

Introducción

Ingeniería de Software

Prof. Gabriel Matonte
matonteg@gmail.com

Temario

- Información de la materia
- Problemática del desarrollo de software
- Calidad e ingeniería de software
- Proceso y ciclo de vida
- Áreas de la ingeniería de software

Información de la Materia

Objetivo

Brindar los conceptos teóricos y prácticos que permitan comprender y ejecutar los distintos procesos que involucra el desarrollo de software.

Información de la Materia

Temario:

1. Introducción
2. Modelos de Proceso
3. Gestión de Proyectos
4. Requerimientos y Análisis
5. Diseño
6. Implementación
7. Verificación

Información de la Materia

Bibliografía

- Fundamentals of Software Engineering
[Ghezzi, Jazayeri, Mandrioli]
- Software Engineering – Theory and Practice
[Pfleiger]
- Software Engineering – Theory and Practice
[Sommerville]

Información de la Materia

Materiales:

- Presentaciones vistas en clase
- Materiales colgados en la web personal del profesor
- Apuntes de clase
- Bibliografía

Información de la Materia

Evaluaciones:

- 2 parciales
- 1 obligatorio

Para exonerar promedio de 9 o más

Para ir reglamentado de 5 a 8

Menos de 5 examen libre

Información de la Materia

Forma de Trabajo:

- Clases teórico/prácticas de 4 hs.
- Clases de Consulta (hs. De departamento)
- Clases de revisión de Obligatorio
- Consultas vía correo electrónico

Problemática del desarrollo de software

Problemática

- ¿Cuáles son los principales problemas del desarrollo de software?

Problemática

- Algunas dificultades comunes:
 - Complejidad del producto
 - Defectos en el producto
 - Es difícil planificar y cumplir plazos
 - Costos altos de los proyectos de software
- Situación:
 - Crisis continua de la producción de software
 - Evolución en los métodos y técnicas

Problemática

- Particularidades de la producción de software
 - Producto es intangible y complejo
 - Implica desarrollo y no manufactura
 - Involucra tareas realizadas por personas
 - Necesidad de alto grado de comunicación
 - Normalmente requiere creatividad

Problemática

- Complejidad del producto
 - Alto cambio tecnológico
 - Dificultad en satisfacer las necesidades de los usuarios
 - Integración de sistemas heterogéneos
 - Intangibilidad del software
 - Sistemas críticos

Problemática

- Contexto del negocio
 - El software es un elemento estratégico
 - Rápidos cambio del contexto del negocio
 - Tiempos de salida al mercado más cortos
 - Alto costo de desarrollo y mantenimiento
 - El software es muy visible
 - Afecta procesos críticos

Problemática

- Profesionalizar el desarrollo de software.
- Transformar el desarrollo de arte a ingeniería.
- Sistematizar actividades posibilitando espacio para la creatividad.
- **Aplicar ingeniería de software.**

Calidad e ingeniería de software

Calidad e ingeniería de software

- Definición de Ingeniería de Software
 - (1) La aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, la aplicación de ingeniería al software.
 - (2) El estudio de los enfoques como en (1).
[Estándar IEEE 610.12]

Calidad e ingeniería de software

- Definición de Ingeniería de Software
 - Disciplina tecnológica y gerencial abocada a la producción y mantenimiento sistemático de productos de software que son desarrollados y modificados en tiempo y bajo estimaciones de costos. [Fairley]

Calidad e ingeniería de software

- Definición de Ingeniería de Software
 - Aplicación disciplinada de **principios de ingeniería, ciencia y matemática, métodos y herramientas** a la producción **económica** de software. [Humphrey]

Calidad e ingeniería de software

- Definición de calidad [Juran 1999]
 - (a) Calidad es el conjunto de **características** de un producto que **satisfacen las necesidades de los clientes** y, en consecuencia, hacen satisfactorio el producto
 - (b) Calidad consiste en **no tener deficiencias** en el producto o en el proceso

Calidad e ingeniería de software

- Concepto moderno de la calidad
 - La calidad no la define ni el productor ni las normas, **la define el cliente.**
 - La calidad es relativa y **surge de la comparación con los mejores.**
 - La calidad que perdura es la que **se logra gracias a valores, organización y sistemas que la propician** y no es el producto de circunstancias afortunadas o de esfuerzos aislados

Calidad e ingeniería de software

- Componentes de la calidad de software
 - Nivel de satisfacción de usuarios
 - Valor que agrega el producto
 - Atributos de calidad del producto
 - Ausencia de defectos
 - Calidad del proceso
- Objetivos de calidad de la organización productora de software
 - Tiempo para salir al mercado
 - Costo
 - Ciclo de vida estimada del producto
 - Mercado objetivo

Calidad e ingeniería de software

- Calidad e ingeniería de software
 - La ingeniería de software, como disciplina tecnológica y gerencial, busca aplicar ciertas prácticas a la producción de software, con el objetivo de lograr la calidad.

