



## **INGENIERIA DE REQUERIMIENTOS (IR)**

### **1.- Fundamentos de ingeniería de Requerimientos.**

- 1.1 Conceptos básicos.
- 1.2 Tipos de requerimientos.
- 1.3 Procesos de la Ing. de Requerimientos.
- 1.4 Problemas en el manejo de requerimientos

## Motivación

Hoy en día el Software es pervasivo en la mayoría de los sectores incluyendo:

Industria automotriz

Servicios financieros

Industria electrónica

Dispositivos médicos, etc.

Esto ha dado origen al termino **Sistemas Intensivos en Software** → *Sistema cuyas partes esenciales o su funcionalidad es realizada por software.*



Se pueden identificar dos tipos de Sistemas Intensivos en Software:

- **Sistemas de Información** → Este sistema recolecta, almacena, transforma, transmite y/o procesa datos o información.

Su principal objetivo es proveer a los usuarios de la información que requiere, en el momento que lo requiere..

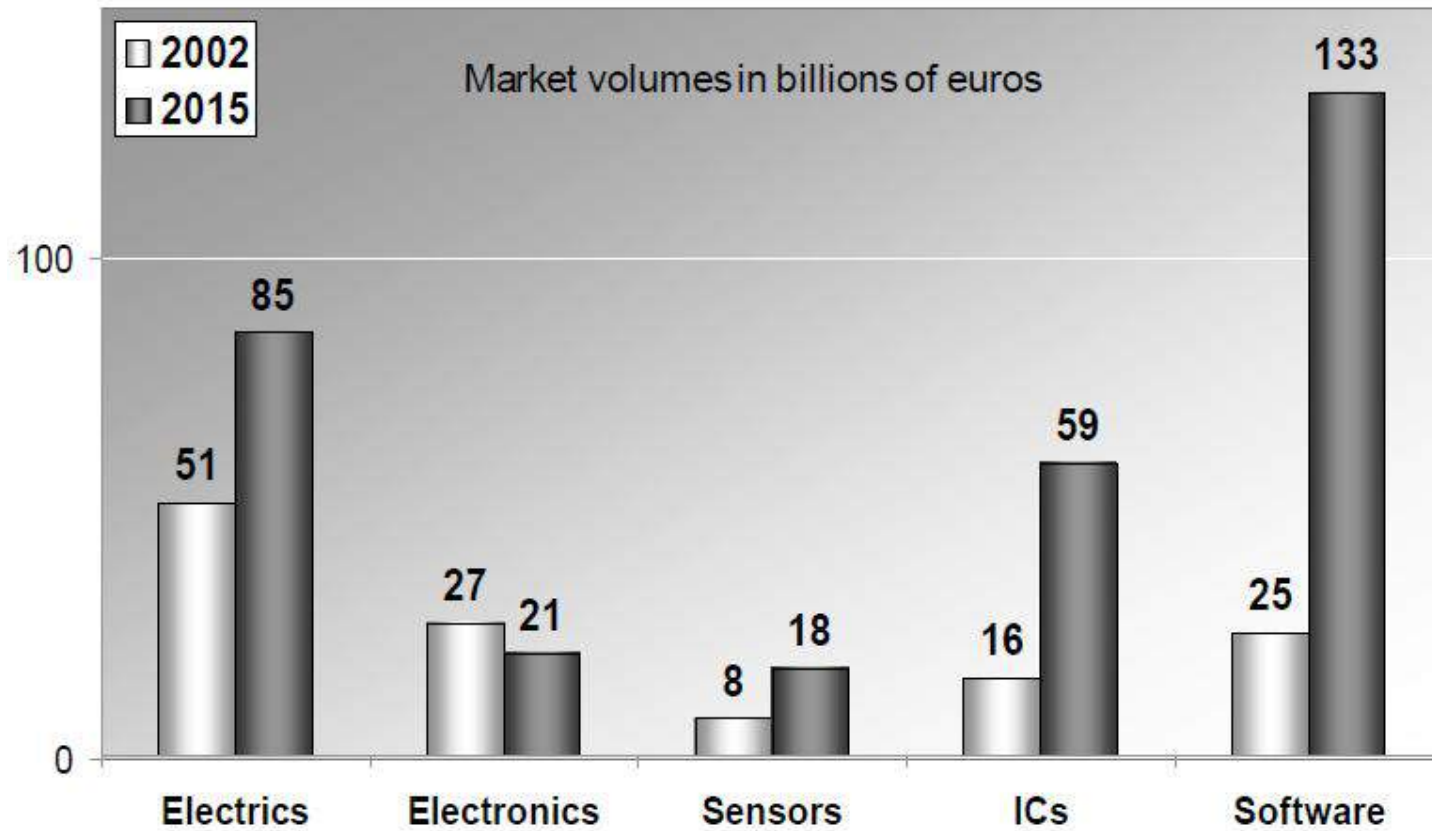
Este tipo de sistemas consiste principalmente en software que se ejecuta en computadoras ordinarias.

Ejemplos¿?

- **Sistemas intensivos en software embebidos** → En este sistema el software es solo una parte, aunque importante del sistema. Este software esta integrado al hardware. Debe existir una comunicación entre ambos para que el sistema funcione.

Ejemplos¿?

## Volumen del mercado de software en la industria automotriz (Honsing 2005)



## Retos en el desarrollo de Sistemas Intensivos en Software

- Innovaciones basadas en software → Los clientes demandan continuamente nuevas innovaciones.
- Incremento en la complejidad → Mayor necesidad de operaciones, interoperabilidad, integración con otros sistemas, etc.
- Presión para reducir costos → Mayor demanda de clientes, mayor competencia.
- Tiempos de desarrollo mas cortos → Reducir el tiempo de entrega sin reducir complejidad.
- Mayor demanda y de mayor calidad → Se requiere software para todo, incluyendo funciones de alta seguridad y funciones criticas de sistemas.

Problema:

El desarrollo de sistemas se ha extendido desde los años 60's.

Desde entonces se han desarrollado infinidad de sistemas con diversos problemas:

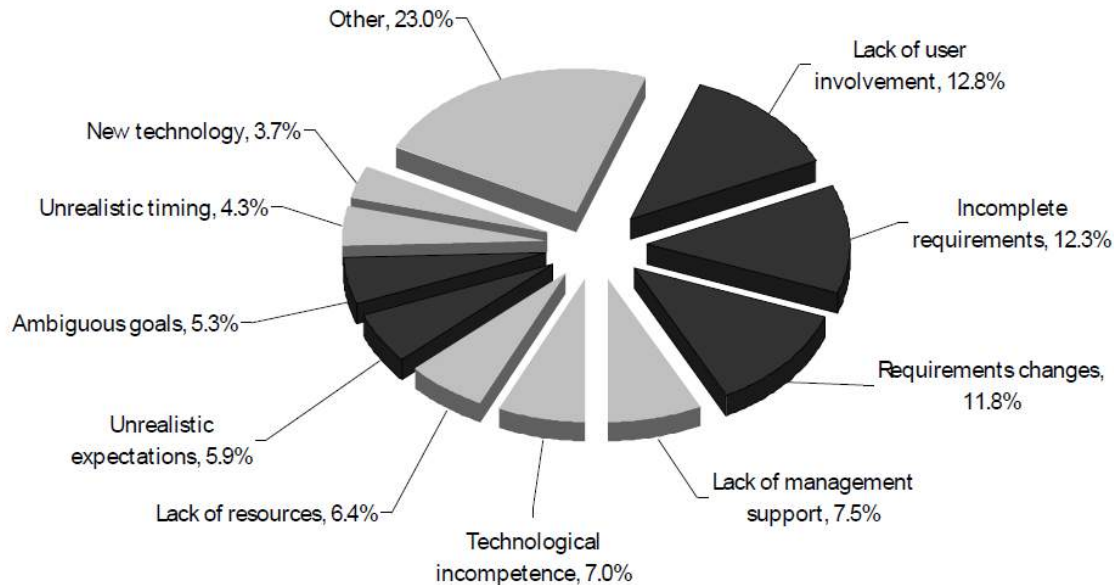
-Se entregaron tarde

-Sobrepasaron el presupuesto

-No realizan lo que los usuarios desean

-No son utilizados o no se utilizan a su máxima capacidad (no reciben lo que pagaron)

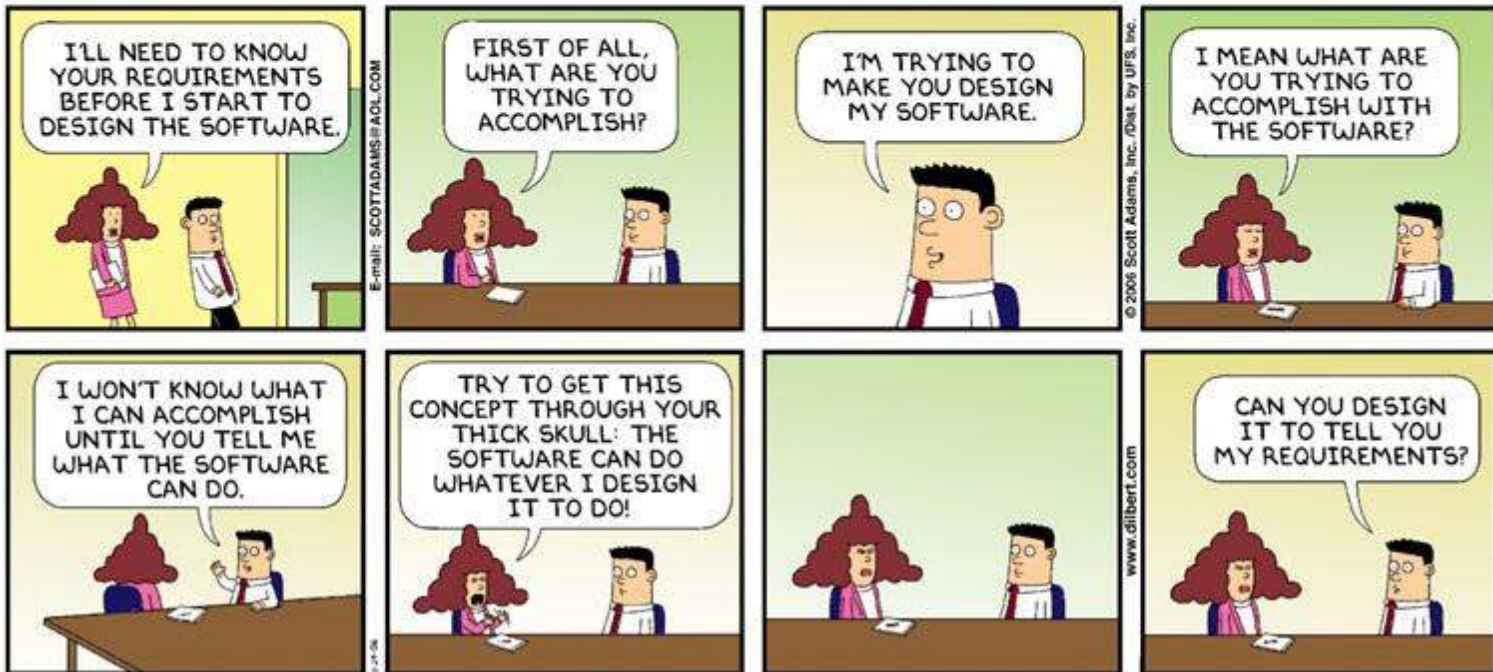
Razones por las que varios proyectos sobrepasaron el presupuesto o no fueron completamente funcionales de acuerdo a: The Standish Group,\* 1995.



La mayor contribución a estos problemas → Mala definición de los **requerimientos del sistema**

\*El Standish Group se encuentra en Boston, Massachusetts es una compañía líder en desempeño y análisis de Tecnologías de la Información.

Es importante definir los requerimientos del sistema., aunque no es fácil

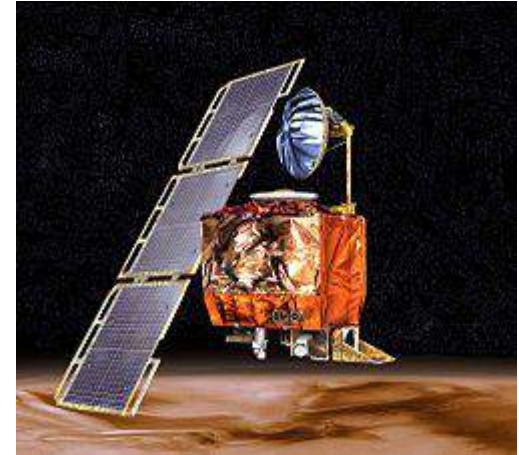


© Scott Adams, Inc./Dist. by UFS, Inc.



## Importancia de la IR

- La **Mars Climate Orbiter (MCO)** fue una sonda de la NASA lanzada desde Cabo Cañaveral el 11 de diciembre de 1998
- La Mars Climate Orbiter se destruyó debido a un error de navegación, consistente en que el equipo de control en la Tierra hacía uso del [Sistema Anglosajón de Unidades](#) para calcular los parámetros de inserción y envió los datos a la nave, que realizaba los cálculos con el [Sistema métrico decimal](#).
- Así, cada encendido de los motores habría modificado la velocidad de la sonda de una forma no prevista y tras meses de vuelo el error se había ido acumulando. Durante los últimos días de vuelo, conforme la gravedad de Marte tenía una creciente influencia, se observó que la sonda se apartaba cada vez más de la trayectoria prevista y se acercaba más y más al planeta, algo que hubiera sido imposible si se hubieran tenido en cuenta bien todos los factores. Finalmente la sonda pasó sobre Marte a sólo 57 km de altura, en lugar de los 140-150 previstos, quedando destruida por la fricción con la atmósfera del planeta.



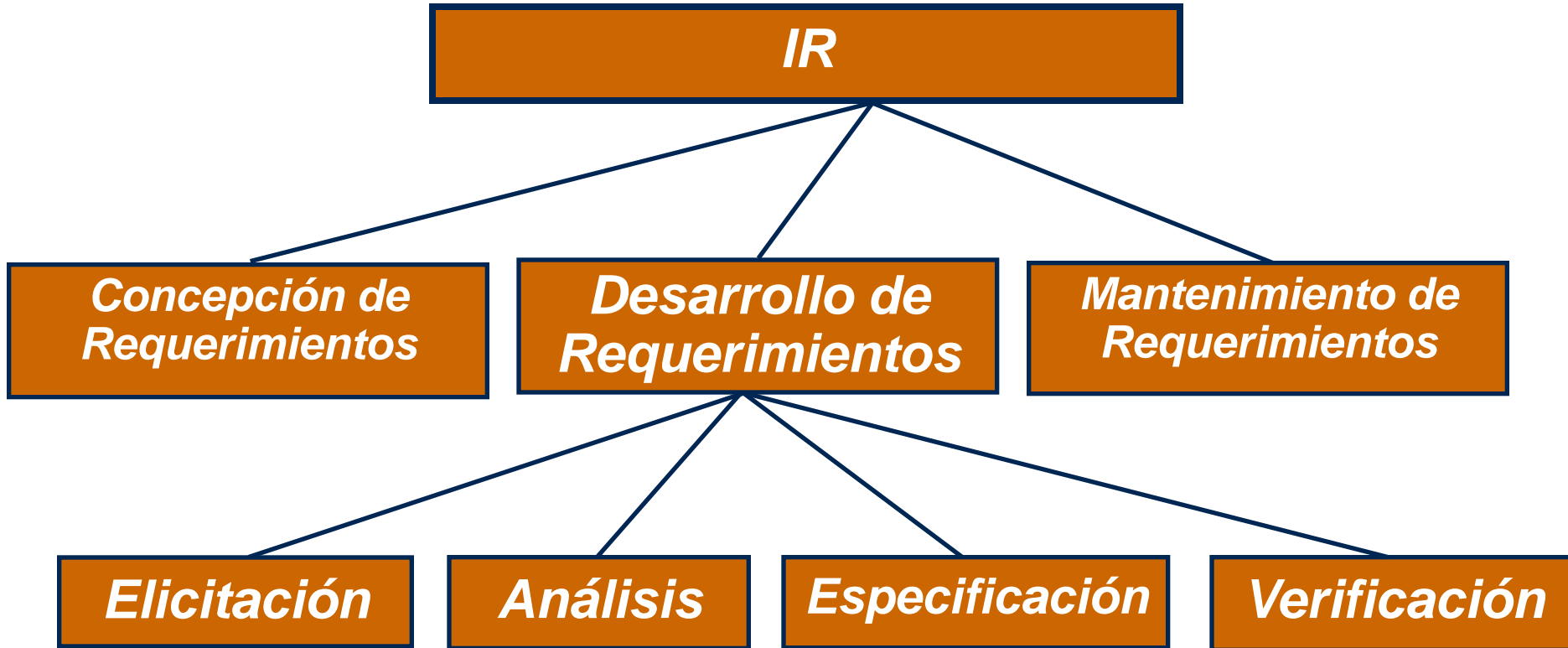
Fuente: wikipedia

# ¿Que es Ingeniería de Requerimientos ?

- **IR es:**
  - Actividad relacionada con el descubrimiento, documentación, y mantenimiento de un conjunto de requerimientos para un sistema intensivo en software.
  - El termino Ingeniería implica que se utilizaran técnicas repetibles y sistemáticas para asegurar que los requerimientos del sistema son completos, consistentes y relevantes.
  - IR consiste en identificar el propósito del sistema de Software así como el contexto en el que será utilizado.
  - IR una rama de la Ingeniería de Software relacionada con las funciones y restricciones de un sistema de software.

- Cómo/Donde se utilizará el sistema
- Captura las necesidades del mundo real que pueden afectar al sistema y las expresa como artefactos que pueden ser implementados en un sistema de Software
- Puente entre el diseño y la construcción.

# Actividades de la Ing. De Requerimientos



# Acerca de estas actividades...

- **Concepción**
  - Iniciar el proceso (necesidad de una compañía, oportunidad en el mercado, buena idea, ...), Estudio de viabilidad, riesgos, etc.
- **Elicitación de requerimientos**
  - Se descubren los requerimientos por medio de consultas con los **stakeholders** (involucrados/interesados, ejem: clientes, usuarios, programadores)
- **Análisis y negociación de requerimientos**
  - Los requerimientos son analizados y los conflictos se resuelven a través de negociación.
- **Especificación de requerimientos**
  - Se produce un documentos precisando los requerimientos.
- **Validación de requerimientos**
  - Se revisa la consistencia y la integridad del documentos de requerimientos.
- **Mantenimiento de requerimientos**
  - Mejora continua, cambios en las necesidades y contextos.

# ¿Que son los requerimientos ?

- Los requerimientos para un sistema computacional son los servicios proporcionados por el sistema y sus restricciones operativas. (sommerville, 2005)
- Los requerimientos pueden variar desde una declaración abstracta de alto nivel o una definición detallada y formal de una función del sistema, esto depende de las necesidades de los stakeholders.

**Algunos de los problemas que surgen durante el proceso de ingeniería de requerimientos son resultado de no hacer una clara separación entre estos diferentes niveles de descripción.**

*Fuente: Ingeniera de software, sommerville*

## Distintos involucrados con los requerimientos.



How the customer explained it



How the project leader understood it



How the analyst designed it



How the programmer wrote it



What the beta testers received



How the business consultant described it



How the project was documented



What operations installed



How the customer was billed



How it was supported



What marketing advertised



When it was delivered



What the customer really needed



What the digg effect can do to your site



The disaster recover plan

## Distintos “Requerimientos”...

Objetivos vs Requerimientos.

- Un **objetivo** todavía no es un requerimiento. Los objetivos son utilizados para guiar el proceso de IR. Pueden ser utilizados para identificar los diversos requerimientos.

Ejemplo:


“Implementar un sistema de acceso a biblioteca que funcione en un dispositivo móvil”

Es un objetivo no un requerimiento.

De acuerdo al estándar IEEE 830 un requerimiento es:

**Una condición o capacidad que debe estar presente en un sistema o componentes de un sistema para satisfacer un contrato, estándar, especificación o cualquier otro documento formal.**

**Una representación documentada de una condición o necesidad de un sistema.**



Todos los requerimientos deben ser verificables.

## Ejemplos de requerimientos:

1. El sistema debe mantener una serie de registros de las búsquedas de material de la biblioteca, incluyendo libros, periódicos, revistas y discos compactos, realizadas por el usuario.
2. El sistema debe permitir que los usuarios busquen un objeto por título, autor o ISBN
3. La interfaz de usuario del sistema debe ser implementada usando un navegador Web
4. El sistema debe mostrar información al usuario en menos de 1 minuto.
5. El sistema debe estar disponible en al menos tres sistemas operativos móviles existentes.

Estos requerimientos son de diferentes tipos:

**1.- Muy general, como el mostrado en 1**

**2.- Requerimientos funcionales, como el 2, que define una parte de la funcionalidad del sistema.**

**3.- Requerimientos de implementación, como el 3.**

**4. Requerimientos de desempeño, como el 4.**

**5.- Requerimientos de usabilidad.**



Los requerimientos pueden describir alguno de los siguientes puntos:

- **Una facilidad a nivel de usuario.**

Ejemplo: El procesador de texto debe incluir un verificador de ortografía y una función de corrección

- **Una propiedad muy general del sistema.**

Ejemplo: El sistema debe asegurar que la información personal nunca se haga disponible sin autorización.

