

Modelando el Ciclo de Vida de Desarrollo del Software

Módulo3: Estructura y las 6 Mejores Prácticas



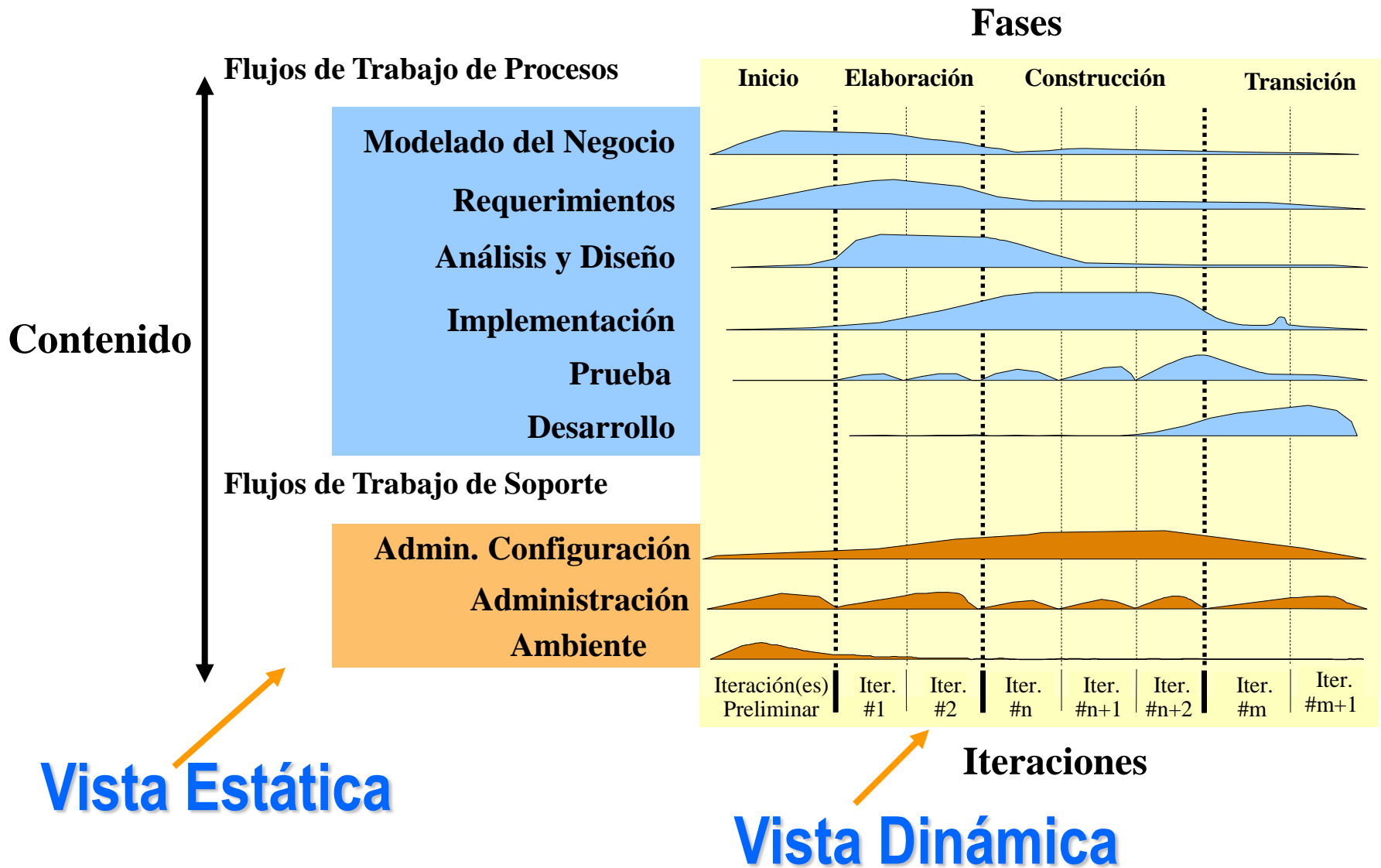
Tópicos

- Estructura del RUP
 - Vista Dinámica
 - Vista Estática
- Las 6 Mejores Prácticas

Estructura del RUP

- El proceso puede describirse en dos dimensiones, o a lo largo de dos ejes:
 - El **eje horizontal** representa *tiempo* y muestra el **aspecto dinámico** del proceso, expresado en términos de ***ciclos, fases, iteraciones, e hitos.***
 - El **eje vertical** representa el **aspecto estático** del proceso; expresado en términos de **actividades, artefactos, trabajadores y flujos de trabajo.**

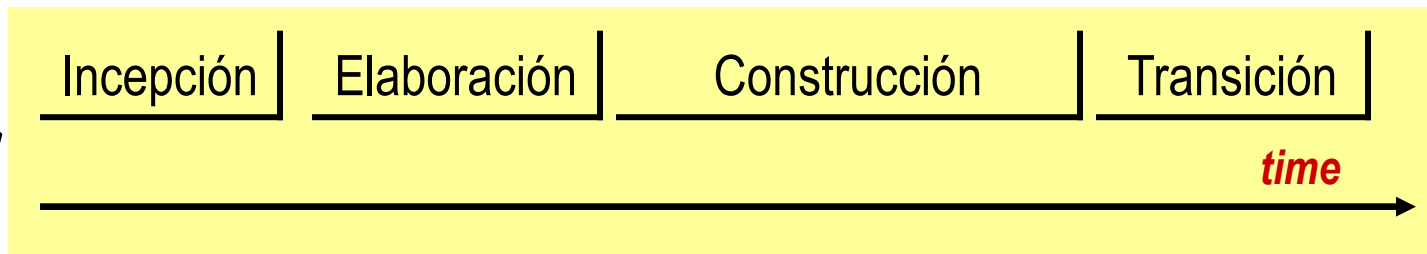
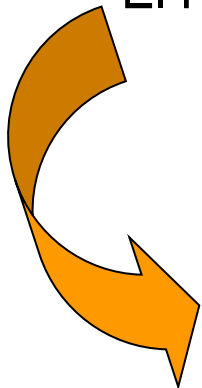
Estructura del RUP



Vista Dinámica

Ciclos y Fases

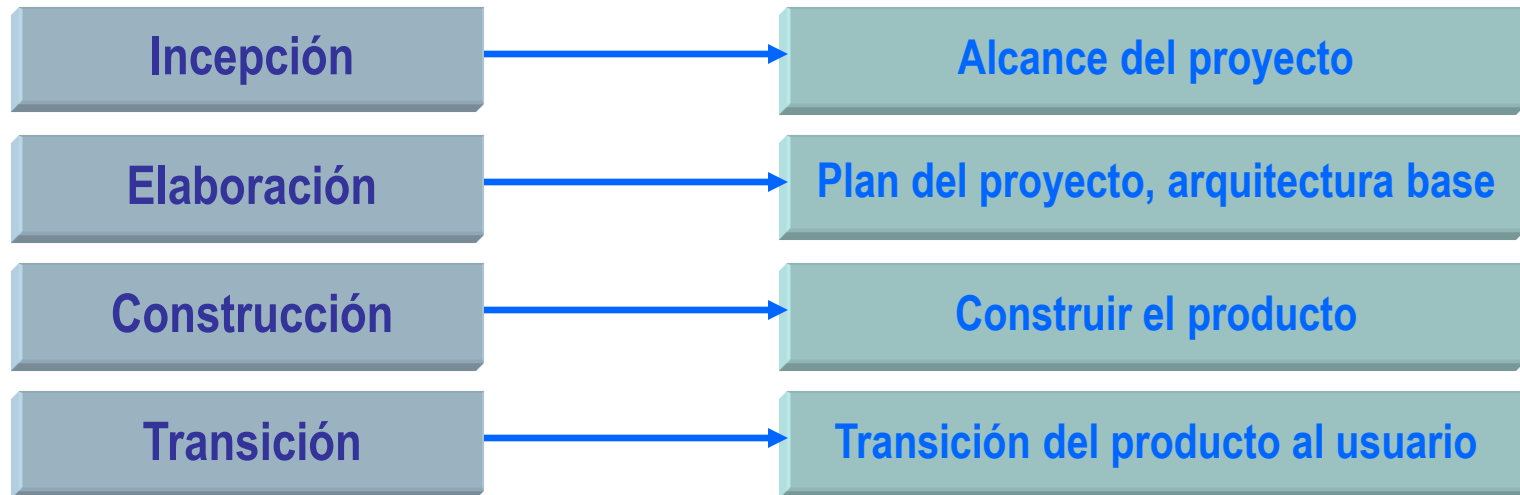
- El ciclo de vida del software está partido en ciclos, y cada ciclo **concluye con una generación del producto para los clientes.**
- El RUP divide cada ciclo de desarrollo en **cuatro fases** consecutivas.



