

# Análisis de requisitos

---

## Qué es un Requisito

Un **requisito de software puede ser** definido como una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo. Un **requisito** es una característica que el sistema **debe** tener o es una restricción que el sistema **debe** satisfacer para ser aceptada por el cliente.

## Especificación de Requisitos

El documento de Especificación de Requisitos utiliza estándares de estructuración de especificaciones de requisitos, aclara el objetivo global a cumplir por el sistema, emplea descripciones textuales y gráficas, ordena y agrupa los requisitos de forma lógica, relaciona unos requisitos con otros para facilitar su entendimiento y relaciona los requisitos con otros elementos. Los pasos para identificar y definir los requisitos de un producto son:

- Observar y entender el trabajo desde **el punto de vista del usuario**.
- Interpretar el trabajo del usuario y la forma en la que él lo describe.
- "Inventar" mejores formas de hacer el trabajo.
- Plasmar estos resultados en forma de especificación de requisitos.

## Problemas al definir los requisitos

Los requisitos no son obvios y vienen de muchas fuentes, pueden ser difíciles de expresar en palabras, la cantidad de requisitos en un proyecto puede ser difícil de manejar, un requisito puede cambiar a lo largo del ciclo de desarrollo. Se tiende a recordar lo excepcional y olvidar lo rutinario, los usuarios tienen distinto vocabulario que los desarrolladores.

## Evolución de los Requisitos

Porque al analizar el problema no se hacen las preguntas correctas a las personas correctas, porque cambió el problema que se estaba resolviendo, porque los usuarios cambiaron su forma de pensar o sus percepciones, porque cambió el mercado en el que se desenvuelve el negocio.

## Tipos de Requisitos

- Requisitos de Negocio
- Requisitos de Usuario
- Requisitos del Sistema/Software

### Requisitos de Negocio

Representan los objetivos generales que quieren alcanzar con el desarrollo del software. Descripción de alto nivel de lo que el sistema debe de hacer. No describen detalles técnicos ni funcionalidades específicas. Se relacionan con las metas estratégicas de la empresa. Son la base de los requisitos de usuario.

### Requisitos de Usuario

Representan las necesidades o metas que tienen los usuarios acerca del sistema. Son más específicos que los de negocio. Se enfocan en las tareas y expectativas del usuario.

### Requisitos del sistema

Detallan las especificaciones técnicas que el software debe cumplir. Son de bajo nivel y específicos. Incluyen detalles técnicos como arquitectura, rendimiento, seguridad y estándares a seguir. Se derivan directamente los requisitos de usuario y de negocio.

### Requisitos funcionales

Describen lo que el sistema debe hacer. Describen la interacción entre el sistema y su entorno independientemente de su implementación. El entorno incluye al usuario y cualquier otro sistema externo que interactúa con el sistema.

### Requisitos no funcionales

Son aquellos que no hacen referencia directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes del mismo como son rendimiento, usabilidad, seguridad o escalabilidad. Los requisitos no funcionales son las **cualidades que debe tener el producto**. Estos requisitos hacen que el producto sea atractivo, útil, rápido, fiable o seguro. Se pueden clasificar de diferentes formas:

- Requisitos de producto
- Requisitos organizacionales
- Requisitos externos

### Características de la descripción de un requisito

- Completo
- Correcto
- Realizable
- Necesario
- Priorizable
- No ambiguo
- Verificable
- Consistente
- Modificable
- Trazable

### Técnicas de recogida de requisitos

- Reuniones / Entrevistas
- Cuestionarios y encuestas
- Brainstorming
- Casos de uso
- Prototipos
- Escenarios

# Modelo de casos de uso

## Introducción

Un diagrama de casos de uso muestra la relación entre los actores y los casos de uso del sistema, representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa, centrándose en lo que debe hacerse y no en la manera de hacerlo. Deben evitarse expresiones imprecisas, se busca la sencillez y la claridad. Elementos que pueden participar en un diagrama de casos de uso:

- Actores
- Casos de uso
- Relaciones

## Actores

Un actor es una entidad externa al sistema que realiza algún tipo de interacción con el mismo, se representa mediante una figura humana; esta representación sirve tanto para actores que son personas como para otro tipo de actores no humanos. Se deben distinguir entre actores:

- Principales: aquellos para los que se construye el sistema
- Secundarios: aquellos que dan soporte al sistema.