- **Una restricción específica del sistema**

Ejemplo: El sensor debe ser presionado diez veces por segundo.

- **Una restricción para el desarrollo del sistema**

Ejemplo: El sistema debe ser desarrollado usando Android.

- **Como realizar una operación**

Ejemplo: La cantidad de registros debe ser obtenida a través de las fechas de acceso, sumando cada una.

## Clasificación de requerimientos (1): Dominios.

Debido a la diversidad de los requerimientos, Sommerville (2005), sugiere organizarlos en tres dominios:

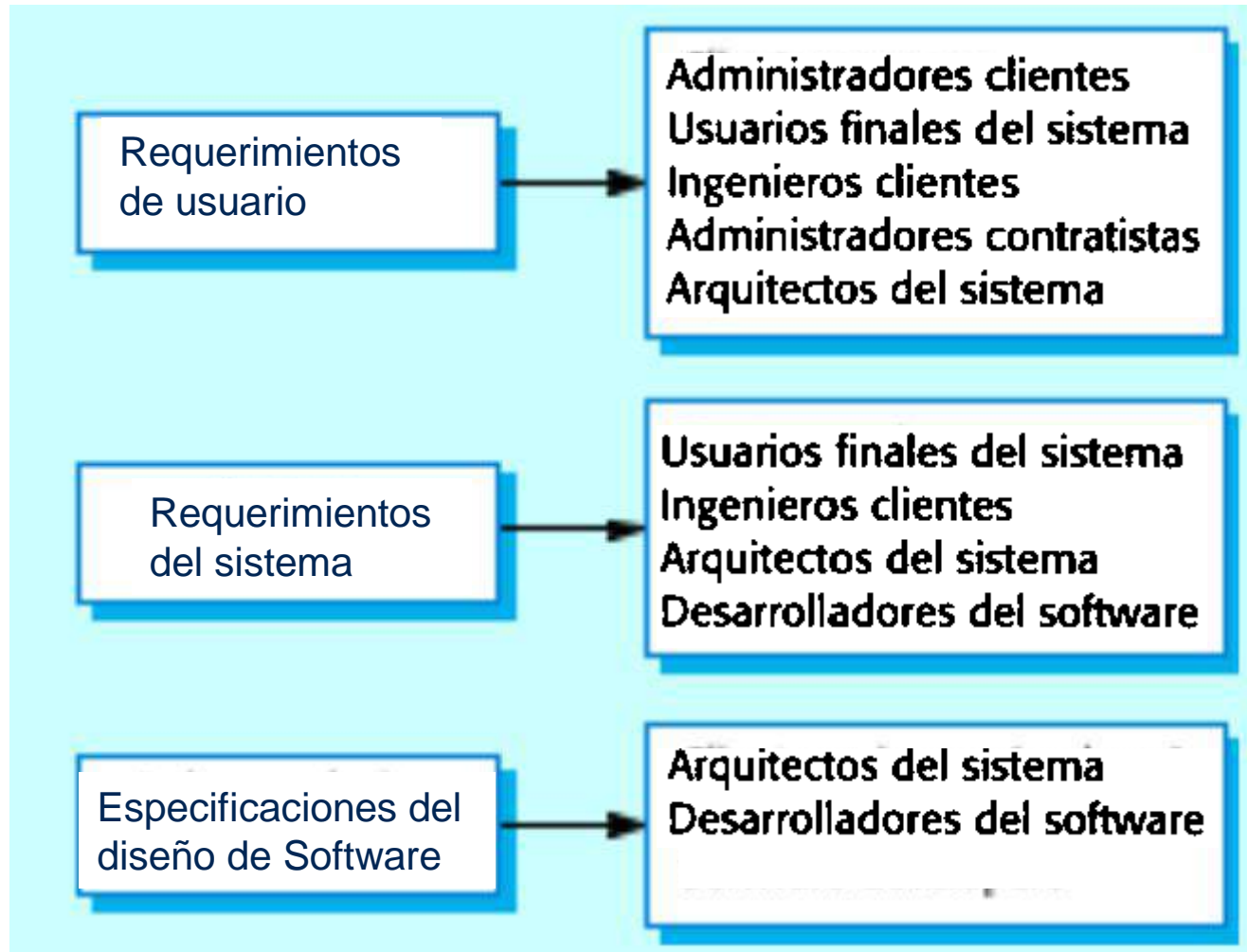
1. Requerimientos del usuario
2. Requerimientos del sistema
3. Especificaciones del diseño de software.

1.- **Requerimientos del usuario**→ Son declaraciones en lenguaje natural y en diagramas informales, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar.

2.- **Requerimientos del sistema**→ Establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requerimientos del sistema debe ser **preciso**. (Se incluye en el contrato)

3.- **Especificaciones del diseño de software**→ Se obtienen de las especificaciones del sistema. Están definidas de manera formal en el documentos de Especificaciones de Requerimientos del Software.

El dominio de los requerimientos se refiere a quien van dirigidos:



## Ejemplo de requerimientos

Sistema para el control de maletas en un aeropuerto.

### **Requerimiento del Usuario:**

- El sistema debe ser capaz de procesar 20 maletas por minuto.

### **Algunos requerimientos del sistema:**

- Cada maleta procesada debe lanzar un evento de almacenamiento.
- El sistema debe ser capaz de manejar 20 eventos de almacenamiento por minuto.

### **Especificaciones del sistema asociadas:**

1.2 El sistema debe ser capaz de procesar 20 eventos de almacenamiento por minuto en modo operacional

1.2.1 Si ocurren más de 20 eventos de almacenamiento en un minuto, el sistema debe ....

1.2.2 [mas excepciones...]

Ejemplo:

## **Sistema de administración de pacientes para apoyar la atención a la salud mental (MHC-PMS)**

### **Definición del requerimiento del usuario**

1. El MHC-PMS elaborará mensualmente informes administrativos que revelen el costo de los medicamentos prescritos por cada clínica durante ese mes.

El requerimiento del usuario es muy general.

### **Especificación de los requerimientos del sistema**

- 1.1 En el último día laboral de cada mes se redactará un resumen de los medicamentos prescritos, su costo y las clínicas que los prescriben.
- 1.2 El sistema elaborará automáticamente el informe que se imprimirá después de las 17:30 del último día laboral del mes.
- 1.3 Se realizará un reporte para cada clínica junto con los nombres de cada medicamento, el número de prescripciones, las dosis prescritas y el costo total de los medicamentos prescritos.
- 1.4 Si los medicamentos están disponibles en diferentes unidades de dosis (por ejemplo, 10 mg, 20 mg) se harán informes por separado para cada unidad de dosis.
- 1.5 El acceso a los informes de costos se restringirá a usuarios autorizados en la lista de control de acceso administrativo.

Los requerimientos del sistema ofrecen información más específica sobre los servicios y las funciones del sistema que se implementará

## Clasificación de requerimientos (2): Tipos.

Otra clasificación de los requerimientos:

- Funcionales
- No Funcionales
- Del Dominio

### **1. *Requerimientos funcionales.***

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares.

En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.

Representación:

Lenguaje natural,

Modelos visuales,

Métodos formales

**Los requerimiento Funcionales definen:**

Cuáles **entradas** debe aceptar el sistema

Cuáles **salidas** debe producir el sistema

Qué **datos** debe almacenar el sistema que utilizarán otros sistemas

Qué **operaciones** debe realizar el sistema

La **sincronización** y **cronometraje** de las actividades anteriores.

**Ejemplos para el sistema de control de maletas**

- El sistema debe manejar hasta 20 maletas por segundo
- Si el suministro de corriente falla, el sistema debe apagarse de manera ordenada en menos de 5 segundo.
- Cada usuario del sistema debe identificarse de manera única utilizando su numero de empleado de 8 dígitos.

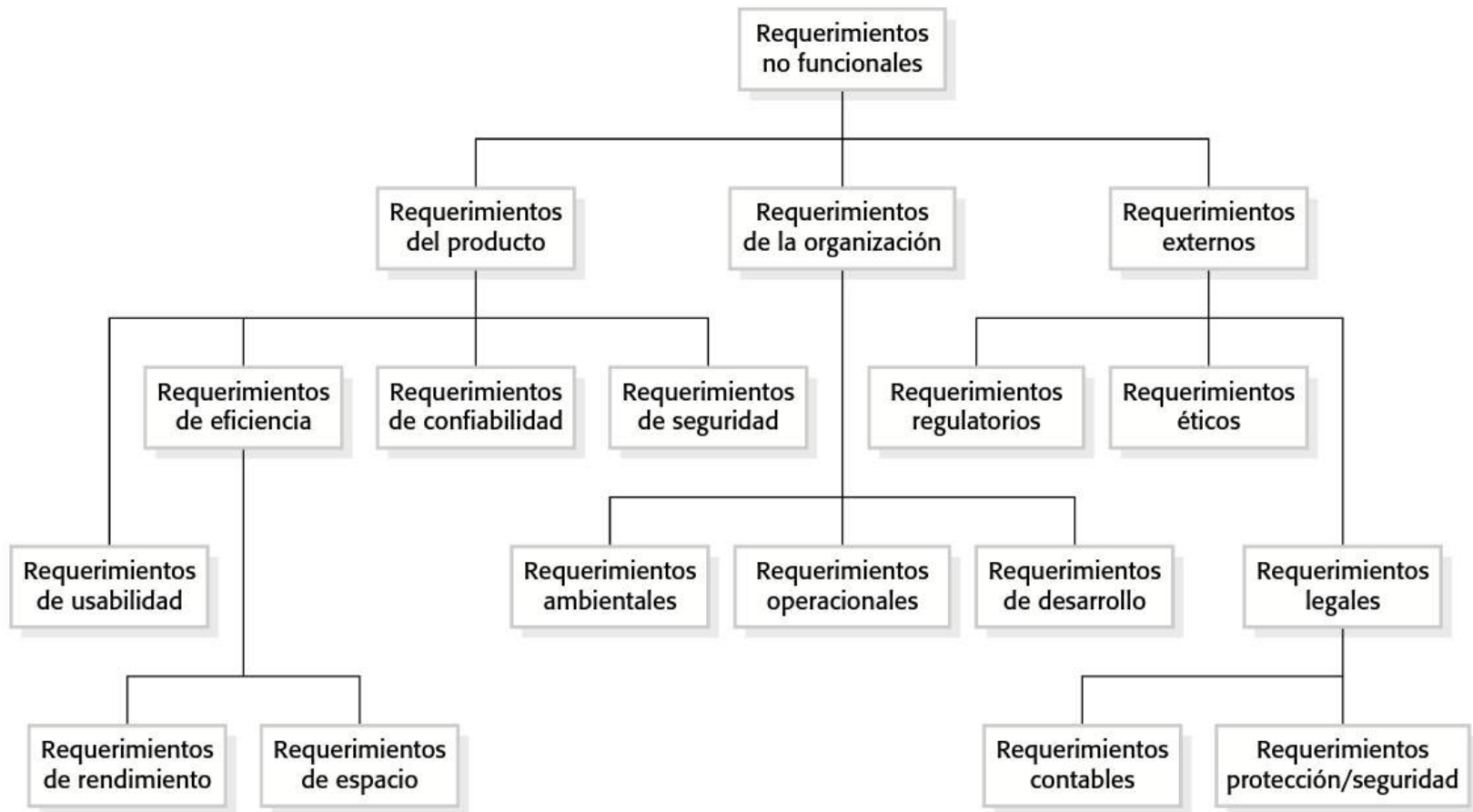
## **2. Requerimientos No Funcionales. (De calidad, Pohl)**

- Los requerimientos no funcionales no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las **propiedades emergentes** de éste como la *fiabilidad*, *el tiempo de respuesta* y la *capacidad de almacenamiento*.
- De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema.
- Pueden especificar: el rendimiento del sistema, la protección, la disponibilidad, y otras propiedades emergentes.
- A menudo son más críticos que los requerimientos funcionales particulares.

Los requerimientos no funcionales surgen de las necesidades del usuario, debido a las restricciones en el presupuesto, a las políticas de la organización, a la necesidad de interoperabilidad con otros sistemas software o hardware, o a factores externos como regulaciones de seguridad o legislaciones sobre privacidad



## Tipos de requerimientos No Funcionales



Fuente (Ing. de Software, Sommerville)

### Requerimientos de Producto

- Eficiencia
  - Desempeño (Ejemplo: Numero de maletas por minuto)
  - Espacio (Ejemplo: Mínima cantidad de memoria requerida)
- Fiabilidad (tiempo mínimo antes de primera falla)
- Portabilidad (Puede usarse con otro S.O o con otro HW?)
- Usabilidad (Tiempo de entrenamiento requerido)

### Requerimientos Organizacionales

- Entrega (ejemplo: Fecha de entrega, fecha cuando estará operacional, sesiones de entrenamiento, actualizaciones)
- Implementación
- Estándares

### Requerimientos Externos

- Interoperabilidad (Ejemplo: Comunicación con otro equipo).
- Éticos (Ejem: Seguridad para los operadores)
- Legislativos (Ejem: Reglas de privacidad)

Ejemplo:

## **Sistema de administración de pacientes para apoyar la atención a la salud mental (MHC-PMS)**

### **REQUERIMIENTO DEL PRODUCTO**

El MHC-PMS estará disponible en todas las clínicas durante las horas de trabajo normales (lunes a viernes, de 8:30 a 17:30). En cualquier día, los tiempos muertos dentro de las horas laborales normales no rebasarán los cinco segundos.

### **REQUERIMIENTOS DE LA ORGANIZACIÓN**

Los usuarios del sistema MHC-PMS se acreditarán a sí mismos con el uso de la tarjeta de identidad de la autoridad sanitaria.

### **REQUERIMIENTOS EXTERNOS**

Como establece la HStan-03-2006-priv, el sistema implementará provisiones para la privacidad del paciente.

Los requerimientos NF son muy importantes

Si no se consiguen, el sistema es inútil

Pueden ser difíciles de establecer (sobre todo al principio), si son imprecisos puede ser difícil verificarlos.

Algunas recomendaciones para determinarlos:

<b>Propiedad</b>	<b>Medida</b>
Rapidez	Transacciones procesadas por segundo Tiempo de respuesta al usuario y a eventos Tiempo de actualización de la pantalla
Tamaño	K Bytes Número de chips de RAM
Facilidad de Uso	Tiempo de entrenamiento Número de pantallas de ayuda
Fiabilidad	Número promedio entre fallos Probabilidad de no disponibilidad Tasa de Ocurrencia de fallos Disponibilidad
Robustez	Tiempo de reinicio entre fallos Porcentaje de eventos que provocan fallos Probabilidad de corrupción de los datos después de fallos
Portabilidad	Porcentaje de declaraciones dependientes de objetivos Número de sistemas objetivo

### **3. *Requerimientos del Dominio***

- Estos requerimientos se derivan del dominio de aplicación del sistema mas que de las necesidades específicas del usuario.
- Incluyen terminología especializada del dominio o referencias a conceptos del dominio.
- Pueden ser:
  - Requerimientos funcionales nuevos,
  - Restringir los existentes o
  - Establecer como se deben ejecutar cálculos particulares.
- Si no se satisfacen puede ser que el sistema no funcione adecuadamente.

Para redactarlos se requieren conocimientos especializados del dominio

### Ejemplo de RD para un sistema de biblioteca:

1.- Deberá existir una interfaz de usuario estándar para todas las bases de datos que estará basada en el estándar Z39.50

### Ejemplo para un sistema de control de trenes.

La desaceleración del tren se calculará como:

$$D_{\text{tren}} = D_{\text{control}} + D_{\text{gradiente}}$$

donde  $D_{\text{gradiente}}$  es  $9.81\text{ms}^2 * \text{gradiente compensado} / \text{alfa}$  y en donde los valores de  $9.81\text{ms}^2 / \text{alfa}$  se conocen para diferentes tipos de trenes.

## Problemas Relacionados con los Requerimientos del Dominio

- **Comprensibilidad**

- Los requerimientos son expresados en el lenguaje del dominio de aplicación.
- Pueden no ser entendidos por los ingenieros de software que desarrollan el sistema.

- **Implicación / Conocimiento tácito**

- Los especialistas del dominio entienden el área tan bien que no consideran necesario explicar los requerimientos del dominio
- Las personas no están conscientes del conocimiento tácito que poseen y no lo expresan a los otros.

Ejercicio:

Considere un sistema de “Casa Inteligente”, el cual consiste en una combinación de hardware y software, en el cual una o mas computadoras controlan varios aspectos de la casa, incluyendo control de clima, seguridad, entretenimiento, entre otros.

Para este sistema defina:

- Dos objetivos que el sistema deba realizar y a partir de estos:
- Tres requerimientos funcionales
- Dos requerimientos No Funcionales de cada tipo (6 en total)
- Tres requerimientos de dominio **(tarea)**.



# ¿Que son los requerimientos ?

- Los requerimientos para un sistema computacional son los servicios proporcionados por el sistema y sus restricciones operativas. (sommerville, 2005)
- Los requerimientos pueden variar desde una declaración abstracta de alto nivel o una definición detallada y formal de una función del sistema, esto depende de las necesidades de los stakeholders.

**Algunos de los problemas que surgen durante el proceso de ingeniería de requerimientos son resultado de no hacer una clara separación entre estos diferentes niveles de descripción.**

*Fuente: Ingeniera de software, sommerville*

## Distintos “Requerimientos”...

Objetivos vs Requerimientos.

- Un **objetivo** todavía no es un requerimiento. Los objetivos son utilizados para guiar el proceso de IR. Pueden ser utilizados para identificar los diversos requerimientos.

Ejemplo:


“Implementar un sistema de acceso a biblioteca que funcione en un dispositivo móvil”

Es un objetivo no un requerimiento.

De acuerdo al estándar IEEE 830 un requerimiento es:

**Una condición o capacidad que debe estar presente en un sistema o componentes de un sistema para satisfacer un contrato, estándar, especificación o cualquier otro documento formal.**

**Una representación documentada de una condición o necesidad de un sistema.**



Todos los requerimientos deben ser verificables.

## Ejemplos de requerimientos:

1. El sistema debe mantener una serie de registros de las búsquedas de material de la biblioteca, incluyendo libros, periódicos, revistas y discos compactos, realizadas por el usuario.
2. El sistema debe permitir que los usuarios busquen un objeto por título, autor o ISBN
3. La interfaz de usuario del sistema debe ser implementada usando un navegador Web
4. El sistema debe mostrar información al usuario en menos de 1 minuto.
5. El sistema debe estar disponible en al menos tres sistemas operativos móviles existentes.

Estos requerimientos son de diferentes tipos:

- 1.- Muy general, como el mostrado en 1**
- 2.- Requerimientos funcionales, como el 2, que define una parte de la funcionalidad del sistema.**
- 3.- Requerimientos de implementación, como el 3.**
- 4. Requerimientos de desempeño, como el 4.**
- 5.- Requerimientos de usabilidad.**

Los requerimientos pueden describir alguno de los siguientes puntos:

- **Una facilidad a nivel de usuario.**

Ejemplo: El procesador de texto debe incluir un verificador de ortografía y una función de corrección

- **Una propiedad muy general del sistema.**

Ejemplo: El sistema debe asegurar que la información personal nunca se haga disponible sin autorización.

- **Una restricción específica del sistema**

Ejemplo: El sensor debe ser presionado diez veces por segundo.

- **Una restricción para el desarrollo del sistema**

Ejemplo: El sistema debe ser desarrollado usando Android.

- **Como realizar una operación**

Ejemplo: La cantidad de registros debe ser obtenida a través de las fechas de acceso, sumando cada una.

