

**PROGRAMA DE ESTUDIOS**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**ANÁLISIS Y DISEÑO  
DE SISTEMAS**

**Tema:**

**ANÁLISIS DE REQUERIMIENTOS  
BASADO EN CASOS DE USO.**

## ANÁLISIS DE REQUERIMIENTOS BASADO EN CASOS DE USO.

### ANALISIS DE REQUISITOS CON CASOS DE USO E HISTORIAS DE USUARIOS

El primer paso para definir un nuevo producto, servicio, proceso o sistema es definir los requisitos, es decir, requisitos funcionales o no funcionales específicos.

- **Los requisitos funcionales** describen cómo el producto o servicio satisface las necesidades del cliente. Esto incluye características y funcionalidades en casos de uso que documentan cómo los usuarios interactuarán con el producto o servicio.
- **Los requisitos no funcionales** son atributos operativos y del producto que a veces no son obvios para el usuario, incluidos el rendimiento, la facilidad de uso, la durabilidad, la seguridad y los aspectos financieros (precio y costo).



Requisitos funcionales frente a requisitos no funcionales

Los requisitos funcionales se pueden considerar como lo que el producto debe hacer para el cliente, mientras que los requisitos no funcionales se pueden

considerar como restricciones que el producto o servicio debe cumplir con el diseño.

Los requisitos funcionales capturan el comportamiento previsto del sistema. Este comportamiento puede expresarse como servicios, tareas o funciones que el sistema debe realizar. En la industria del desarrollo de software, el enfoque de casos de uso se ha convertido rápidamente en una práctica generalizada para capturar requisitos funcionales. Esto es especialmente cierto en la comunidad UML y orientada a objetos donde se originaron, pero su aplicabilidad no se limita a los sistemas orientados a objetos.

### **¿Qué técnicas para capturar los requisitos funcionales?**

Los requisitos funcionales generalmente se capturan en forma de casos de uso o escenarios de usuario. Estos términos a veces se usan indistintamente, pero en realidad significan cosas ligeramente diferentes.

- **Los casos de uso** se centran más en el sistema y en lo que debe hacer para satisfacer las necesidades de los usuarios.
- **Las historias de usuario**, por otro lado, muestran la funcionalidad del producto desde el punto de vista del usuario, especificando roles de usuario y objetivos específicos que quieren lograr.

### **Captura de requisitos funcionales mediante historias de usuarios**

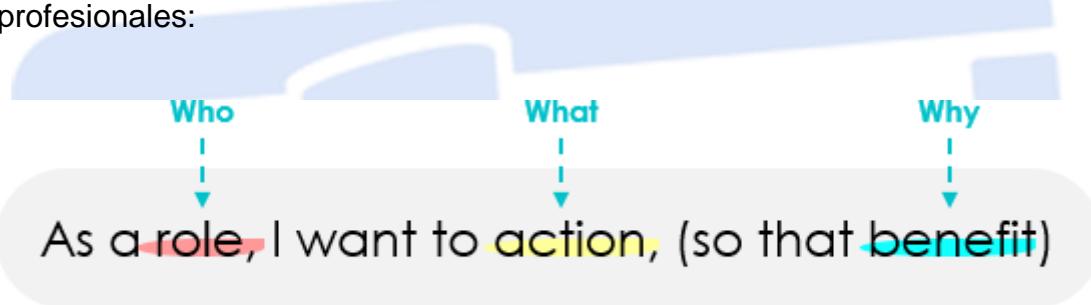
Las historias de usuarios son un método ligero para capturar rápidamente el «quién», «qué» y «por qué» de los requisitos de un producto. En pocas palabras, las historias de usuarios son ideas que expresan las necesidades que desean los usuarios. Las historias de usuario son breves y cada elemento suele contener menos de 10 o 15 palabras. Las historias de usuario son listas de «cosas por hacer» que lo ayudan a identificar los pasos a lo largo de la ruta del proyecto. Ayudan a garantizar que su proceso y el producto resultante cumplan con sus requisitos.

### Plantilla de historia de usuario

Las historias de usuario solo capturan los elementos esenciales de un requisito:

- ¿Para quién es?
- ¿Qué espera del sistema?
- ¿Por qué es importante (¿opcional)?