Calidad e ingeniería de software

- La IS es tecnología multicapa:
 - Capa 1: enfoque de calidad.
 - Capa 2: procesos.
 - Capa 3: métodos.
 - Capa 4: herramientas.

Calidad e ingeniería de software

- Enfoque de calidad.
 - Es la forma de gestionar la calidad para mejorar los procesos, ej: ISO, CMM, etc.
- Procesos.
 - Delimitan marco de trabajo.
 - Definen pasos a dar.
 - Consideran las áreas clave para definirlo, ejecutarlo y mejorarlo.
- Métodos.
 - Indican como construir técnicamente el sw.
 - Existen distintos métodos para cada actividad.
 - Diseño, programación, gestión, etc.
- Herramientas.
 - Automatizan los métodos.
 - CASEs, gestores, workflows, etc.

Proceso y ciclo de vida

Proceso y ciclo de vida

- Proceso
 - Una serie de pasos que involucran actividades, restricciones y recursos con el fin de producir un resultado esperado de algún tipo. [Pfleger]
- Proceso de Software
 - Define las **actividades, prácticas, técnicas, roles, herramientas** necesarias para transformar los requerimientos del usuario en un producto de software. [Jacobson]

Proceso y ciclo de vida

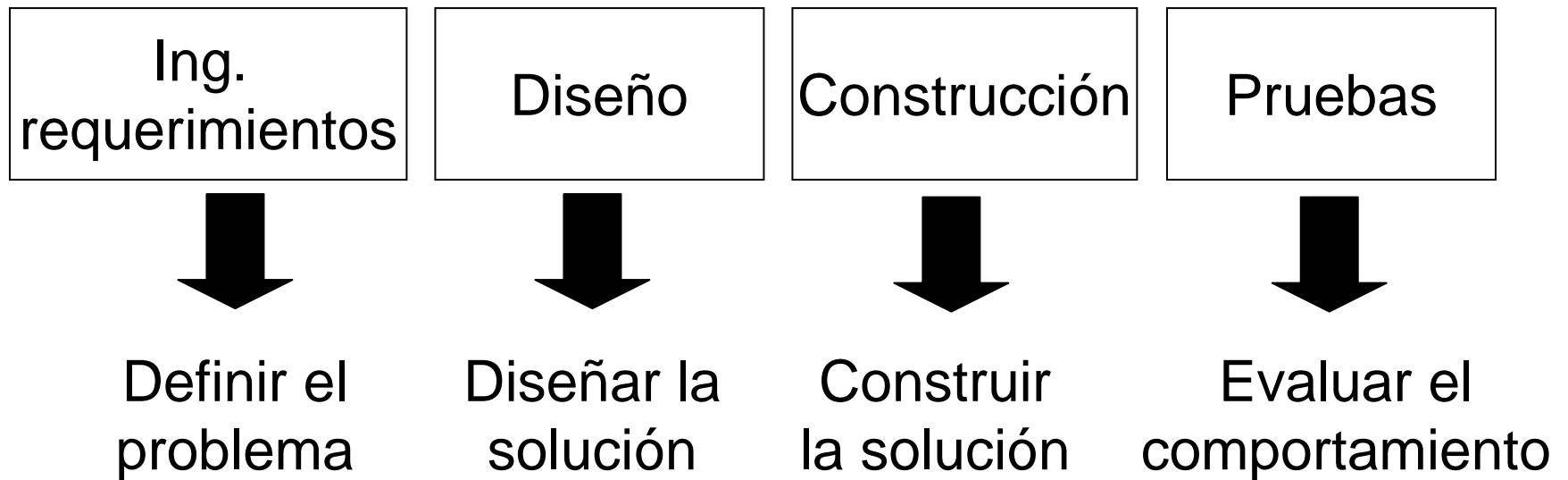
- Metodología
 - Una serie de métodos o técnicas relacionadas.)
 - Es el estudio de los métodos.
- Método
 - Un procedimiento sistemático. [Cockburn]
 - Método o técnica es un procedimiento formal para producir algún resultado. [Pfleger]

