
Tema 2: Ingeniería de requisitos

Tema 2.1: Introducción a la Ingeniería de Requisitos

Tema 2.2: Obtención de requisitos

Tema 2.3: Modelo de Casos de Uso

Tema 2.4: Análisis y Especificación de requisitos

Bibliografía: [ARLO05 parte 2 y 3].
[LARM03 capítulos 5-13]
[PRES13 capítulos 5, 6 y 7]
[SOM11 capítulos 4 y 5]

Tema 2.1: Introducción a las ingeniería de requisitos





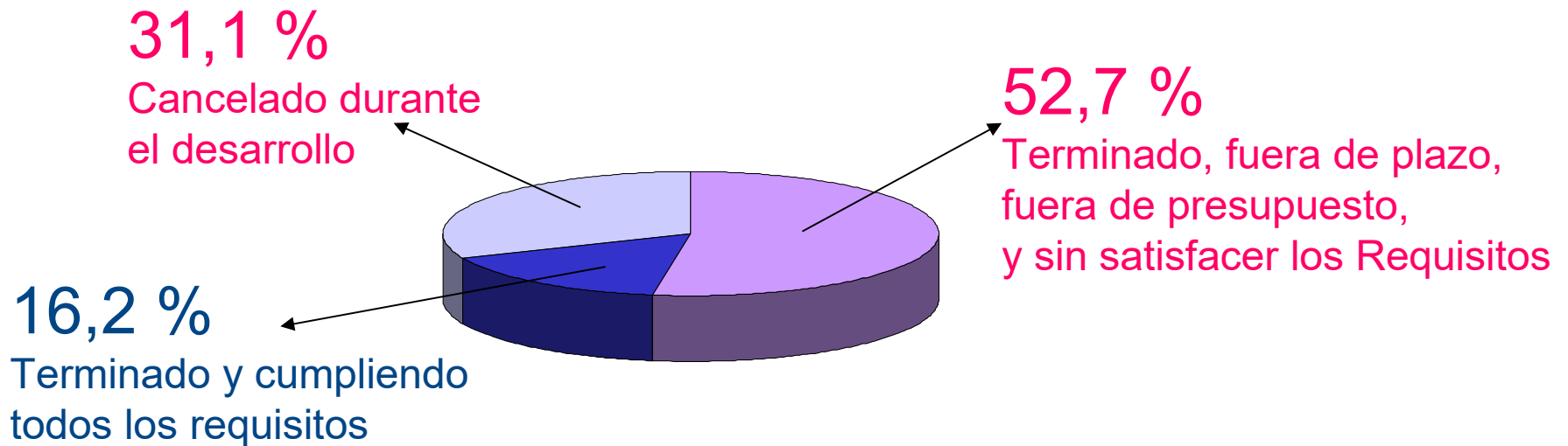
Tema 2.1. Introducción a la Ingeniería de Requisitos

- ✓ Introducción. [PRES13 página 102] y [LARM03 capítulo 5]
- ✓ Concepto de requisito y tipos. [SOMM páginas 83-90]
- ✓ Propiedades de los requisitos. [ARLO05 página 82-83]
- ✓ Tareas de la Ingeniería de requisitos.
[PRES13 páginas 102-106]
- ✓ Roles. [PRES13 página 92]
- ✓ Problemas de la Ingeniería de requisitos.
[LARM03 página 40]



Introducción

En 1995 se realizó el informe CHAOS sobre los **resultados** obtenidos en diversos proyectos software



Factores de fracaso:

- Falta de información por parte de los usuarios.
- Especificación de requisitos incompleta.
- Continuos cambios de los requisitos.
- Pobres habilidades técnicas en la especificación de requisitos.



Introducción: ¿Qué es la IR?