Aquí hay un formato simple de historia de usuario utilizado por el 70% de los profesionales:



**Rol** : el usuario debe ser un ser humano real que interactúe con el sistema.

- Sea lo más específico posible
- El equipo de desarrollo NO es un usuario

**Acción** : el comportamiento del sistema debe escribirse como una acción.

- Por lo general, único para cada historia de usuario
- El «sistema» está implícito y no se escribe en la historia.
- Voz activa, no voz pasiva (“Puedo ser notificado”)

**Beneficios** : el beneficio debe ser un resultado del mundo real que no sea funcional o sea externo al sistema.

- Muchas historias pueden compartir la misma declaración de beneficios.
- El beneficio puede ser para otros usuarios o clientes, no solo para el usuario de la historia.

### ¿Cómo identificar los requisitos funcionales con el caso de uso?

Para comprender completamente los requisitos funcionales del sistema, debe saber para quién es el sistema, es decir, ¿quién usará el sistema?

La respuesta a esta pregunta es: el **actor** en el análisis de casos de uso

Los casos de uso o las historias de usuarios capturan los requisitos funcionales cuyo comportamiento puede expresarse como servicios, tareas o funciones que el sistema debe realizar. Los casos de uso definen la interacción entre el usuario y el servicio del sistema que puede ayudar a definir los requisitos funcionales del sistema en desarrollo. O en otras palabras, qué debe hacer el producto o servicio para satisfacer las necesidades y deseos del cliente.

Un caso de uso comienza con un “actor” o “quién”, un usuario específico del producto o servicio.

Un actor es una persona o un sistema externo que desempeña un papel en la interacción con el sistema. Las instancias de actores pueden ser individuos o sistemas externos; sin embargo, cada actor proporciona una perspectiva única e importante del sistema, una perspectiva que es común a cada instancia (persona real/usuario) del actor.

### Usuario real frente a actor de caso de uso

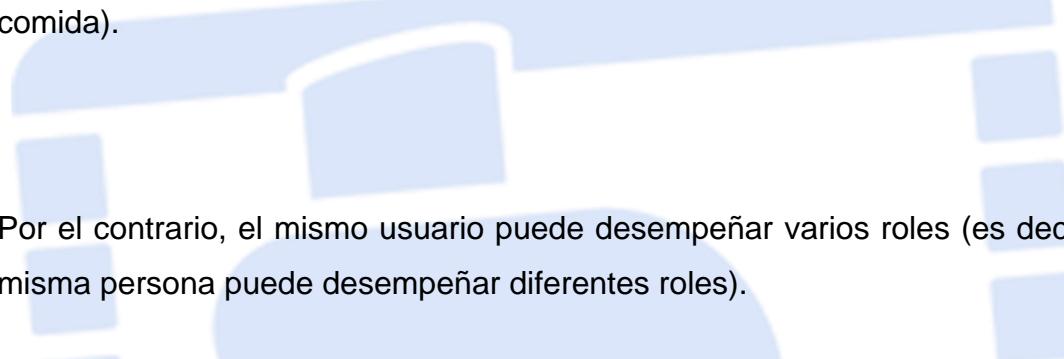
Para comprender completamente el propósito del sistema, debe saber para quién es el sistema, es decir, quién lo usará. Los diferentes tipos de usuarios se representan como Actor (roles).

La diferencia entre un rol y un usuario individual es que un rol representa una clase específica de usuarios, en lugar de un usuario real. Diferentes usuarios

pueden desempeñar el mismo rol, en cuyo caso cada usuario constituye una instancia de un actor.

Esta distinción entre actores e instancias de actores se ilustra a continuación:

La siguiente figura muestra un caso en el que Mary y John son clientes de una máquina expendedora. Cuando utilizan la máquina expendedora, cada uno está representado por una instancia de un actor llamado cliente que espera tener acceso a ciertas funciones del sistema (en este caso la impresión de comprar comida).



Por el contrario, el mismo usuario puede desempeñar varios roles (es decir, la misma persona puede desempeñar diferentes roles).

Por ejemplo, el Dr. Gates, que es miembro del comité de la Computer Society. Es responsable de administrar el sistema de administración de membresía, como agregar y eliminar cuentas de usuario. Cuando hace esto, desempeña un papel llamado Administrador (Actor). Sin embargo, el mismo Dr. Gates también puede ser miembro de la Computer Society. En este caso también desempeñaría un papel llamado "Miembro" (Actor)