Proceso y ciclo de vida

- Proceso de software
 - En general las actividades del proceso de software se dividen en dos grandes grupos:
 - Procesos de ingeniería
 - Comprenden actividades inherentes a la construcción del producto.
 - Procesos de apoyo
 - Comprenden actividades inherentes a la gestión de los procesos de construcción del producto.

Proceso y ciclo de vida

Procesos o actividades típicas de ingeniería



Proceso y ciclo de vida

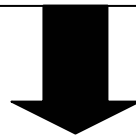
Procesos o actividades típicas de apoyo

Gestión de proyectos



Planificar y controlar el proyecto

Aseguramiento de calidad



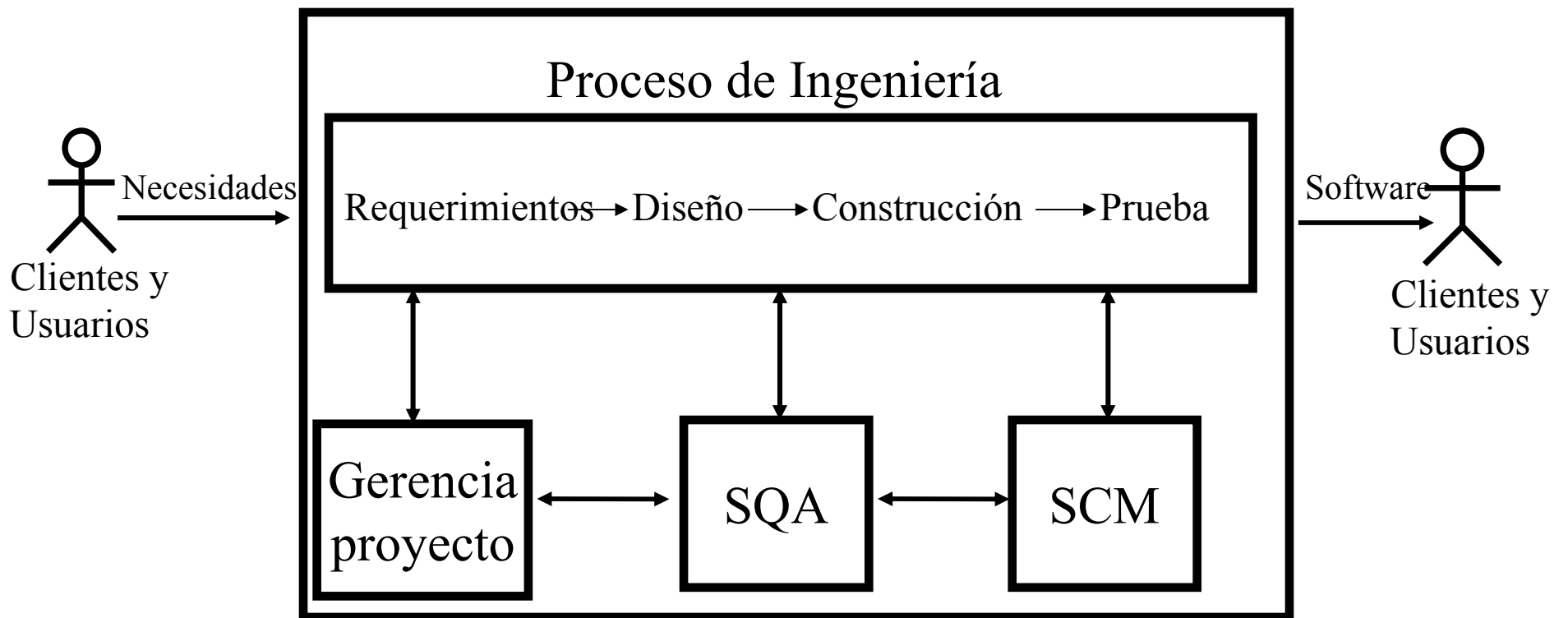
Asegurar la calidad del producto y del proceso

