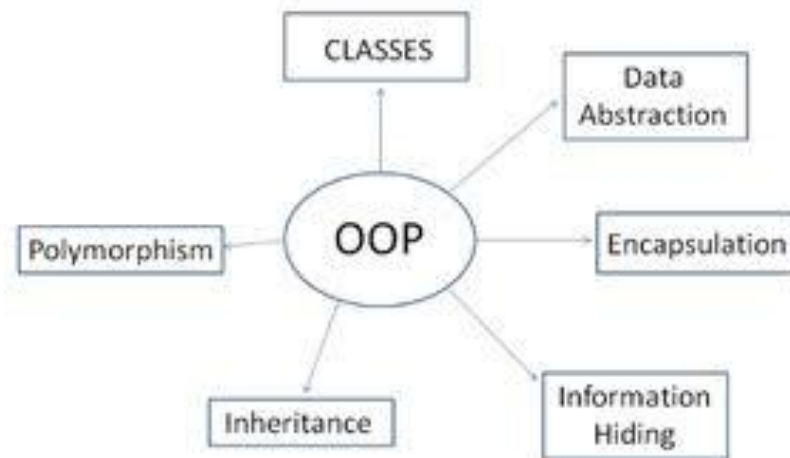
- Cada problema tiene normalmente varias soluciones alternativas
 - Opciones de diseño
- El diseñador toma una decisión de diseño para resolver cada problema
 - Este proceso implica elegir la mejor opción entre las alternativas

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

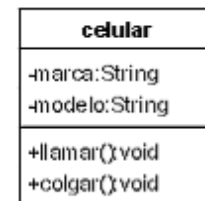
Diseño orientado a Objetos

- Es un enfoque de la ingeniería de software que modela un sistema como un grupo de objetos que interactúan entre sí
- Es también visto como una metodología de programación que se define en términos de clases de objetos: entidades que combinan datos, procedimientos y propiedades

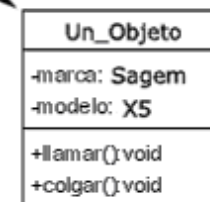


Diseño orientado a Objetos

- A diferencia con otros métodos de diseño, el DOO produce un diseño que interconecta objetos y operaciones de procesamiento para esos objetos, de forma que se modulariza la información y el procesamiento, en lugar de aislar el procesamiento.
- Proporciona un mecanismo que permite al diseñador conseguir apropiadamente las características de abstracción, ocultamiento de información y modularidad.



Creando un objeto de la Clase





Diseño orientado a Objetos

Características

- Diseños modulares
- Efectos laterales mínimos (encapsulamiento)
- Extensibilidad
- Fácil de modificar
- Orientado a datos
- Explota la herencia (jerárquico)
- Reutilización de clases

Diseño orientado a Objetos

Beneficios

- Mantenimiento más fácil.
- Los objetos son componentes que permiten la reusabilidad.
- Implementar una biblioteca de componentes.
- Para algunos sistemas, puede haber una obvia correspondencia entre entidades del mundo real y objetos del sistema
- Mejor comprensión y mantenimiento.
- Permite crear sistemas desacoplados.



Diseño orientado a Objetos

Inconvenientes

- Forma de pensar diferente: curva de aprendizaje lenta
- Difícil seguir el flujo de ejecución
- Si no se lleva una buena administración se generan problemas de mantenimiento.





Diseño orientado a Objetos

Conclusiones

- La identificación de objetos (o clases) es la parte mas difícil del DOO.
- No existe una fórmula mágica para la identificación de objetos. Depende de la experiencia y conocimiento del dominio de los arquitectos.
- La identificación de objetos es un proceso iterativo.

Para reflexionar

“La mayoría de los ejecutivos tienen una capacidad notoriamente subdesarrollada para entender y tratar las emociones”.

Manfred F. R. Kets de Vries
INSEAD, Francia



“Las emociones descontroladas pueden hacer estúpido al inteligente”

Doug Lennick
American Express

Y... ¿en el Perú tenemos este problema?



Resumen

- El proceso de desarrollo del software tiene como propósito la producción eficaz y eficiente de un producto de software que reúna los requisitos del cliente.
- Las metodologías son modelos del proceso o modelo del ciclo de vida de desarrollo de software, y las podemos agrupar en tradicionales, evolutivas, especializados y ágiles
- Podemos definir Diseño de Sistemas como el proceso de describir, organizar y estructurar los componentes de un sistema, teniendo en cuenta los aspectos de Hardware, Software, comunicaciones, etc.
- A diferencia con otros métodos, el Diseño Orientado a Objetos produce un diseño que interconecta objetos y operaciones de procesamiento para esos objetos.



¿Preguntas?



- ¿Cuál es la importancia del Diseño de Sistemas?



Bibliografía

- Ingeniería de Software, Una perspectiva orientada a objetos, 1ra. edición, Eric J. Braude
 - Capítulo 1, Proceso; parte 1, Esencial
- Ingeniería de Software, 1ra. edición, Guillermo Pantaleo y Ludmila Rinaudo
 - Capítulo 4, Metodologías de desarrollo de software
- Ingeniería del Software, 9na. edición, Ian Sommerville
 - Capítulo 3: Desarrollo ágil de software
- Ingeniería de Software: Un enfoque práctico, 7ma. edición, Roger S. Pressman
 - Capítulo 8: Conceptos de diseño
- Ingeniería de Software. Un enfoque desde la guía SWEBOK 1ra. edición, S.Sánchez - M.Sicilia - D.Rodríguez
 - Capítulo 5: Diseño