
**Trabajo integrador Programación con Objetos 2**

**Integrantes:**
**-Tomás Centurión: ferc721@gmail.com**
**-Martin Licciardello: martinlicciardello7@gmail.com**

**Profesores:**

- **Diego Torres**
- **Diego Cano**
- **Matias Butti**

**Fecha de entrega:**

- **23/11/2023**

## Decisiones de modelo:

Decidimos utilizar los patrones detallados en el siguiente apartado de “Patrones Encontrados”. Las búsquedas mediante filtros, se detallan a continuación:

Las búsquedas de rutas marítimas se realizan a través de varios filtros individuales:

- -Fitro And
- -Filtro Or
- -Filtro que incluye una fechaDeSalida
- -Filtro que incluye una fechaDeLlegada
- -Filtro que incluye un puertoDestino

Para realizar las búsquedas, nosotros elegimos el patrón de diseño composite, debido a que teníamos que cumplir varias condiciones acerca de buscar una ruta marítima, para que no se vuelva tan tedioso. Nos pareció que con el Patrón Composite se solucionan todos los conflictos posibles. El filtrado utiliza ciertas condiciones iniciales y se indican además los tipos de filtros.
Estos filtros pueden ser combinables mediante los filtros binarios And y Or los cuales se implementan a través de la interfaz Filtro.

**Organización:**

En nuestro equipo desglosamos las tareas y nos dividimos para hacer las clases y los tests. Por otro lado,utilizamos el framework mockito para realizar los tests, ya que este framework nos facilitó hacer los test de forma más eficiente y más rápida. Además siempre nos juntábamos los dos mediante Google Meet para poder codear juntos y mostrar lo que codeo cada uno(cuando no nos podíamos juntar).

## Dificultades que tuvimos:

Las dificultades que encontramos fueron las siguientes:

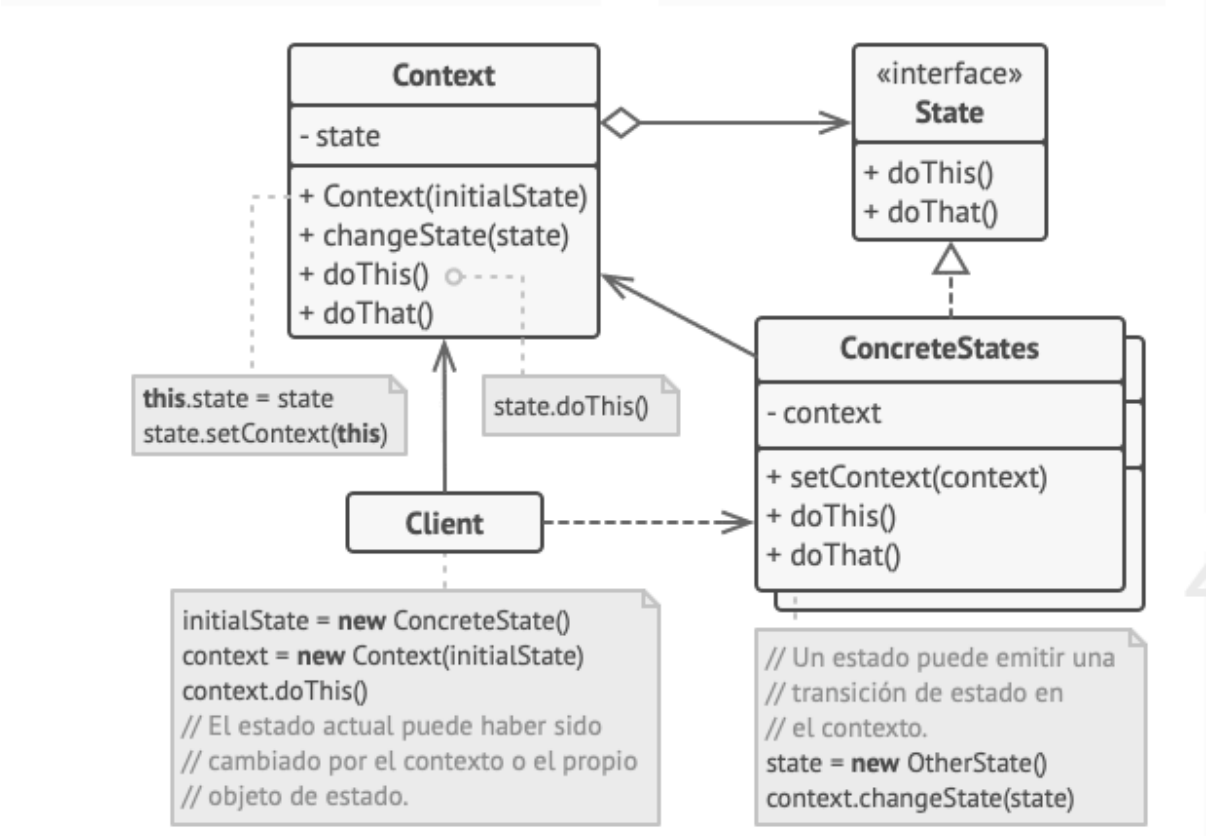
-Implementación de las operaciones

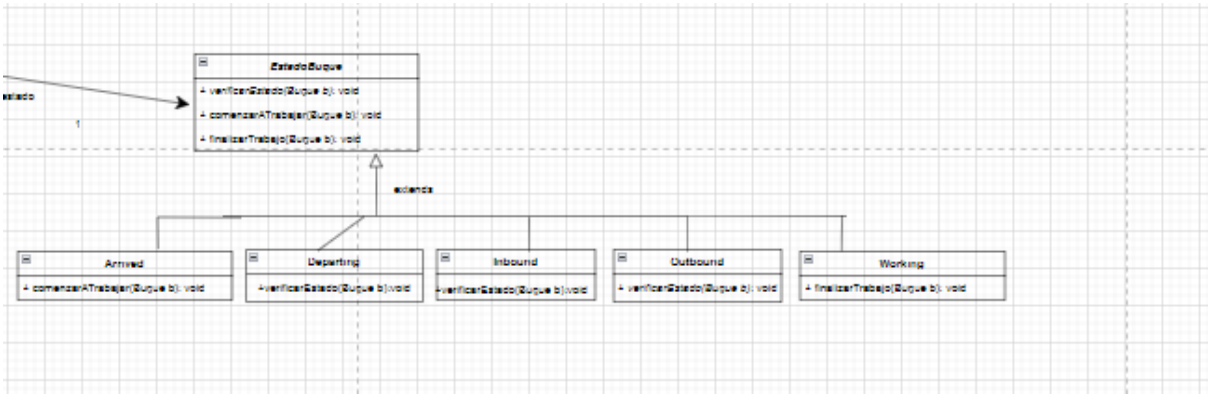
