



TECNOLÓGICO NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO DEL VALLE DE OAXACA

DEPARTAMENTO DE: CIENCIAS ECONÓMICO – ADMINISTRATIVA

PROGRAMA EDUCATIVO: INGENIERÍA INFORMÁTICA

ASIGNATURA: DESARROLLO BACK-END

UNIDAD 1. PATRONES DE DISEÑO CREACIONALES

ACTIVIDAD 1.1 DIAGRAMAS DE CLASE

DOCENTE: CARDOSO JIMÉNEZ AMBROSIO

ESTUDIANTE: ESTHER MEJÍA ANTONIO

OCTAVO SEMESTRE GRUPO: I8A

CICLO: ENERO - JUNIO DE 2023

EX HACIENDA DE NAZARENO, XOXOCOTLÁN, OAXACA.

03 DE MARZO DEL 2023

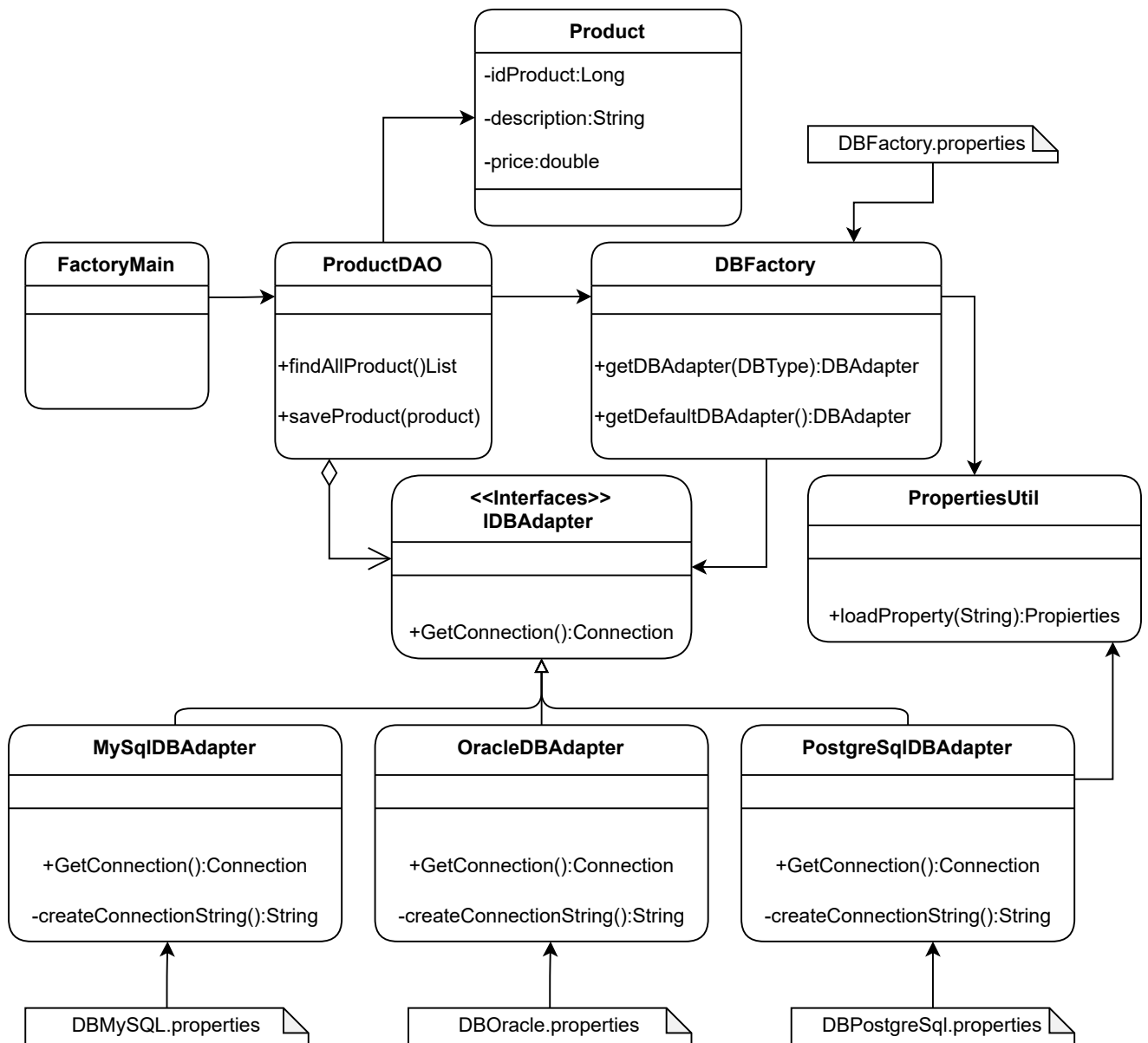
Introducción

El patrón de diseño es la solución a un problema de diseño, el cual debe haber comprobado su efectividad resolviendo problemas similares en el pasado, también