### Cómo obtener los requisitos funcionales identificando los casos de uso del sistema

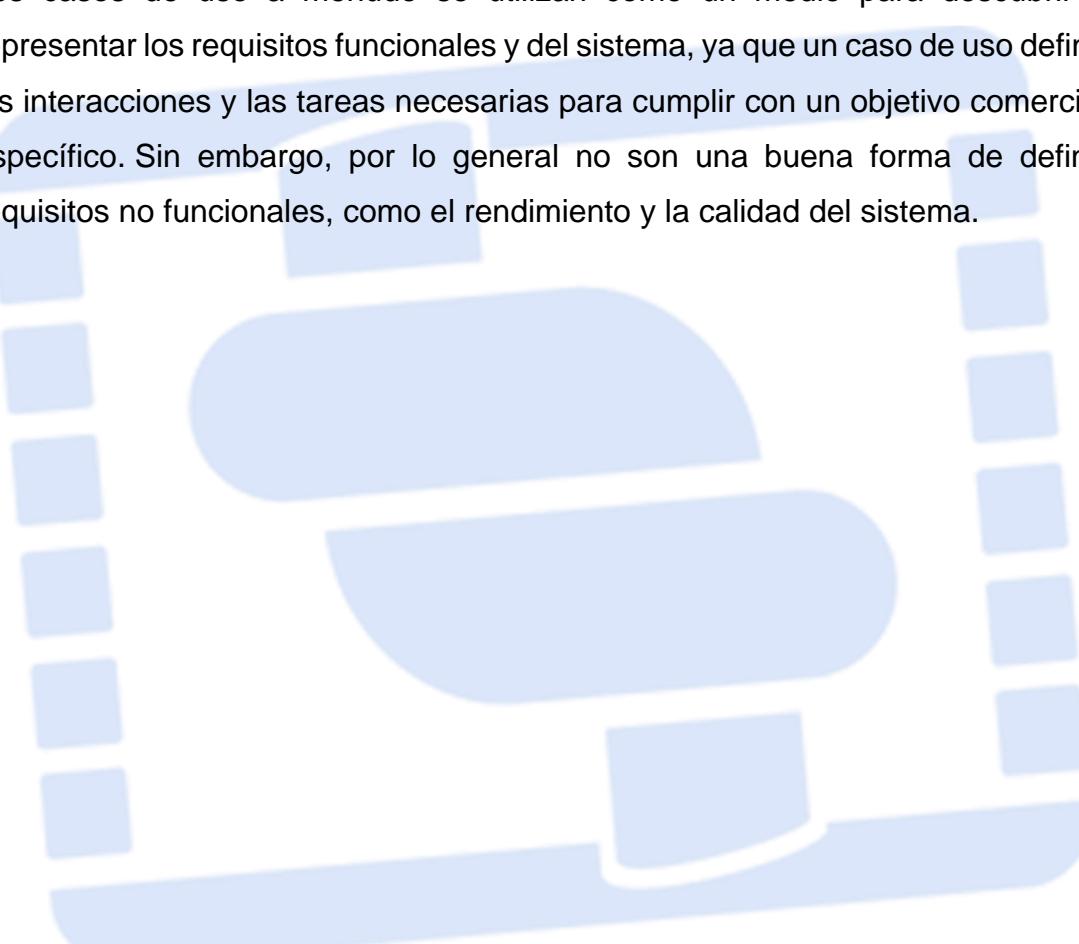
Se puede identificar un caso de uso haciendo a las partes interesadas los siguientes tipos de preguntas (a las que deben responder desde el punto de vista de los actores):

- ¿Qué intentan lograr los usuarios en este rol?
- Para cumplir con este rol, ¿qué deben poder hacer los usuarios?
- ¿Cuáles son las principales tareas de los usuarios en este rol?

- ¿Qué información necesitan los usuarios en este rol para examinar, crear o cambiar?
- ¿De qué debe informar el sistema a los usuarios con este rol?
- ¿Sobre qué deben informar al sistema los usuarios con este rol?

**Tenga en cuenta que:**

Los casos de uso a menudo se utilizan como un medio para descubrir y representar los requisitos funcionales y del sistema, ya que un caso de uso define las interacciones y las tareas necesarias para cumplir con un objetivo comercial específico. Sin embargo, por lo general no son una buena forma de definir requisitos no funcionales, como el rendimiento y la calidad del sistema.



## FUENTE:

- <https://www.uml.org/>
- <https://www.cybermedian.com/es/capturing-functional-requirements-with-use-cases-and-user-stories>
- <https://es.scribd.com/document/522919869/Guia-4-1-Casos-de-Uso-y-Requisitos>
- <https://www.oocities.org/gusdelact/ingsoft/temall.html>
- BURCH, John; GRUDNISKY, Gary. "Diseño de Sistemas de Información", Grupo Noriega editores.
- SENN, James A. "Análisis y diseño de sistemas de información", 2da. ed., McGraw-Hill.