Tarea

Tipos de requerimientos

## Clasificación de requerimientos (1): Dominios.

Debido a la diversidad de los requerimientos, Sommerville (2005), sugiere organizarlos en tres dominios:

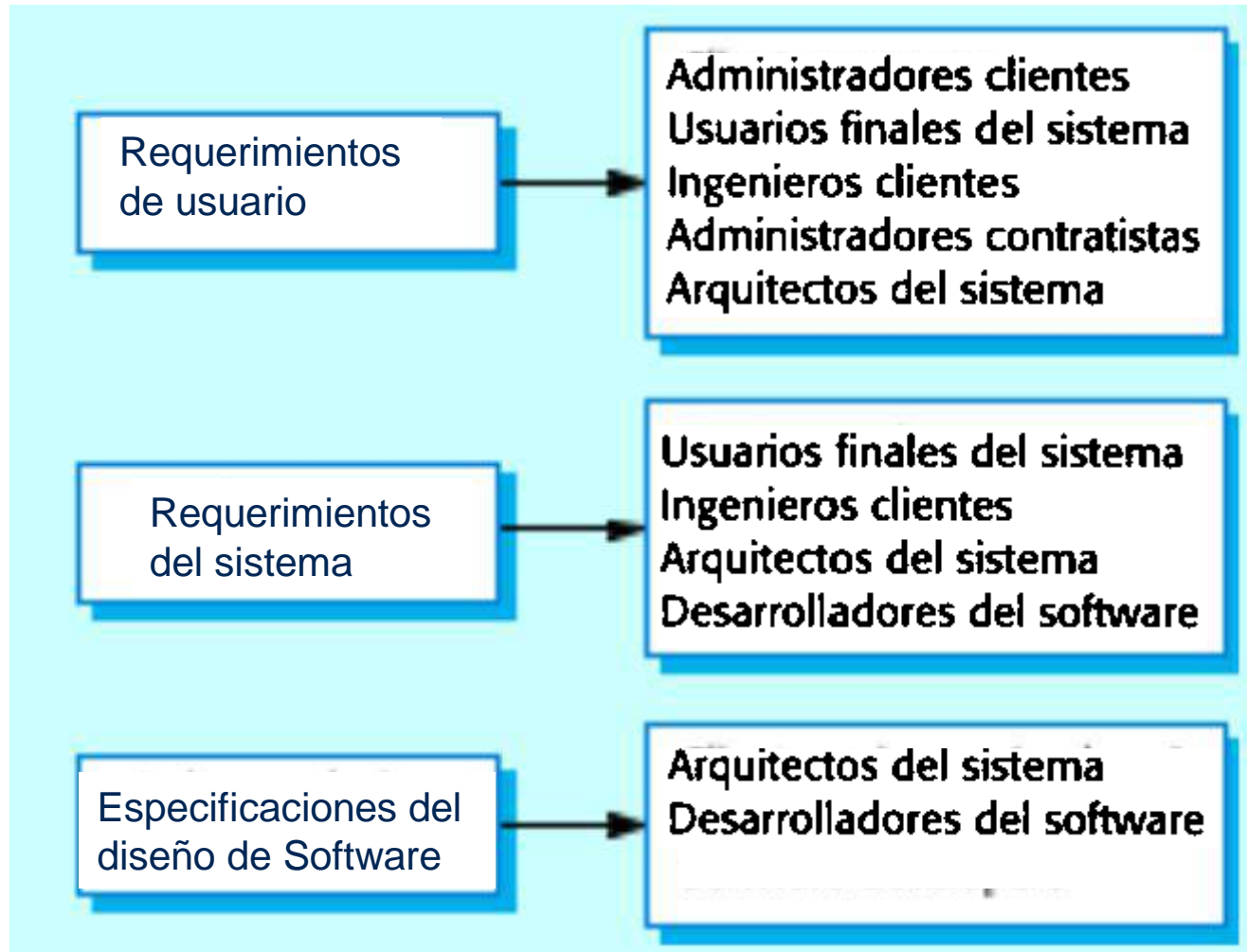
1. Requerimientos del usuario
2. Requerimientos del sistema
3. Especificaciones del diseño de software.

1.- **Requerimientos del usuario**→ Son declaraciones en lenguaje natural y en diagramas informales, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar.

2.- **Requerimientos del sistema**→ Establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requerimientos del sistema debe ser **preciso**. (Se incluye en el contrato)

3.- **Especificaciones del diseño de software**→ Se obtienen de las especificaciones del sistema. Están definidas de manera formal en el documentos de Especificaciones de Requerimientos del Software.

El dominio de los requerimientos se refiere a quien van dirigidos:



## Ejemplo de requerimientos

Sistema para el control de maletas en un aeropuerto.

### **Requerimiento del Usuario:**

- El sistema debe ser capaz de procesar 20 maletas por minuto.

### **Algunos requerimientos del sistema:**

- Cada maleta procesada debe lanzar un evento de almacenamiento.
- El sistema debe ser capaz de manejar 20 eventos de almacenamiento por minuto.

### **Especificaciones del sistema asociadas:**

1.2 El sistema debe ser capaz de procesar 20 eventos de almacenamiento por minuto en modo operacional

1.2.1 Si ocurren más de 20 eventos de almacenamiento en un minuto, el sistema debe ....

1.2.2 [mas excepciones...]



Ejemplo:

## **Sistema de administración de pacientes para apoyar la atención a la salud mental (MHC-PMS)**

### **Definición del requerimiento del usuario**

1. El MHC-PMS elaborará mensualmente informes administrativos que revelen el costo de los medicamentos prescritos por cada clínica durante ese mes.

El requerimiento del usuario es muy general.

### **Especificación de los requerimientos del sistema**

- 1.1 En el último día laboral de cada mes se redactará un resumen de los medicamentos prescritos, su costo y las clínicas que los prescriben.
- 1.2 El sistema elaborará automáticamente el informe que se imprimirá después de las 17:30 del último día laboral del mes.
- 1.3 Se realizará un reporte para cada clínica junto con los nombres de cada medicamento, el número de prescripciones, las dosis prescritas y el costo total de los medicamentos prescritos.
- 1.4 Si los medicamentos están disponibles en diferentes unidades de dosis (por ejemplo, 10 mg, 20 mg) se harán informes por separado para cada unidad de dosis.
- 1.5 El acceso a los informes de costos se restringirá a usuarios autorizados en la lista de control de acceso administrativo.

Los requerimientos del sistema ofrecen información más específica sobre los servicios y las funciones del sistema que se implementará

## Clasificación de requerimientos (2): Tipos.

Otra clasificación de los requerimientos:

- Funcionales
- No Funcionales
- Del Dominio

### **1. *Requerimientos funcionales.***

Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares.

En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.

Representación:

Lenguaje natural,

Modelos visuales,

Métodos formales

**Los requerimiento Funcionales definen:**

Cuáles **entradas** debe aceptar el sistema

Cuáles **salidas** debe producir el sistema

Qué **datos** debe almacenar el sistema que utilizarán otros sistemas

Qué **operaciones** debe realizar el sistema

La **sincronización** y **cronometraje** de las actividades anteriores.

**Ejemplos para el sistema de control de maletas**

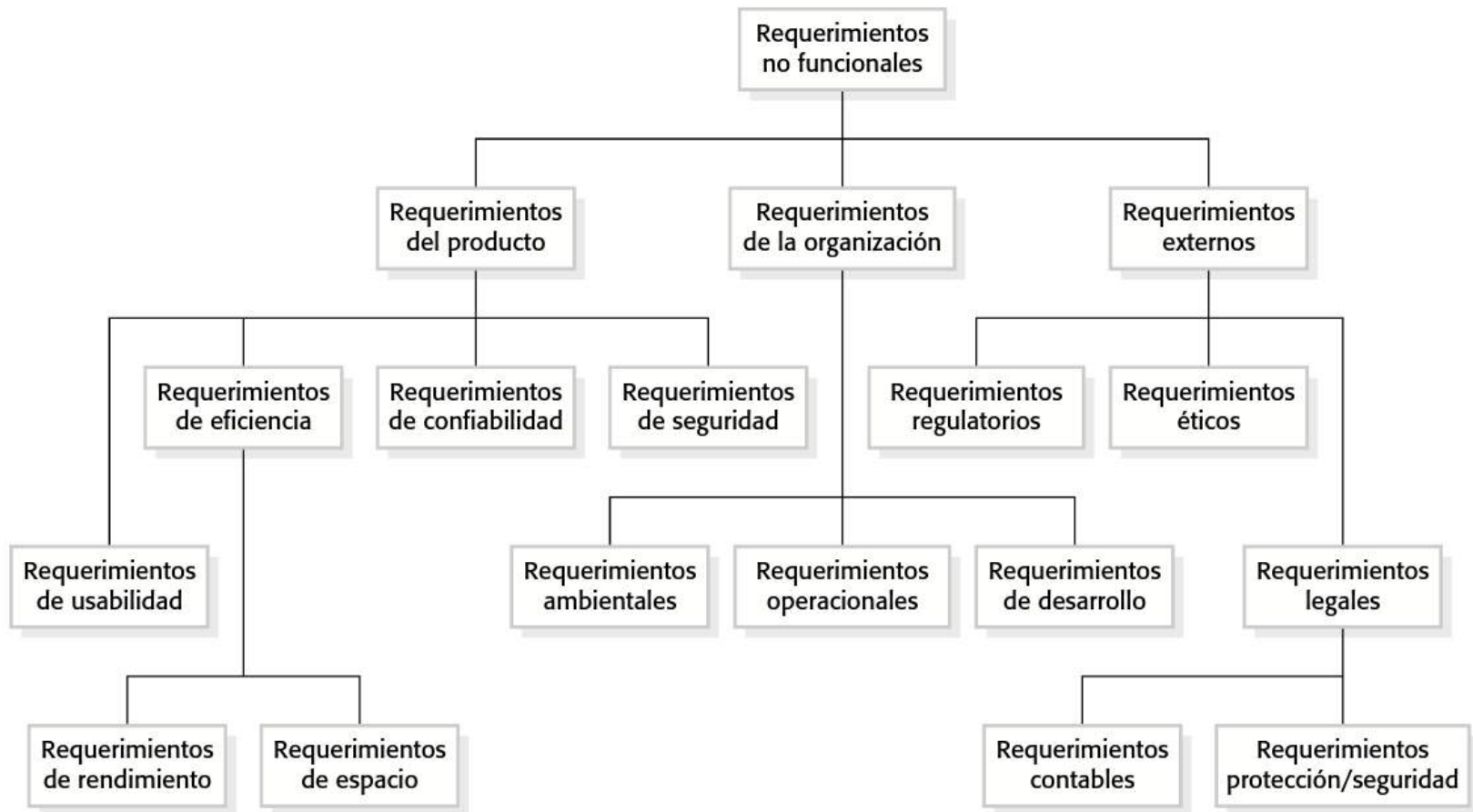
- El sistema debe manejar hasta 20 maletas por segundo
- Si el suministro de corriente falla, el sistema debe apagarse de manera ordenada en menos de 5 segundo.
- Cada usuario del sistema debe identificarse de manera única utilizando su numero de empleado de 8 dígitos.

## **2. Requerimientos No Funcionales. (De calidad, Pohl)**

- Los requerimientos no funcionales no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las **propiedades emergentes** de éste como la *fiabilidad*, *el tiempo de respuesta* y la *capacidad de almacenamiento*.
- De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema.
- Pueden especificar: el rendimiento del sistema, la protección, la disponibilidad, y otras propiedades emergentes.
- A menudo son más críticos que los requerimientos funcionales particulares.

Los requerimientos no funcionales surgen de las necesidades del usuario, debido a las restricciones en el presupuesto, a las políticas de la organización, a la necesidad de interoperabilidad con otros sistemas software o hardware, o a factores externos como regulaciones de seguridad o legislaciones sobre privacidad

## Tipos de requerimientos No Funcionales



Fuente (Ing. de Software, Sommerville)

### Requerimientos de Producto

- Eficiencia
  - Desempeño (Ejemplo: Numero de maletas por minuto)
  - Espacio (Ejemplo: Mínima cantidad de memoria requerida)
- Fiabilidad (tiempo mínimo antes de primera falla)
- Portabilidad (Puede usarse con otro S.O o con otro HW?)
- Usabilidad (Tiempo de entrenamiento requerido)

### Requerimientos Organizacionales

- Entrega (ejemplo: Fecha de entrega, fecha cuando estará operacional, sesiones de entrenamiento, actualizaciones)
- Implementación
- Estándares

### Requerimientos Externos

- Interoperabilidad (Ejemplo: Comunicación con otro equipo).
- Éticos (Ejem: Seguridad para los operadores)
- Legislativos (Ejem: Reglas de privacidad)

Ejemplo:

## **Sistema de administración de pacientes para apoyar la atención a la salud mental (MHC-PMS)**

### **REQUERIMIENTO DEL PRODUCTO**

El MHC-PMS estará disponible en todas las clínicas durante las horas de trabajo normales (lunes a viernes, de 8:30 a 17:30). En cualquier día, los tiempos muertos dentro de las horas laborales normales no rebasarán los cinco segundos.

### **REQUERIMIENTOS DE LA ORGANIZACIÓN**

Los usuarios del sistema MHC-PMS se acreditarán a sí mismos con el uso de la tarjeta de identidad de la autoridad sanitaria.

### **REQUERIMIENTOS EXTERNOS**

Como establece la HStan-03-2006-priv, el sistema implementará provisiones para la privacidad del paciente.

Los requerimientos NF son muy importantes

Si no se consiguen, el sistema es inútil

Pueden ser difíciles de establecer (sobre todo al principio), si son imprecisos puede ser difícil verificarlos.

Algunas recomendaciones para determinarlos:

<b>Propiedad</b>	<b>Medida</b>
Rapidez	Transacciones procesadas por segundo Tiempo de respuesta al usuario y a eventos Tiempo de actualización de la pantalla
Tamaño	K Bytes Número de chips de RAM
Facilidad de Uso	Tiempo de entrenamiento Número de pantallas de ayuda
Fiabilidad	Número promedio entre fallos Probabilidad de no disponibilidad Tasa de Ocurrencia de fallos Disponibilidad
Robustez	Tiempo de reinicio entre fallos Porcentaje de eventos que provocan fallos Probabilidad de corrupción de los datos después de fallos
Portabilidad	Porcentaje de declaraciones dependientes de objetivos Número de sistemas objetivo



### **3. *Requerimientos del Dominio***

- Estos requerimientos se derivan del dominio de aplicación del sistema mas que de las necesidades específicas del usuario.
- Incluyen terminología especializada del dominio o referencias a conceptos del dominio.
- Pueden ser:
  - Requerimientos funcionales nuevos,
  - Restringir los existentes o
  - Establecer como se deben ejecutar cálculos particulares.
- Si no se satisfacen puede ser que el sistema no funcione adecuadamente.

Para redactarlos se requieren conocimientos especializados del dominio

### Ejemplo de RD para un sistema de biblioteca:

1.- Deberá existir una interfaz de usuario estándar para todas las bases de datos que estará basada en el estándar Z39.50

### Ejemplo para un sistema de control de trenes.

La desaceleración del tren se calculará como:

$$D_{\text{tren}} = D_{\text{control}} + D_{\text{gradiente}}$$

donde  $D_{\text{gradiente}}$  es  $9.81\text{ms}^2 * \text{gradiente compensado} / \text{alfa}$  y en donde los valores de  $9.81\text{ms}^2 / \text{alfa}$  se conocen para diferentes tipos de trenes.

## Problemas Relacionados con los Requerimientos del Dominio

- **Comprensibilidad**

- Los requerimientos son expresados en el lenguaje del dominio de aplicación.
- Pueden no ser entendidos por los ingenieros de software que desarrollan el sistema.

- **Implicación / Conocimiento tácito**

- Los especialistas del dominio entienden el área tan bien que no consideran necesario explicar los requerimientos del dominio
- Las personas no están conscientes del conocimiento tácito que poseen y no lo expresan a los otros.

Ejercicio:

Considere un sistema de “Casa Inteligente”, el cual consiste en una combinación de hardware y software, en el cual una o mas computadoras controlan varios aspectos de la casa, incluyendo control de clima, seguridad, entretenimiento, entre otros.

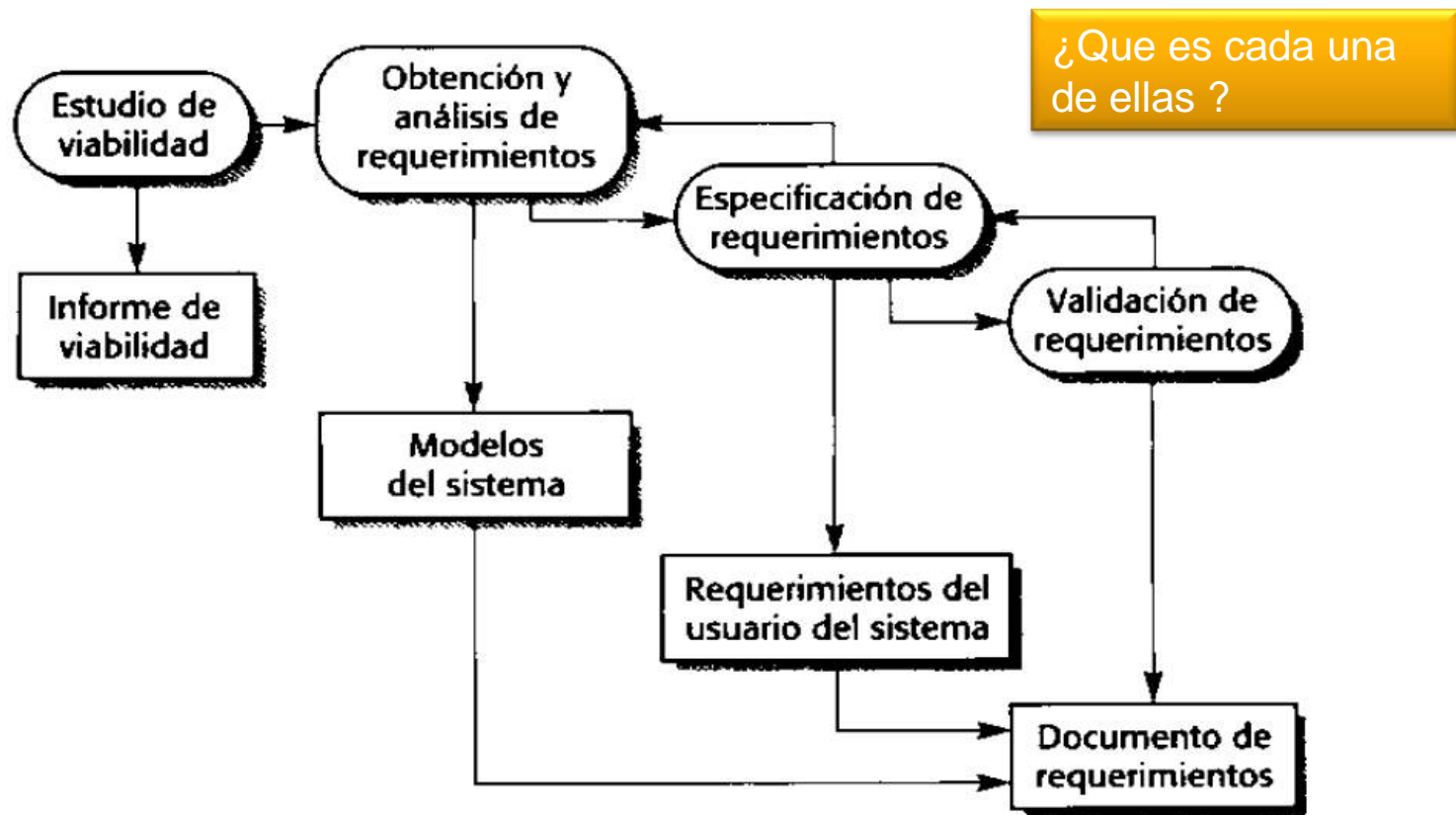
Para este sistema defina:

- Dos objetivos que el sistema deba realizar y a partir de estos:
- Tres requerimientos funcionales
- Tres requerimientos No Funcionales
- Tres requerimientos de dominio **(tarea)**.

Cada organización implementa la IR de manera diferente de acuerdo a sus características.

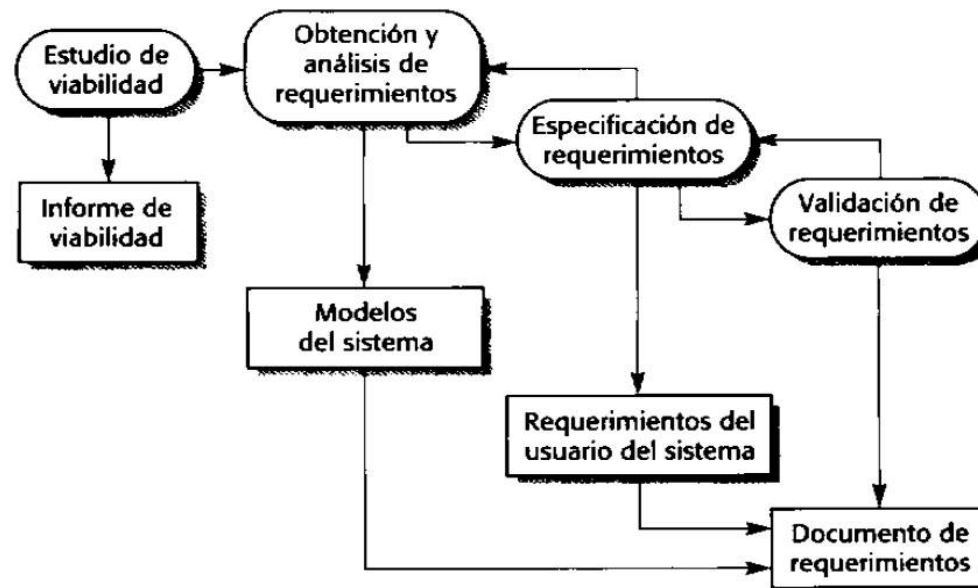
Estas diferencias se acentúan en la descripción detallada de los procesos.