tiene que ser reutilizable, por lo que se deben poder usar para resolver problemas parecidos en contextos diferentes.

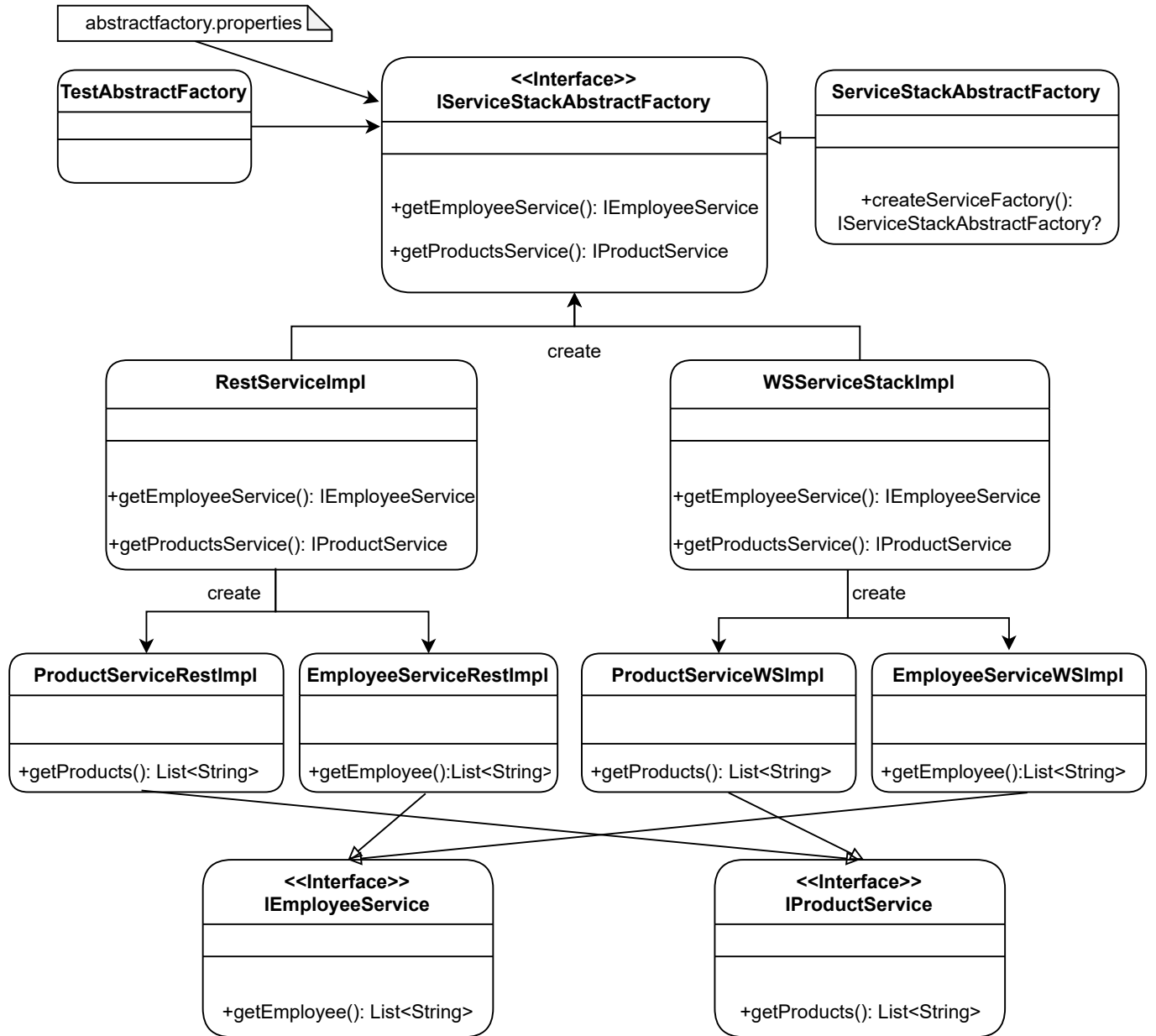
Existen tres tipos de patrones de diseño que son los creacionales, estructurales y de comportamiento, en ese caso se enfocara solo en los patrones creacionales, que son patrones de diseño relacionados con la creación o construcción de objetos, estos patrones intentan controlar la forma en que los objetos son creados implementando mecanismos que eviten la creación directa de objetos. Existen varios patrones, pero en ese documento estarán los diagramas de tres patrones en específico, la de Factory Method, Abstract Factory y Singleton.

