



Tema 2.2: Conceptos básicos de la Ingeniería de Requisitos

Ingeniería de Requisitos

Raquel Martínez España

Grado en Ingeniería Informática



Índice

1. Concepto de requisito
2. Tipos de requisitos



Objetivos

- Definir las diferencias entre requisitos del cliente y del desarrollador.
- Conocer el enfoque orientado a objetos en la IR.



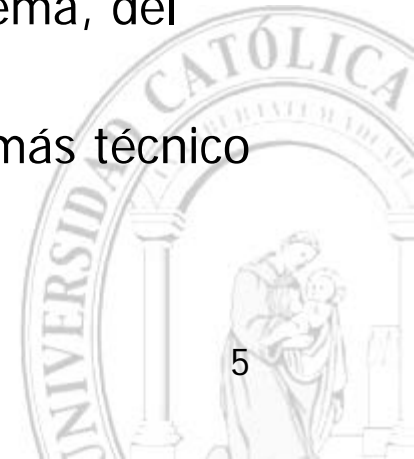
Índice

1. Concepto de requisito
2. Tipos de requisitos
 1. Requisitos funcionales y no funcionales. Dominio.
 2. Requisitos de usuario y sistema. Software.
 3. Requisitos de cliente y requisitos de desarrollador.



2.3. Requisitos de cliente y de desarrollador

- Dos niveles en los requisitos:
 - ¿Del cliente o del sistema, del usuario o del software?
 - Puente a medio camino: del cliente para el desarrollador
 - Primer nivel: **requisitos del cliente** (o del usuario)
 - Deseos y necesidades del cliente, expresados en lenguaje comprensible por él
 - Audiencia primaria: cliente
 - Audiencia secundaria: desarrollador
 - Segundo nivel: **requisitos del desarrollador** (o del sistema, del software, detallados...)
 - Forma estructurada y específica, carácter mucho más técnico
 - Audiencia primaria: desarrollador
 - Audiencia secundaria: cliente



2.3. Requisitos de cliente y de desarrollador

- La finalidad última es la misma.
- La distinción entre los dos niveles no es muy clara:
 - Forma: no estructurada / estructurada
 - Audiencia: cliente / desarrollador
 - Contenido: mayor o menor nivel de detalle
 - Texto / Texto + Diagramas
 - Requisitos en bruto / Requisitos depurados



2.3. Requisitos de cliente y de desarrollador

- Distintas nomenclaturas y clasificaciones, misma idea de fondo:
 - Clásica/IEEE:
 - Una única fase: Análisis de Requisitos
 - Único documento Especificación de Requisitos (ERS) con los requisitos C y D
 - USDP (Unified Software Development Process):
 - Dos fases: Requisitos + Análisis
 - Dos documentos: Requisitos del Usuario (RU) + Requisitos del Sistema (RS)



2.3. Requisitos de cliente y de desarrollador

- Enfoque OO
 - **Requisitos C**
 - Expresión:
 - Lenguaje natural
 - Casos de uso
 - (Diagramas de estado y flujo de datos)
 - Incluye interfaces de usuario.
 - **Requisitos D**
 - **Consistentes con, y son un refinamiento de, los requisitos C**
 - (D)etallados
 - Expresión:
 - Diagramas de secuencia por caso de uso
 - Clases



2.3. Requisitos de cliente y de desarrollador

- **Requisitos C**

1. Identificar al “cliente”

2. Entrevistar representantes del cliente

3. Escribir requisitos C en formato de documento estándar / **Casos de Uso**

4. Inspeccionar los requisitos C

Con la aprobación del cliente....

5. Construir los requisitos D

Revisión con el cliente



2.3. Requisitos de cliente y de desarrollador

- Requisitos D

Obtener requisitos C

1. Seleccionar la organización de requisitos D

2. Crear diagramas de secuencia a partir de los casos de uso

En paralelo

3 a. Obtener requisitos D a partir de C y del cliente

3 b. Describir planes de prueba

3 c. Inspeccionar

4. Validar con el cliente

Aplicar retroalimentación del cliente

Cuando el cliente aprueba la unidad

5. Liberar

Ingeniería de Requisitos



2.3. Requisitos de cliente y de desarrollador

- **Requisitos D** (Enfoque OO)

Obtener requisitos C

1. Clases de dominio y objetos a partir de casos de uso

2. Agregar clases de dominio esenciales adicionales. Inspeccionar.

3 Para cada clase:

- atributos y funcionalidad
- objetos requeridos
- reacción objetos/caso
- planes de pruebas
- inspeccionar resultados

4. Inspeccionar contra Requisitos C

5. Verificar con el cliente

Al terminar:

6. Liberar



2.3. Requisitos de cliente y de desarrollador

- Organización de **Requisitos D**
 - Por clase: Orientado a Objetos.
 - Por caso de uso (“por escenario”): USDP.
 - Por característica: servicio deseado percibido en el exterior, pares estímulo respuesta
 - Por modo: distintos perfiles
 - Por estado: requisitos para cada estado y eventos que afectan en cada estado.
 - Por jerarquía de función: descomposición en funciones, subfunciones, etc.