Vista Dinámica

Fases en RUP

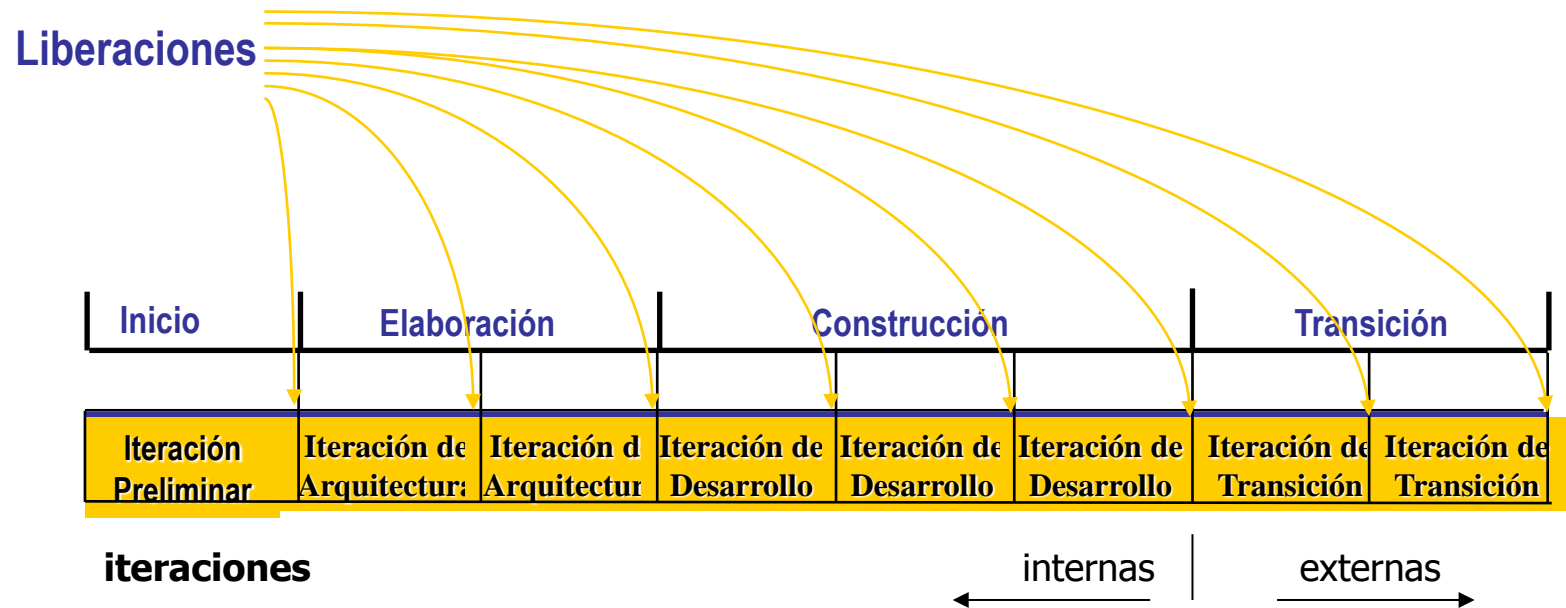
- Cada fase tiene propósitos específicos



Vista Dinámica

Iteraciones

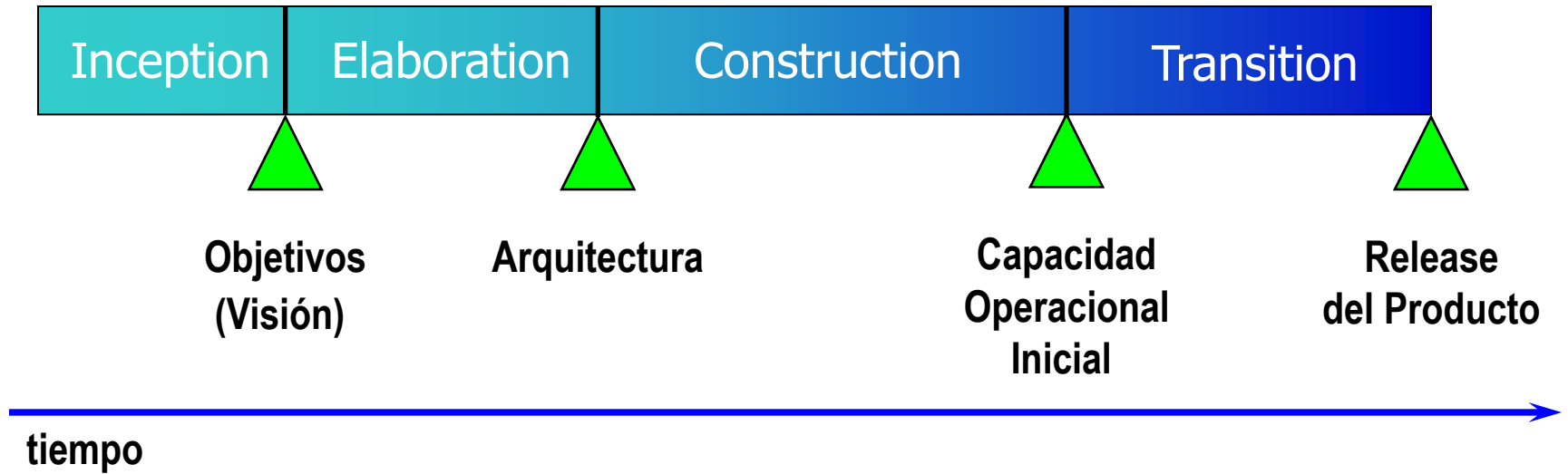
- Cada fase en RUP puede descomponerse en iteraciones.
- El número de iteraciones en cada fase es variable.



Una *iteración* es un ciclo de desarrollo completo dando como resultado una entrega de producto ejecutable (interna o externa).

Vista Dinámica

Fases e Hitos (Milestones)



Cada fase se concluye con un hito bien definido, un punto en el tiempo en el cual se deben tomar ciertas decisiones críticas y alcanzar las metas clave antes de pasar a la siguiente fase.