- Patrón Factory Method: Patrón que se centra en la creación de una clase fábrica la cual tiene métodos que permiten crear objetos de un subtipo determinado.
- Patrón Abstract Factory: Patrón muy similar al Factory Method, sin embargo, este patrón permite crear objetos de una determinada familia de clases.
- Patrón Singleton: Patrón utilizado para controlar la creación de una clase determinada, de esta forma sólo se puede crear una única instancia en toda la aplicación.

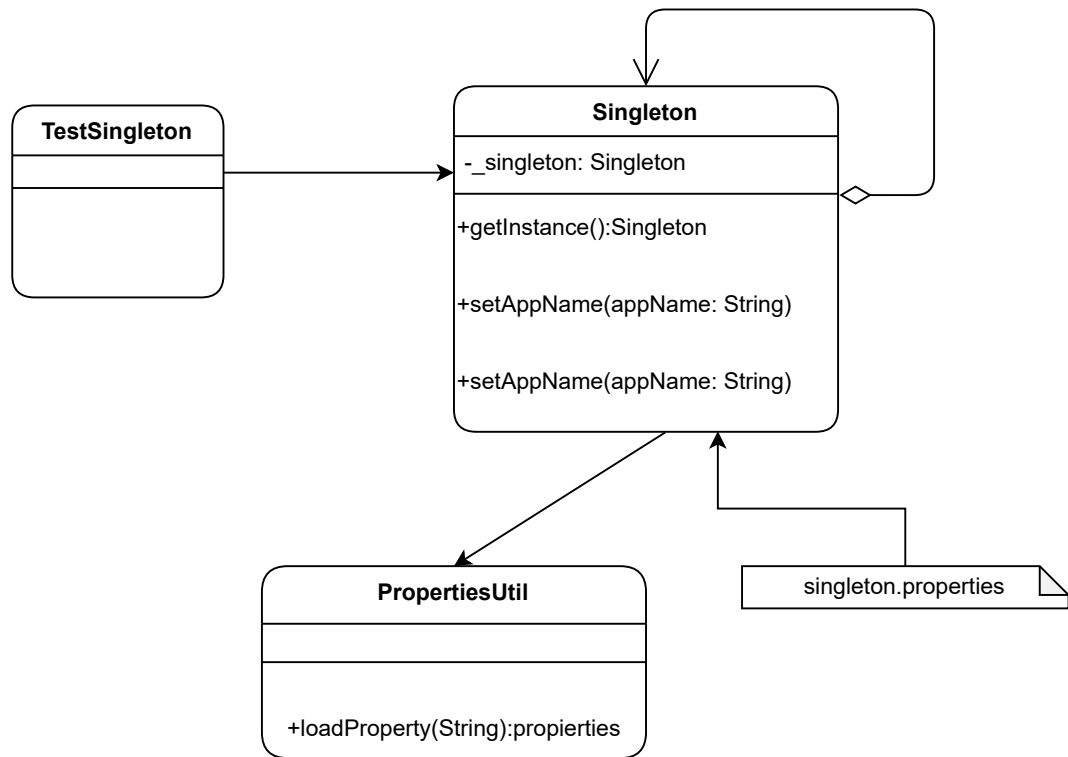
Patrón de diseño FactoryMethod - Implementación