## Identificación de Actores

¿Qué usuarios están soportados por el sistema para desarrollar su trabajo? ¿Qué usuarios ejecutan las funciones principales del sistema? ¿Qué usuarios desempeñan funciones secundarias, como mantenimiento y administración? ¿El sistema interactúa con hardware externo o software?

## Casos de uso

Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea determinada. Expresa una unidad coherente de funcionalidad, y se representa mediante una elipse con el nombre del caso de uso en su interior. El nombre del caso de uso debe reflejar la tarea específica que el actor desea llevar a cabo usando el sistema.

## Identificación de casos de uso

Capturar el comportamiento deseado del sistema en desarrollo, sin tener que especificar cómo se implementa este comportamiento. Los casos de uso proporcionan un medio para que los desarrolladores, los usuarios finales del sistema y los expertos del dominio lleguen a una comprensión común del sistema. Un escenario de la instancia de un caso de uso. Los casos de uso no deben ser excesivamente genéricos ni demasiado específicos. Requisitos funcionales del sistema.

## Relaciones

En un diagrama de casos de uso pueden aparecer las siguientes relaciones:

- Asociación (-----)
- Extiende (- - - - ->) <>

- Generalización (-----|>)
- Inclusión (- - - - ->) <>

## Identificar relaciones entre Casos de uso

### Inclusión <>

- Indica que el flujo de eventos del caso de uso base se incluye en el comportamiento del otro caso de uso.
- Factoriza comportamiento común.
- Solamente se hace cuando la parte común es utilizada por otro caso de uso o cuando es utilizada por otro actor.

### Extensión <>

- Un caso de uso extiende otro caso de uso, si el caso de uso extendido incluye el comportamiento del otro bajo ciertas condiciones.
- Se utiliza para modelar la parte de un caso de uso que el usuario puede ver como comportamineto opcional del sistema.
- Se separa el comportamiento opcional del obligatorio

### Generalización

- Cuando algunos casos de uso tienen algo en común y puede ser abstraído a otro, mucho más general.
- El caso de uso hijo hereda el comportamiento y el significado del caso de uso padre.
- El hijo puede añadir o redefinir el comportamineto y el significado del padre.
- El hijo puede ser colocado en cualquier lugar donde aparezca el padre. algunos casos de uso tienen algo en común y puede ser abstraído a otro, mucho más general.
- El caso de uso hijo hereda el comportamiento y el significado del caso de uso padre.
- El hijo puede añadir o redefinir el comportamineto y el significado del padre.
- El hijo puede ser colocado en cualquier lugar donde aparezca el padre.

## Construcción

### Identificación de casos de uso:

- Proceso iterativo en el que se van descubriendo escenarios desde el punto de vista del usuario.
- Pueden utilizarse diversas técnicas: observación, entrevista estructurada, etc.

### Descripción de los casos de uso:

- Descripción inicial: se describe en un par de frases las principales etapas de cada caso de uso; se distingue un escenario principal y se identifican los escenarios alternativos y excepciones.
- Se establece un proceso iterativo en el cual los casos de uso se amplían, profundizándose en su descripción, buscando etapas comunes y alternativas.

## El proceso de Análisis de Requisitos con Casos de Uso

1. Elaborar el diagrama de casos de uso de trazo grueso: 1.1. Identificar los actores 1.2. Identificar los principales casos de uso de cada actor. 1.3. Identificar los nuevos casos de uso a partir de los

existentes.

2. Crear descripciones de casos de uso de trazo grueso.
3. Definir prioridades y seleccionar casos de uso de cada incremento
4. Se abordan los diferentes incrementos: 4.1. Se profundiza en el conocimiento de los casos de uso involucrados. 4.2. Se refina el diagrama de casos de uso. 4.3. Se escriben los casos de trazo fino.