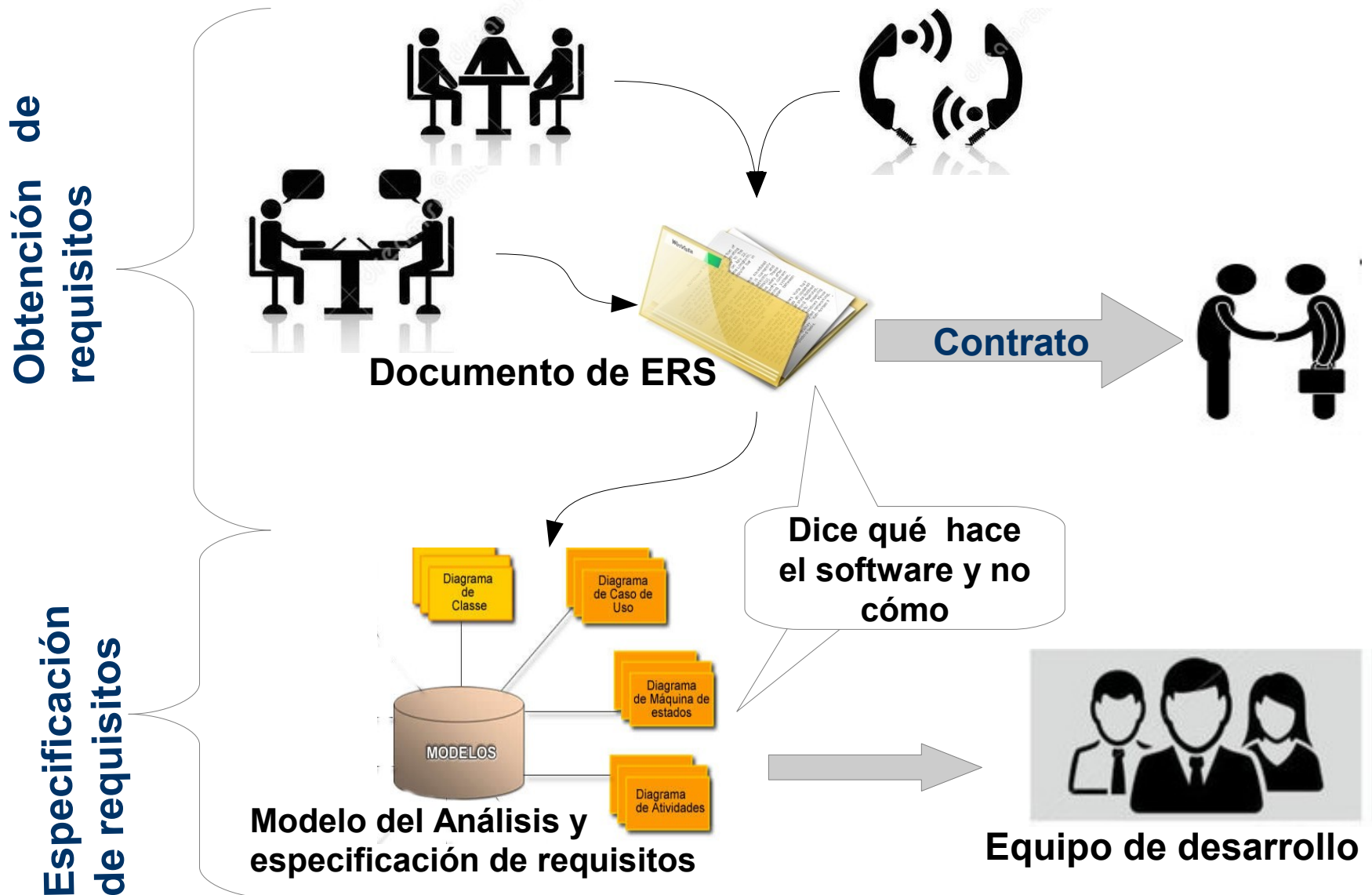
La **Ingeniería de Requisitos (IR)** cubre las tareas y proporciona las técnicas y mecanismos apropiados para:

- Entender y analizar las **necesidades del cliente**.
- Evaluar la **viabilidad** de las necesidades.
- Negociar una **solución razonable**.
- **Especificar la solución** sin ambigüedades. Como resultado se tendrá un documento que describa la solución acordada.
- **Validar y analizar la especificación** reflejada en el documento de especificación de requisitos. Como resultado se obtendrá el modelo del análisis.
- **Administrar** y controlar los **requisitos** a lo largo del proceso de desarrollo.

El proceso de construcción de una “especificación de Requisitos” es un proceso iterativo, en el que partimos de especificaciones iniciales incompletas, poco claras o ambiguas y llegamos a especificaciones finales **completas, claras, documentadas y validadas**.



Introducción: ¿Qué es la IR?