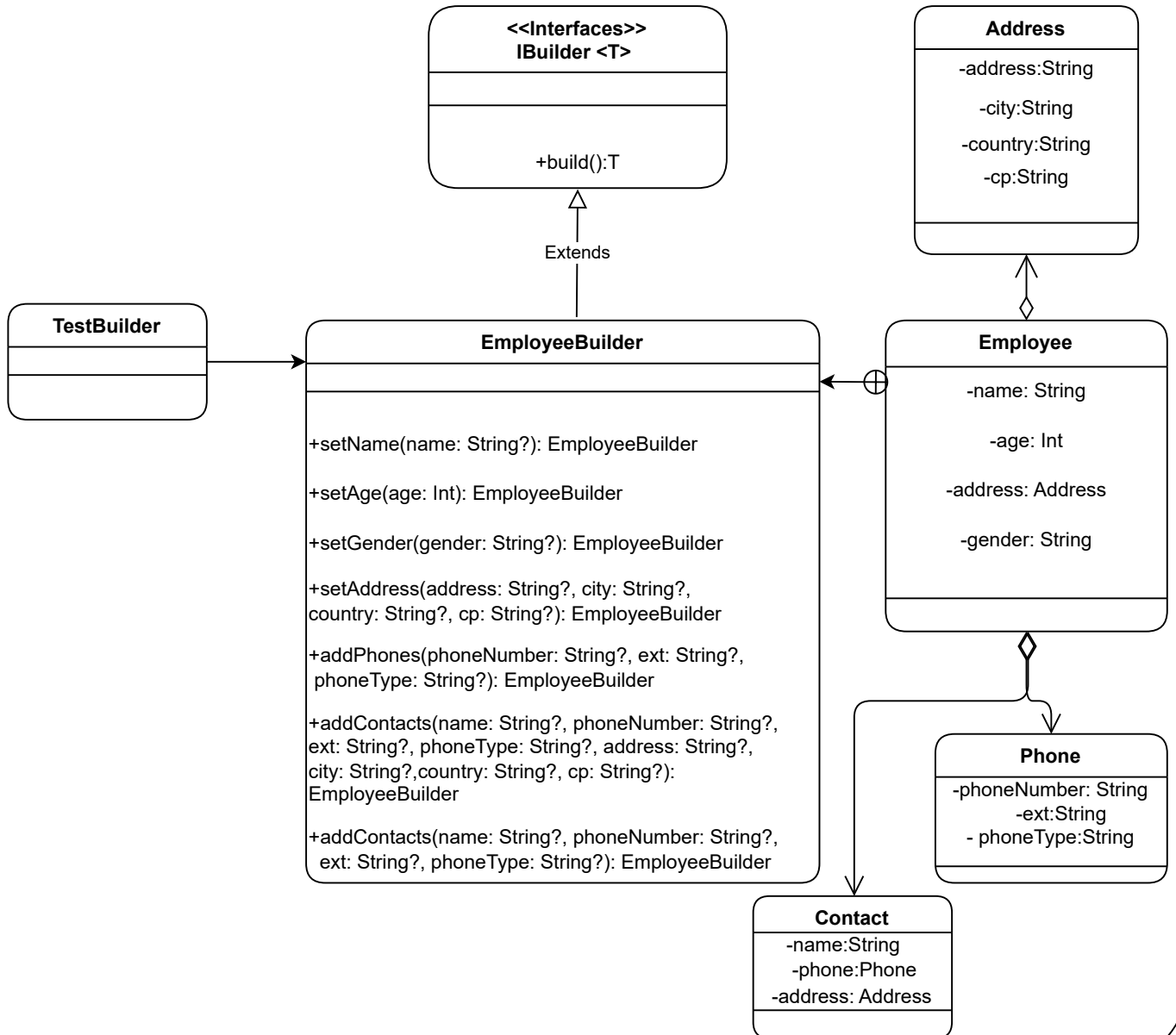
Patrón de diseño Abstract Factory- Implementación