Vista Estática

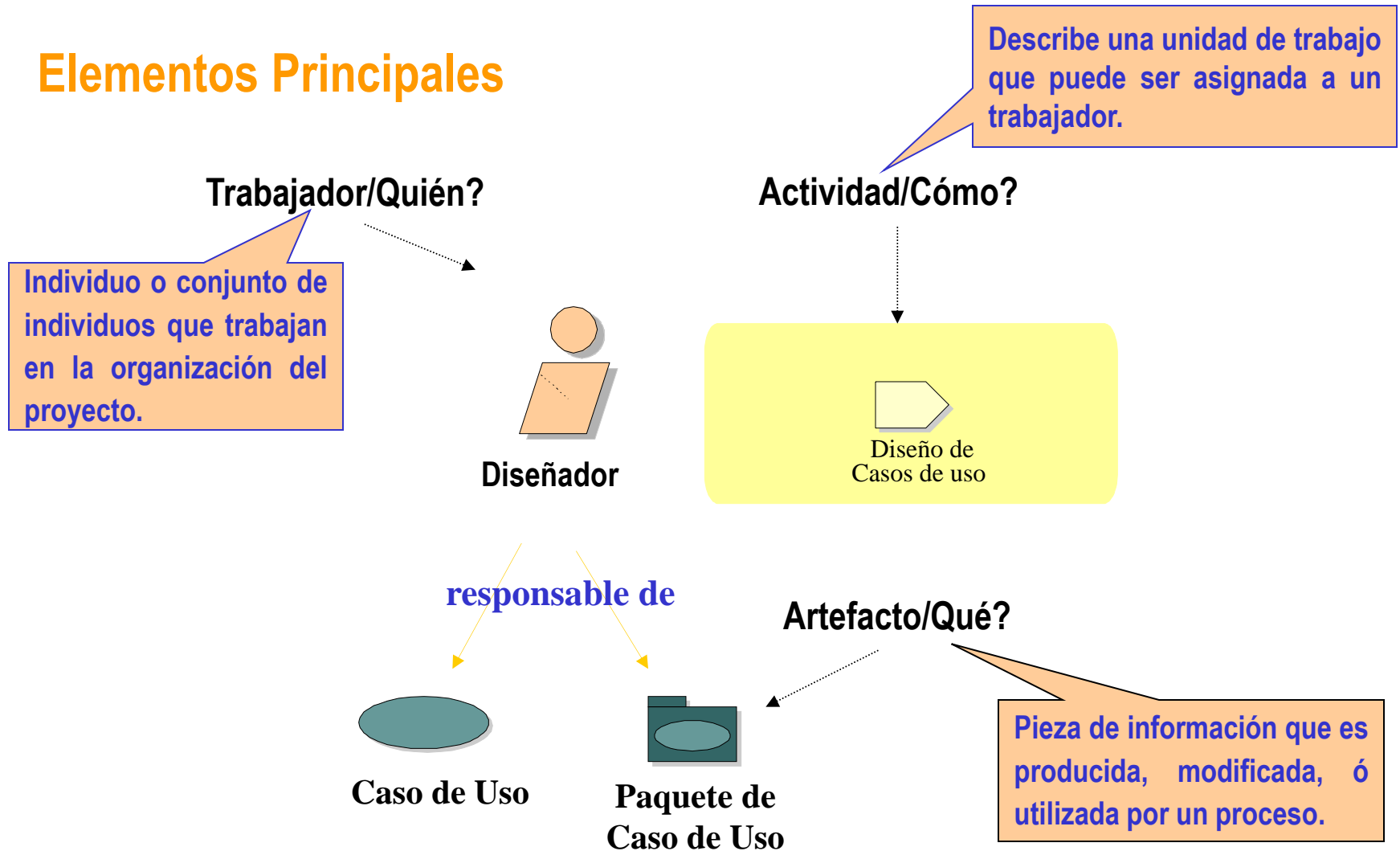
Elementos Principales

- El RUP es representado con el uso de los cuatro principales elementos de modelado.

Trabajadores	-	el Quién
Actividades	-	el Cómo
Artefactos	-	el Qué
Flujos de Trabajo	-	el Cuándo

Vista Estática

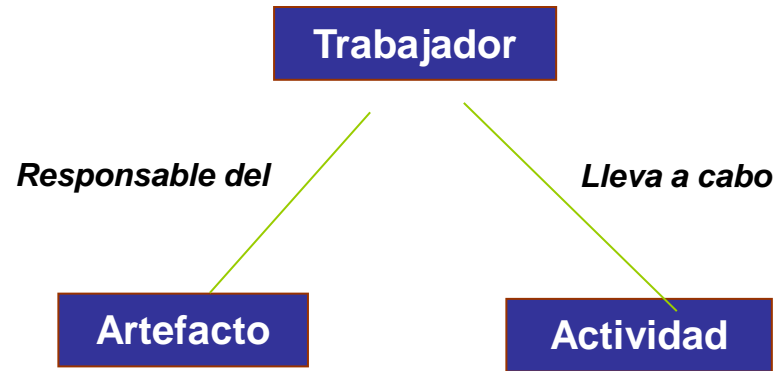
Elementos Principales



Vista Estática

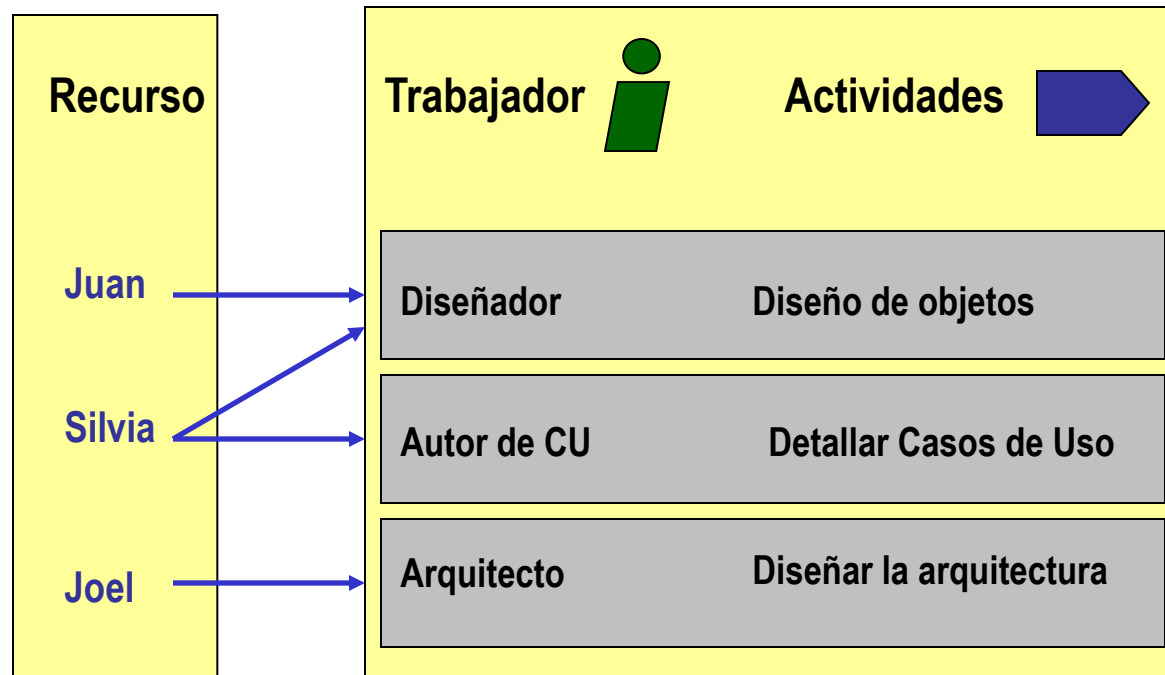
Trabajador o Worker

- El término Worker define el comportamiento y responsabilidades de un individuo o un grupo de individuos trabajando juntos como equipo.