De manera general, en un nivel abstracto la mayoría de los procesos de IR siguen las mismas etapas:

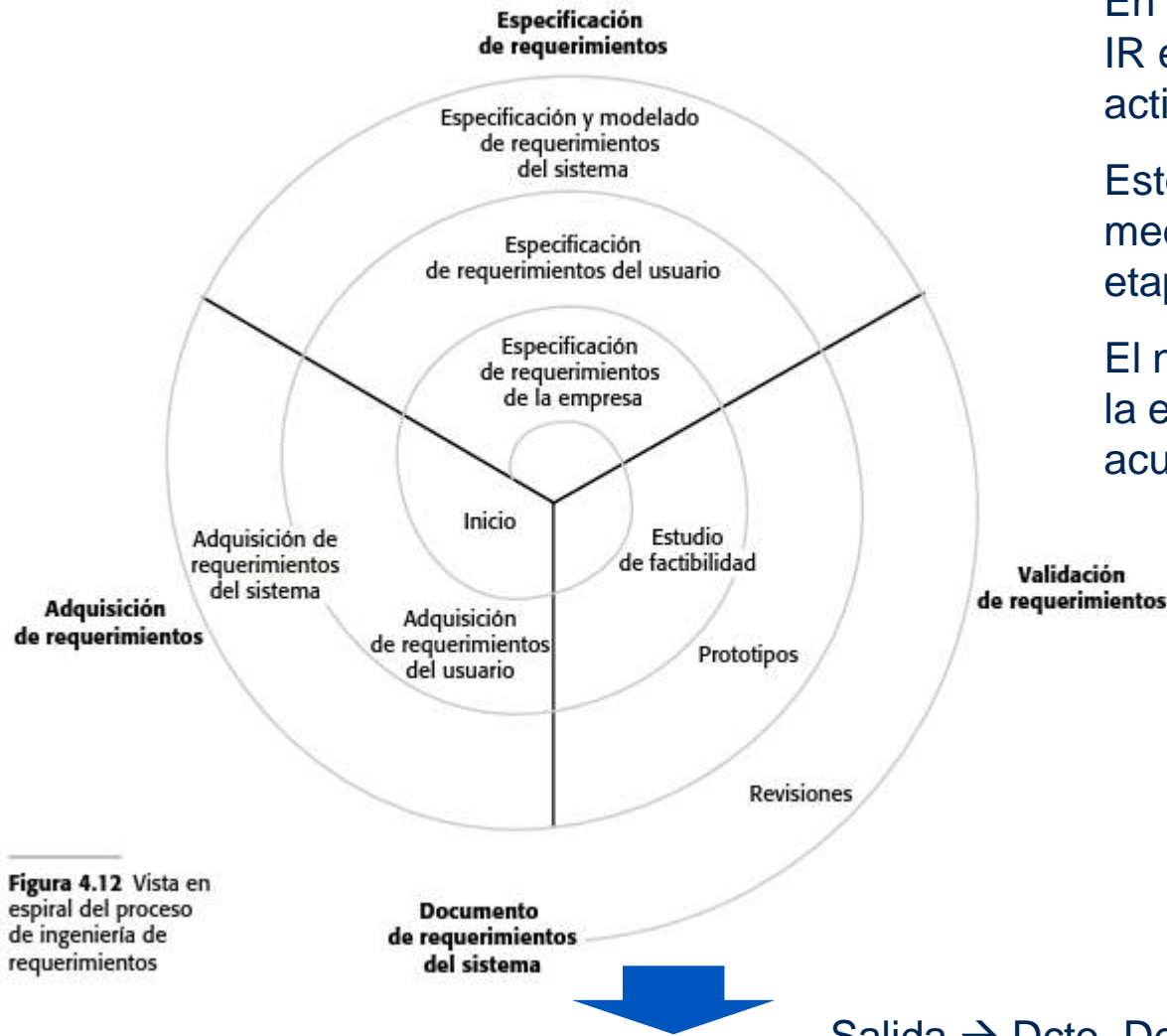


Los procesos de ingeniería de Requerimientos incluyen las siguientes **etapas básicas**

- \*Estudio de viabilidad o factibilidad → Determina si el sistema es útil para la empresa
- \*Obtención y análisis de requerimientos → Descubrimiento de los requerimientos
- \*Especificación de requerimientos → Transformación de requerimientos en estándares.
- \*Validación de requerimientos. → Verificar que los requerimientos realmente definen el sistema que el cliente requiere



## Perspectiva alternativa en espiral.



**Figura 4.12** Vista en espiral del proceso de ingeniería de requerimientos

En la práctica el proceso de IR es un proceso iterativo con actividades entrelazadas.

Esto se puede modelar por medio de una espiral con tres etapas.

El número de iteraciones en la espiral es variable de acuerdo a la organización.

Salida → Dcto. De requerimientos del sistema

## Etapas del proceso de IR

### 1.- Estudio de factibilidad (viabilidad):

#### **Entradas:**

- Conjunto de requerimientos de negocio preliminares
- Descripción resumida del sistema
- Contribución pretendida del sistema a los procesos del negocio

#### **Salida:**

Informe que indique si se debe realizar o no el sistema.



**Un estudio de viabilidad es un estudio corto y orientado a resolver varias cuestiones:**

- 1. ¿Contribuye el sistema a los objetivos generales de la organización?**
- 2. ¿Se puede implementar el sistema utilizando la tecnología actual y dentro de las restricciones de coste y tiempo?**
- 3. ¿Puede integrarse el sistema con otros sistemas existentes en la organización?**




## Estudio de viabilidad

Llevar a cabo un estudio de viabilidad comprende la evaluación y recopilación de la información, y la redacción de informes. La fase de evaluación de la información identifica la información requerida para contestar las tres preguntas anteriores. Una vez que dicha información se ha identificado, se debería hablar con las fuentes de información para descubrir las respuestas a estas preguntas. Algunos ejemplos de preguntas posibles son:

1. ¿Cómo se las arreglaría la organización si no se implementara este sistema?
2. ¿Cuáles son los problemas con los procesos actuales y cómo ayudaría un sistema nuevo a aliviarlos?
3. ¿Cuál es la contribución directa que hará el sistema a los objetivos y requerimientos del negocio?
4. ¿La información se puede obtener y transferir a otros sistemas de la organización?
5. ¿Requiere el sistema tecnología que no se ha utilizado previamente en la organización?
6. ¿A qué debe ayudar el sistema y a qué no necesita ayudar?

## Estudio de factibilidad

### Fuentes

- 
- Jefes de departamentos donde se utilizara el sistema
  - Ingenieros de software que están familiarizados
  - Expertos en tecnología
  - Expertos en el área
  - Usuarios finales

### Informe final:

- Recomendación para realizar o no el sistema
- Proponer cambios en el alcance, presupuesto y/o agenda
- Sugerir requerimientos adicionales

## Etapas del proceso de IR

## 2.- Adquisición y análisis de requerimientos.

Se trabaja con los *stakeholders* para definir:

- Dominio de aplicación
- Servicios que debe proporcionar el sistema
- Desempeño requerido
- Restricciones, etc.



En su proyecto ¿quienes son los stakeholders?

¿Cuál es el dominio de aplicación ?

1. *Descubrimiento de requerimientos* Éste es el proceso de interactuar con los participantes del sistema para descubrir sus requerimientos. También los requerimientos de dominio de los participantes y la documentación se descubren durante esta actividad. Existen numerosas técnicas complementarias que pueden usarse para el descubrimiento de requerimientos, las cuales se estudian más adelante en esta sección.

2. *Clasificación y organización de requerimientos* Esta actividad toma la compilación no estructurada de requerimientos, agrupa requerimientos relacionados y los organiza en grupos coherentes. La forma más común de agrupar requerimientos es usar un modelo de la arquitectura del sistema, para identificar subsistemas y asociar los requerimientos con cada subsistema. En la práctica, la ingeniería de requerimientos y el diseño arquitectónico no son actividades separadas completamente.

## 2.- Adquisición y análisis de requerimientos.

### Razones que dificultan el proceso de adquisición de requerimientos

1. Los participantes con frecuencia no saben lo que quieren de un sistema de cómputo, excepto en términos muy generales; pueden encontrar difícil articular qué quieren que haga el sistema; pueden hacer peticiones inalcanzables porque no saben qué es factible y qué no lo es.
2. Los participantes en un sistema expresan naturalmente los requerimientos con sus términos y conocimientos implícitos de su trabajo. Los ingenieros de requerimientos, sin experiencia en el dominio del cliente, podrían no entender dichos requerimientos.
3. Diferentes participantes tienen distintos requerimientos y pueden expresarlos en variadas formas. Los ingenieros de requerimientos deben descubrir todas las fuentes potenciales de requerimientos e identificar similitudes y conflictos.
4. Factores políticos llegan a influir en los requerimientos de un sistema. Los administradores pueden solicitar requerimientos específicos del sistema, porque éstos les permitirán aumentar su influencia en la organización.
5. El ambiente económico y empresarial donde ocurre el análisis es dinámico. Inevitablemente cambia durante el proceso de análisis. Puede cambiar la importancia de requerimientos particulares; o bien, tal vez surjan nuevos requerimientos de nuevos participantes a quienes no se consultó originalmente.

## Etapas del proceso de IR

## 2.- Adquisición y análisis de requerimientos.

4. *Especificación de requerimientos* Los requerimientos se documentan e ingresan en la siguiente ronda de la espiral. Pueden producirse documentos de requerimientos formales o informales



3. *Priorización y negociación de requerimientos* Inevitablemente, cuando intervienen diversos participantes, los requerimientos entrarán en conflicto. Esta actividad se preocupa por priorizar los requerimientos, así como por encontrar y resolver conflictos de requerimientos mediante la negociación. Por lo general, los participantes tienen que reunirse para resolver las diferencias y estar de acuerdo con el compromiso de los requerimientos.

Tarea → Próxima clase

Preparar una presentación (equipos de 2 o individual) sobre las siguientes técnicas de descubrimiento de requerimientos:

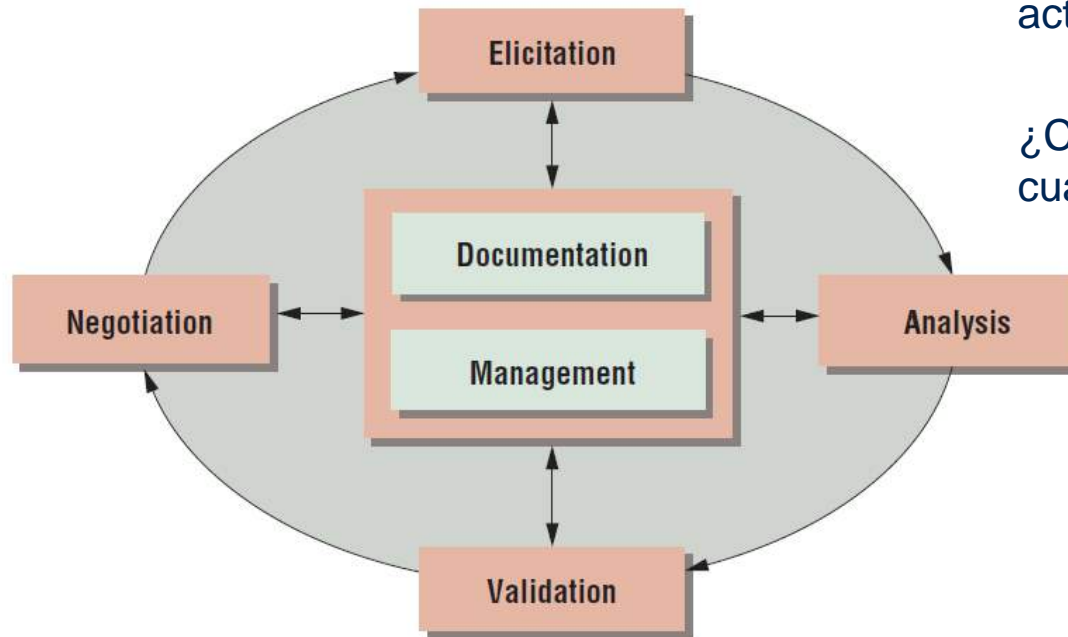
- Entrevistas (incluir tipos de entrevistas)
- Casos de uso
- Etnografía
- Puntos de vista
- Joint application design (JAD)
- Escenarios
- Prototipos

La presentación debe incluir : Descripción de la técnica, ejemplos de su uso para la obtención de requ., ventajas y desventajas.

Según Sommerville:

¿Cuales son los cambios en la actualidad?

¿Cuales son las tendencias en cuanto a IR ?



## Etapas del proceso de IR

### 3- Especificación de requerimientos.

La especificación de requerimientos es el proceso de escribir, en un documento de requerimientos, los requerimientos del usuario y del sistema. De manera ideal, los requerimientos del usuario y del sistema deben ser claros, sin ambigüedades, fáciles de entender, completos y consistentes. Esto en la práctica es difícil de lograr, pues los participantes interpretan los requerimientos de formas diferentes y con frecuencia en los requerimientos hay conflictos e inconsistencias inherentes.

Requerimientos del usuario → Se expresan en lenguaje natural

Incluyen requerimientos funcionales y no funcionales



## Etapas del proceso de IR

## 3- Especificación de requerimientos.

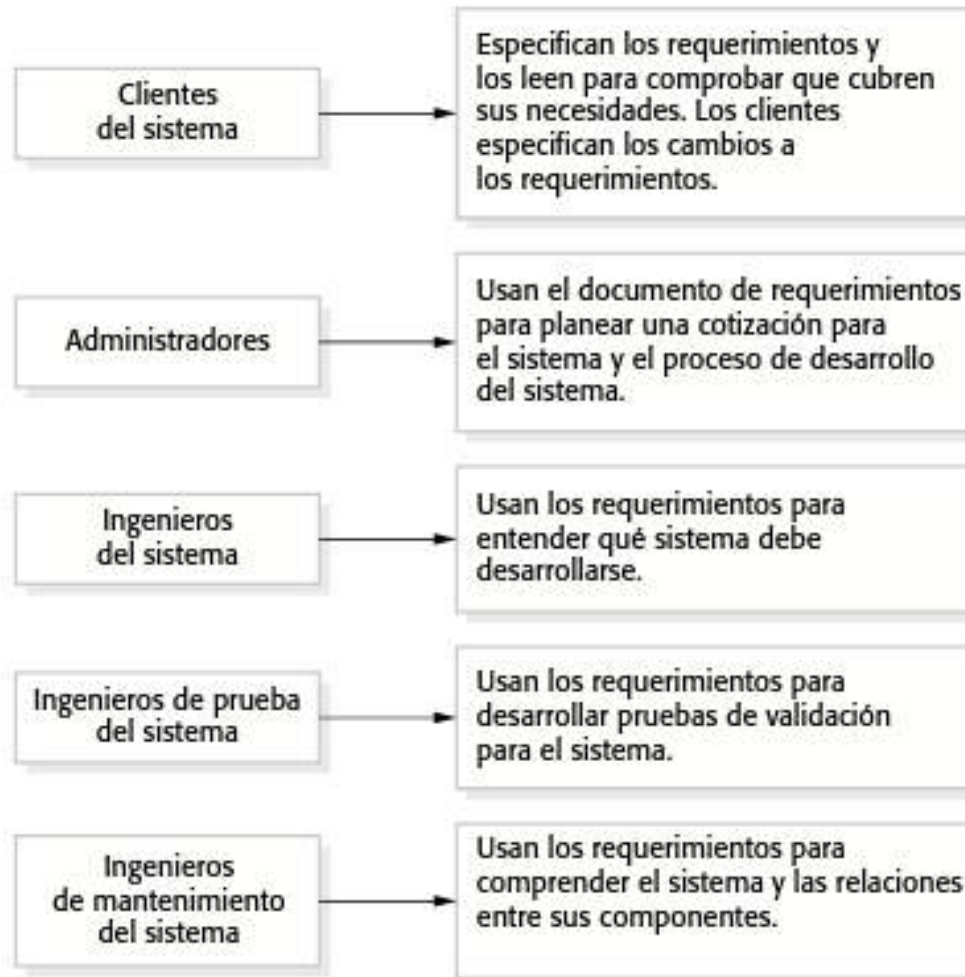
Requerimientos del sistema → describen el comportamiento del sistema, pueden requerir diferentes técnicas de representación.

Notación	Descripción
Enunciados en lenguaje natural	Los requerimientos se escriben al usar enunciados numerados en lenguaje natural. Cada enunciado debe expresar un requerimiento.
Lenguaje natural estructurado	Los requerimientos se escriben en lenguaje natural en una forma o plantilla estándar. Cada campo ofrece información de un aspecto del requerimiento.
Lenguajes de descripción de diseño	Este enfoque usa un lenguaje como un lenguaje de programación, pero con características más abstractas para especificar los requerimientos al definir un modelo operacional del sistema. Aunque en la actualidad este enfoque se usa raras veces, aún tiene utilidad para especificaciones de interfaz.
Anotaciones gráficas	Los modelos gráficos, complementados con anotaciones de texto, sirven para definir los requerimientos funcionales del sistema; los casos de uso del UML y los diagramas de secuencia se emplean de forma común.
Especificaciones matemáticas	Dichas anotaciones se basan en conceptos matemáticos como máquinas o conjuntos de estado finito. Aunque tales especificaciones sin ambigüedades pueden reducir la imprecisión en un documento de requerimientos, la mayoría de los clientes no comprenden una especificación formal. No pueden comprobar que representa lo que quieren y por ello tienen reticencia para aceptarlo como un contrato de sistema.

Capítulo	Descripción
Prefacio	Debe definir el número esperado de lectores del documento, así como describir su historia de versiones, incluidas las causas para la creación de una nueva versión y un resumen de los cambios realizados en cada versión.
Introducción	Describe la necesidad para el sistema. Debe detallar brevemente las funciones del sistema y explicar cómo funcionará con otros sistemas. También tiene que indicar cómo se ajusta el sistema en los objetivos empresariales o estratégicos globales de la organización que comisiona el software.
Glosario	Define los términos técnicos usados en el documento. No debe hacer conjeturas sobre la experiencia o la habilidad del lector.
Definición de requerimientos del usuario	Aquí se representan los servicios que ofrecen al usuario. También, en esta sección se describen los requerimientos no funcionales del sistema. Esta descripción puede usar lenguaje natural, diagramas u otras observaciones que sean comprensibles para los clientes. Deben especificarse los estándares de producto y proceso que tienen que seguirse.
Arquitectura del sistema	Este capítulo presenta un panorama de alto nivel de la arquitectura anticipada del sistema, que muestra la distribución de funciones a través de los módulos del sistema. Hay que destacar los componentes arquitectónicos que sean de reutilización.
Especificación de requerimientos del sistema	Debe representar los requerimientos funcionales y no funcionales con más detalle. Si es preciso, también pueden detallarse más los requerimientos no funcionales. Pueden definirse las interfaces a otros sistemas.
Modelos del sistema	Pueden incluir modelos gráficos del sistema que muestren las relaciones entre componentes del sistema, el sistema y su entorno. Ejemplos de posibles modelos son los modelos de objeto, modelos de flujo de datos o modelos de datos semánticos.
Evolución del sistema	Describe los supuestos fundamentales sobre los que se basa el sistema, y cualquier cambio anticipado debido a evolución de hardware, cambio en las necesidades del usuario, etc. Esta sección es útil para los diseñadores del sistema, pues los ayuda a evitar decisiones de diseño que restringirían probablemente futuros cambios al sistema.
Apéndices	Brindan información específica y detallada que se relaciona con la aplicación a desarrollar; por ejemplo, descripciones de hardware y bases de datos. Los requerimientos de hardware definen las configuraciones, mínima y óptima, del sistema. Los requerimientos de base de datos delimitan la organización lógica de los datos usados por el sistema y las relaciones entre datos.
Índice	Pueden incluirse en el documento varios índices. Así como un índice alfabético normal, uno de diagramas, un índice de funciones, etcétera.

Documento de requerimientos basado en el estándar IEEE830

## Usuarios del documento de requerimientos:



Fuente: Ing. de software, sommerville,  
9na edición

## Etapas del proceso de IR

### 4- Validación de requerimientos.

Es el proceso de verificar que los requerimientos definan realmente lo que el cliente desea

Esta etapa es muy importante, ya que resulta muy costoso corregir errores en el sistema implementado !

**Debe realizarse en cada etapa del proceso para asegurarse que lo que se esta realizando se esta haciendo bien.**

## Etapas del proceso de IR

### 4- Validación de requerimientos.

Durante el proceso de validación deben realizarse varias comprobaciones.

1. *Comprobaciones de validez* Un usuario quizá crea que necesita un sistema para realizar ciertas funciones. Sin embargo, con mayor consideración y análisis se logra identificar las funciones adicionales o diferentes que se requieran. Los sistemas tienen diversos participantes con diferentes necesidades, y cualquier conjunto de requerimientos es inevitablemente un compromiso a través de la comunidad de participantes.
2. *Comprobaciones de consistencia* Los requerimientos en el documento no deben estar en conflicto. Esto es, no debe haber restricciones contradictorias o descripciones diferentes de la misma función del sistema.

## Etapas del proceso de IR

### 4- Validación de requerimientos.

3. *Comprobaciones de totalidad* El documento de requerimientos debe incluir requerimientos que definan todas las funciones y las restricciones pretendidas por el usuario del sistema.
4. *Comprobaciones de realismo* Al usar el conocimiento de la tecnología existente, los requerimientos deben comprobarse para garantizar que en realidad pueden implementarse. Dichas comprobaciones también tienen que considerar el presupuesto y la fecha para el desarrollo del sistema.
5. *Verificabilidad* Para reducir el potencial de disputas entre cliente y contratista, los requerimientos del sistema deben escribirse siempre de manera que sean verificables. Esto significa que usted debe ser capaz de escribir un conjunto de pruebas que demuestren que el sistema entregado cumpla cada requerimiento especificado.

