



# Análisis de Requisitos

El proceso de estudio de las necesidades  
de los usuarios

# Requisitos

**Requirimiento:** Condición o capacidad que debe tener un sistema para satisfacer un contrato.

**Gestión de requerimientos:** Conjunto de actividades que intentan entender las necesidades de los usuarios y traducirlas en afirmaciones precisas.

# Requisitos

El análisis de requisitos es la piedra angular en la construcción del Software

Los problemas con los requisitos constituyen la principal fuente de problemas (37%)

El costo de solucionar un error en la etapa de mantenimiento es 200 veces mayor que en la etapa de requerimientos

# Actividades generales del análisis de requisitos

Se basa en técnicas para obtener información

- . Análisis de requisitos
- . Especificación de requisitos
- . Validación de los requisitos (por parte de los usuarios comprobando validez, consistencia y lo completo que sean)

# Ciclo de Vida

## Ciclo de Vida de la IR (Según Boehm, 1988 - Kontoya/Somerville, 1997)



# Actividades y Entidades



# ¿Quienes hacen la ingeniería de requerimientos?

Es muy difícil encontrar a una persona que sepa: entrevistar, escuchar, cuestionar de manera crítica, modelar, analizar, facilitar discusiones y negociaciones, observar, comunicar de manera verbal, relacionarse con gente, etc.

¿Existen?

# Requisitos Funcionales

Describen la funcionalidad o los servicios que se espera que el sistema proveerá, sus entradas y salidas, excepciones, etc.

El requerimiento o requisito funcional describe que debe hacer el sistema respecto a su entorno. En otras palabras, refleja las necesidades de los usuarios o la interacción con otros sistemas.



# Requisitos No Funcionales

Se refieren a las propiedades emergentes del sistema como la fiabilidad, el tiempo de respuesta, la capacidad de almacenamiento, la capacidad de dispositivos de entrada/salida y la representación de datos que se utilizan en las interfaces del sistema.

# Clasificación de Requerimientos no funcionales

Clasificación modelo FURPS, definición de Larman en 2002.

F: Functionality

U: Usability

R: Reliability

P: Performance

S: Supportability

# Requerimientos de Funcionabilidad (Functionality)

Son aquellos requerimientos que reflejan las características fundamentales (requerimiento funcional o funcionabilidades del sistema), además de capacidades y seguridad.

# Requerimientos de usabilidad o Capacidad de uso (Usability)

Son aquellos requerimientos que representan facilidad o nivel de uso del producto. Es decir, el grado en el que el diseño de un elemento facilita o dificulta su manejo. Se incluyen factores humanos, estética, consistencia de la interfaz, ayuda en línea, agentes y wizards, documentación y material de entrenamiento. Por ej, visibilidad del texto a una cierta distancia y combinación de colores del texto

# Requerimientos de Fiabilidad (Reliability)

Muestran la capacidad de un sistema o componente para ejecutar las funciones requeridas bajo condiciones normales en un periodo de tiempo específico.

Por ejemplo: Disponibilidad (porcentaje de tiempo disponible, horas de uso, etc), tiempo medio entre fallas y para reparación, cuanto tiempo es posible que el sistema esté inoperante después que falla, exactitud que se requiere para las salidas del sistema, cantidad máxima de errores o porcentaje de defectos, etc.

# Requerimientos de Desempeño(Performance)

Se refieren a las características de rendimiento del sistema. Incluye tiempos de respuesta específicos. Por ej: Tiempo de respuesta para una transacción, transacciones por segundo, capacidad, como opera el sistema si es degradado de alguna manera, como utiliza los recursos, etc.

# Requerimientos de Capacidad de Soporte(Supportability)

Son requerimientos que refuerzan el soporte y mantenimiento del sistema que está siendo construido, incluyendo normas de codificación, convenciones de nombres, librerías, acceso para mantenimiento, utilidades de mantenimiento si las hay. Como requerimiento que ayuda al mantenimiento se debe hacer referencia al uso de nomenclatura común para el desarrollo del sistema y a la metodología de desarrollo.

# FURPS

## Classifying Requirements

- FURPS

- ▶ Functionality
- ▶ Usability
- ▶ Reliability
- ▶ Performance
- ▶ Supportability

- +

- ▶ Design requirements
- ▶ Implementation requirements
- ▶ Interface requirements
- ▶ Physical requirements



**Functional requirements**



**Non-functional requirements**



# ¿Cómo debe ser un requerimiento?

Un requerimiento debe ser:

Especificado por escrito, como todo contrato o acuerdo entre dos partes.

Posible de probar o verificar, si un requerimiento no se puede comprobar entonces, como se sabe si se cumplió con él o no?

Conciso, debe ser fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en el futuro.

# Documentos de especificación de requisitos

Luego de dar el informe de necesidad y comenzado el proyecto, se crea el SyRS (System Requirements Specification) y el SRS (Software Requirements Specification)

Se espera que una especificación de requerimientos que fue aprobada por clientes y/o usuarios tenga al menos las siguientes características:

- \* Que contenga todos los requerimientos deseados
- \* Que cada requerimiento solo tenga una interpretación posible
- \* Que el cumplimiento de cualquier requerimiento no provoque conflictos con el cumplimiento de otro requerimiento, es decir, que sea consistente
- \* Que se definan prioridades

# Como formalizar la especificación de requisitos

1) Misión y Alcance	Pequeño párrafo que sintetice claramente qué se espera del proyecto
2) Determinar quiénes son los actores relevantes	Identificar todos los posibles beneficiarios / usuarios del software
3) Determinar objetivos	Definir objetivos generales y avanzar paulatinamente en la definición de objetivos particulares
4) Descripción del circuito funcional	Describir correctamente este circuito administrativo como marco de referencia
5) Detallar principales prestaciones funcionales	Una lista de prestaciones funcionales que el se deben cumplir, agrupadas por usuarios
6) Especificar requerimientos	Describir el comportamiento esperado para cada una de las prestaciones funcionales detalladas en el punto anterior
7) Determinar requerimientos no funcionales	En general se dice que los requerimientos no funcionales son “restricciones” EJ: “el desarrollo debe ser en PHP y PSQL”
8) Determinar prioridades	Se recomienda que las prioridades queden formalmente definidas por los propios interesados