Vista Estática

Trabajador o Worker



Vista Estática

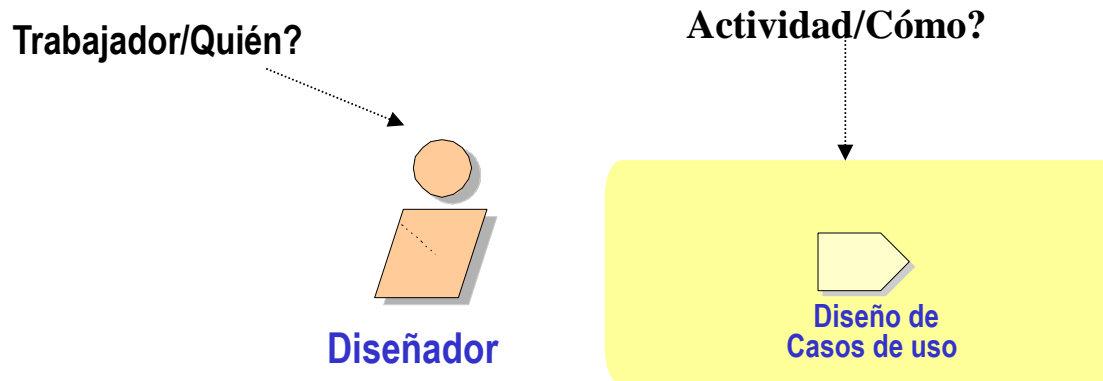
Trabajador o Worker

- **Analyst workers**
 - Business-Process Analyst
 - Business Designer
 - Requirements Reviewer
 - System Analyst
 - Use-Case Specifier
 - User-Interface Designer
- **Developer workers**
 - Architect
 - Architecture Reviewer
 - Code Reviewer
 - Database Designer
 - Designer
 - Implementer
 - Integrator
- **Testing professional workers**
 - Test Designer
 - Tester
- **Manager workers**
 - Change Control Manager
 - Configuration Manager
 - Deployment Manager
 - Process Engineer
 - Project Manager
- **Other workers**
 - Course Developer
 - Graphic Artist
 - Stakeholder
 - System Administrator
 - Technical Writer
 - Tool Specialist

Vista Estática

Actividades

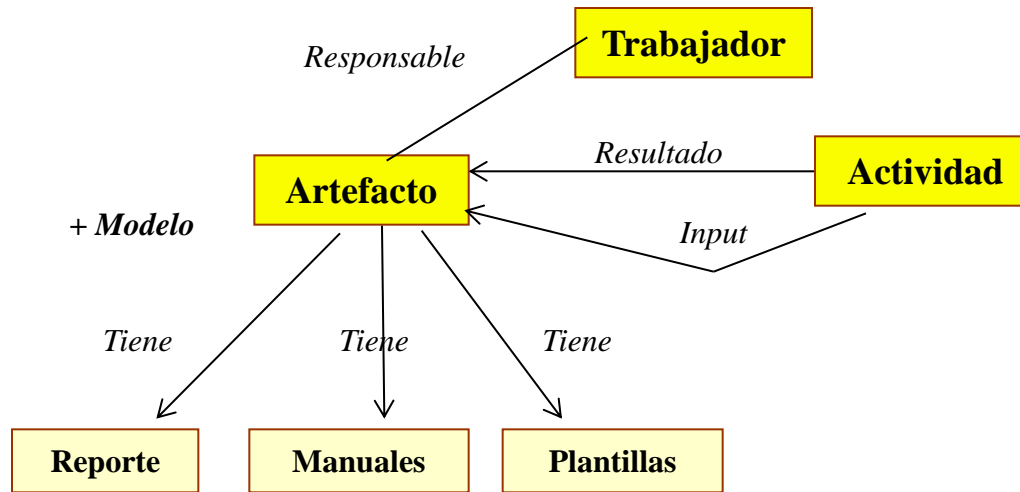
- Es una unidad de trabajo de un Trabajador que
 - tiene un propósito claro.
 - es expresada normalmente en términos de creación o actualización de artefactos.
 - es asignada a un trabajador específico.



Vista Estática

Artefactos

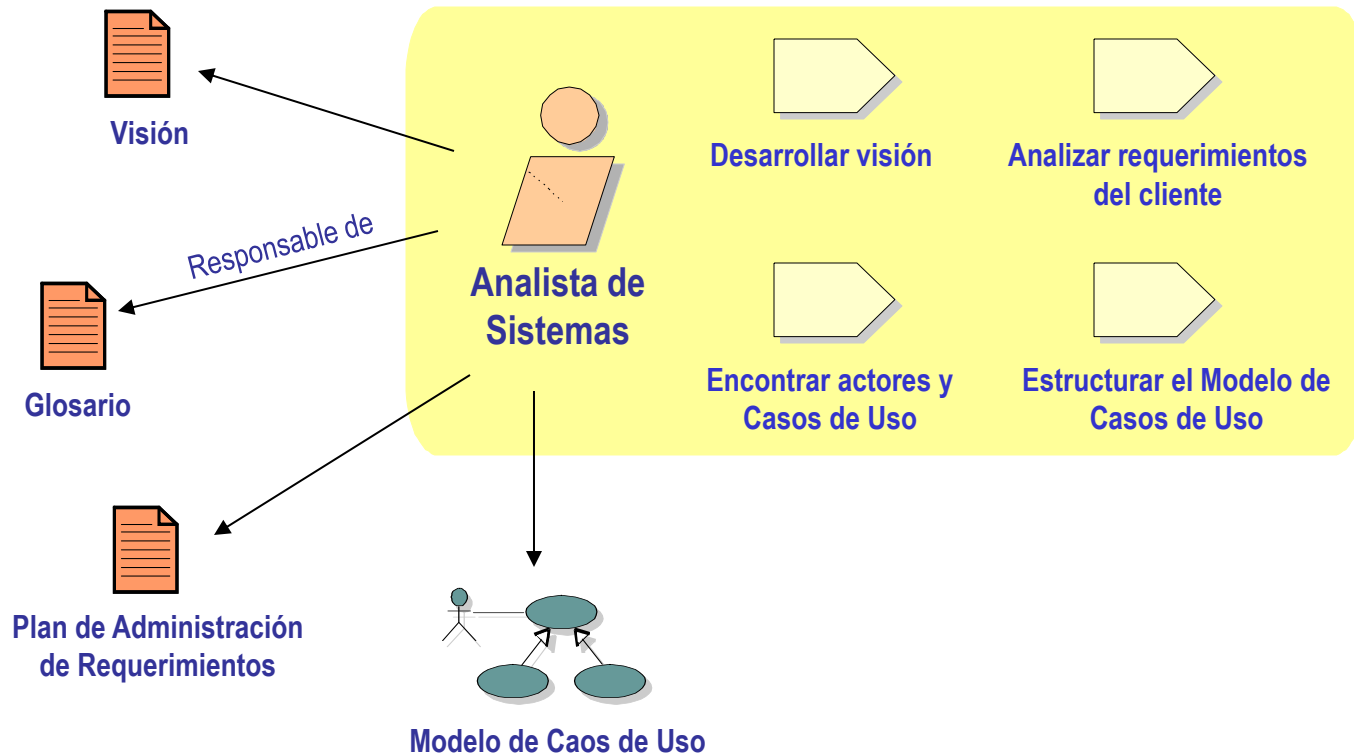
- Es una pieza de información que es producida, modificada o usada por un proceso.