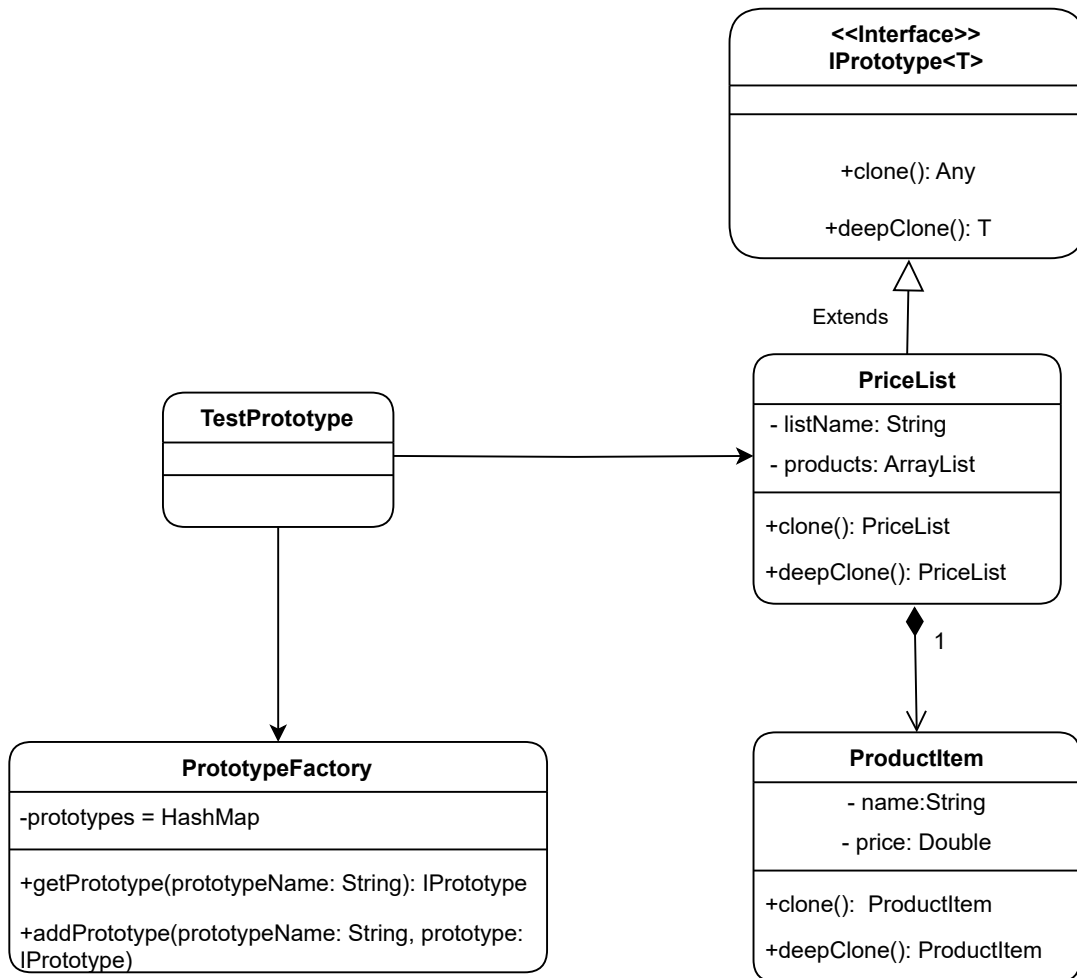
Patrón de diseño Singleton - Implementación