-Gestión de importación y exportación

-Se nos dificultó la división de tareas.

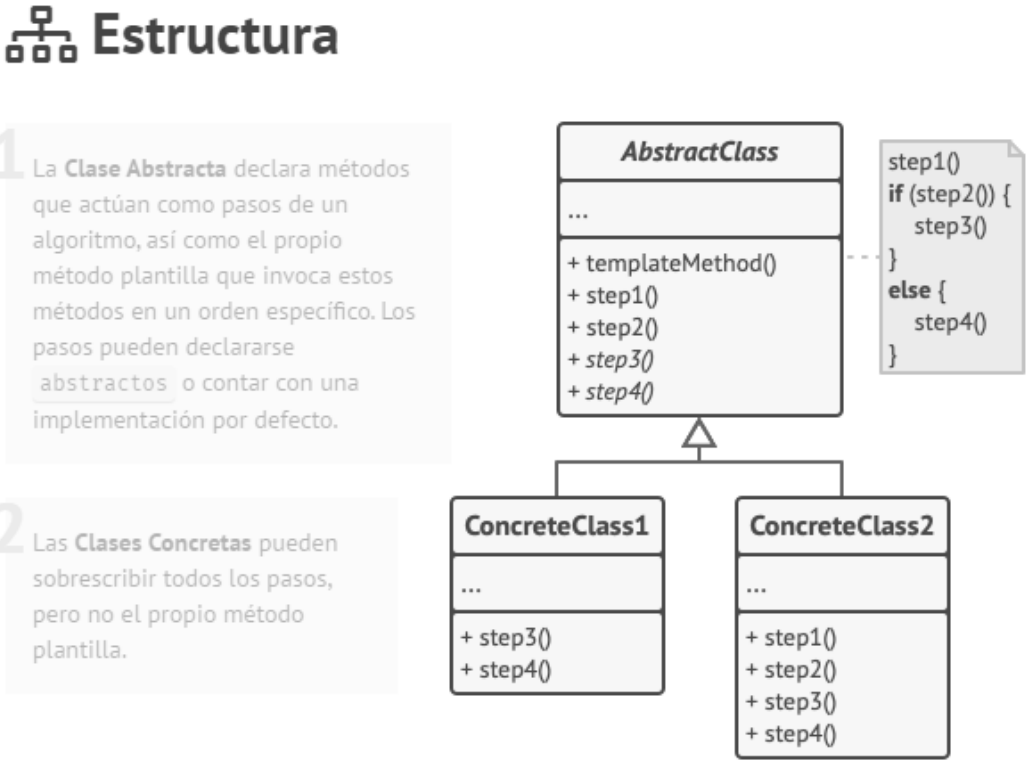
## Patrones encontrados:

## Patrón State



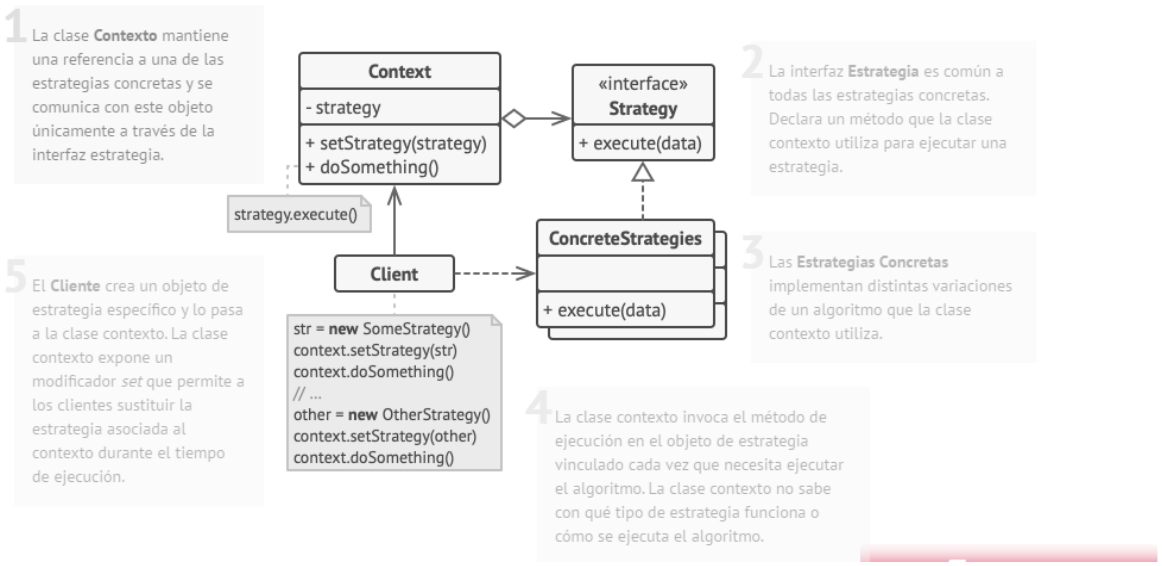


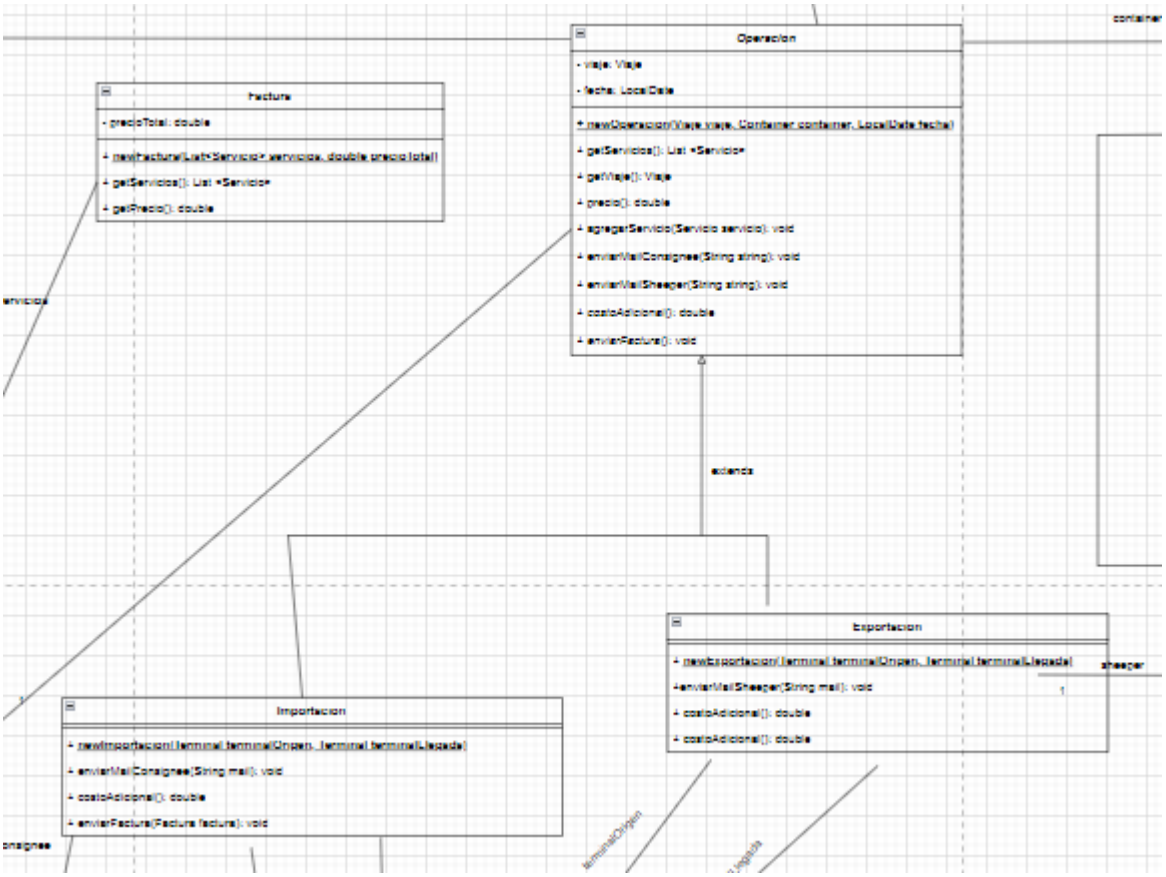

**Roles:**
**Cliente: Buque**
**Estado: EstadoBuque**
**EstadosConcretos: Inbound, Outbound, Working, Departing , Arrived.**