Vista Estática

Trabajador, Actividades, Artefactos

- Ejemplo: Trabajador Analista de Sistemas



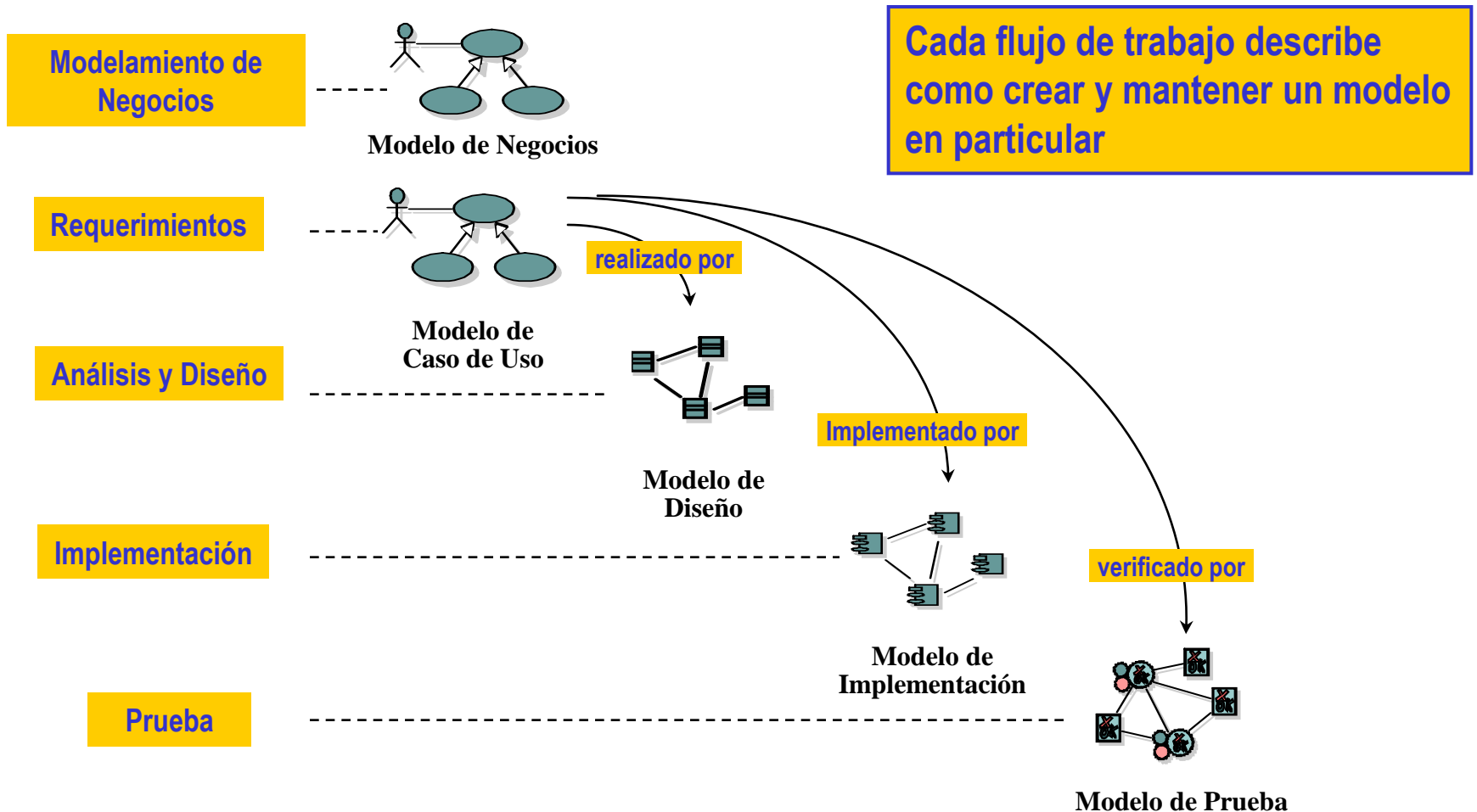
Vista Estática

Flujos de Trabajo o Workflows

- Con la enumeración de trabajadores, actividades y artefactos no se define un proceso.
- Necesitamos contar con una secuencia de actividades (workflows) realizadas por los diferentes trabajadores, así como la relación entre los mismos.

Vista Estática

Flujos de Trabajo o Workflows



Vista Estática

Flujos de Trabajo o Workflows

- **Workflows Primarios**
 - Modelamiento de Negocios
 - Requerimientos
 - Análisis y Diseño
 - Implementación
 - Pruebas
 - Despliegue
- **Workflows de Apoyo**
 - Entorno
 - Gestión del Proyecto
 - Gestión de Configuración y Cambios

Vista Estática

Workflow de Modelamiento de Negocios

- Describe los procesos de una organización.
- Usado cuando se identifican requerimientos.
- El resultado es un modelo del negocio.

Workflows detallados

- Identificar Procesos de Negocio.
- Refinar las Definiciones del Proceso de Negocio.
- Identificar Roles y Responsabilidades.
- Refinar Roles y Responsabilidades.

Vista Estática

Workflow de Requerimientos

- Describe lo que el sistema debe hacer.
- El resultado es un modelo de casos de uso.

Workflows detallados

- Analizar el Problema.
- Entender las Necesidades de los Stakeholders.
- Definir el Sistema.
- Administrar el Alcance del Sistema.
- Refinar la Definición del Sistema.
- Administrar Cambios de Requerimientos.

Vista Estática

Workflow de Análisis y Diseño

- Muestran como se verá el sistema en la implementación.
- El modelo de casos de uso es la base para el diseño.

Workflows detallados

- Diseño de la Arquitectura.

Vista Estática

Workflow de Implementación

- El modelo de implementación describe el sistema ejecutable.
- El modelo de diseño se usa como base para la implementación.

Workflows detallados

- Estructurar el Modelo de Implementación.
- Planear la Integración dentro de una Iteración.
- Implementar Clases dentro de una Iteración.
- Integrar cada Subsistema dentro de una Iteración.
- Integrar el Sistema dentro de una Iteración.

Vista Estática

Workflow de Pruebas

- El modelo de Pruebas describe como se realizará la prueba del sistema.
- Verifica la interacción entre objetos, la integración de todos los componentes del software, que todos los requerimientos se hallan implementado correctamente.



Vista Estática

Workflow de Despliegue

- Producir exitosamente releases del producto y entregar el software a sus usuarios finales.
- Cubre actividades como:
 - Producir releases externas del software.
 - Empaquetar software.
 - Distribuir software.
 - Instalar el software.
 - Suministrar ayuda y asistencia a los usuarios.

Vista Estática

Workflow de Entorno

- Provee a la organización de desarrollo de software el ambiente necesario (procesos y herramientas) para el soporte del equipo de desarrollo.
- Incluye:
 - ✓ Configuración del proceso.
 - ✓ Mejorar el proceso.
 - ✓ Selección y adquisición de herramientas.
 - ✓ Apoyo del desarrollo.
 - ✓ Capacitación.

Vista Estática

Workflow de Gestión del Proyecto

- Provee una estructura para administrar los proyectos de software.
- Provee guías prácticas para planificar, proveer de personal, ejecutar y monitorear proyectos.
- Provee una estructura para administrar riesgos.



Vista Estática

Workflow de Gestión de Configuración y Cambios

- Rastrear y mantener la integridad de recursos de los proyectos como su evolución en presencia de cambios.
- Ayuda a evitar
 - Problemas de actualización simultánea de componentes.
 - Notificación limitada de cambios.
 - Múltiples versiones.
- Administrar desarrollo paralelo, realizado en distintos sitios.