Gestión de la configuración



Identificar y controlar los productos del desarrollo

Proceso y ciclo de vida



Proceso y ciclo de vida

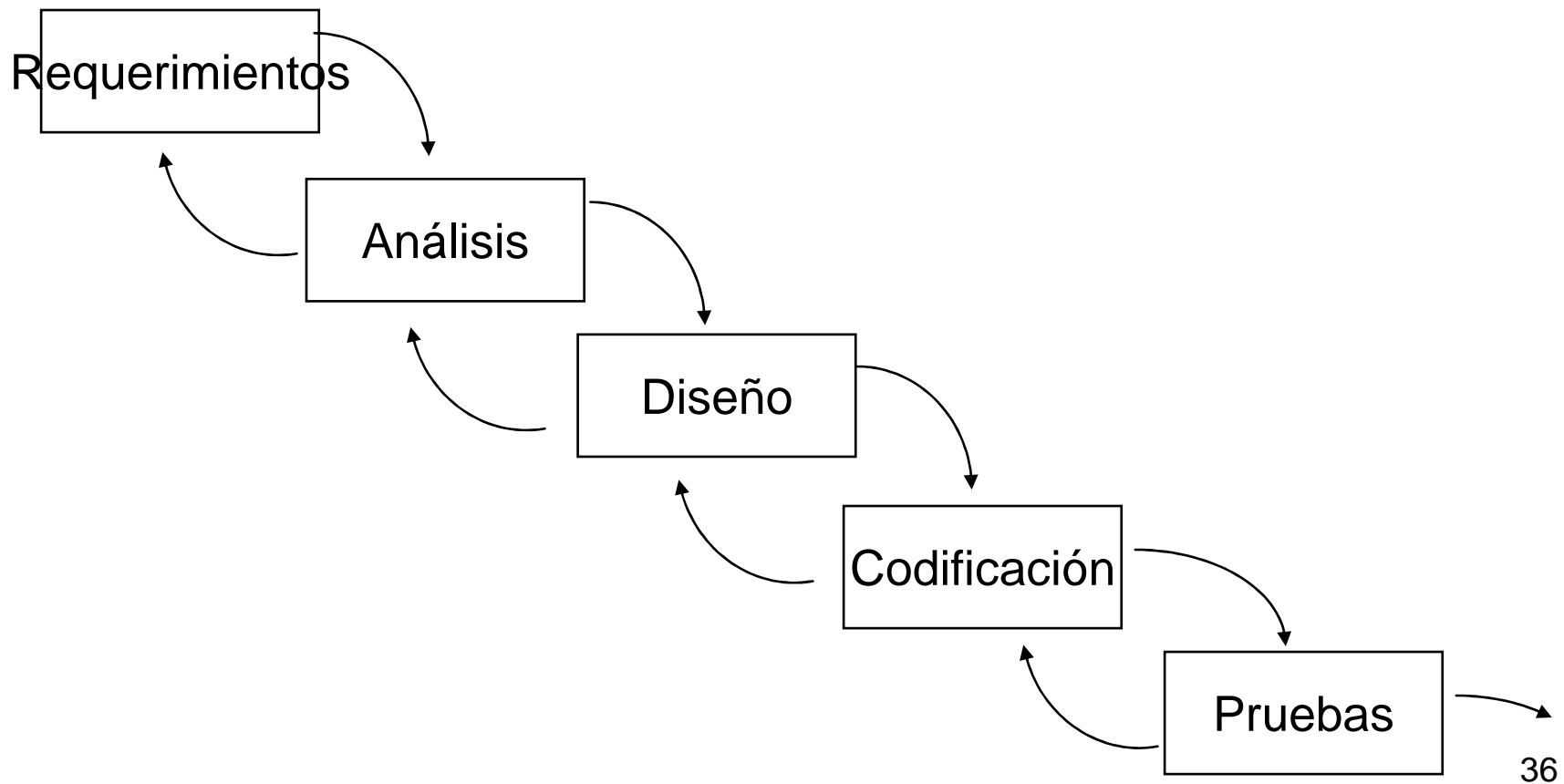
- Un proceso debe definir QUE hacer, QUIEN debe hacerlo, CUANDO y COMO hacerlo para alcanzar una meta.
- Componentes de un proceso
 - Marco común de trabajo, una filosofía de calidad.
 - Tareas (análisis, diseño, prueba, etc.) y actividades de apoyo (mediciones, revisiones, versionado, etc.)
 - Roles y responsabilidades
 - Métodos técnicos
 - Ciclo de vida del desarrollo

Proceso y ciclo de vida

- Ciclo de vida
 - Es un modelo de referencia de alto nivel de las actividades necesarias para el desarrollo de software.
 - Un modelo de ciclo de vida:
 - Describe las principales fases y actividades del desarrollo.
 - Define el orden de las fases.
 - Sirve para “saber donde estamos” durante un proyecto.