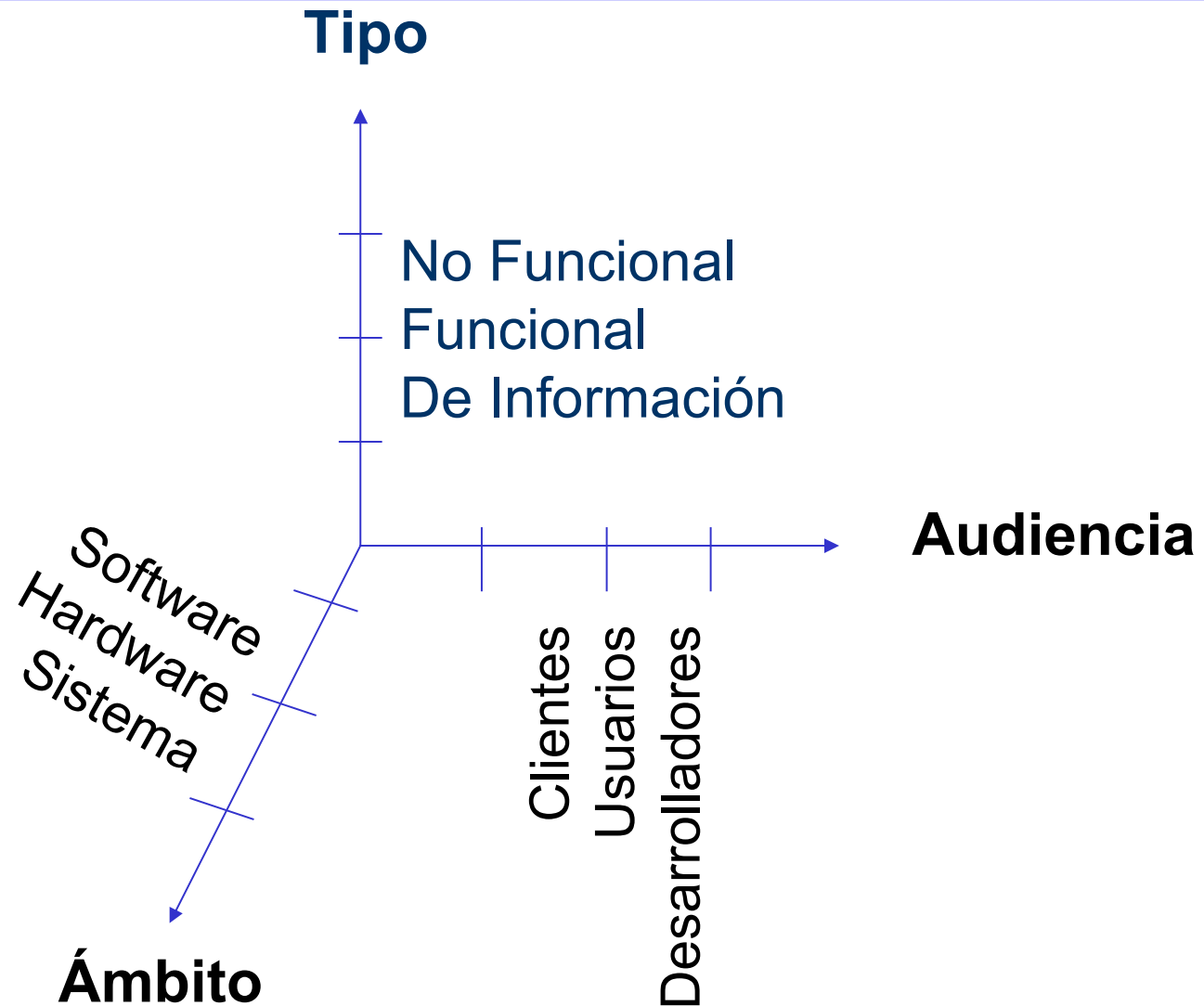


Concepto de requisito

- **Condición o capacidad** que debe tener un producto software para resolver una necesidad expresada por un usuario.
- **Representación** en forma de documento de una capacidad o condición que debe tener un producto software.
- **Característica** de un producto software que es condición para su **aceptación** por parte del cliente.
- ***Propiedad o restricción, determinada con precisión, que un producto software debe satisfacer.***



Tipos de requisitos: Clasificación





Tipos de requisitos

- **Funcionales:** Describen la interacción entre el sistema y su entorno, proporcionando servicios que proveerá el sistema o indicando la manera en que éste reaccionará ante determinados estímulos.
- **No funcionales o atributo de calidad:** Describen cualidades o restricciones del sistema que no se relacionan de forma directa con el comportamiento funcional del mismo.
- **De Información:** Describen necesidades de almacenamiento de información en el sistema.



