

# Diagramas para la especificación y análisis de requisitos

José Ignacio Botero Osorio  
**Instructor**

Lyam Acosta Forero  
**Aprendiz**

SENA – Servicio Nacional de Aprendizaje  
Tecnología en análisis y Desarrollo de Software  
3118526  
Bogotá D.C.

## **Contenido**

<b>Diagramas para la especificación y análisis de requisitos .....</b>	<b>3</b>
<b>Tipos de diagramas.....</b>	<b>3</b>
<b>Diagrama.....</b>	<b>4</b>

# Diagramas para la especificación y análisis de requisitos

La especificación y el análisis de requisitos para nuestro videojuego es una parte importante por ello utilizar diagramas nos ayudaría a una identificación clara de funcionalidades, actores y como interactúan entre si dentro del sistema del videojuego sirviendo de base para el diseño, la estimación y la validación del producto a desarrollar.

## Tipos de diagramas

- **“Diagrama de clases**

Un [diagrama de clases UML](#) representa un sistema estático orientado a objetos. Define proyectos por clases, atributos y funciones; como tal, es un bloque de construcción fundamental de cualquier solución orientada a objetos. Muestra las clases de un sistema y las operaciones de cada una de ellas.

- **Diagrama de paquetes**

Los [diagramas de paquetes](#) agrupan las clases en paquetes. Muestran las distintas dependencias y relaciones entre los paquetes de un sistema.

- **Diagrama de objetos**

Los [diagramas de objetos](#) son similares a los diagramas de clases en el sentido de que muestran las relaciones entre los objetos de un software. La diferencia es que los diagramas de objetos utilizan ejemplos del mundo real. Los diagramas de objetos también se denominan diagramas de instancia porque muestran el aspecto del sistema en un momento determinado.

- **Diagrama de componentes**

Un [diagrama de componentes UML](#) ofrece a los desarrolladores una comprensión global de los objetos físicos de un sistema. Este tipo de diagrama UML muestra la relación estructural entre cada componente físico y subcomponente de un sistema de software complejo. Esto ayuda a las partes interesadas a comprender cómo se organizan y conectan los componentes.

- **Diagrama de estructura compuesta**

Los diagramas de estructura compuesta visualizan la estructura interna de una clase. Estos diagramas desglosan la red de clases, interfaces y componentes. Este tipo de diagrama UML también muestra cómo interactúan estos elementos entre sí y por qué son esenciales para la estructura general del software.

- **Diagrama de despliegue**

Un diagrama de despliegue muestra la relación entre los componentes de software y hardware de un sistema. Representa la disposición física de los nodos en un sistema distribuido. Estos diagramas son especialmente útiles cuando el software que se está desarrollando funcionará en varios sistemas de hardware diferentes.”<sup>1</sup>

## Diagrama

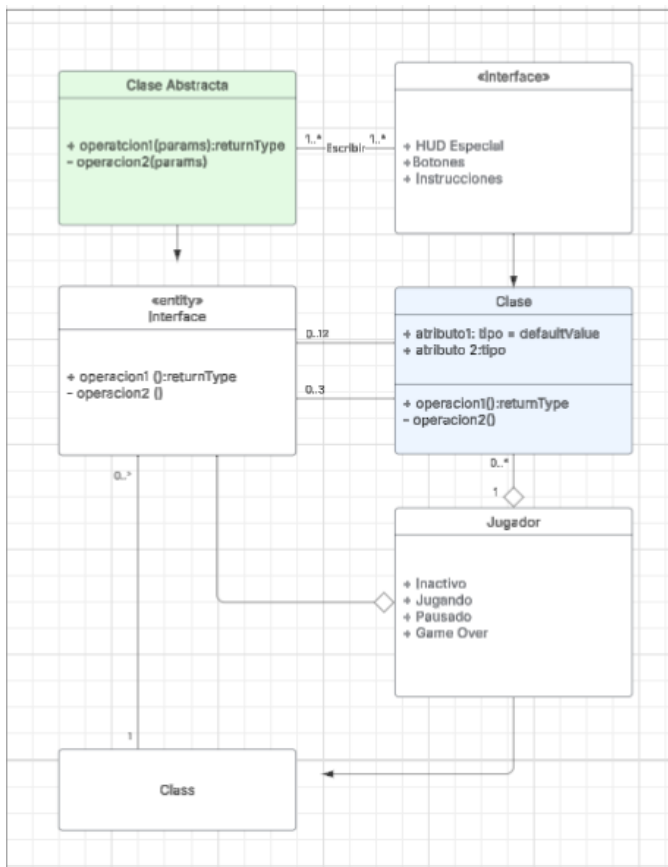


Ilustración 1 Diagrama UML

<sup>1</sup> <https://miro.com/es/diagrama/que-es-diagrama-uml/#diagramas-estructurales>