## Etapas del proceso de IR

### 4- Validación de requerimientos.

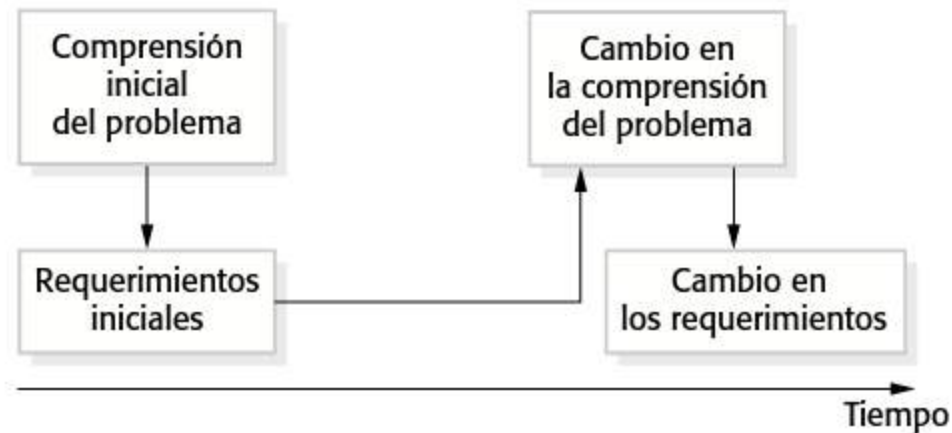
Hay algunas técnicas de validación de requerimientos que se usan individualmente o en conjunto con otras:

1. *Revisiones de requerimientos* Los requerimientos se analizan sistemáticamente usando un equipo de revisores que verifican errores e inconsistencias.
2. *Creación de prototipos* En esta aproximación a la validación, se muestra un modelo ejecutable del sistema en cuestión a los usuarios finales y clientes. Así, ellos podrán experimentar con este modelo para constatar si cubre sus necesidades reales.
3. *Generación de casos de prueba* Los requerimientos deben ser comprobables. Si las pruebas para los requerimientos se diseñan como parte del proceso de validación, esto revela con frecuencia problemas en los requerimientos. Si una prueba es difícil o imposible de diseñar, esto generalmente significa que los requerimientos serán difíciles de implementar, por lo que deberían reconsiderarse. El desarrollo de pruebas a partir de los requerimientos del usuario antes de escribir cualquier código es una pieza integral de la programación extrema.

## Etapas del proceso de IR

### 6- Administración de requerimientos.

Los requerimientos iniciales pueden cambiar, es necesario llevar a cabo la administración/ actualización de requerimientos



La administración de requerimientos tiene dos etapas:

- 1.- Planeación de la administración de requerimientos.
2. Administración del cambio de requerimientos.



## 6- Administración de requerimientos.

### Planeación de la administración de requerimientos

Establece el nivel de detalle que se desea en la administración de requerimientos. En esta etapa se define lo siguiente:

1. *Identificación de requerimientos* Cada requerimiento debe identificarse de manera exclusiva, de forma que pueda tener referencia cruzada con otros requerimientos y usarse en las evaluaciones de seguimiento.
2. *Un proceso de administración del cambio* Éste es el conjunto de actividades que valoran el efecto y costo de los cambios. En la siguiente sección se estudia con más detalle este proceso.
3. *Políticas de seguimiento* Dichas políticas definen las relaciones entre cada requerimiento, así como entre los requerimientos y el diseño del sistema que debe registrarse. La política de seguimiento también tiene que definir cómo mantener dichos registros.
4. *Herramientas de apoyo* La administración de requerimientos incluye el procesamiento de grandes cantidades de información acerca de los requerimientos. Las herramientas disponibles varían desde sistemas especializados de administración de requerimientos, hasta hojas de cálculo y sistemas de bases de datos simples.

## 6- Administración de requerimientos.

La administración de requerimientos necesita apoyo automatizado y herramientas de software, para lo cual deben seleccionarse durante la fase de planeación. Se necesitan herramientas de apoyo para:

1. *Almacenamiento de requerimientos* Los requerimientos tienen que mantenerse en un almacén de datos administrado y seguro, que sea accesible para todos quienes intervienen en el proceso de ingeniería de requerimientos.
2. *Administración del cambio* El proceso de administración del cambio se simplifica si está disponible la herramienta de apoyo activa.
3. *Administración del seguimiento* Como se estudió anteriormente, la herramienta de apoyo para el seguimiento permite la identificación de requerimientos relacionados. Algunas herramientas que están disponibles usan técnicas de procesamiento en lenguaje natural, para ayudar a descubrir posibles relaciones entre los requerimientos.

Sistemas pequeños pueden no requerir estas herramientas

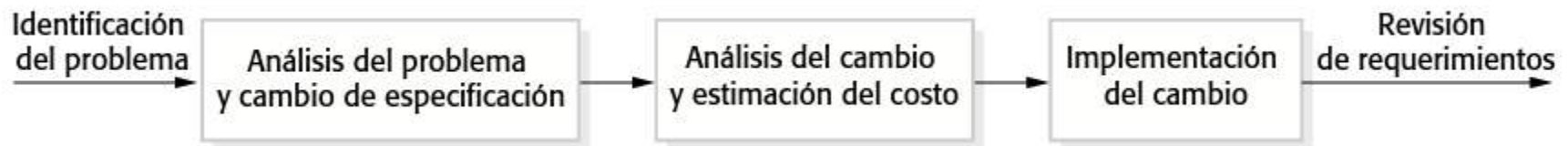
## 6- Administración de requerimientos.

### Administración del cambio de requerimientos

Debe aplicarse a todos los cambios propuestos después de elaborarse y aprobarse el documento de requerimientos.

La administración del cambio es esencial ya que debe determinarse si se justifica o no la realización del cambio

Se tienen tres etapas principales:



## 6- Administración de requerimientos.

1. *Análisis del problema y especificación del cambio* El proceso comienza con la identificación de un problema en los requerimientos o, en ocasiones, con una propuesta de cambio específica. Durante esta etapa, el problema o la propuesta de cambio se analizan para comprobar que es válida. Este análisis retroalimenta al solicitante del cambio, quien responderá con una propuesta de cambio de requerimientos más específica, o decidirá retirar la petición.
2. *Análisis del cambio y estimación del costo* El efecto del cambio propuesto se valora usando información de seguimiento y conocimiento general de los requerimientos del sistema. El costo por realizar el cambio se estima en términos de modificaciones al documento de requerimientos y, si es adecuado, al diseño y la implementación del sistema. Una vez completado este análisis, se toma una decisión acerca de si se procede o no con el cambio de requerimientos.
3. *Implementación del cambio* Se modifican el documento de requerimientos y, donde sea necesario, el diseño y la implementación del sistema. Hay que organizar el documento de requerimientos de forma que sea posible realizar cambios sin reescritura o reorganización extensos. Conforme a los programas, la variabilidad en los documentos se logra al minimizar las referencias externas y al hacer las secciones del documento tan modulares como sea posible. De esta manera, secciones individuales pueden modificarse y sustituirse sin afectar otras partes del documento.

Tarea → Próxima clase

Preparar una presentación sobre las siguientes técnicas de descubrimiento de requerimientos:

- Entrevistas (incluir tipos de entrevistas)
- Casos de uso
- Etnografía
- Puntos de vista
- Joint application design (JAD)
- Escenarios
- Prototipos

La presentación debe incluir : Descripción de la técnica, ejemplos de su uso para la obtención de requ., ventajas y desventajas.



## **INGENIERIA DE REQUERIMIENTOS (IR)**

### **2. Técnicas de recolección (elicitación) de requerimientos**

2.1 Entrevista

2.2 Casos de Uso

2.3 Prototipos

2.4 Etnografía

2.5 JAD (Joint Application Development) y otras

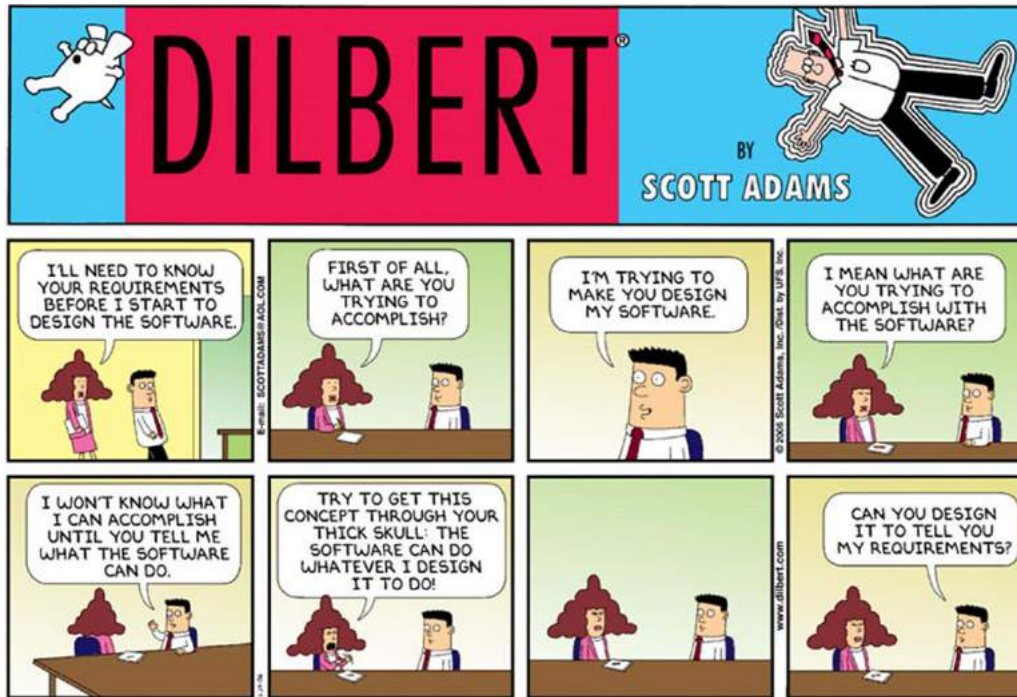
2.6 Análisis y Negociación

## Adquisición y análisis de requerimientos

Adquisición (elicitación) y análisis de requerimientos se refieren a un conjunto de actividades realizadas para descubrir los requerimientos.

Se realiza después del estudio de factibilidad o viabilidad

No consiste solamente en preguntar a los usuarios lo que quieren !



© Scott Adams, Inc./Dist. by UFS, Inc.

Requiere un análisis minucioso de la organización, el dominio de aplicación, y los procesos que el sistema utilizará

## La adquisición de requerimientos debe cubrir cuatro dominios

1.- Entender el dominio de aplicación → Se refiere adquirir conocimiento o comprender el funcionamiento de términos relacionados con la aplicación del sistema. Ejemplo: Para realizar el sistema de biblioteca deberán entenderse algunos términos como índices de clasificación de libros, etc.



2.- Entender el problema → Se deben entender los detalles del problema donde se utilizara el sistema. Ejemplo: Entender la organización particular de una biblioteca

3.- Conocimiento de la organización → Se debe comprender el funcionamiento del sistema dentro de la organización y como interactúa con otros sistemas ya existentes.

4.- Necesidades y restricciones de stakeholders → Se debe comprender las funciones de todos los involucrados, así como el rol de cada uno en el sistema.



La adquisición **efectiva** de requerimientos es muy importante



Si no se descubren los requerimientos reales del usuario, nunca estará satisfecho con el producto final.

El descubrimiento de requerimientos (llamado a veces adquisición de requerimientos) es el proceso de recopilar información sobre el sistema requerido y los sistemas existentes, así como de separar, a partir de esta información, los requerimientos del usuario y del sistema.

Las fuentes de información durante la fase de descubrimiento de requerimientos incluyen documentación, participantes del sistema y especificaciones de sistemas similares.

Debido a las múltiples dimensiones en la obtención de requerimientos se pueden presentar varios problemas:

- 1.- El conocimiento del dominio de aplicación no se obtiene en un solo lugar. Es necesario consultar varias fuentes: libros de textos, manuales operativos, las cabezas de las personas involucradas.
- 2.- Las personas que entienden el problema están muy ocupadas resolviendo el problema como para detenerse a explicarlo (y muchas veces no requieren un nuevo sistema).
- 3.- Los requerimientos pueden estar influenciados por factores políticos u organizacionales, no tanto por necesidades de los usuarios.
- 4.- Los stakeholders no saben exactamente lo que requieren de un sistema de software.
- 5.- Stakeholders diferentes tienen requerimientos diferentes.

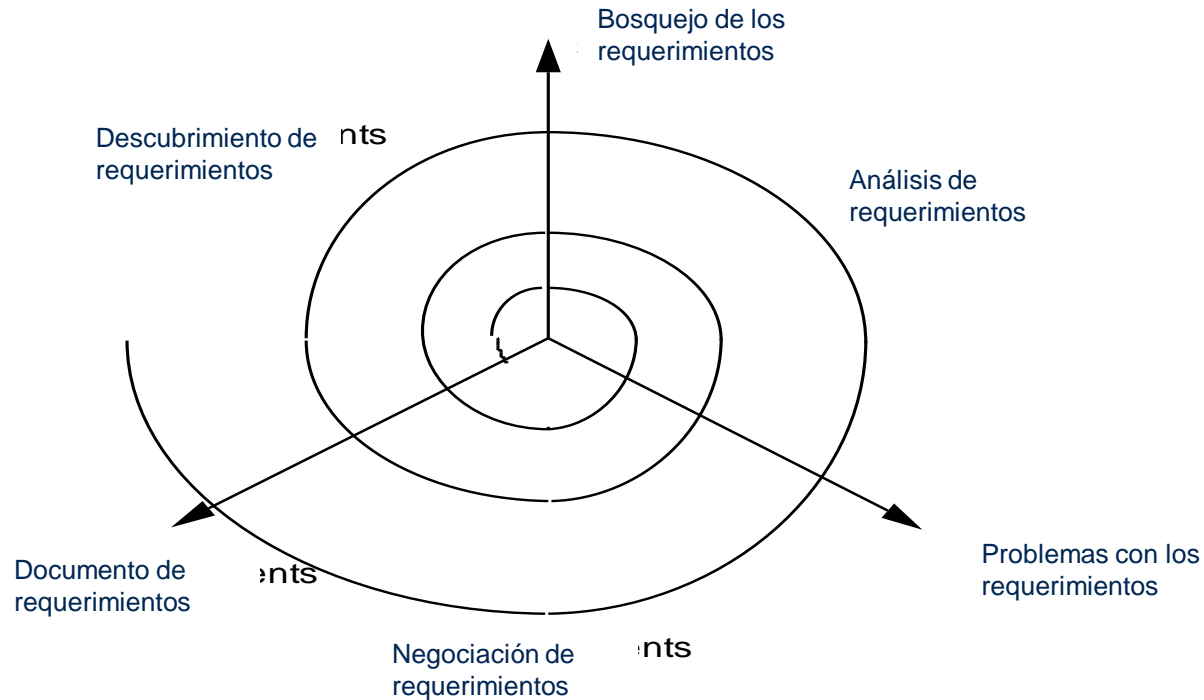
Los *stakeholders* o participantes en el sistema son muy variados.

-Administradores, usuarios finales, participantes externos, certificadores, etc.

Por ejemplo, los participantes que se incluyen para el sistema de información de pacientes en atención a la salud mental son:

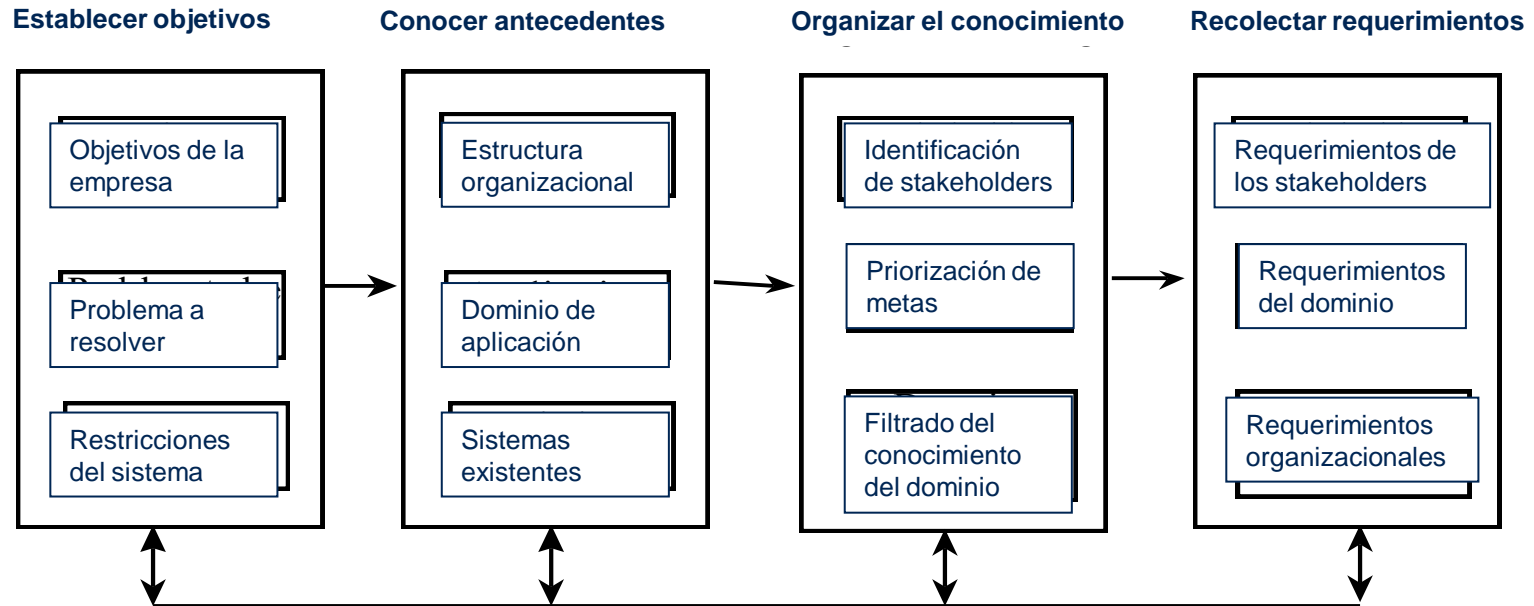
1. Pacientes cuya información se registra en el sistema.
2. Médicos que son responsables de valorar y tratar a los pacientes.
3. Enfermeros que coordinan, junto con los médicos, las consultas y suministran algunos tratamientos.
4. Recepcionistas que administran las citas médicas de los pacientes.
5. Personal de TI que es responsable de instalar y mantener el sistema.
6. Un director de ética médica que debe garantizar que el sistema cumpla con los lineamientos éticos actuales de la atención al paciente.
7. Encargados de atención a la salud que obtienen información administrativa del sistema.
8. Personal de archivo médico que es responsable de garantizar que la información del sistema se conserve, y se implementen de manera adecuada los procedimientos de mantenimiento del archivo.

La adquisición y el análisis de requerimientos están muy ligados. Conforme los requerimientos se descubren requieren ser analizados y negociados. Este proceso puede verse como una espiral.



Típicamente durante el proceso un IR descubre requerimientos, los analiza, y realiza negociaciones con los involucrados, después inicia un nuevo segmento de la espiral. Esto continua hasta que los involucrados están satisfechos con los requerimientos.

Existen varios procesos para llevar a cabo la adquisición y análisis de requerimientos. Un proceso general puede incluir las siguientes etapas:



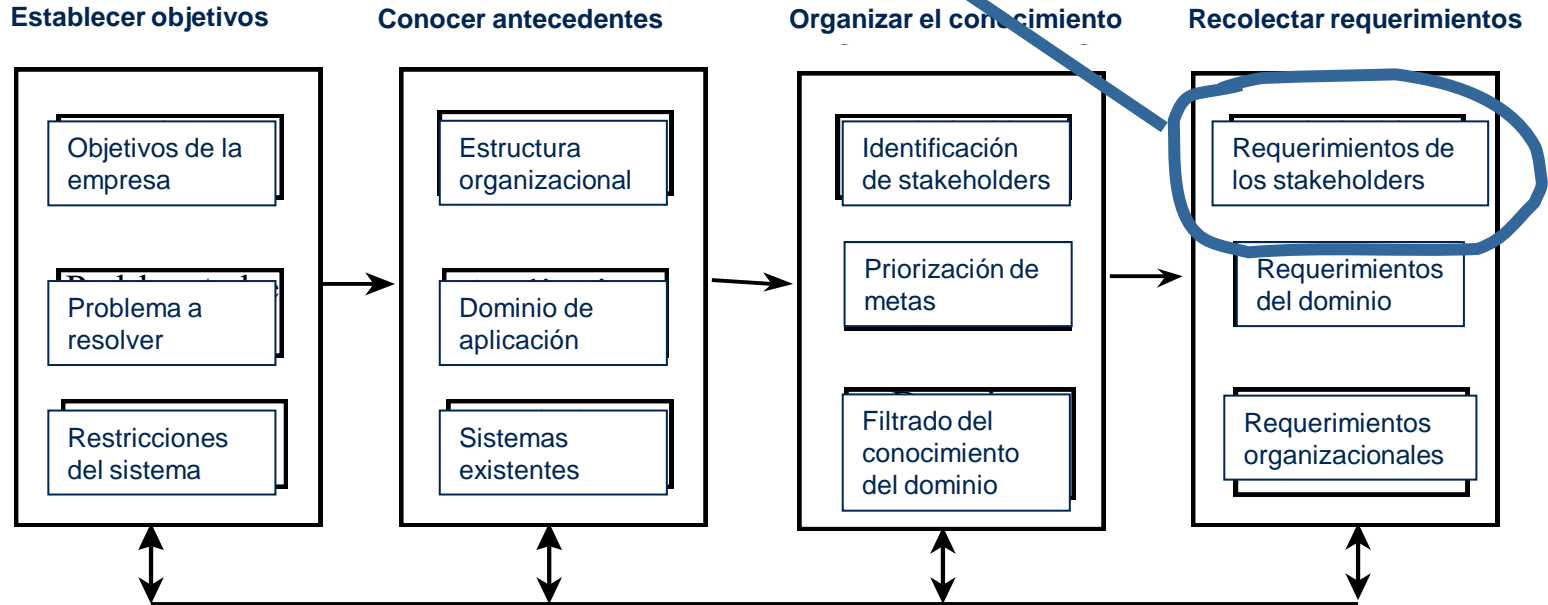
La adquisición de requerimientos abarca entonces cuatro etapas básicas.