Las 6 Mejores Prácticas del RUP



Las 6 Mejores Prácticas

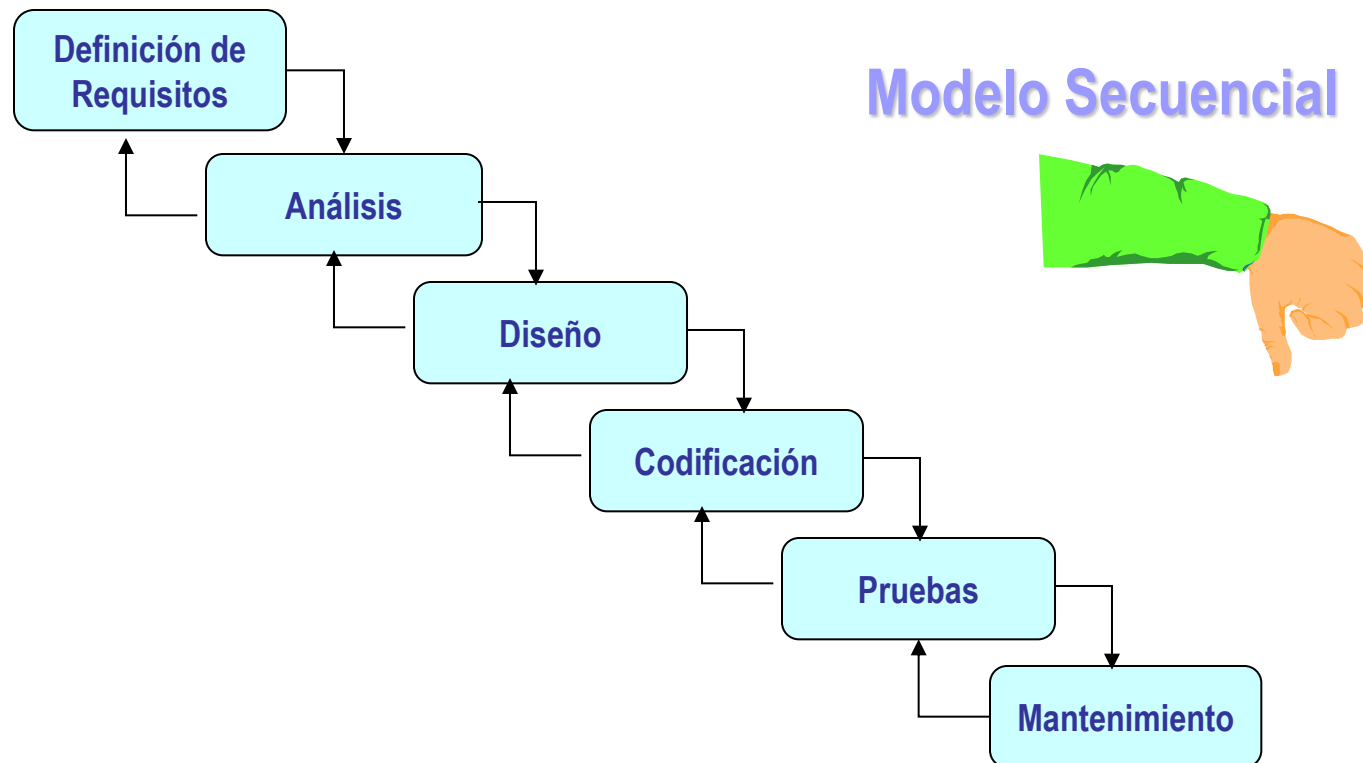
- RUP pretende implementar las mejores prácticas actuales en ingeniería de software:



Las 6 Mejores Prácticas

Desarrollo Iterativo del Software

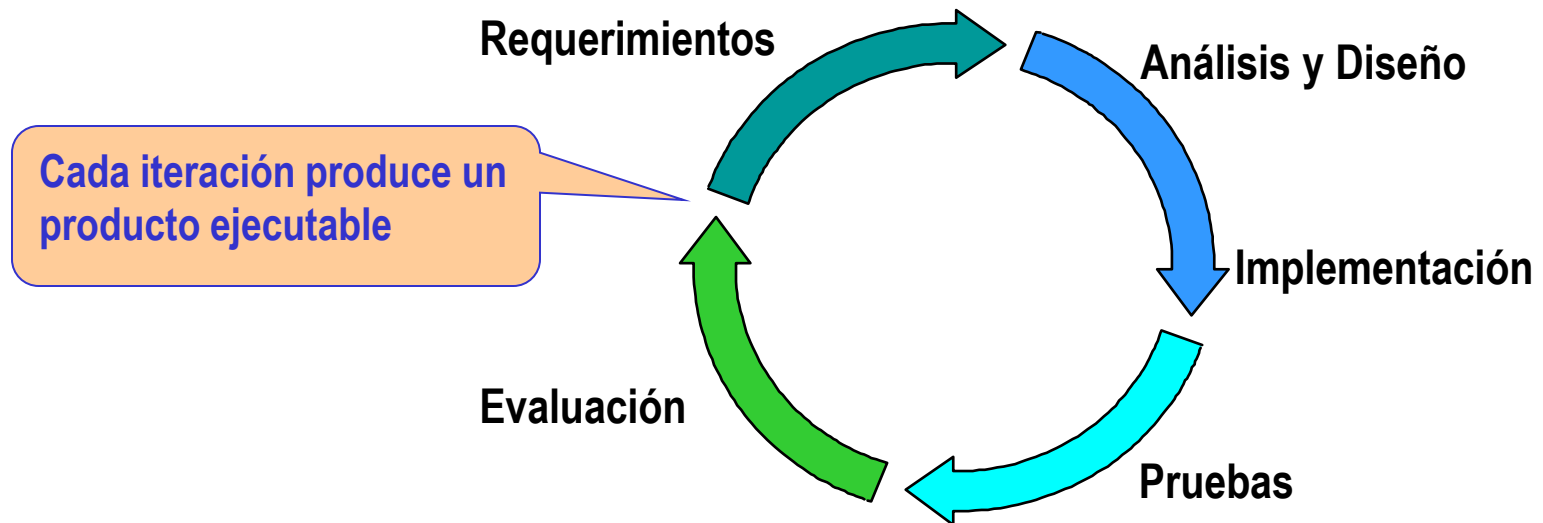
- Dados los sistemas de software sofisticados de la actualidad, no es posible hacer de manera secuencial el Proceso de Desarrollo de SW.



Las 6 Mejores Prácticas

Desarrollo Iterativo del Software

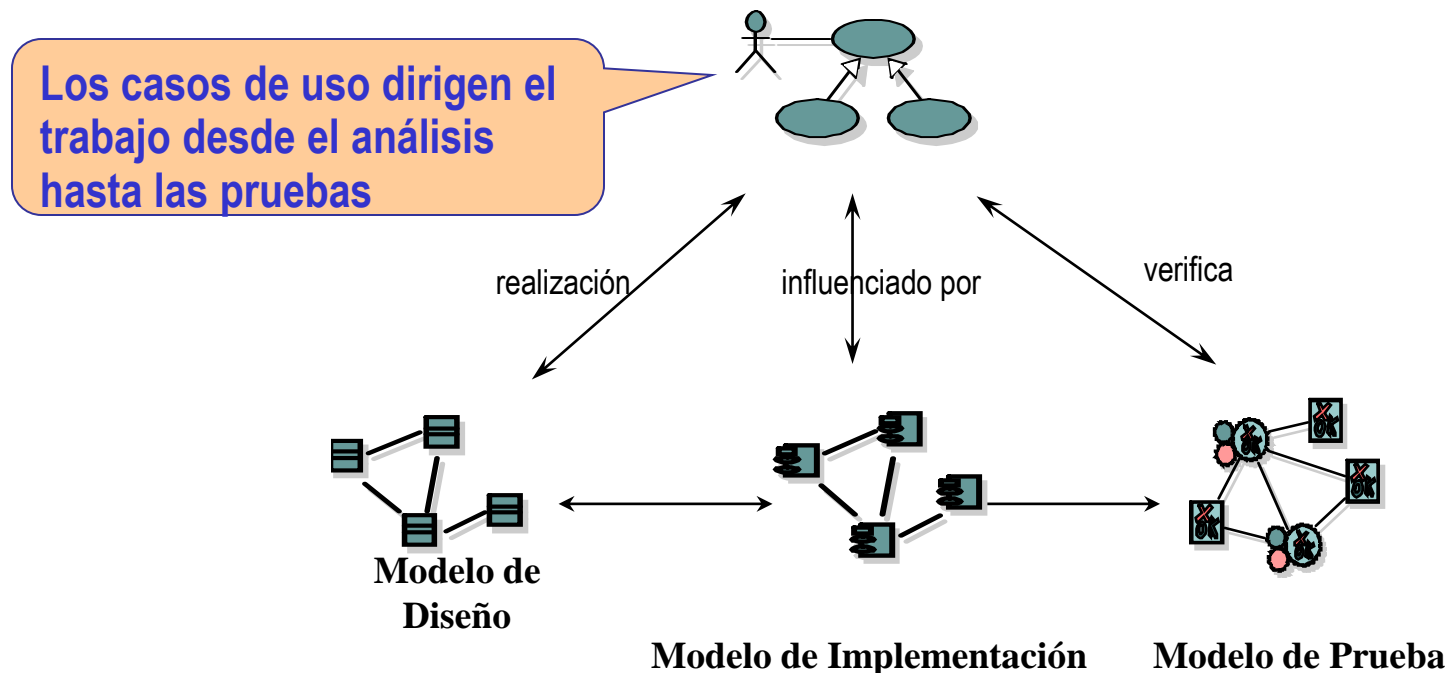
- Un proceso iterativo permite una comprensión creciente de los requerimientos a la vez que se va haciendo crecer el sistema.
- Con esto se logra reducir los riesgos del proyecto y tener un subsistema ejecutable tempranamente.