Tipos de requisitos: Requisitos no funcionales

“Limitaciones sobre servicios y funciones que ofrece el sistema, suelen aplicarse al sistema como un todo”

- Restringen los tipos de soluciones que podemos tomar y suelen restringir el diseño que se realice.
- No describen funciones sino propiedades (rendimiento, fiabilidad, seguridad, capacidad de almacenamiento...).
- Son los que garantizan la calidad del software.
- Pueden ser requisitos del producto, requisitos de la organización o requisitos externos. **[SOMM12 página 88]**

Dificultades para determinarlos:

- Las metodologías no proveen herramientas ni formas de abordar de forma directa su obtención.
- Suelen aparecer al estudiar los posibles diseños.
- Aumentan la complejidad del diseño.
- Uso del lenguaje natural para su especificación.



Tipo de requisitos: Clasificación FURPS+

FURPS+

[Grady-1992]

- **Funcionalidad** (**F**unctionality): requisito funcional.
- **Facilidad de uso** (**U**sability): Factores humanos, ayuda, documentación.
- **Fiabilidad** (**R**eliability): Frecuencia de fallos, disponibilidad, capacidad de recuperación de un fallo y grado de previsión.
- **Rendimiento** (**P**erformance): Tiempos de respuesta, productividad, precisión, velocidad, uso de los recursos.
- **Soporte** (**S**upportability): Adaptabilidad, facilidad de mantenimiento, internacionalización, configurabilidad.



Tipo de requisitos: Clasificación FURPS+

Pseudorrequisitos o restricciones de diseño(+):

- **Implementación:** Limitación de recursos, lenguajes y herramientas, hardware, etc.
- **Interfaz:** Restricciones impuestas para la interacción con sistemas externos.
- **Operación:** Gestión del sistema en su puesta en marcha y a nivel operacional.
- **Empaquetamiento:** Formas de distribución, restricciones de instalación, etc.
- **Legales:** Licencias, derechos de autor, etc.



Ejemplos de requisitos

- El sistema debe validar la tarjeta en menos de 3 segundos.
- El sistema debe insertar las palabras en el orden correcto
- El sistema debe contar el número de palabras procesadas
- El sistema se diseñará para un terminal CRT monocromo
- Los usuarios del sistema serán en su mayoría novatos
- La cantidad que pagan los socios debe ser almacenada como dato de tipo real
- Deben de producirse informes útiles
- El sistema no deberá revelar a los operadores información personal de los clientes que no sea el nombre y referencia
- Debe existir un interfaz de usuario para las bases de datos que siga el estándar de la biblioteca general



Propiedades de los requisitos

Para que sean de calidad tienen que satisfacer:

- **Completos:** Todos los aspectos del sistema están representados en el modelo de requisitos.
- **Consistentes:** Los requisitos no se contradicen entre sí.
- **No ambiguos:** No es posible interpretar los requisitos de dos o más formas diferentes.
- **Correctos:** Representan exactamente el sistema que el cliente necesita y que el desarrollador construirá.
- **Realistas:** Los requisitos se pueden implementar con la tecnología y presupuesto disponible.
- **Verificables:** Se pueden diseñar pruebas para demostrar que el sistema satisface los requisitos.
- **Trazables:** Cada requisito puede rastrearse a través del desarrollo del software hasta su correspondiente funcionalidad del sistema.



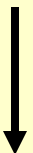
Tareas de la Ingeniería de requisitos

(0) Estudio de Viabilidad: Técnico, Económico y Jurídico

Descripción reducida
del sistema



Estudio de
viabilidad



Informe de
viabilidad

¿Es conveniente realizar el desarrollo del Sistema/Software?

- ¿Soluciona el Software los problemas existentes?
- ¿Se puede desarrollar con la tecnología actual?
- ¿Se puede desarrollar con las restricciones de costo y tiempo?
- ¿Puede integrarse con otros existentes en la organización?



Tareas de la Ingeniería de requisitos

(1) Obtención de Requisitos (Elicitación).