- 1.- Determinación de objetivos
- 2.- Conocimiento de antecedentes.
- 3.- Organización de conocimiento
- 4.- Recolectar requerimientos

## Etapas de la adquisición

- Determinación de objetivos
  - Los objetivos organizacionales deben establecerse, incluyendo las metas generales de la organización, una descripción del problema a resolver, la justificación y necesidad del sistema, las restricciones del sistema.
- Conocimiento de antecedentes
  - Empaparse de información sobre el sistema, incluyendo conocer la organización donde se instalara el sistema, el dominio de aplicación y los sistemas existentes.
- Organización del conocimiento
  - Las etapas anteriores generan una gran cantidad de conocimiento el cual debe ser filtrado y organizado. Esto incluye identificar a los *stakeholders*, sus roles en la organización, priorización de metas y descartar conocimiento que no contribuye a los requerimientos del sistema.
- Recolección de requerimientos
  - Se consultan los *stakeholders* del sistema para recolectar requerimientos, utilizando una técnica adecuada.

**Muchos piensan que solo esto es obtención de requerimientos. [enfoque limitado]**



El proceso no es ideal, en la realidad es proceso puede complicarse, sobre todo si no se realizan las primeras etapas, lo que puede generar problemas importante.

## Técnicas para el descubrimiento de requerimientos.

El descubrimiento de requerimientos incluye recabar información sobre:

- El dominio de la aplicación
- El problema específico
- La organización o proceso
- Necesidades de stakeholders.

Se requiere diferentes técnicas para descubrir toda esta información.

Cada técnica puede cubrir uno o varios dominios diferentes.



Técnicas específicas para el descubrimiento de requerimientos.

- Entrevistas (incluir tipos de entrevistas)
- Casos de uso
- Etnografía
- Puntos de vista
- Joint application design (JAD)
- Escenarios
- Prototipos

## Equipos y temas

-De Jesús Asunción Daniel

Temas: Prototipos y Etnografía

-Espinosa Ramírez Eneida

Tema: Casos de uso

-Lizarraga Alcaraz Enrique

Escenarios y JAD

-Lopez Juarez Bonifacio

Temas: Entrevistas (incluir tipos) y Puntos de Vista



## **INGENIERIA DE REQUERIMIENTOS (IR)**

### **3.- Análisis y Presentación de Requerimientos**

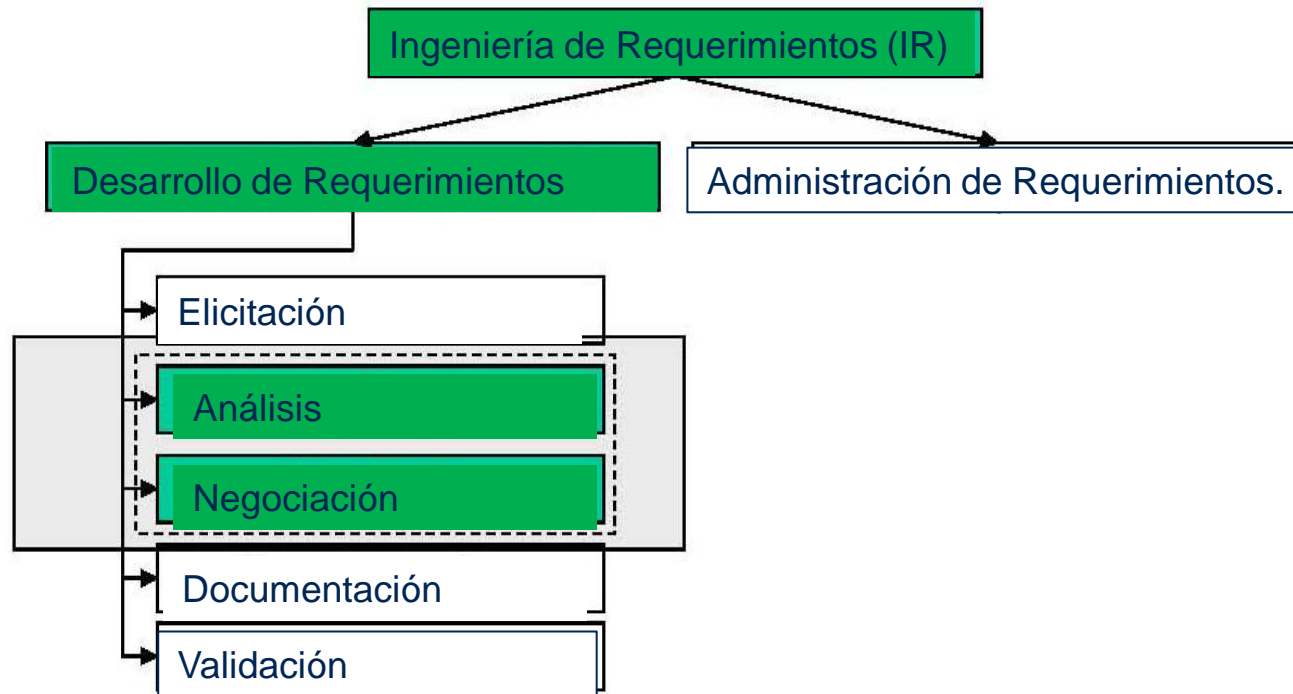
#### **3.1 Análisis y Negociación**

#### **3.2 Enfoques y estándares de escritura de req.**

#### **3.3 Métodos formales**

#### **3.4 Requerimientos para metodologías ágiles**

## Análisis y negociación de requerimientos



- **Análisis** – Proceso de evaluar valor/costo de diferentes requerimientos, identificar dependencias entre requerimientos, etc.
- **Negociación** – Proceso de resolver conflictos entre requerimientos, decidir cual aceptar, determinar prioridades.


## Negociación de requerimientos.

- El objetivo de la negociación de requerimientos es lograr un acuerdo para realizar los cambios que satisfagan a todos los stakeholders.
- La etapa de negociación debe considerar a los diferentes stakeholders para resolver todos los conflictos y traslapes.
  - Los conflictos no son fallas pero reflejan las prioridades y necesidades diferentes de los stakeholders.

### Negociación

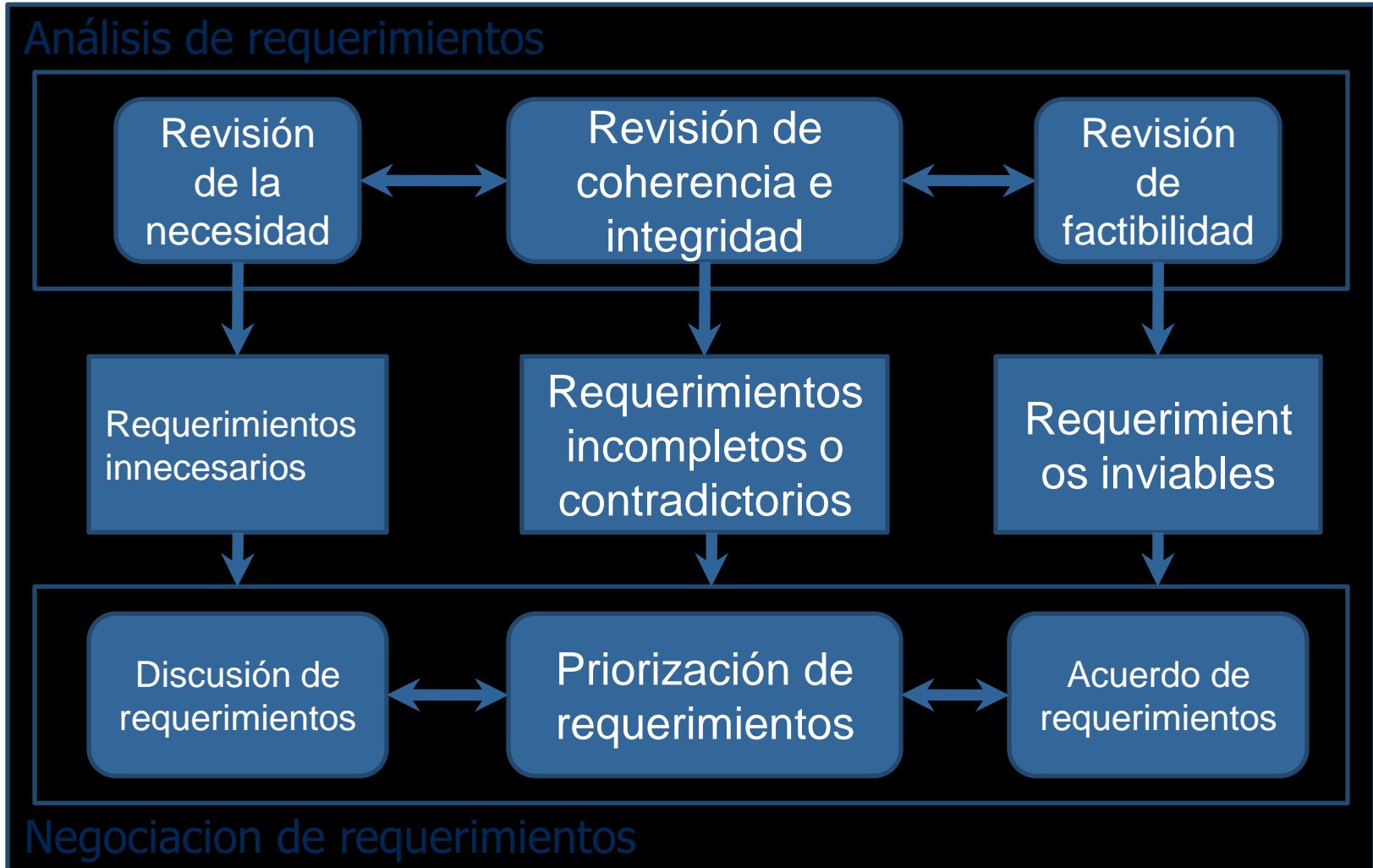
**Entrada** – Conjunto de requerimientos en traslape o con conflictos

**Salida** –Un conjunto de requerimientos en comun acuerdo

- La etapa de negociación esta intercalada con la obtención y el análisis  

- Se descubren problemas y se acuerdan posibles soluciones al momento de obtener los requerimientos.
- Encontrar soluciones aceptables puede resulta una tarea que consuma mucho tiempo.

La negociación de requerimientos es la parte del proceso de IR encargada de:

- 1) Identificar y analizar la causa de cada conflicto
- 2) Resolver los conflictos utilizando las estrategias apropiadas.
- 3) Documentar la resolución de los conflictos

**Actividades del análisis y negociación de requerimientos.**

Los conflictos de requerimientos pueden incluso presentarse en todas las etapas de la IR.

-Durante la obtención de requerimientos, en una lluvia de ideas dos stakeholders pueden tener requerimientos completamente contradictorios entre ellos.

-Al documentar entrevistas realizadas por diferentes personas pueden identificarse requerimientos contradictorios.

-Pueden incluso identificarse en etapas posteriores a la documentación, como la validación y administración de requerimientos.

Es importante analizar los requerimientos en cada etapa del proceso, entre mas tarde se detecten será mas complicado (tiempo/esfuerzo/costo) resolverlo.



## 1) Identificación de conflictos

Existen diferentes tipos de conflictos. Se pueden clasificar como :

**-Conflictos de datos:** Estos conflictos se pueden deber a faltas de información a malentendidos o a interpretaciones diferentes de algún punto.

Ejemplo:

*R4: El reproductor de DVD debe ser capaz de manejar CDs (CD-RW) y DVDs (DVD-RW) regrabables.*

Un *stakeholder* no está de acuerdo con el requerimiento. En su opinión no tiene sentido que el reproductor de DVDs del carro sea capaz de grabar CDs o DVDs

Conflicto → mala interpretación.

Se requiere reescribir el requerimiento.

**Conflicto de intereses:** Se presenta cuando los objetivos o intereses con respecto al sistema de dos o mas stakeholders se contradicen entre ellos.

Ejemplo:

Un stakeholder desea que el sistema de entretenimiento del nuevo modelo de automóvil este equipado con funcionalidades MP3, disco duro opcional y una interfaz USB para atraer a los clientes interesados en la tecnología.

Otro stakeholder desea que el sistema solo este equipado con un reproductor de CD standard y el radio. Su objetivo es reducir costos para atraer a clientes preocupados por el presupuesto.

**Conflicto de valores:** Se presenta por diferentes criterios que los stakeholders utilizan para evaluar un punto (ejemplo: diferencias culturales). A unos les parecen mas importantes unos aspectos que otros.

**Conflicto de relaciones:** Problemas personales entre stakeholders.

**Conflicto estructural:** Se presenta por problemas de equidad entre los stakeholder.

Estos dos últimos conflictos difícilmente pueden ser resueltos en las etapas de IR.

## Procedimiento para el análisis de conflictos

Un conflicto puede tener múltiples causas.

Recomendaciones:

-Primero revisar los conflictos de datos, después de interés y finalmente los de valor.

Para determinar el tipo de conflicto:  
Para comprobar si hay conflictos de datos

–Solicitar a los stakeholders que escriban o expliquen las interpretaciones de los requerimientos.

–Identifique desviaciones. **Si hay interpretaciones diferentes, existen conflictos de datos.**

Si no hay conflictos de datos o todavía hay conflictos sin identificar, revisar conflictos de interés:

–Solicitar a los stakeholders que escriban o expliquen sus objetivos.

–Realizar comparaciones entre ellas.

Si no se detectaron conflictos de los tipos anteriores o aun existen conflictos después de resolver los otros tipos

Buscar **conflictos de valores**:

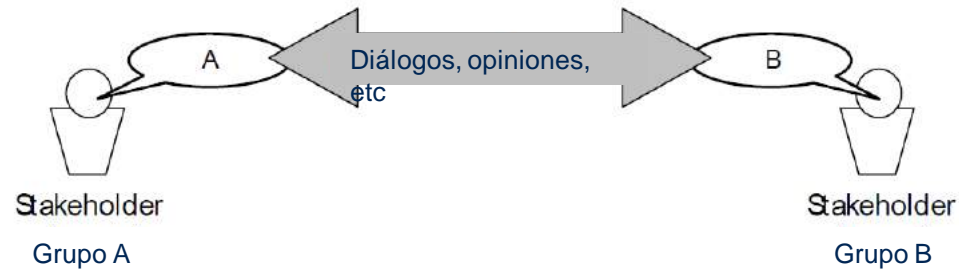
- Solicitar a los stakeholders que definan como evalúan los requerimientos.
- Revisar y clarificar los antecedentes de los stakeholders.
- Definir porque los stakeholders evalúan los requerimientos de manera diferente.

**Si definen puntos contradictorios de evaluación, existe un conflicto de valores.**

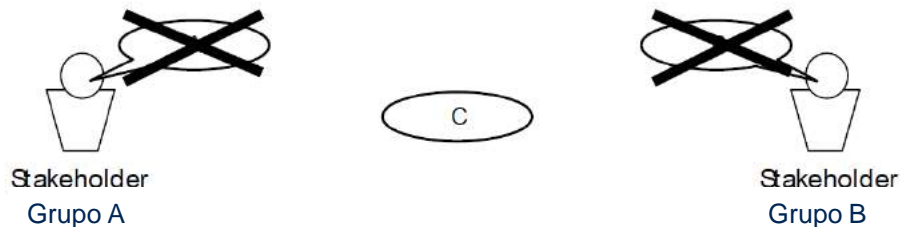
## 2) Resolver conflictos

Existen tres estrategias básicas para resolver conflictos:

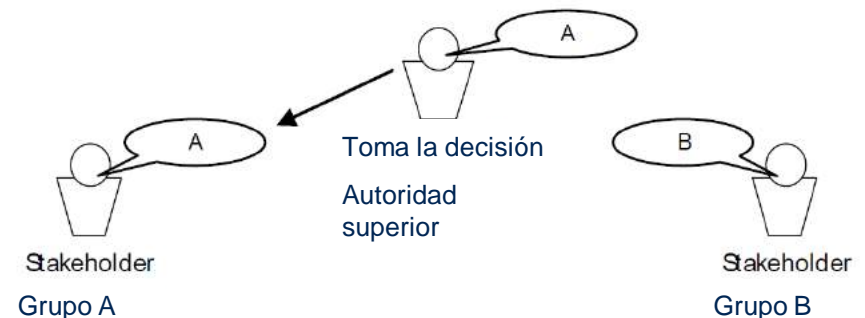
- 1) Negociación y priorización. Las partes en conflicto llegan a un acuerdo por medio de negociación



- 2) Solución creativa. Se descartan los puntos originando el conflicto y se crea una nueva solución completamente diferente.



- 3) Decisión superior: Una autoridad superior toma la decisión en favor de alguno de los puntos en conflicto.



## Evaluación de las estrategias de resolución de conflictos.

	Negociación	Solución creativa	Decisión
Conflicto de datos	Adecuada (Recomendada)	No adecuada	No adecuada
Conflicto de intereses	Adecuada (Recomendada)	Adecuada condicionalmente	Adecuada
Conflicto de valores	Adecuada condicionalmente	Adecuada (Recomendada)	Adecuada condicionalmente

La negociación es la estrategia mas adecuada, a excepción de los conflictos de valores en donde los antecedentes de los stakeholders pueden estar muy arraigados y difícilmente cambiaran su punto de vista.

En ese caso la solución creativa (una solución diferente) es la mas adecuada.

La decisión por una autoridad mas alta es recomendada como la ultima estrategia, cuando las demás fallan, ya que siempre habrá inconformes.

## Reuniones de negociación

- Las reuniones con los involucrados son clave para negociar y resolver conflictos de requerimientos.
- Cada conflicto de requerimientos debe ser discutido y evaluado individualmente.
- Participantes
  - Los analistas que descubrieron los traslapes, omisiones y/o conflictos de requerimientos.
  - --Stakeholders que pueden ayudar a resolver los problemas encontrados.
  - Un mediador independiente.





# Tres etapas de una reunión de negociación

## 1. Etapa de información

Explicación de cada uno de los problemas encontrados con los requerimientos.

## 2. Etapa de Discusión

- Todos los stakeholders interesados deben tener oportunidad de comentar.



requerimientos.



## 3. Etapa de Resolución

Se acuerdan acciones con respecto a los requerimientos problemáticos

Eliminar el requerimiento

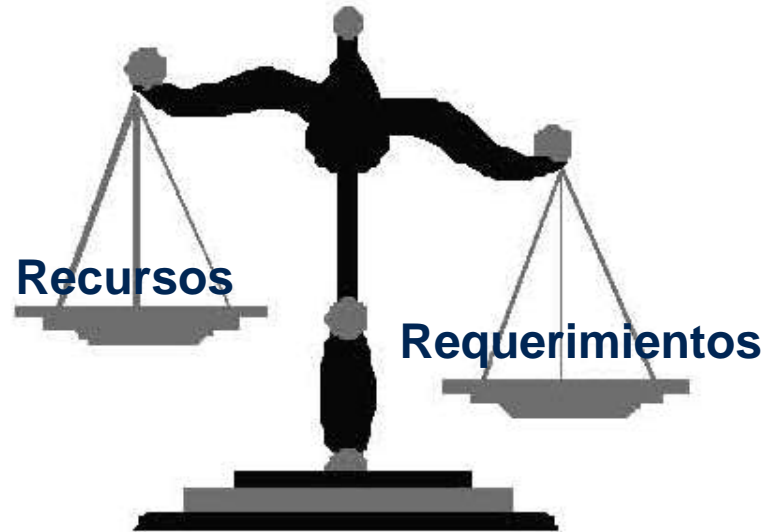
Realizar modificaciones específicas

Obtener más información acerca del requerimiento

# Priorización de requerimientos ¿?

- **Recursos limitados**

- Tiempo
- Presupuesto
- Esfuerzo



- **Expectativas altas de los cliente**
  - Demasiados requerimientos
  - **Requerimientos conflictivos.**

Todos los requisitos son necesarios, pero algunos son esenciales / críticos, mientras que otros no lo son.