2.3. Requisitos de cliente y de desarrollador

- Tipo de **Requisitos D**
 - Requisitos funcionales
 - Requisitos no funcionales
 - Rendimiento
 - Confiabilidad y disponibilidad
 - Manejo de errores
 - Interfaz
 - Restricciones
 - Requisitos **inversos**: qué no debe hacer el sw (eliminar malentendidos)



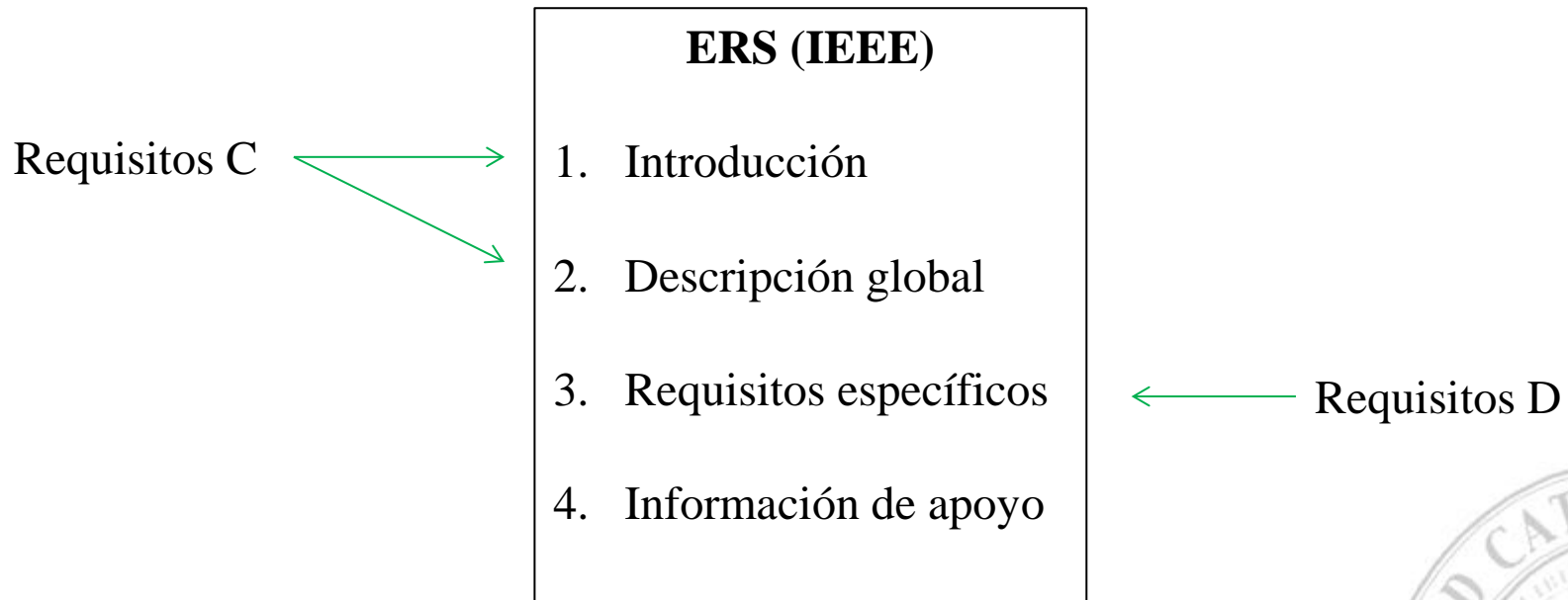
2.3. Requisitos de cliente y de desarrollador

- Tipo de **Requisitos C**
 - En realidad son los mismo que para los D
 - Distinción en segundo plano
 - Objetivo principal: obtener información del cliente



2.3. Requisitos de cliente y de desarrollador

- Correspondencia con el estándar IEEE-830
(requisitos específicos con organización OO)



2.3. Requisitos de cliente y de desarrollador

- Correspondencia con el estándar IEEE-830
(requisitos específicos con organización OO)

3. Requisitos específicos

3.1. Requisitos de interfaz externa

3.1.1. Interfaces de usuario

3.1.2. Interfaces de hardware

3.1.3. Interfaces de software

3.1.4. Interfaces de comunicación

3.2. Clases/Objetos

3.3. Requisitos de rendimiento

3.4. Restricciones de diseño

3.5. Atributos del sistema de sw

3.6. Otros requisitos

Requisitos de interfaz

Requisitos funcionales

Requisitos inversos

Otros requisitos no
funcionales

2.3. Requisitos de cliente y de desarrollador

- Evaluación de Análisis Orientado a Objetos (AOO) en IR:
 - Ventajas de OO:
 - Se acomoda bien para el diseño y la implementación → continúa una forma de pensamiento y notación
 - No pone énfasis en la función como lo hace AE
 - Evita la fragmentación que produce el AE
 - Desventajas de OO:
 - Complejo para rescatar características dinámicas de los objetos
 - No es claro que siempre se quiera modelar objetos, servicios y relaciones
 - Tendencia a pasar rápidamente al diseño
 - No es la bala de plata pensada por muchos



Puntos clave

- Requisitos Cliente (C) →
 - Principalmente para el cliente
 - Especificación clara: lenguaje natural, casos de uso, interfaces
- Requisitos Desarrollador (D) →
 - Orientados al Diseñador/Desarrollador
 - También para el cliente
 - Detallan requisitos C
 - Diferentes organizaciones según uso.