Proceso y ciclo de vida

- Ciclo de vida en cascada



Proceso y ciclo de vida

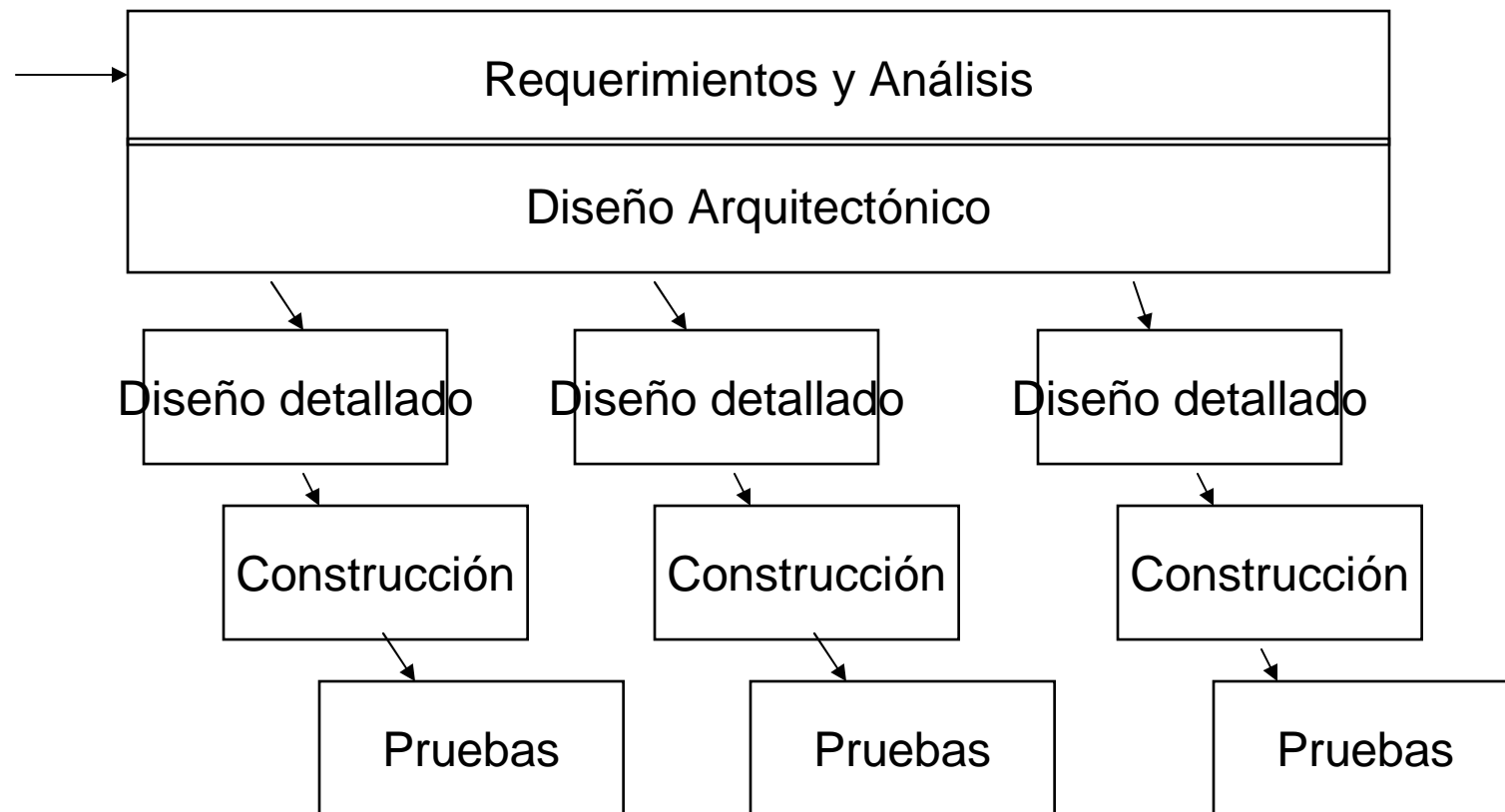
- Ciclo de vida en cascada
 - El proyecto progresa a través de una secuencia ordenada de fases.
 - Revisión al final de cada fase para determinar si se pasa a la siguiente.
 - El software como producto tangible se obtiene al final del ciclo.