## Priorización

- Priorizar significa listar o calificar en orden de prioridad.
- La priorización de requerimientos significa balancear el beneficio de cada requerimiento contra sus costos.

### Retos de la priorización

- Los *stakeholders* tienen diferentes puntos de vista sobre la importancia y necesidad de los requerimientos.
- Los clientes tratan de evitar la priorización por miedo a que los requerimientos de baja prioridad no se implementen.
- Los desarrolladores pueden tratar de evitar la priorización por que puede significar que son incapaces de implementar todos los requerimientos.
- Muchas técnicas de priorización son o muy complicados, o requieren mucho tiempo o resultan insuficientes.

Para que las comparaciones y la toma de decisiones sean mas efectivas, la priorización debe basarse en los siguientes puntos:

- a. Las prioridades deben ordenarse por niveles, cada nivel conteniendo requerimientos de la misma prioridad. Para facilitar la tarea el numero de niveles debe ser pequeño.
- b. La caracterización de los niveles debe ser cualitativa en lugar de cuantitativa. La calificación relativa en lugar de absoluta. Ejemplo: “Mas alta que”, en lugar de “Alta”, “Mediana”, “Baja”
- c. Los requerimientos comparados deben ser comparables. Deben comprender el mismo nivel de granularidad y abstracción.
- d. Los requerimientos siendo comparados deben ser independientes, o al menos no mutuamente dependientes, de manera que se pueden conservar un requerimiento y descartar otro.

# Técnicas de priorización

- Escalas numéricas
  - Asignación numérica.
  - Método de los 100 puntos.
- Modelo de priorización- Enfoque costo-beneficio
  - **Proceso analítico jerárquico (AHP)**
  - Modelo de estimación costo, valor y riesgo.
- Método Kano
- Otros enfoques
  - **Despliegue de la función de calidad(QFD)**

# Escalas numéricas

## Método de Asignación Numérica

Método → Cada stakeholder asigna un valor numérico (1-5) a cada requerimiento.

Resultado → Lista por orden de prioridad de requerimientos.

Ejemplo de escala:

5. Obligatorio (el cliente no puede prescindir de él).
4. Muy importante (el cliente no quiere estar sin él).
3. Importante (el cliente apreciaría tenerlo).
2. No es importante (el cliente aceptaría su ausencia).
1. No importa.

Otra escala (ejemplo: RequisitePro) con respecto a implementarse o no:

4. Debe
3. Debería
2. Podría
1. No

# Escalas numéricas

Método de 100 puntos

Método → Cada stakeholder recibe 100 puntos para asignar a los requerimientos como el/ella considere.

Ejemplo: 50 puntos a dos requerimientos, si solo dos son importantes.

10 puntos a 10 requerimientos (casi todos son importantes)

Resultado → Lista por orden de prioridad de requerimientos.

## Escalas Numéricas

Ventaja → Sencillos de implementar,

Desventaja → Requiere participación directa de *stakeholders* y negociación posterior

# Escalas numéricas

- Ventajas
  - Económicos
  - Fácil de implementar y entender
  - Convenientes para proyectos pequeños.
- Desventajas
  - Los resultados pueden ser solo estimaciones
  - Puede requerirse negociación posterior
  - Dependen de la participación de los *stakeholders*
  - Los clientes estiman 85% de requerimientos de alta prioridad, 15% prioridad media y 5% baja prioridad
    - No desean cambios en el proyecto.





## **INGENIERIA DE REQUERIMIENTOS (IR)**

### **4.- Escritura, Validación y Administración de Requerimientos**

- 4.1 Enfoques y estándares de escritura de req.
- 4.2 Criterios de calidad
- 4.3 Administración de Requerimientos
- 4.4 Validación de requerimientos

## Importancia de la documentación de requerimientos

La documentación de requerimientos es la actividad que documenta la información más importante adquirida durante el proceso de elicitación.

Las principales razones de la importancia de la etapa de documentación son:

- Persistencia
- Referencias en común
- Promover comunicación
- Promover objetividad
- Dar soporte para el entrenamiento de nuevos empleados
- Preservar el conocimiento
- Ayuda a entender el problema

Cada actividad de IR produce una multitud de información diferente

Ejemplos de piezas de información que requieren ser documentadas:

- Acuerdos alcanzados sobre los requerimientos
- Resultados de lluvias de ideas, de entrevistas, entre otros
- Puntos de vistas
- Contradicciones identificadas
- Priorización de requerimientos
- Personas responsables de cada actividad
- Resultados de prototipos utilizados
- Deseos y necesidades de los *stakeholders*
- Nuevas fuentes de requerimientos
- Peticiones de cambios
- Soluciones alternativas para los conflictos
- Resultados de inspecciones
- Etc.

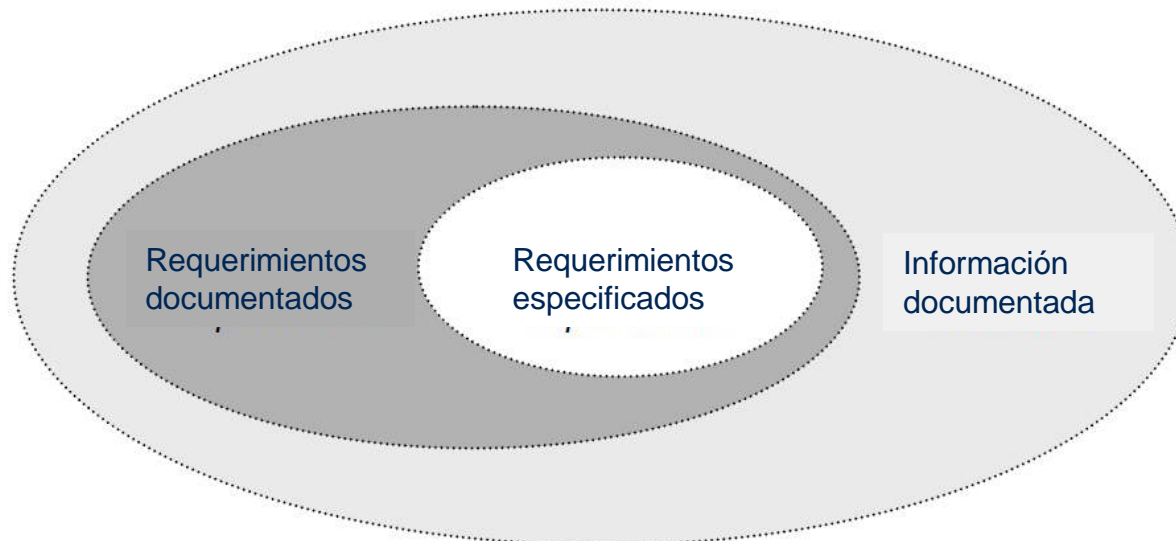
## Documentación vs Especificación

La información recabada debe ser documentada en diferentes formatos dependiendo del propósito de la documentación.

Dependiendo del uso se utilizan diferentes formatos y diferentes niveles de detalles.

La especificación de requerimientos es un tipo específico de documentación

- El objetivo de la especificación es definir los requerimientos en conjunto con las reglas, las guías y el resto de los documentos de requerimientos.

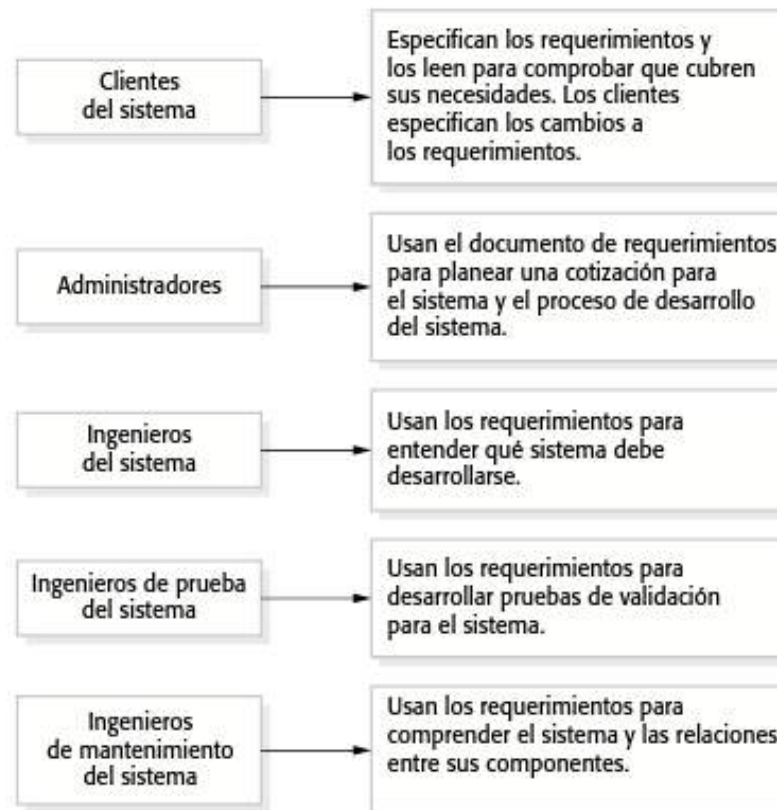


El proceso de documentación se basa en una serie de reglas. Básicamente pueden definirse tres:

- **Reglas de documentación general** → Se aplican a toda la información que se documenta (entrevistas, protocolos de reuniones, minutas, etc). Estas reglas identifican, por ejemplo: formato de documentos, encabezados requeridos en los documentos, etc.
- **Reglas de documentación para los requerimientos** → Definen como se deben documentar los requerimientos de acuerdo a los propósitos de la documentación. Incluyen: Lenguaje natural o formal, nivel de formalización, nivel de detalle, etc.
- **Reglas de especificación de requerimientos** → Estas reglas se aplican a los requerimientos que se incluyen en el documento de especificación. Pueden incluir restricciones adicionales como criterios de calidad, consistencia, etc.

## Especificación de requerimientos.

La especificación de requerimientos es el proceso de escribir, en un documento de requerimientos, los requerimientos del usuario y del sistema. De manera ideal, los requerimientos del usuario y del sistema deben ser claros, sin ambigüedades, fáciles de entender, completos y consistentes. Esto en la práctica es difícil de lograr, pues los participantes interpretan los requerimientos de formas diferentes y con frecuencia en los requerimientos hay conflictos e inconsistencias inherentes.



¿Qué técnicas de escritura existen ?

Es posible utilizar diferentes técnicas para la escritura de requerimientos:

Notación	Descripción
Enunciados en lenguaje natural	Los requerimientos se escriben al usar enunciados numerados en lenguaje natural. Cada enunciado debe expresar un requerimiento.
Lenguaje natural estructurado	Los requerimientos se escriben en lenguaje natural en una forma o plantilla estándar. Cada campo ofrece información de un aspecto del requerimiento.
Lenguajes de descripción de diseño	Este enfoque usa un lenguaje como un lenguaje de programación, pero con características más abstractas para especificar los requerimientos al definir un modelo operacional del sistema. Aunque en la actualidad este enfoque se usa raras veces, aún tiene utilidad para especificaciones de interfaz.
Anotaciones gráficas	Los modelos gráficos, complementados con anotaciones de texto, sirven para definir los requerimientos funcionales del sistema; los casos de uso del UML y los diagramas de secuencia se emplean de forma común.
Especificaciones matemáticas	Dichas anotaciones se basan en conceptos matemáticos como máquinas o conjuntos de estado finito. Aunque tales especificaciones sin ambigüedades pueden reducir la imprecisión en un documento de requerimientos, la mayoría de los clientes no comprenden una especificación formal. No pueden comprobar que representa lo que quieren y por ello tienen reticencia para aceptarlo como un contrato de sistema.

Estas técnicas se encuentran en tres clasificaciones para la escritura de requerimientos:

- Formal → Siguen una base rigurosa, notación matemática.
- Informal → Lenguaje natural, diagramas flujos, diagramas personalizados
- Semi-Formal → Tienen una estructura informal, pero utilizan bases mas estructuradas, como lenguaje UML

Normalmente se usan al menos dos enfoques, es decir, incluso un documento formal puede hacer uso del enfoque informal en algunas ocasiones (lenguaje natural)

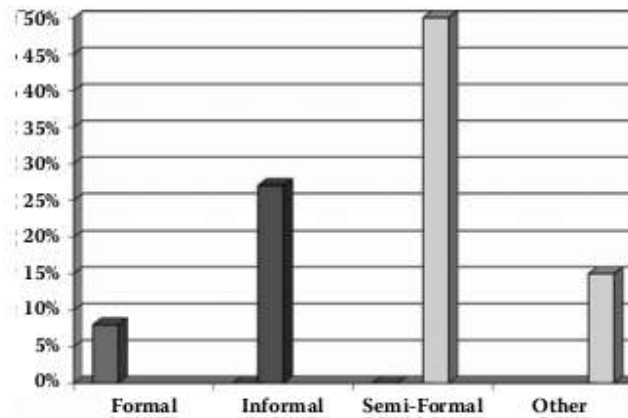


Figure 4.1 Reported requirements notation used (Neill and Laplante 2003).



## Especificación en lenguaje natural

- Consiste en utilizar el lenguaje natural de los *stakeholders* (ingles, español, francés) para escribir los requerimientos.
- Es el método mas utilizado para la especificación de requerimientos.

### Ventajas:

- Es universal → Puede ser usado para cualquier problema o dominio.
- Es flexible → Permite abstracciones arbitrarias y refinamientos durante el proceso de documentación.
- Es fácil de entender → La mayoría de los *stakeholders* puede comprender los requerimientos sin conocimientos previos o especiales.

**Desventajas:**

-Es ambiguo → Puede haber diferentes interpretaciones para un mismo requerimiento.

Razones que pueden provocar esta desventaja:

-Falta de una especificación clara → Se omiten detalles, esto es independiente del lenguaje. Esa falta de detalles puede ser llenada de manera diferente por cada lector.

-Defectos del lenguaje natural → El lenguaje natural puede causar confusiones por naturaleza.

Se distinguen dos elementos que pueden provocar estos problemas:

**-ambigüedad léxica,**

**-ambigüedad sintáctica.**

**-ambigüedad semántica**

\*Ambigüedad léxica:

Es causada por dos factores principalmente:

-Sinónimos

-Homónimos

-Sinónimos: Ejemplos: Medio/moderado/forma, pequeño/chico/niño,

-Homónimos: Ejemplos: vela → de barco, vela → de cera

Estos problemas pueden presentarse sobre todo cuando se utiliza vocabulario técnico que puede ser distinto entre compañías diferentes o en países diferentes que hablan el mismo idioma.

Se puede presentar incluso en diferentes departamentos de la misma organización.

**\*Ambigüedad sintáctica:**

Es causada cuando existen dos interpretaciones sintácticas validos diferentes que pueden ser asignados a la misma frase y para cada uno la frase tiene un significado distinto.

**Ejemplo:**

*El usuario introduce el código de acceso con la tarjeta de acceso para ingresar al sistema*

**¿Interpretación ?**

**El usuario introduce la tarjeta de acceso y además el código de acceso**

**El usuario introduce el código de acceso, utilizando la tarjeta de acceso (es decir, el código esta dentro de la tarjeta)**

## \*Ambigüedad semántica:

Se presenta cuando una frase tiene mas de una interpretación en un contexto específico, aun cuando no presenta ambigüedad léxica ni sintáctica.

## Ejemplo:

Si la ventana de un carro esta dañada y el sistema interior de vigilancia del carro detecta un intruso o si se abre la puerta del carro sin la llave, el sistema de seguridad debe activar una alarma.

¿Interpretación? ¿Cuándo se activa la alarma?

Ambigüedad → Cual es lazo es mas fuerte “o” o “y”

Interpretación 1 (y mas fuerte que o)

Si [ (1) la ventana de un carro esta dañada y (2) el sistema interior de vigilancia del carro detecta un intruso ] o (3) si se abre la puerta del carro sin la llave, el sistema de seguridad debe activar una alarma.

- Con abrir una puerta se activa la alarma

Interpretación 2 (o mas fuerte que y)

Si (1) la ventana de un carro esta dañada y [ (2) el sistema interior de vigilancia del carro detecta un intruso ] o (3) si se abre la puerta del carro sin la llave ], el sistema de seguridad debe activar una alarma.

La ventana debe estar dañada y además debe detectarse al intruso o alternativamente abrir la puerta sin la llave para activar la alarma

## Técnicas para evitar ambigüedades:

Es posible reducir el riesgo de ambigüedades en los requerimientos utilizando dos técnicas:

-Glosarios

--Lenguajes controlados

### 1.- Glosarios.

Colección de términos técnicos que son parte de la terminología de un lenguaje. Define el significado específico de ese término.

Es muy útil agregar ejemplos en cada definición, así como posibles sinónimos



## Estructuras de un glosario\*

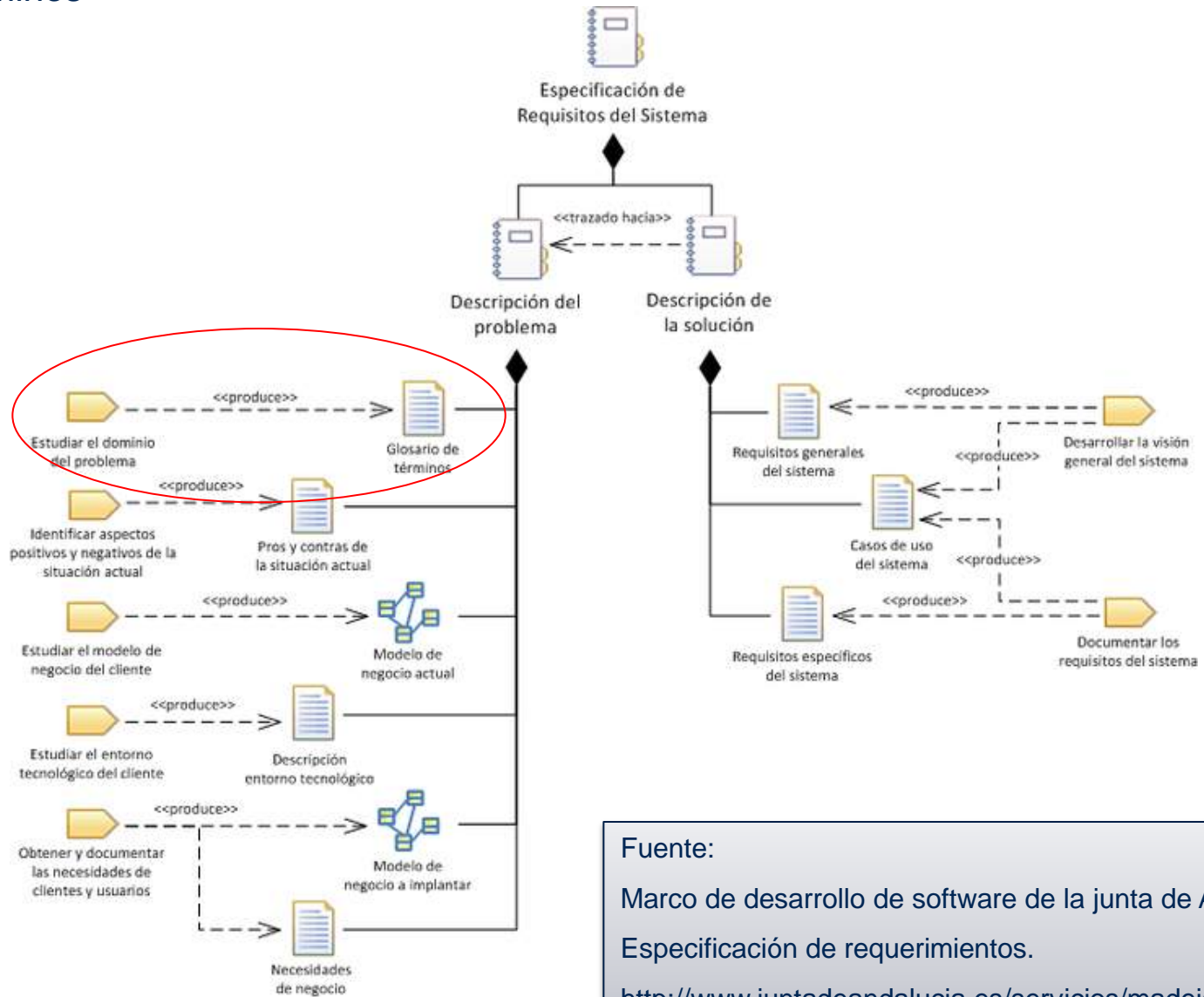
<b>Término</b>	nombre del termino
<b>Definición</b>	texto de la definición
<b>Sinónimos</b>	lista de sinónimos
<b>Términos relacionados</b>	lista de términos
<b>Ejemplos</b>	

**Término [nombre del termino] : Definición** texto de la definición. **Ejemplos**

## Ejemplo:

**Atributo:** Una parte de información de un *Elemento de Configuración*. Ejemplos: nombre, ubicación, Versión, número y Costo. Los Atributos se registran en la Base de Datos de la Configuración (CMDB).

El glosario puede ser definido en la etapa inicial de IR para acordar términos



Fuente:  
 Marco de desarrollo de software de la junta de Andalucía  
 Especificación de requerimientos.  
<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/407>



**Beneficios de los glosarios:**

- 1) Reducen el riesgo de que al desconocer un termino en el documento, este sea interpretado de manera diferente.
- 2) Reducen el riesgo de confusión por sinónimos y homónimos
- 3) Reducen la posibilidad de que haya interpretaciones desconocidas de algunos términos.

