

# TP FINAL POO2

## TERMINAL PORTUARIA

**Universidad Nacional De Quilmes | 2023**



### Integrantes:

- Gomez, Dario Gabriel – [gabigomez45@hotmail.com](mailto:gabigomez45@hotmail.com)
- Garcia, Axel – [axlgarcia080498@gmail.com](mailto:axlgarcia080498@gmail.com)

### Diseño:

Estructura de paquetes:

## Buque:

### State

En este paquete se encuentra la clase Buque, y sus distintas Fases. Además, cuenta con la clase GPS la cual modela la posición de dicho buque.

- **FaseBuque:** es la interfaz que se utiliza como estado, la cual posee un método abstracto (actualizarEstado) el cual debe ser implementado por las clases que la implementen.
- **OutBound:** es la fase inicial de todo buque.
- **Inbound:** esta fase es en la que se encuentra todo buque el cual su posición se encuentra a menos de 50km de la terminal de destino.
- **Arrived:** esta es la fase en la que se encuentra el buque el cual su posición coincide con la posición de la terminal de destino.
- **Working:** esta fase indica que el buque se encuentra en trabajo de carga y descarga
- **Departing:** esta es la fase en la que el buque se encuentra en una posición mayor a 1km de distancia de la terminal una vez concluidas las tareas.

## Circuito:

En este paquete se encuentran las clases **Circuito**, **Tramo**, y **Viaje**.

- **Viaje:** Posee un Circuito, un Buque en específico para ese viaje, un Puerto de origen, un Puerto de destino, una fecha de salida y una fecha de llegada.
- **Circuito:** Posee una lista de tramos, su puerto origen, la fechaYHoraDeSalida de dicho circuito, y su id.

Puede contestar a cuál es la duración de todo el circuito, a la cantidad de tramos que la componen, al costo total del circuito, y el tiempo que le toma llegar desde una terminal hacia otra.

- **Tramo:** Posee un puertoOrigen, unPuertoDestino, una duracion, un costo y una fechaYHoraDeSalida.

### Cliente:

En este paquete se encuentra las clases: **Cliente**, **Consignee**, **Shipper**, **Mail**, **Turno**.

- **Consignee:** Es quien recibe la carga. Posee una lista de buques los cuales son los que está interesado en recibir información (tiene una lista de Mails como notificación de dicha información sobre los buques).
- **Shipper:** Es quien envía la carga. Posee un Turno que es el día y hora en el que el camión debe llevar la carga el puerto. También posee una lista de Mails como notificación en el momento en que la carga este saliendo de la terminal.
- **Mail:** Posee un Buque, el cual es sobre el que se notifica. También tiene un nombre y una descripción.
- **Turno:** Posee una Orden a la cual está asociado dicho turno y una fechaYHora del turno.

### Container:

En este paquete se encuentran las clases **Container**, **ContainerDry**, **ContainerReefer**, **ContainerTanque**.

- **Container:** Es una clase abstracta la cual modela un container que posee un ancho, alto, largo. Y puede contestar a su tamaño total.
- **ContainerDry:** Es una subclase la cual extiende de la clase Container, y modela un tipo de container.
- **ContainerReefer:** Es una subclase la cual extiende de la clase Container, y modela un tipo de container, además este container posee una cantidad de consumo de kW por hora.
- **ContainerTanque:** Es una subclase la cual extiende de la clase Container, y modela un tipo de container.

### EmpresaTransportista:

En este paquete se encuentran las clases **Camión**, **Chofer**, **EmpresaTransportista**.

- **Camión:** Posee un Chofer y un Container y puede descargar el container en un puerto.
- **Chofer:** Posee un nombre.
- **EmpresaTransportista:** Posee una lista de camiones pertenecientes a dicha empresa.

### EstrategiaMejorRuta:

## Strategy

En este paquete se encuentran las clases **EstrategiaMejorRuta**, **EstrategiaMenorCantidadDeTerminales**, **EstrategiaMenorPrecio**, **EstrategiaMenorTiempo**.