Proceso y ciclo de vida

- Ciclo de vida en cascada
 - Buena performance en proyectos con especificaciones estables y bien comprendidas. (Por ejemplo: proyectos de mantenimiento)
 - No es flexible
 - Los Requerimientos detectados tardíamente suelen ser muy costosos
 - Se dificulta la vuelta a una etapa anterior
 - Feedback de defectos tardío. El costo de corrección puede ser enorme dependiendo del tipo de defecto encontrado

Proceso y ciclo de vida

- Ciclo de vida incremental

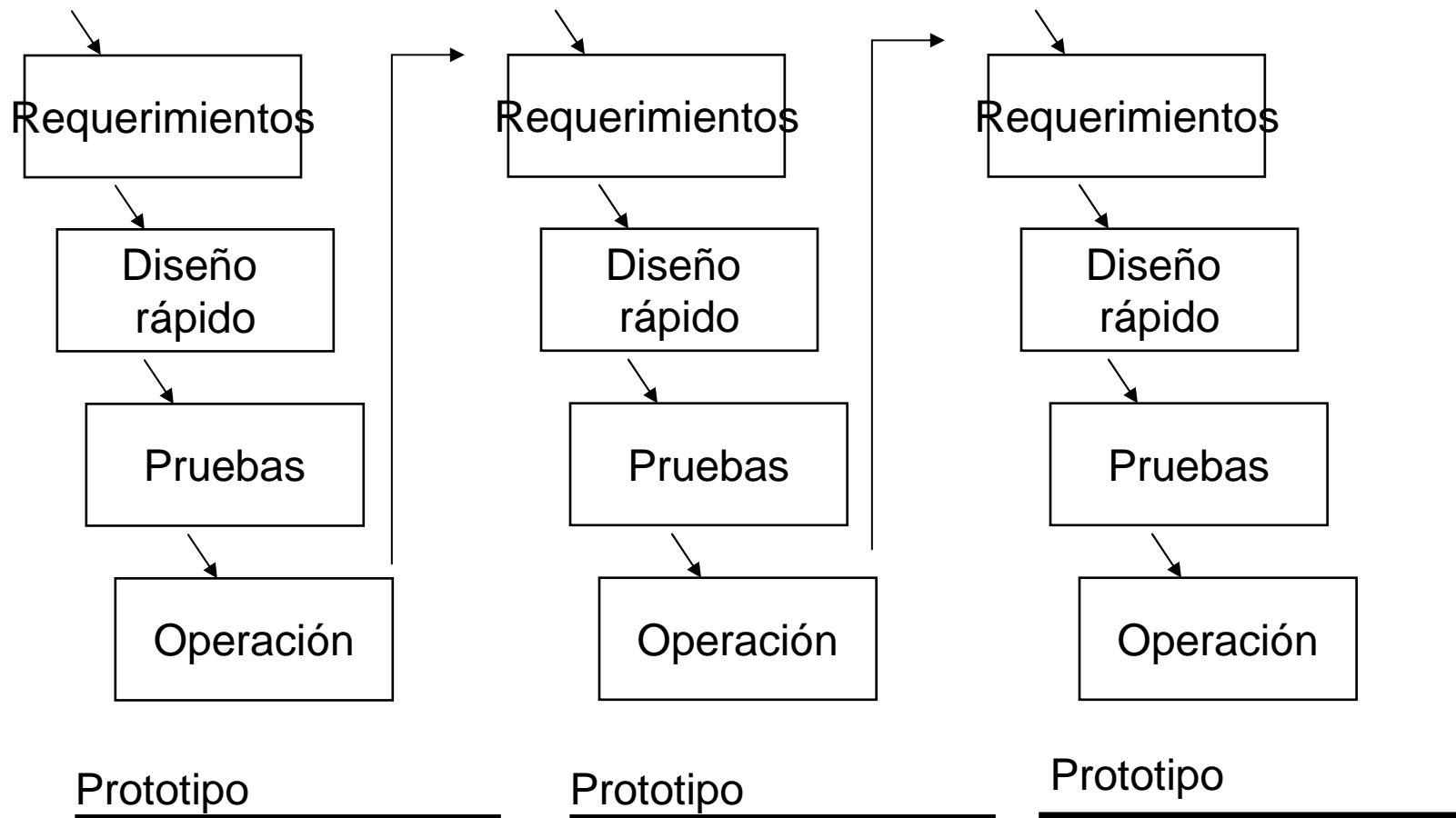


Proceso y ciclo de vida

- Ciclo de vida incremental
 - En la primera vuelta se hace casi todo el relevamiento y análisis.
 - Se itera sobre Diseño, Codificación y Pruebas
 - El cliente va viendo el avance.

Proceso y ciclo de vida

- Ciclo de vida evolutivo



Proceso y ciclo de vida

- Ciclo de vida evolutivo
 - Las iteraciones incluyen análisis.
 - Útil cuando
 - Requerimientos poco definidos
 - El dominio de la aplicación es poco conocido para el desarrollador y para el cliente
 - Produce resultados visibles en forma temprana
 - Puede ser difícil de determinar cuando el producto es aceptable

Proceso y ciclo de vida

- Beneficios de un proceso y ciclo de vida para el software:
 - Estandarizar
 - Entrenar y comunicar
 - Gestionar y realizar seguimiento de proyectos
 - Mejorar con la experiencia de cada proyecto
- No existe un proceso o ciclo de vida estándar para el desarrollo de software

Áreas de la ingeniería de software

Áreas de la ingeniería de software

- Ingeniería de requerimientos de software
 - Definición o identificación de los servicios o capacidades que debe proveer el software para satisfacer las necesidades de los usuarios.
 - Definición del problema a resolver.

Áreas de la ingeniería de software

- Diseño de software
 - Diseño arquitectónico
 - Determinar la estructura general del sistema.
 - Asignar los requerimientos del software a los subsistemas.
 - Definir interfaces de los subsistemas.
 - Etc.

