



**Universidad**  
Internacional  
de Valencia

**21GIIN - PROYECTOS DE PROGRAMACIÓN**

**Diagrama de clases y casos de uso**

**GRUPO 11**

Luis Valbuena Arribas

Benjamín Miguel Miralpeix

# DIAGRAMA DE CLASES

## Introducción

El siguiente diagrama de clases representa un modelo conceptual para un sistema de gestión logística. Está diseñado para organizar, gestionar y registrar actividades relacionadas con la logística, incluyendo operaciones, rutas, empaquetado, facturación y usuarios.

## Clases Principales

1. **Usuario:** Representa la información de un usuario en el sistema logístico. Contiene atributos como nombre, clave, tipo de usuario, teléfono y email. Incluye métodos CRUD para la gestión de usuarios.
2. **Operacion:** Clase que integra diferentes aspectos de la gestión logística. Tiene relación con clases de Ruta, Empaquetado y Facturacion. Implementa métodos CRUD para la gestión de operaciones logísticas.
3. **Ruta:** Atributos: id\_ruta, tipo de ruta, puertos de origen y destino, fechas de salida y llegada, coste de la ruta. Incluye método **agregarTrayecto()** para la gestión de trayectos adicionales a una ruta existente. Este método es particularmente útil cuando una ruta cambia de medio de transporte (por ejemplo, de marítimo a terrestre o aéreo) para que la mercancía pueda llegar a su destino final. Permite la flexibilidad de modificar el trayecto en cualquier punto, agregando nuevos segmentos o cambiando el tipo de transporte según las necesidades logísticas.
4. **RutaTerrestre, RutaMaritima, RutaAerea:** Representan distintos tipos de rutas logísticas con atributos y métodos específicos según su naturaleza (terrestre, marítima, aérea). Las subclases heredan atributos y métodos de la clase base Ruta. Implementan la interfaz **GestorDatos** para CRUD
5. **Empaquetado:** Contiene detalles del empaquetado de mercancías como el número de contenedores, peso y tamaño. Implementa métodos CRUD para la gestión de datos de empaquetado.
6. **Facturacion:** Representa información de facturación con atributos como código, fecha, tipo, importe, entre otros. Implementa métodos CRUD para la gestión de facturación.

## Relaciones y Herencia

- **Herencia:** Las subclases de Ruta (terrestre, marítima, aérea) heredan atributos y métodos de la clase base Ruta, reflejando diferentes tipos de rutas logísticas.
- **Asociaciones:** La clase Operacion está relacionada con las clases Ruta, Empaquetado y Facturacion, lo que indica la asociación de estas clases con las operaciones logísticas.

## Interfaces

- **GestorDatos:** Esta interfaz ofrece funcionalidades típicas de un sistema CRUD (Crear, Leer, Actualizar y Eliminar) para todas las clases que la implementen.

## Conclusiones

El diagrama de clases representa un sistema de gestión logística con clases para usuarios, operaciones, rutas, empaquetado y facturación. La implementación de interfaces y herencia proporciona una estructura modular y consistente para operaciones CRUD y la representación de diferentes tipos de rutas

# DIAGRAMA DE CASOS DE USO

## Estructura de las gráficas de casos de uso

Debido a la cantidad de casos de uso, hemos decidido crear una única gráfica, pero creando un conjunto de relaciones por cada uno de los actores, mostrándolo en imágenes diferentes, como si de gráficas diferentes se tratara.

### *Actores Internos del Sistema*

Los actores internos en este sistema de gestión logística son:

- **Administrador:** Encargado de la gestión de usuarios y tareas administrativas.
- **Coordinador de Rutas:** Responsable de la creación, modificación y gestión de rutas de transporte.
- **Contable:** Encargado de la gestión de facturas y operaciones contables.
- **Jefe de Almacén:** Responsable del manejo y gestión del empaquetado de la mercancía.

### ***Actores Externos***

El **Operador externo** tiene acceso a la consulta de información relevante, como facturas, rutas y estado del empaquetado.

### ***Funcionalidades Principales***

#### **Seguridad**

Este conjunto de casos de uso permite la gestión de seguridad del sistema, incluyendo la identificación, introducción, consulta, modificación y eliminación de usuarios. Es el Administrador el que puede acceder a todos ellos, aunque todo actor podrá acceder al caso de uso IDENTIFICARSE, para poder, después acceder a cualquier caso de uso que le competa.

#### **Rutas**

El Coordinador de Rutas es, principalmente, puede generar, modificar, eliminar y consultar rutas.

#### **Facturación**

El Contable será, principalmente, quien puede generar, modificar, eliminar y consultar facturas.

#### **Empaquetado**

El Jefe de Almacén gestiona el empaquetado, permitiendo la generación, modificación, eliminación y consulta de la información relacionada con los productos y su empaquetado. Este línea de casos de uso, incluye un paso extra diferente al resto de líneas, ya que un empaquetado incluye muchos productos y eso implica un bucle en el que poder ir añadiendo diferentes productos

### ***Interacciones entre Actores y Casos de Uso***

Los actores internos interactúan directamente con las distintas funcionalidades del sistema. Por ejemplo:

- El Administrador gestiona usuarios y realiza acciones de identificación, consulta, modificación y eliminación. Aunque no sea parte de sus funciones, realizar tareas del resto de actores, como administrador tiene acceso a todas las acciones que le compete a sus subalternos. Para evitar una sobre carga de flechas, hemos marcado este control sobre todas las acciones creando un control directo sobre los actores en las gráficas.

- El Jefe de Almacén interactúa con las operaciones relacionadas con el empaquetado.
- El Coordinador de Rutas lleva a cabo las operaciones de creación, modificación y consulta de rutas.
- El Contable maneja la gestión de facturas.

### ***Flujo de Acciones***

El flujo comienza con la generación o consulta de datos, pasando por la introducción o modificación y terminando con la verificación o confirmación de dichas operaciones. La selección de datos. Confirmación de operación o confirmación de eliminación de datos, se conciben como casos de usos comunes a todos los actores en sus diferentes responsabilidades, con el objetivo de evitar la duplicación de código en el momento de ponernos a programar.

Este diagrama de casos de uso proporciona una visión completa de las funcionalidades ofrecidas por el sistema, así como las interacciones entre los actores y las distintas operaciones que pueden llevarse a cabo en el entorno de gestión logística.

Enlaces: <https://github.com/BenjaLuke/ProyectoProgramacion>