Información de
clientes y usuarios



Obtención



Requisitos

Trabajo con los clientes y usuarios para:

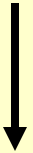
- Estudiar el funcionamiento del sistema
- Descubrir las **necesidades reales**
- Consensuar los requisitos entre las distintas partes
- Proceso difícil apoyado por técnicas:
 - Entrevista
 - Escenarios/Puntos de vista
 - **Casos de uso**
 - Prototipado
 - Análisis Etnográfico



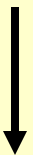
Tareas de la Ingeniería de requisitos

(2) Análisis de Requisitos

Lista de requisitos e
información del usuario



Análisis



Requisitos

Actividad más importante de todas

Objetivos:

- Detectar **conflictos** entre los requisitos
- **Profundizar** en el conocimiento del sistema
- Establecer las **bases para el diseño**
- Construcción de modelos abstractos

Sistema — — — — — ➔ Software

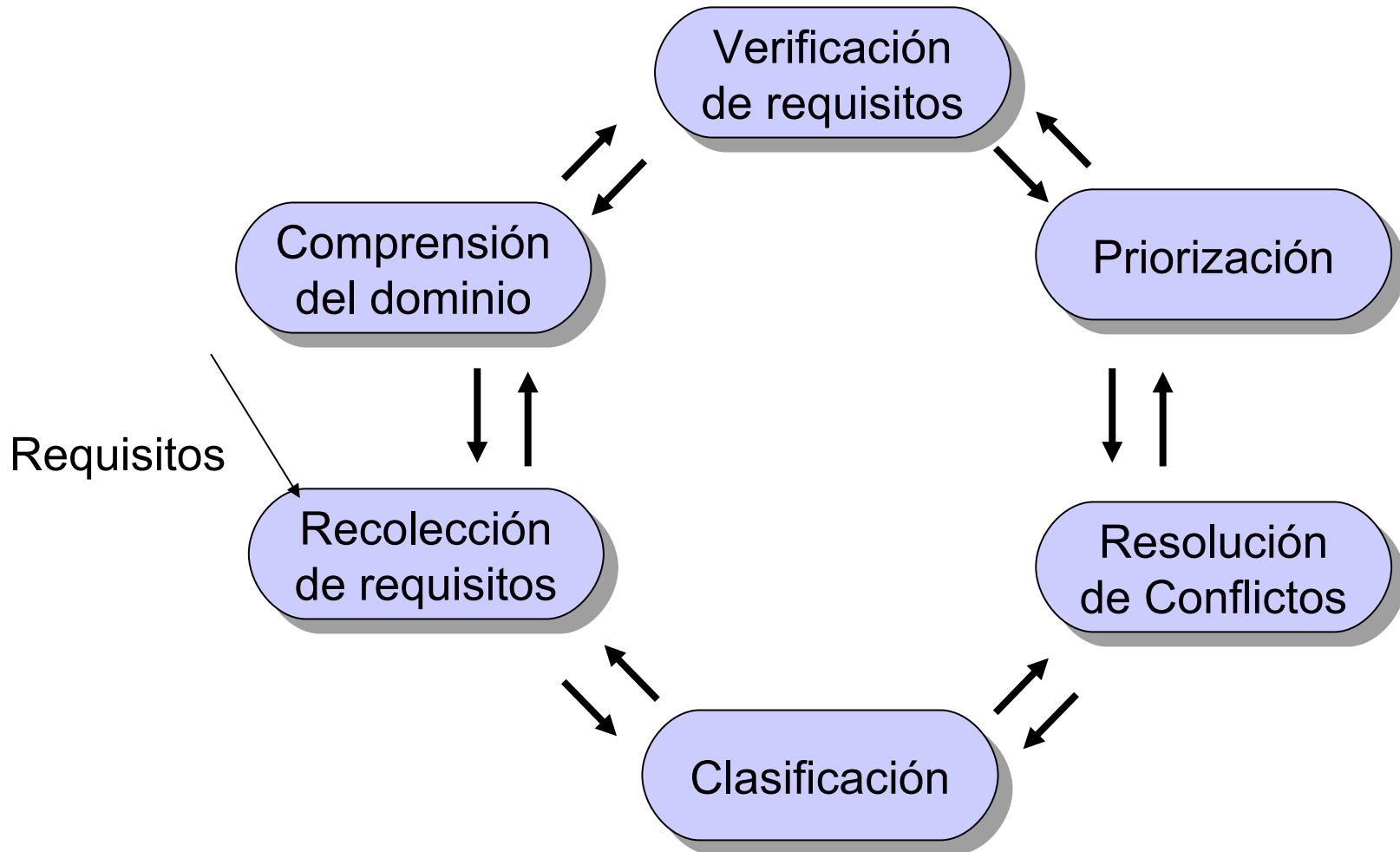


“Estudio de Soluciones”