Áreas de la ingeniería de software

- Diseño de software
 - Diseño detallado
 - Descomponer el sistema en clases o módulos.
 - Seleccionar mecanismos y algoritmos.
 - Asignar responsabilidades a módulos.
 - Etc.
- Construcción del software
 - Codificar, validar y probar unitariamente el software.

Áreas de la ingeniería de software

- Pruebas del software
 - Integración de los distintos componentes del software.
 - Verificación dinámica del software contra la especificación del comportamiento esperado.
 - Existen distintos niveles de prueba y distintos objetivos de las pruebas

Áreas de la ingeniería de software

- Mantenimiento del software
 - Modificar el software una vez entregado para corregir fallas o incorporar a cambios.

Áreas de la ingeniería de software

- Gestión de proyectos de software.
 - Planificación y seguimiento de las actividades necesarias para la construcción y mantenimiento del software.
- Gestión de la configuración del software
 - Identificación y control del software y sus componentes durante su construcción, mantenimiento y uso.

Áreas de la ingeniería de software

- Calidad del software
 - Asegurar la calidad del producto y de los procesos durante el ciclo de vida del proyecto.
- Procesos de ingeniería del software
 - Definición, implantación y mejora del proceso de software.

Áreas de la ingeniería de software

- Herramientas y métodos.
 - Ambientes de desarrollo para software y métodos para la realización de las distintas actividades del desarrollo del software.

Bibliografía

- Capítulos 1 y 2, Ingeniería del Software, 4ta. Edición. R. S. Pressman, 1998
- Capítulos 1 y 2, Software Engineering. Shari Pfleeger, 2001.
- Software Engineering Body of knowledge. IEEE, 2003.
- Software Engineering 2 ed., vol. 1 y 2. R. Thayer, I. Sommerville, 2002.
- Object-Oriented Software Engineering. I. Jacobson, 1993.
- Agile Software Development. A. Cockburn, 2002.
- Material del curso ISO 9000 en Informática. Amalia Alvarez, 2001.