Las 6 Mejores Prácticas

Administración de Requerimientos

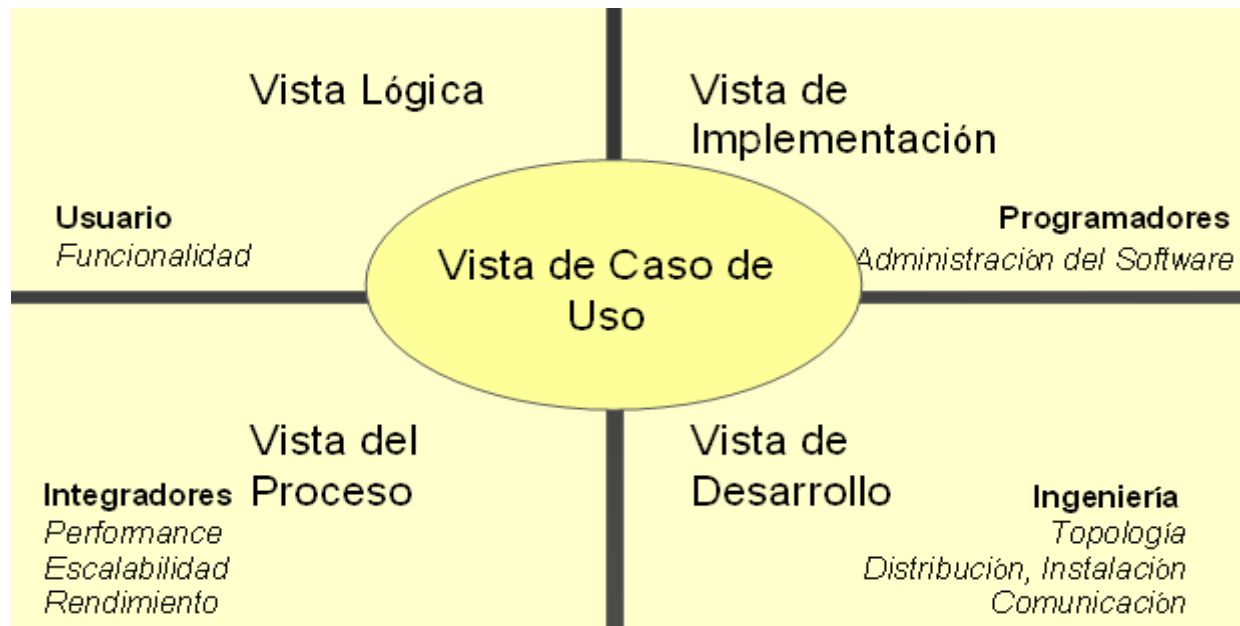
- Involucra obtener, organizar y documentar la funcionalidad y restricciones requeridas a un sistema.
- Los requerimientos son capturados y realizados con Casos de Uso.



Las 6 Mejores Prácticas

Arquitectura Basada en Componentes

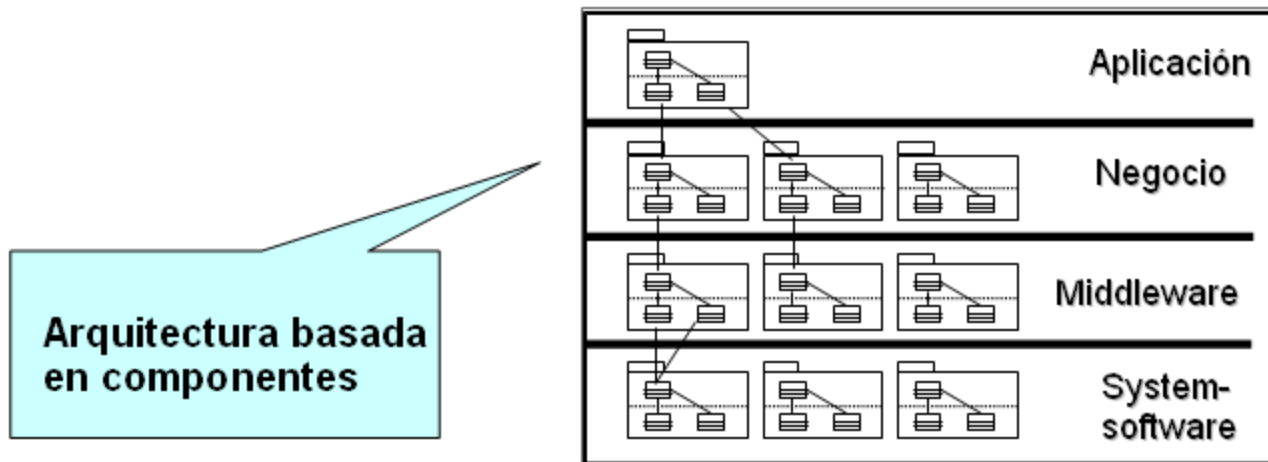
- Se enfoca en el desarrollo de una arquitectura ejecutable robusta. Resistente al cambio mediante el uso de interfaces bien definidas.
 - Intuitivamente comprensible.
 - Es derivada a partir de los casos de uso más importantes.