Tareas de la Ingeniería de requisitos

Actividades del **Análisis de Requisitos**

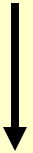




Tareas de la Ingeniería de requisitos

(3) Especificación de Requisitos

Lista de requisitos



Especificación



Documento de
Especificación

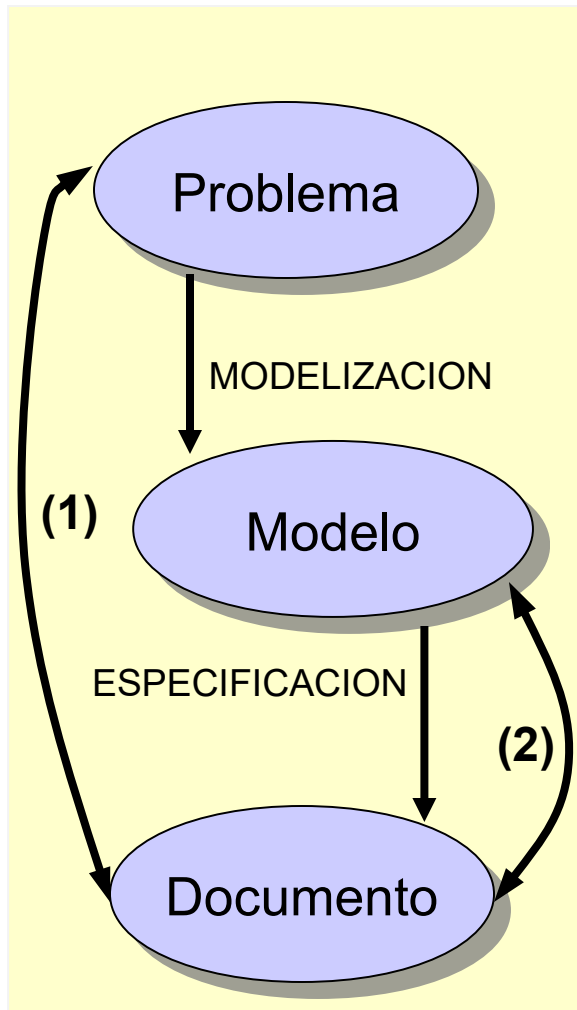
- Representación de los requisitos en base al modelo creado en la etapa de análisis (documento escrito, conjunto de diagramas, modelo matemático, simulación, prototipo)
- Utilización de herramientas y de estándares
- Manual preliminar del usuario

“la idea es correcta pero no es la forma en lo que yo me imaginaba que se iba a poder realizar”



Tareas de la Ingeniería de requisitos

(4) Revisión de requisitos



Validación (1): Ver que los requisitos documentados representan el problema que se desea representar

Verificación (2): La representación es correcta

Proceso continuo durante todo el desarrollo

Facilitar la revisión

- Crear prototipos
- Crear simulaciones
- Revisión automática (técnicas formales)
- Apoyarse en herramientas



Tareas de la Ingeniería de requisitos

Productos generados

(1) En la Obtención de Requisitos:

- Documentos de entrevistas
- Lista Estructurada de Requisitos
- Diagramas de casos de uso + Plantillas de casos de uso
+ Diagramas de actividad

(2) En la especificación de Requisitos:

- Modelo arquitectónico (subsistemas) **====>** Diagrama de paquetes
- Modelos estático (Conceptual) **====>** Diagrama de Clases
- Modelo dinámico (Funcional) **====>** Diagrama de secuencia del sistema + contratos



¿Qué roles pueden distinguirse en el proceso de ingeniería de requisitos?

- Stakeholder (Personas que tienen relación con el sistema)
- Ingeniero de requisitos
- Analista de sistemas
- Arquitecto del software (Diseño)
- Documentalista
- Diseñador de Interfaces de Usuario
- Gestor de proyecto
- Revisor



Problemas de la Ingeniería de Requisitos

Podemos agruparlos en 3 áreas:

- Dificultades para **obtener información**
- Manejo de la **complejidad del problema**
- Dificultades para la **integración de los cambios**

Posibles causas

- Pobre comunicación
- Uso de técnicas inapropiadas
- Tendencias a acortar el análisis
- No considerar alternativas