

## 1) Definiciones generales

### a) Describir qué es el desarrollo centrado en el usuario.

El desarrollo centrado en el usuario es un enfoque de diseño y desarrollo de sistemas que pone a las personas y sus necesidades en el centro del proceso. Se basa en comprender a fondo a los usuarios finales, sus contextos y las tareas que realizarán con el sistema. El objetivo es crear productos y servicios que sean intuitivos, eficientes y efectivos para los usuarios. Este enfoque incluye prácticas como la investigación de usuarios, pruebas de usabilidad, prototipado iterativo y evaluación continua con retroalimentación directa de los usuarios.

### b) Definir qué son los casos de uso y describa cómo se utilizan.

Los casos de uso son descripciones de cómo los usuarios interactúan con un sistema para lograr un objetivo específico. Son una herramienta en el análisis de requerimientos de sistemas que detalla las acciones que deben realizarse para que el sistema cumpla una función.

#### Utilización:

- Se utilizan para capturar los requisitos funcionales del sistema.
- Describen las interacciones entre los actores (usuarios u otros sistemas) y el sistema.
- Ayudan a los desarrolladores a entender lo que debe hacer el sistema.
- Proveen una base para las pruebas, garantizando que el sistema funcione según lo esperado.

### c) Definir qué es un actor y un escenario.

**Actor:** En un caso de uso, un actor es cualquier entidad que interactúa con el sistema. Puede ser un usuario, otro sistema o cualquier entidad externa que necesite interactuar con el sistema para lograr un objetivo. Por ejemplo, un “cliente” o un “administrador” puede ser un actor en un sistema de ventas.

**Escenario:** Un escenario es una secuencia específica de pasos que describa una interacción particular entre un actor y el sistema. Cada caso de uso puede tener múltiples escenarios, por ejemplo, escenarios principales (lo que sucede normalmente) y alternativos (lo que sucede en situaciones excepcionales o no estándar).

### d) Definir las relaciones que pueden presentarse en el diagrama de casos de uso. Describa cuándo se utiliza cada una.

**Inclusión (include):** Se utiliza cuando un caso de uso incluye explícitamente el comportamiento de otro caso de uso. El caso de uso base “incluye” el caso de uso común.

Cuando varias funciones comparten acciones comunes, y estas acciones pueden ser modeladas como un caso de uso independiente. Ejemplo: Un caso de uso “procesar algo” podría incluir un caso de uso “verificar saldo disponible”.

**Extensión (extends):** Se utiliza cuando un caso de uso opcional puede agregar comportamiento a otro caso de uso en ciertos puntos, sin modificar el flujo principal.

Cuando hay comportamiento adicional opcional o condicional que solo se ejecuta bajo ciertas circunstancias. Ejemplo: Un caso de uso “hacer una compra” podría extenderse con “ofrecer descuento”, que solo ocurre si el cliente tiene un código de promoción.

**Generalización:** Indica una relación de herencia entre casos de uso o actores. Un caso de uso general puede ser especializado por otros casos de uso, que heredan las características del caso de uso general.

Cuando varios actores o casos de uso comparten comportamientos similares. Ejemplo: Un actor “Administrador” podría ser una generalización de “Usuario”, ya que el administrador hereda las acciones del usuario y tiene permisos adicionales.

e) Enumerar los beneficios de modelar requerimientos del sistema con casos de uso.

- **Claridad en los requerimientos:** Los casos de uso ayudan a definir claramente las funcionalidades que el sistema debe ofrecer, desde la perspectiva del usuario.
- **Comunicación efectiva:** Facilitan la comunicación entre los equipos técnicos (desarrolladores, testers) y no técnicos (clientes, usuarios).
- **Facilitan la creación de pruebas:** Proveen una base clara para diseñar pruebas funcionales, asegurando que el sistema funcione según las expectativas.
- **Focalización en los usuarios:** Se enfocan en lo que los usuarios finales necesitan, lo que ayuda a desarrollar un sistema que sea relevante y útil.
- **Modularidad:** Permiten descomponer el sistema en partes manejables, lo que facilita su desarrollo y mantenimiento.
- **Identificación de excepciones y alternativas:** A través de los escenarios alternativos, los casos de uso permiten anticipar y manejar situaciones excepcionales que podrían surgir en el uso del sistema.