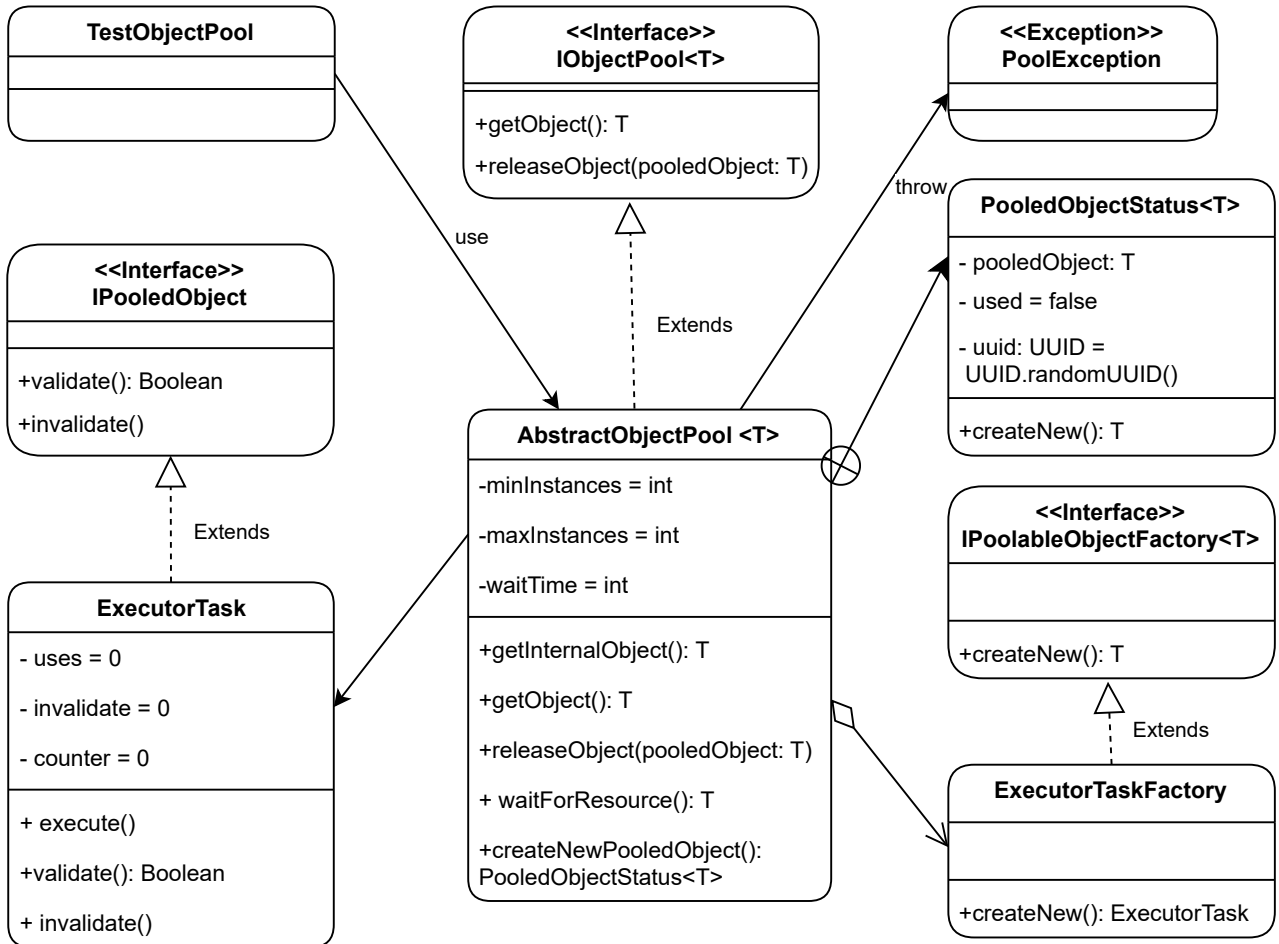
Patrón de diseño Builder - Implementación