**Creación de glosarios:**

- 1) Definir la estructura de las entradas del glosario, la cual será utilizada por todos los autores del documento.
- 2) Consultar a todos los stakeholders con diferentes conocimientos sobre la definición de un termino y obtener un consenso general entendido por todos.
- 3) Si hay dudas sobre incluir o no un termino en el glosario, es mejor incluirlo
- 4) El glosario debe ser revisado por stakeholders con diferentes conocimientos antes de ser aprobado de manera final.

Herramientas de ayuda:

Wikis, intranet, foros, etc.

## 2.- Lenguajes controlados.

Un lenguaje controlado define, para un dominio específico, la sintaxis o gramática de un lenguaje natural restringido, así como el conjunto de términos que se utilizarán dentro de ese lenguaje.

### Ventajas:

- Las expresiones son más fáciles de entender, ya que son muy similares al lenguaje natural.
- Se reduce la ambigüedad debido al uso de gramática simplificada, y un vocabulario predefinido con una semántica precisa.
- Las expresiones son pre verificadas para ser válidas dentro del lenguaje controlado.

## Definición del lenguaje controlado:



Ejemplo de un lenguaje controlado para definición de requerimientos →

[Attempto](#)

## Consejos para evitar interpretaciones erróneas en lenguaje natural:

1. Elabore un formato estándar y asegúrese de que todas las definiciones de requerimientos se adhieran a dicho formato. Al estandarizar el formato es menos probable cometer omisiones y más sencillo comprobar los requerimientos. El formato que usa el autor expresa el requerimiento en una sola oración. A cada requerimiento de usuario se asocia un enunciado de razones para explicar por qué se propuso el requerimiento. Las razones también pueden incluir información sobre quién planteó el requerimiento (la fuente del requerimiento), de modo que usted conozca a quién consultar en caso de que cambie el requerimiento.
2. Utilice el lenguaje de manera clara para distinguir entre requerimientos obligatorios y deseables. Los primeros son requerimientos que el sistema debe soportar y, por lo general, se escriben en futuro “debe ser”. En tanto que los requerimientos deseables no son necesarios y se escriben en tiempo pospretérito o como condicional “debería ser”.
3. Use texto resaltado (negrilla, cursiva o color) para seleccionar partes clave del requerimiento.
4. No deduzca que los lectores entienden el lenguaje técnico de la ingeniería de software. Es fácil que se malinterpreten palabras como “arquitectura” y “módulo”. Por lo tanto, debe evitar el uso de jerga, abreviaturas y acrónimos.
5. Siempre que sea posible, asocie una razón con cada requerimiento de usuario. La razón debe explicar por qué se incluyó el requerimiento. Es particularmente útil cuando los requerimientos cambian, pues ayuda a decidir cuáles cambios serían indeseables.

Independientemente de la técnica....

### **Criterios de calidad de la especificación de requerimientos**

**Completo:** No hay información faltante

- Trazable:** Es posible trazar su fuente, evolución, impacto y uso
- Correcto:** confirmado por los *stakeholders*
- No ambiguo:** Solo tiene una interpretación valida
- Comprensible:** Es fácil de entender
- Consistente:** Los enunciados en el documento no se contradicen entre ellos
- Verificable:** El sistema implementado puede ser revisado
- Calificable:** Es posible determinar la relevancia del documento obtenido
- Actual:** Refleja el estado actual del sistema
- Atómico:** Un documento describe el sistema.

**Requerimiento ambiguo:**

*Con el fin de autenticarse, el conductor debe introducir una tarjeta electrónica y un PIN. Si es invalido, el motor no se enciende.*

Es ambiguo porque deja duda “si es invalido” (la tarjeta, el pin o los dos)

**Requerimiento verificable:**

*El sistema debe responder al evento ES-2 en al menos 80% de los casos en 2 s, y en todos los casos en 3 s como máximo para una carga del sistema entre 80% y 90% de su carga máxima, como se especifica en la restricción C14 .*

**Requerimiento No verificable:**

*El tiempo de respuesta normal del sistema debe ser menor de 2 s*

No puede ser verificado por que no queda claro a que se refiere la expresión “tiempo de respuesta normal” (en que condiciones, cuando puede ser mayor a 2 s, etc)

## Especificación en lenguaje estructurado

Consiste en utilizar una estructura de referencia ( plantilla ) para todos los documentos relacionados con la especificación de requerimientos

Se conserva la expresividad del lenguaje natural, pero se asegura que exista uniformidad en la especificación de requerimientos.

### **Ventajas:**

- Estructura probada: Se utiliza una estructura de referencia definida por expertos en la estructuración de documentos de requerimientos.
- Referencia para revisar integridad: La estructura de referencia facilita revisar si la documentación esta completa o no.
- Facilita centrarse en el contenido. La estructura define los elementos a describir, por lo que los IR se centran en obtener y documentar esa información.
- Misma información en el mismo lugar. Es fácil comprender documentos aunque se trate de diferentes proyectos.
- Herramientas de soporte. Existen diversas herramientas para el llenado de estructuras predefinidas.

## Algunos documentos de estructuras de referencia son:

**\*\*Plantillas Volere**

<http://www.volere.co.uk>

**\*\*Proyecto MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre )**

<http://merinde.net/>

**\*\*Referencias IEEE**

IEEE estandar 830-1998

IEEE Estandar 1233-1998

<http://standards.ieee.org/findstds/standard/830-1993.html>



## Especificación en lenguaje estructurado

Consiste en utilizar una estructura de referencia ( plantilla ) para todos los documentos relacionados con la especificación de requerimientos

Se conserva la expresividad del lenguaje natural, pero se asegura que exista uniformidad en la especificación de requerimientos.

### **Ventajas:**

- Estructura probada: Se utiliza una estructura de referencia definida por expertos en la estructuración de documentos de requerimientos.
- Referencia para revisar integridad: La estructura de referencia facilita revisar si la documentación esta completa o no.
- Facilita centrarse en el contenido. La estructura define los elementos a describir, por lo que los IR se centran en obtener y documentar esa información.
- Misma información en el mismo lugar. Es fácil comprender documentos aunque se trate de diferentes proyectos.
- Herramientas de soporte. Existen diversas herramientas para el llenado de estructuras predefinidas.

**Algunos documentos de estructuras de referencia son:**

**\*\*Plantillas Volere**

<http://www.volere.co.uk>

**\*\*Proyecto MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre )**

<http://merinde.net/>

**\*\*Referencias IEEE**

IEEE estandar 830-1998

IEEE Estandar 1233-1998

<http://standards.ieee.org/findstds/standard/830-1993.html>

## Plantilla VOLERE

-Definida El proceso de requisitos Volere está descrito en el libro Mastering the Requirements Process—Second

Edition por Suzanne Robertson y James Robertson, Addison-Wesley, 2006. ISBN 0-321-41949-9

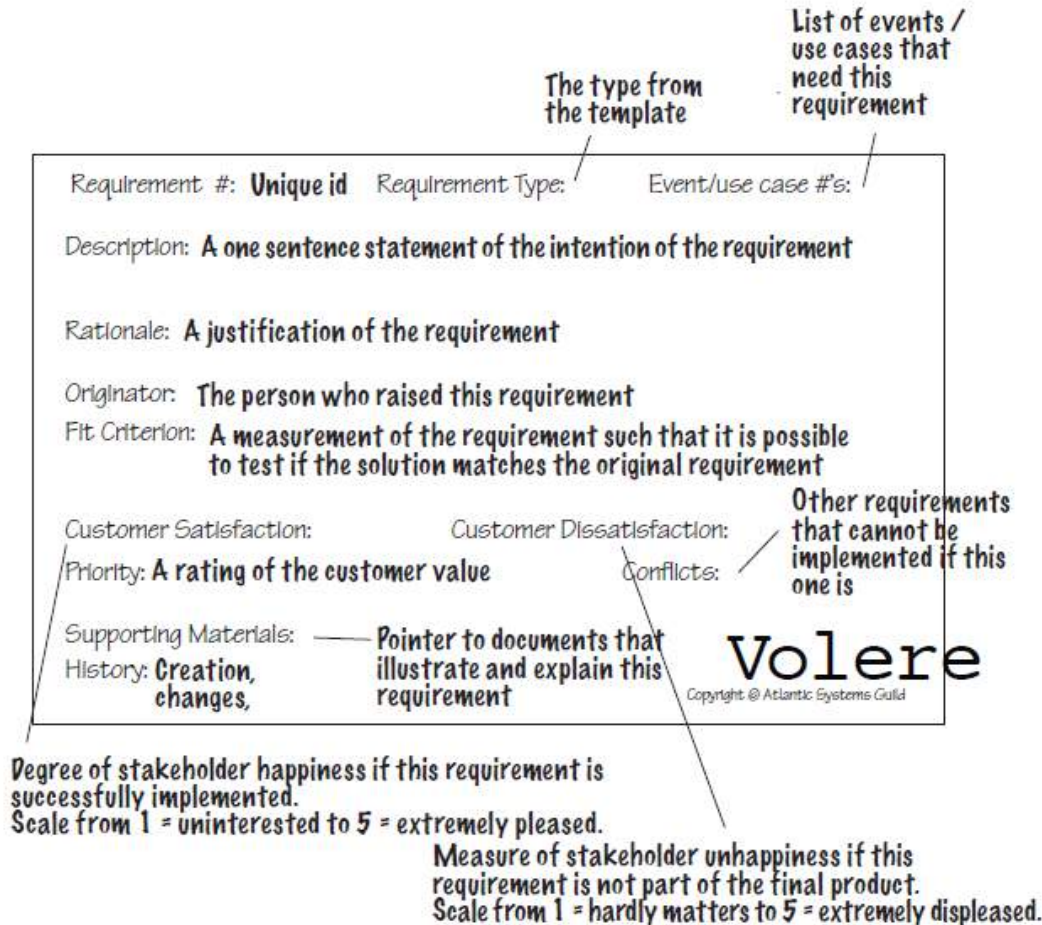
La plantilla VOLERE define la utilización de tarjetas para definir los requerimientos. (Una tarjeta por requerimiento)

Utiliza además estructuras de lenguaje de programación para definir los requerimientos.

La plantilla proponen algunos campos en cada tarjeta, tales como:

- razones de los requerimientos,
- -dependencias en otros requerimientos,
- fuente de los requerimientos,
- materiales de apoyo, etcétera.

Plantilla VOLERE



## Plantilla VOLERE

### Información incluida en las tarjetas

1. Una descripción de la función o entidad a especificar.
2. Una descripción de sus entradas y sus procedencias.
3. Una descripción de sus salidas y a dónde se dirigen.
4. Información sobre los datos requeridos para el cálculo u otras entidades en el sistema que se utilizan (la parte “requiere”).
5. Una descripción de la acción que se va a tomar.
6. Si se usa un enfoque funcional, una precondition establece lo que debe ser verdadero antes de llamar a la función, y una postcondición especifica lo que es verdadero después de llamar a la función.
7. Una descripción de los efectos colaterales (si acaso hay alguno) de la operación.

Ejemplo:

### Sistema de control para una bomba de Insulina\*

Una bomba de insulina es un sistema médico que simula la función del páncreas (un órgano interno). El software que controla este sistema es un sistema embebido, que recopila información de un sensor y controla una bomba que entrega al usuario una dosis regulada de insulina.

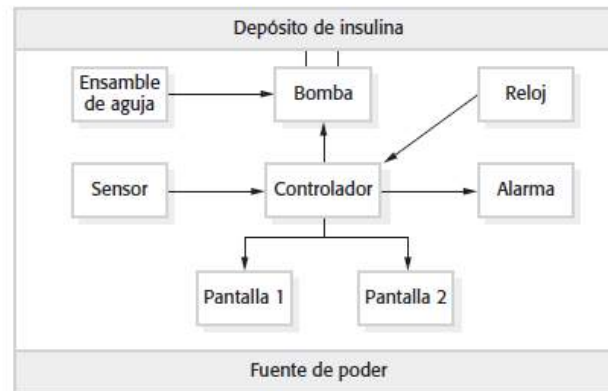
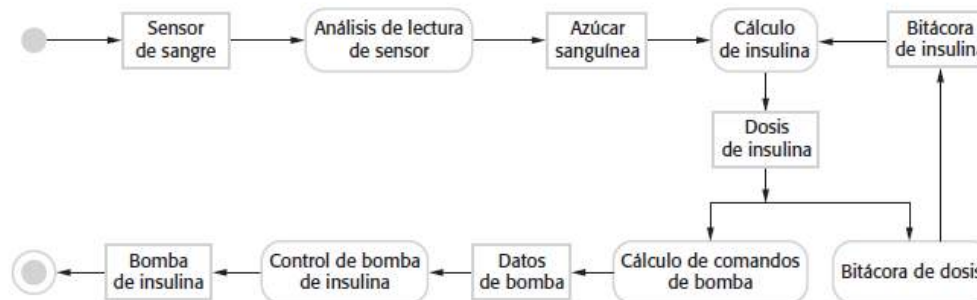


Figura 1.4 Hardware de bomba de insulina



\*Fuente ing. software, sommerville

## Ejemplo: Requerimiento funcional en lenguaje natural

3.2 Si se requiere, cada 10 minutos el sistema medirá el azúcar en la sangre y administrará insulina. (Los cambios de azúcar en la sangre son relativamente lentos, de manera que no son necesarias mediciones más frecuentes; la medición menos periódica podría conducir a niveles de azúcar innecesariamente elevados.)

3.6 Cada minuto, el sistema debe correr una rutina de autoevaluación, con las condiciones a probar y las acciones asociadas definidas en la tabla 1. (Una rutina de autoevaluación puede detectar problemas de hardware y software, y prevenir al usuario sobre el hecho de que la operación normal puede ser imposible.)

## Ejemplo: Requerimiento funcional en lenguaje estructurado (plantilla VOLERE)

### Bomba de insulina/Software de control/SRS/3.3.2

<b>Función</b>	Calcula dosis de insulina: nivel seguro de azúcar.
<b>Descripción</b>	Calcula la dosis de insulina que se va a suministrar cuando la medición del nivel de azúcar actual esté en zona segura entre 3 y 7 unidades.
<b>Entradas</b>	Lectura del azúcar actual (r2), las dos lecturas previas (r0 y r1).
<b>Fuente</b>	Lectura del azúcar actual del sensor. Otras lecturas de la memoria.
<b>Salidas</b>	CompDose: la dosis de insulina a administrar.
<b>Destino</b>	Ciclo de control principal.
<b>Acción</b>	CompDose es cero si es estable el nivel de azúcar, o cae o si aumenta el nivel pero disminuye la tasa de aumento. Si el nivel se eleva y la tasa de aumento crece, CompDose se calcula entonces al dividir la diferencia entre el nivel de azúcar actual y el nivel previo entre 4 y redondear el resultado. Si la suma se redondea a cero, en tal caso CompDose se establece en la dosis mínima que puede entregarse.
<b>Requerimientos</b>	Dos lecturas previas, de modo que puede calcularse la tasa de cambio del nivel de azúcar.
<b>Precondición</b>	El depósito de insulina contiene al menos la dosis individual de insulina máxima permitida.
<b>Postcondición</b>	r0 se sustituye con r1, luego r1 se sustituye con r2.
<b>Efectos colaterales</b>	Ninguno.

Condición	Acción
Nivel de azúcar en descenso ( $r2 < r1$ )	CompDose = 0
Nivel de azúcar estable ( $r2 = r1$ )	CompDose = 0
Nivel de azúcar creciente y tasa de incremento decreciente ( $(r2 - r1) < (r1 - r0)$ )	CompDose = 0
Nivel de azúcar creciente y tasa de incremento estable o creciente ( $(r2 - r1) \geq (r1 - r0)$ )	CompDose = round $((r2 - r1)/4)$ If resultado redondeado = 0 then CompDose = MinimumDose

## Proyecto MeRinde (**Metodología de la Red Nacional de Integración y Desarrollo de Software Libre** )

MeRinde es un proyecto que propone un estándar abierto para el proceso de desarrollo de software

**Dentro del proyecto esta definida una [plantilla](#) para la especificación de requerimientos.**



## Estándares IEEE

Son los mas utilizados para la definición de requerimientos mundialmente.

[IEEE 1233-1988](#) →

### **IEEE Guide for Developing System Requirements Specifications**

Guía para el desarrollo de Especificaciones de Requerimientos de Sistemas

Esta guía nos da la pauta para el desarrollo de un conjunto de requerimientos que satisfarán una necesidad específica. En esta guía, a ese conjunto de requerimientos se le denomina Especificación de Requerimientos del Sistema (System Requirements Specification, SyRS)

El desarrollo de una SyRS incluye la identificación, organización, presentación y modificación de los requerimientos. Esta guía trata las condiciones necesarias para incorporar conceptos operacionales, restricciones de diseño, y requerimientos de la configuración del diseño en la especificación. Además, trata las características y cualidades necesarias de los requerimientos individuales y del conjunto de todos los requerimientos

[IEEE 830-1988](#) →

## IEEE Recommended Practice for Software Requirements Specifications

Prácticas Recomendadas para la Especificación de los Requisitos del Software

<b>Table of Contents</b>	
1.	Introduction
1.1	Purpose
1.2	Scope
1.3	Definitions, acronyms, and abbreviations
1.4	References
1.5	Overview
2.	Overall description
2.1	Product perspective
2.2	Product functions
2.3	User characteristics
2.4	Constraints
2.5	Assumptions and dependencies
3.	Specific requirements (See 5.3.1 through 5.3.8 for explanations of possible specific requirements. See also Annex A for several different ways of organizing this section of the SRS.)
	Appendixes
	Index

**Figure 1—Prototype SRS outline**

Independientemente de la técnica....

### **Criterios de calidad de la especificación de requerimientos**

**Completo:** No hay información faltante

- **Trazable:** Es posible trazar su fuente, evolución, impacto y uso
- **Correcto:** confirmado por los *stakeholders*
- **No ambiguo:** Solo tiene una interpretación valida
- **Comprensible:** Es fácil de entender
- **Consistente:** Los enunciados en el documento no se contradicen entre ellos
- **Verificable:** El sistema implementado puede ser revisado
- **Calificable:** Es posible determinar la relevancia del documento obtenido
- **Actual:** Refleja el estado actual del sistema
- **Atómico:** Un documento describe el sistema.

**Requerimiento ambiguo:**

*Con el fin de autenticarse, el conductor debe introducir una tarjeta electrónica y un PIN. Si es invalido, el motor no se enciende.*

Es ambiguo porque deja duda “si es invalido” (la tarjeta, el pin o los dos)

**Requerimiento verificable:**

*El sistema debe responder al evento ES-2 en al menos 80% de los casos en 2 s, y en todos los casos en 3 s como máximo para una carga del sistema entre 80% y 90% de su carga máxima, como se especifica en la restricción C14 .*

**Requerimiento No verificable:**

*El tiempo de respuesta normal del sistema debe ser menor de 2 s*

No puede ser verificado por que no queda claro a que se refiere la expresión “tiempo de respuesta normal” (en que condiciones, cuando puede ser mayor a 2 s, etc)

## Criterios de aceptación de requerimientos

- Un criterio de aceptación define las reglas que deben seguirse para definir si un requerimiento se acepta o no en el proceso de revisión.
- Para ser aceptado el requerimiento debe cubrir todos los criterios definidos.
- Algunos requerimientos pueden ser adaptados para cumplir con los criterios.
- Se definen criterios de aceptación para:
  - Artefactos de requerimientos
  - Requerimientos individuales
  - Documentos de requerimientos.

### Criterios para artefactos de requerimientos:

Se refiere a aquellos documentos generales que se recaban durante el proceso de IR.

Los criterios definen las reglas de aceptación tanto de los documentos generales, como del documento de requerimientos.

*Ejemplo: En el documento de requerimientos no debe tener ninguna contradicción.*

### Criterios para requerimientos individuales:

Refinan los lineamientos generales de calidad de los requerimientos individuales. Se aplican a cada requerimiento de manera independiente.

*Ejemplo: Cada requerimiento debe tener un identificador válido. El identificador debe ser único para cada requerimiento y estructurado de acuerdo al esquema:  
<Categoría><Número>.*

### Criterios del documento de requerimientos:

Se aplican a todo el documento y no a cada requerimiento individual.

*Ejemplo:*

*Para aprobar el documento de requerimientos, 98% de los requerimientos funcionales deben ser aceptados de acuerdo a los criterios de aceptación de los requerimientos funcionales*

## Ejercicios.

Escriba dos requerimientos en lenguaje natural y reescríbalos en lenguaje estructurado (plantilla Volere) para:

- Un sistema de bombeo de petróleo (gasolina) no asistido que incluya un lector de tarjeta de crédito. El cliente pasa la tarjeta en el lector, luego especifica la cantidad de combustible requerido. Se suministra el combustible y se deduce de la cuenta del cliente.
- La función de dispensar efectivo en un cajero automático.
- La función de revisión y corrección ortográfica en un procesador de textos.

Asegúrese que los requerimientos cumplan los requisitos de calidad.