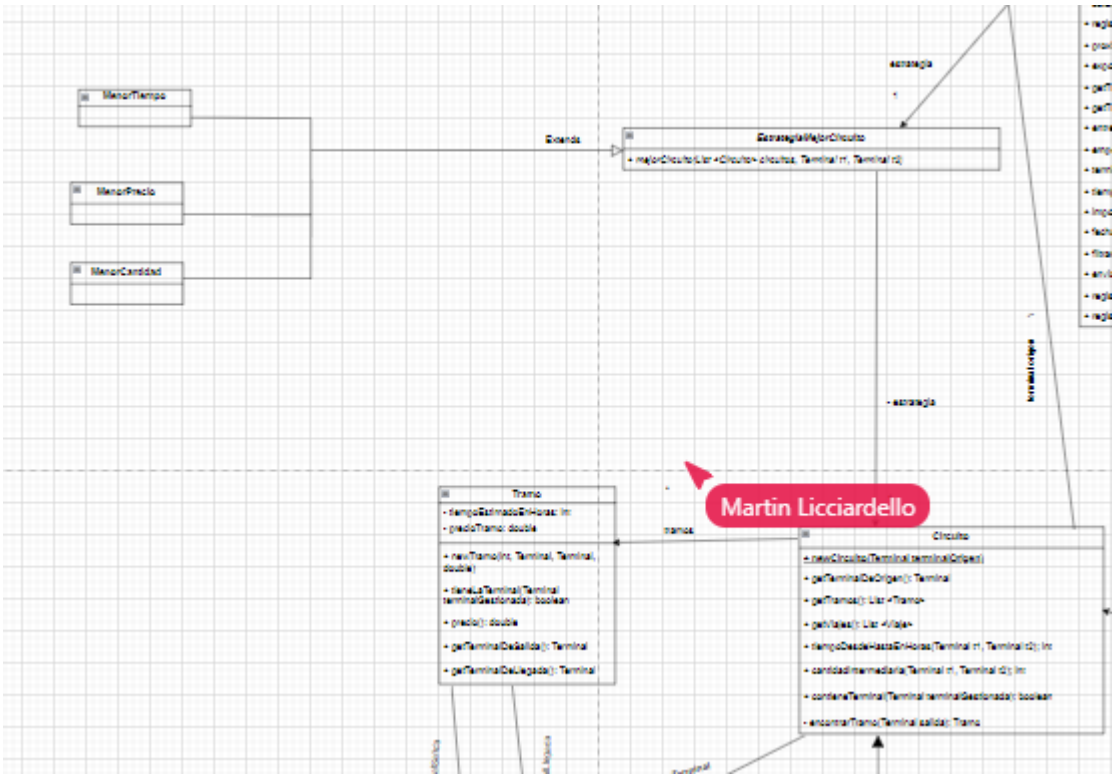

**Roles:**

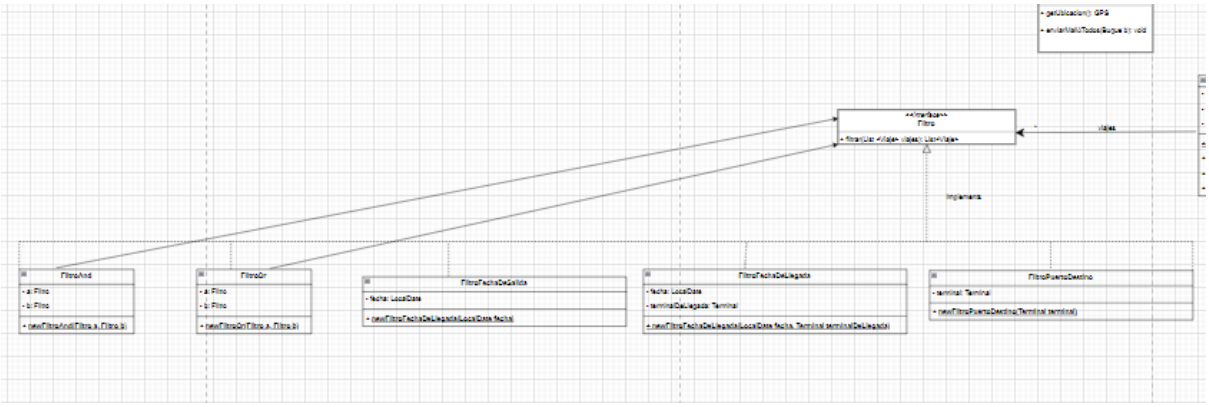
## Clase Abstracta: Operación
## Clases Concretas:Exportación e importación

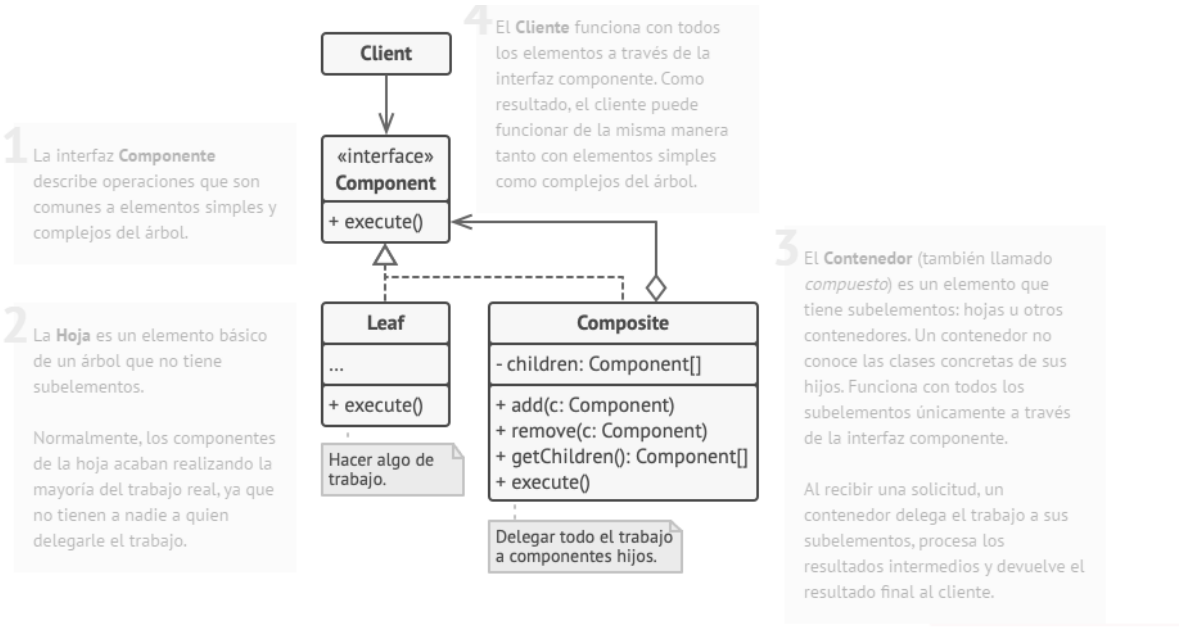


**Patrón Strategy**
**Roles:**
**Cliente: TerminalGestionada**
**Estrategia: EstrategiaMejorCircuito**
**EstrategiasConcretas: MenorTiempo, MayorTiempo, MenorCantidad**








**Roles:**
**Component:**
**Filtro**
**Composite: FiltroAnd, FiltroOr**
**Leaf: FiltroFechaDeSalida, FiltroFechaDeLlegada, FiltroFechaPuertoDestino.**