- **EstrategiaMejorRuta**: Es una clase abstracta la cual tiene como método abstracto `elMejorCircuito()`, el cual retorna un `Circuito`. Además, puede responder a todos los circuitos filtrados por un destino en específico.
  - **EstrategiaMenorCantidadDeTerminales**: Es una subclase que extiende de **EstrategiaMejorRuta**, y en base a los circuitos con destino en un puerto específico puede retornar cual es el mejor circuito con menor cantidad de terminales.
  - **EstrategiaMenorPrecio**: Es una subclase que extiende de **EstrategiaMejorRuta**, y en base a los circuitos con destino en un puerto específico puede retornar cual es el mejor circuito con menor precio.
  - **EstrategiaMenorTiempo**: Es una subclase que extiende de **EstrategiaMejorRuta**, y en base a los circuitos con destino en un puerto específico puede retornar cual es el mejor circuito con menor tiempo.
- 
- **Strategy** -> `EstrategiaMejorRuta`
  - **ConcreteStrategy** -> `EstrategiaMenorCantidadDeTerminales`, `EstrategiaMenorPrecio`, `EstrategiaMenorTiempo`.
  - **Context** -> `Puerto`

## Filtros:

En este paquete se encuentran las clases: **Filtro**, **FiltroCombinado**, **FiltroFechaLlegada**, **FiltroFechaSalida**, **FiltroPuertoDestino**, **OperadorLogico**, **OperadorAND**, **OperadorOR**.

## **Composite:**

- Al momento de realizar la búsqueda se utilizan filtros, estos pueden ser simples o compuestos. Es por esto que se decidió utilizar dicho patrón de diseño.
- **Cliente** -> Viaje
- **Component** -> Filtro
- **Composite** -> FiltroCombinado
- **Leaf** -> FiltroFechaLlegada, FiltroFechaSalida, FiltroPuertoDestino

## **Strategy:**

- En el caso del FiltroCombinado nos encontramos que dependiendo de cuál sea el conector (**OR** o **AND**) se siguen diferentes estrategias, es por esta razón que se utilizó este patrón de diseño.
- **Strategy** -> OperadorLogico
- **ConcreteStrategy** -> OperadorOR, OperadorAND
- **Context** -> FiltroCombinado

## Naviera:

En este paquete se encuentra la clase **Naviera**

- **Naviera:** Esta clase posee una lista de buques, una lista de viajes y una lista de circuitos. Puede agregarse nuevos buques y nuevos circuitos en ella.

### Orden:

En este paquete se encuentra la clase **Orden**, **OrdenExportacion** y **OrdenImportacion**.

- **Orden:** Posee un Viaje, un Container, un Cliente, un Puerto, una lista de Servicios, un Chofer asignado, un Camión asignado y un Turno. Esta clase puede responder a la fechaSalidaCarga (fecha de salida del viaje) y a la fechaLlegadaCarga (fecha de llegada del viaje), y al total del costo de los servicios que la componen.
- **OrdenExportacion:** Es una subclase que extiende de **Orden** y puede crear un tipo de orden de exportación.
- **OrdenImportacion:** Es una subclase que extiende de **Orden** y puede crear un tipo de orden de exportación.

### Servicio:

En este paquete se encuentra la interfaz **Servicio**, y las clases **Pesado**, **Lavado**, **Electricidad**, y **AlmacenamientoExcedente**.

- **Servicio:** Es una interfaz la cual tiene como método costoServicio(Orden orden) el cual es implementado por cada clase que implemente dicha interfaz.
- **Electricidad:** Implementa la interfaz servicio y le otorga una implementación al método costoServicio(Orden orden).

- **Lavado:** Implementa la interfaz servicio y le otorga una implementación al método costoServicio(Orden orden). En base al tamaño del container declarado en la orden, su costo varía.
- **Pesado:** Implementa la interfaz servicio y le otorga una implementación al método costoServicio(Orden orden). Posee un costo de 200
- **AlmacenamientoExcedente:** Implementa la interfaz servicio y le otorga una implementación al método costoServicio(Orden orden). Posee un costoPorDia de 30.

### TerminalPortuaria:

En este paquete se encuentran la clase **Puerto** y **Terminal**.

- **Puerto:** Posee un nombre, una Lista de choferes habilitados, una Lista de consignee's, una Ubicación, una Lista de Navieras una Estrategia de mejor Ruta, Una lista de Ordenes y una Lista de buques. Puede dar aviso a clientes consignee's sobre el estado de buques de tipo inbound. Puede responder a cuál es el mejor circuito en base a una estrategia de mejor ruta. Y puede responder a su ubicación en un Point2D.
- **Terminal:** Posee un Puerto y puede validar la entrega de una carga en base a una orden, validando que el camión, el chofer y la hora de entrega sean válidas.