

Adrián Tirado Ramos

1º De DAW

Entorno Desarrollo

TAREA 5.4



# ÍNDICE

DIAGRAMAS DE CASOS DE USO .....	2
INTRODUCCIÓN.....	2
¿QUÉ ES UN DIAGRAMA DE CASOS DE USO? .....	2
COMPONENTES DEL DIAGRAMA DE CASOS DE USO.....	2
<b>Actor</b> .....	2
<b>Caso de uso</b> .....	2
<b>Relaciones</b> .....	2
UTILIDAD DE LOS DIAGRAMAS DE CASOS DE USO .....	3
<b>Comunicación</b> .....	3
<b>Identificación de requisitos</b> .....	3
<b>Diseño Preliminar</b> .....	3
<b>Base para pruebas</b> .....	3
<b>Gestión de cambios</b> .....	3
CONCLUSIÓN.....	4

## Diagramas de casos de uso

### Introducción

En el mundo del desarrollo de software, una comprensión clara y concisa de los requisitos del sistema es esencial para el éxito del proyecto, para ello están los diagramas de casos de uso que son la herramienta básica para esta tarea.

El propósito de este estudio es investigar en detalle qué es un diagrama de casos de uso, cómo se utiliza en el desarrollo de software y por qué es esencial para este proceso.

### ¿Qué es un diagrama de casos de uso?

Un diagrama de casos de uso es una técnica de modelado utilizada en el desarrollo de software para representar las interacciones entre los actores (usuarios o sistemas externos) y el sistema de desarrollo.

Estos se centran en la funcionalidad del sistema desde la perspectiva del usuario y le ayudan a comprender qué puede hacer el sistema y cómo los usuarios pueden interactuar con él para lograr sus objetivos.

### Componentes del diagrama de casos de uso

Los elementos principales de un diagrama de casos de uso son:

#### Actor

Representa cualquier entidad que interactúa con el sistema, podría ser un usuario humano, otro sistema o un dispositivo externo.

#### Caso de uso

Describe la función o acción que realiza el sistema en respuesta a una solicitud de un actor, cada caso de uso debe proporcionar un valor agregado significativo a al menos un actor.

#### Relaciones

Las relaciones entre actores y casos de uso se representan mediante líneas que muestran las interacciones entre ellos.

Las más comunes incluyen asociaciones, extensiones e inclusiones.

## Utilidad de los diagramas de casos de uso

Los diagramas de casos de uso proporcionan numerosos beneficios y usos en el desarrollo de software.

### Comunicación

Proporciona un lenguaje común para discutir y comprender los requisitos del sistema entre desarrolladores, clientes y otras partes interesadas.

Y así, las presentaciones visuales y fáciles de entender facilitan la comunicación efectiva entre todos los involucrados en el proyecto.

### Identificación de requisitos

Ayudan a identificar y comprender los requisitos funcionales del sistema y cosas como qué acciones necesita realizar el sistema y quiénes son los usuarios involucrados en estas acciones.

Esto le da al equipo de desarrollo una comprensión clara de lo que se espera del sistema.

### Diseño Preliminar

Permite realizar un diseño preliminar del sistema identificando las principales características e interacciones entre los usuarios y el sistema.

Lo que ayuda a proporcionar una base sólida para el diseño y la implementación del sistema.

### Base para pruebas

Los casos de uso identificados en el diagrama forman una base sólida diseñar casos de prueba.

Cada caso de uso se puede traducir directamente a un escenario de prueba, lo que facilita la creación y ejecución de pruebas para garantizar la calidad del software.

### Gestión de cambios

Facilita la gestión de cambios en los requisitos del sistema al proporcionar una descripción clara de cómo los cambios afectan otras partes del sistema.

Esto minimiza los riesgos asociados con los requisitos cambiantes y mantiene a todas las partes informadas y capaces de tomar decisiones informadas.

### Conclusión

En resumen, los diagramas de casos de uso son una herramienta invaluable en el desarrollo de software.

Estos permiten la recopilación, el análisis y la comunicación eficaces de los requisitos del sistema, garantizando que el producto final cumpla con las expectativas del cliente y del usuario final.

Su capacidad para representar visualmente las interacciones entre usuarios y sistemas lo convierte en una herramienta esencial para los equipos de desarrollo de software.