Las 6 Mejores Prácticas

Arquitectura Basada en Componentes

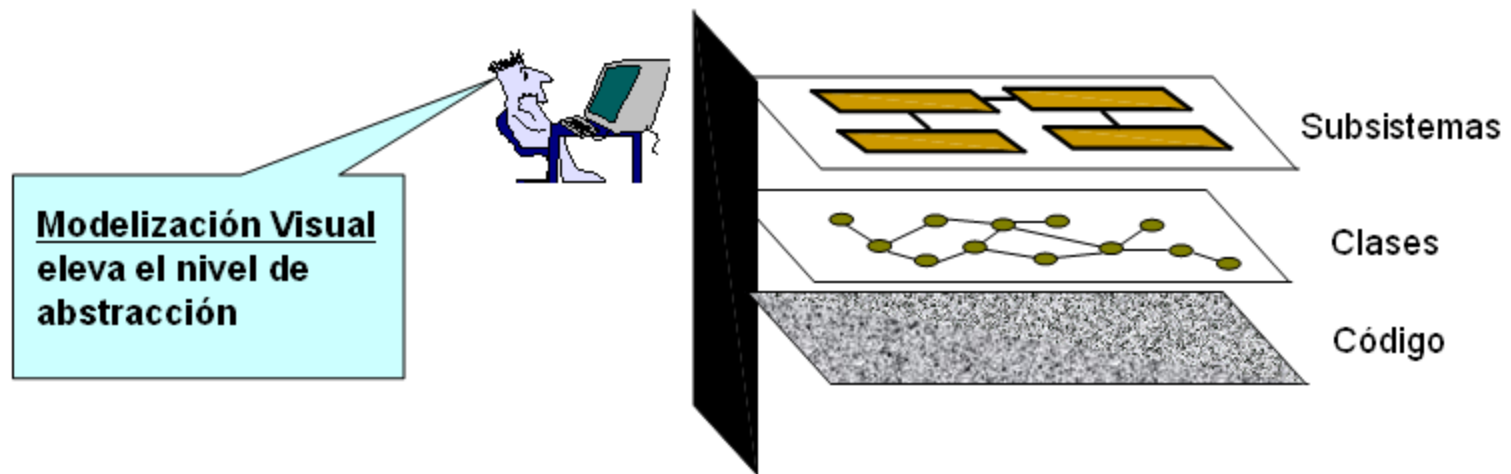
- Definir arquitecturas muy modulares e identificar, aislar, diseñar, desarrollar y probar componentes bien formados.
- Desarrollar componentes para ser reutilizados.



Las 6 Mejores Prácticas

Modelamiento Visual del Software

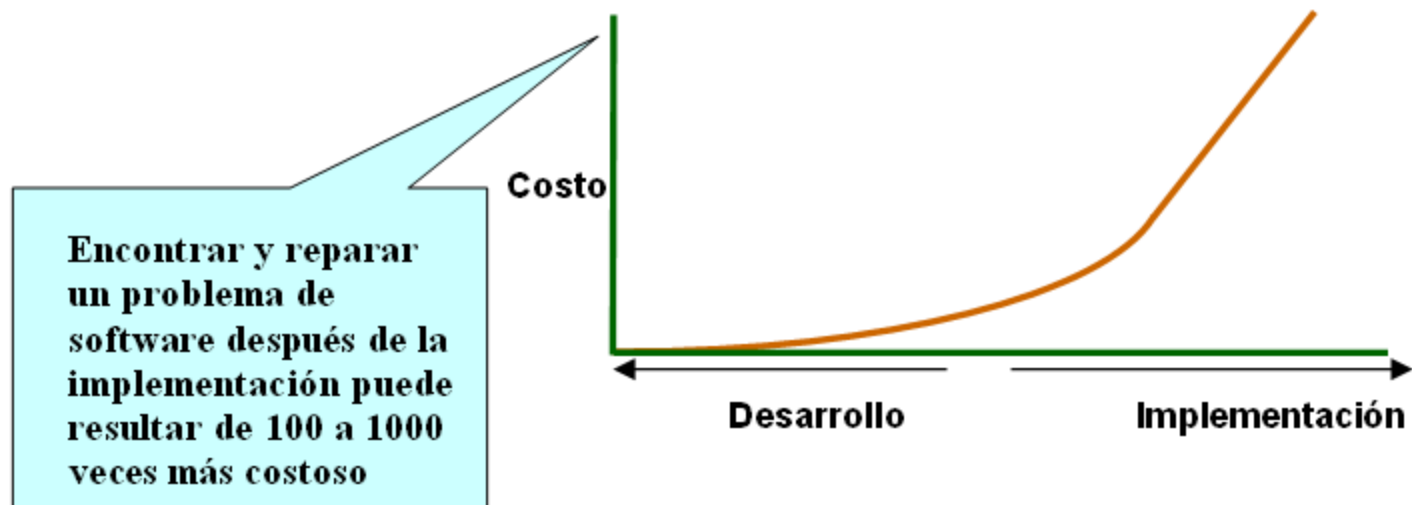
- Muestra como encajan de forma conjunta los elementos del sistema.
 - Mantiene la consistencia entre el diseño e implementación.
-
- ✓ Diagramas de Casos de Uso
 - ✓ Diagramas de Clases
 - ✓ Diagramas de Estados
 - ✓ Diagramas de Componentes
 - ✓ Diagramas de Despliegue
 -



Las 6 Mejores Prácticas

Verificación de la Calidad del Software

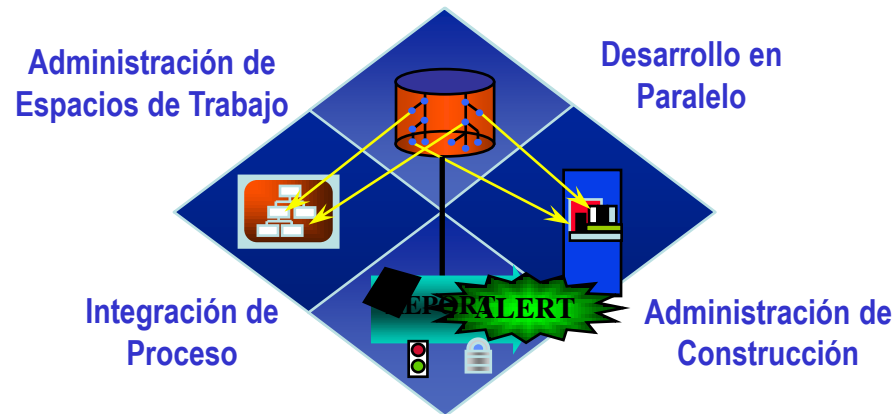
- La actividad fundamental de esta práctica es el testing.
- Verifica la calidad del software con respecto a los requerimientos basados en la confiabilidad, funcionalidad, desempeño de la aplicación y del sistema.



Las 6 Mejores Prácticas

Control de Cambios del Software

- Controlar, registrar y monitorear los cambios para posibilitar el desarrollo iterativo.
- Establece espacios de trabajo seguros para cada desarrollador
 - Provee aislamiento de cambios hechos en otros espacios de trabajo.
 - Controla todos los artefactos de software, modelos, código, documentos, etc.



Resumen

- Vista Dinámica del RUP

Cada una de las fases del RUP tiene propósitos definidos:

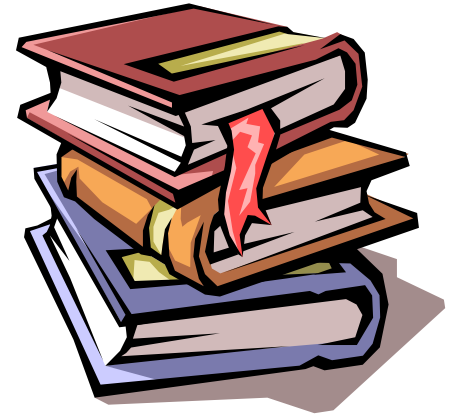
- Incepción ----- Alcance del proyecto.
- Elaboración ----- Arquitectura base.
- Construcción ----- Construir el producto.
- Transición ----- Transición del producto al usuario.

- Vista Estática del RUP

- Trabajadores ----- el Quién.
- Actividades ----- el Cómo.
- Artefactos ----- el Qué.
- Flujos de Trabajo ----- el Cuándo.

Resumen

- RUP contempla las 6 Mejores Prácticas para el desarrollo de software:
 - Desarrollo iterativo.
 - Administración de Requerimientos.
 - Uso de arquitecturas basadas en componentes.
 - Modelamiento Visual.
 - Verificación continua de la calidad.
 - Gestión y control de cambios.



Laboratorio

Laboratorio 3.1

- Identificar trabajadores, artefactos y actividades del caso de estudio.
- Establecer los objetivos por fases del caso de estudio.