Patrón de diseño Prototype - Implementación



Patrón de diseño Object Pool- Implementación





TECNOLÓGICO NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO DEL VALLE DE OAXACA

DEPARTAMENTO DE: CIENCIAS ECONÓMICO – ADMINISTRATIVA

PROGRAMA EDUCATIVO: INGENIERÍA INFORMÁTICA

ASIGNATURA: DESARROLLO BACK-END

UNIDAD 2. PATRONES DE DISEÑO ESTRUCTURALES

ACTIVIDAD 2.2 DIAGRAMA DE CLASES

DOCENTE: CARDOSO JIMÉNEZ AMBROSIO

ESTUDIANTE: ESTHER MEJÍA ANTONIO

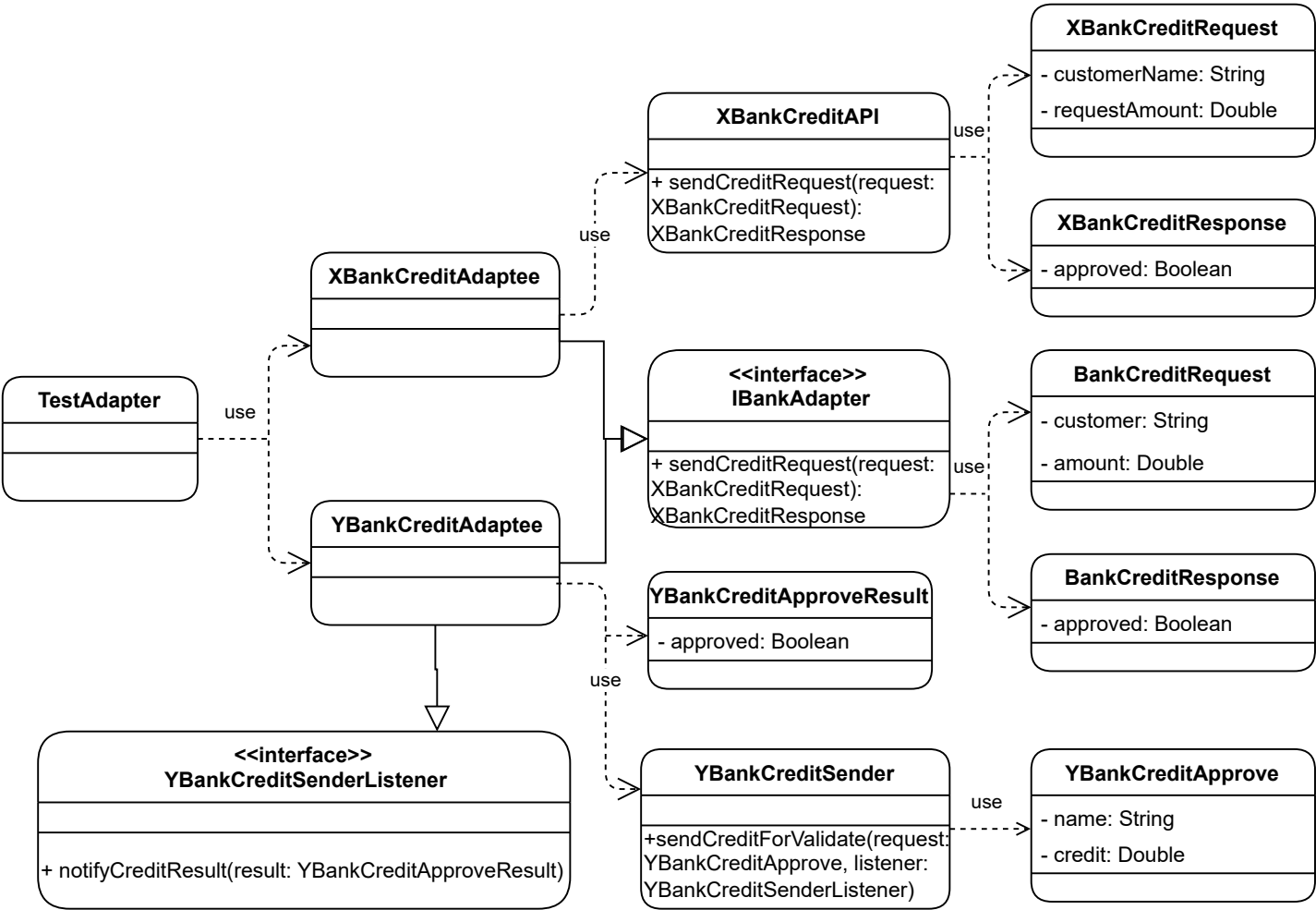
OCTAVO SEMESTRE GRUPO: I8A

CICLO: ENERO - JUNIO DE 2023

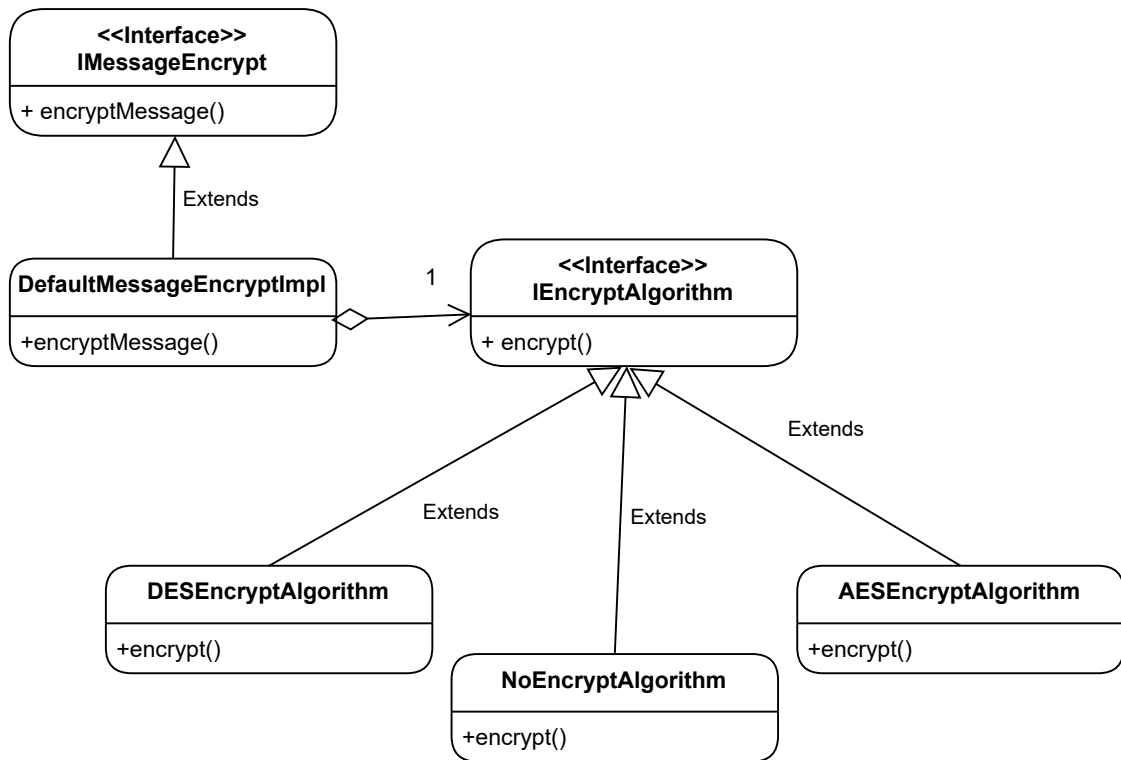
EX HACIENDA DE NAZARENO, XOXOCOTLÁN, OAXACA.

09 DE MAYO DEL 2023.

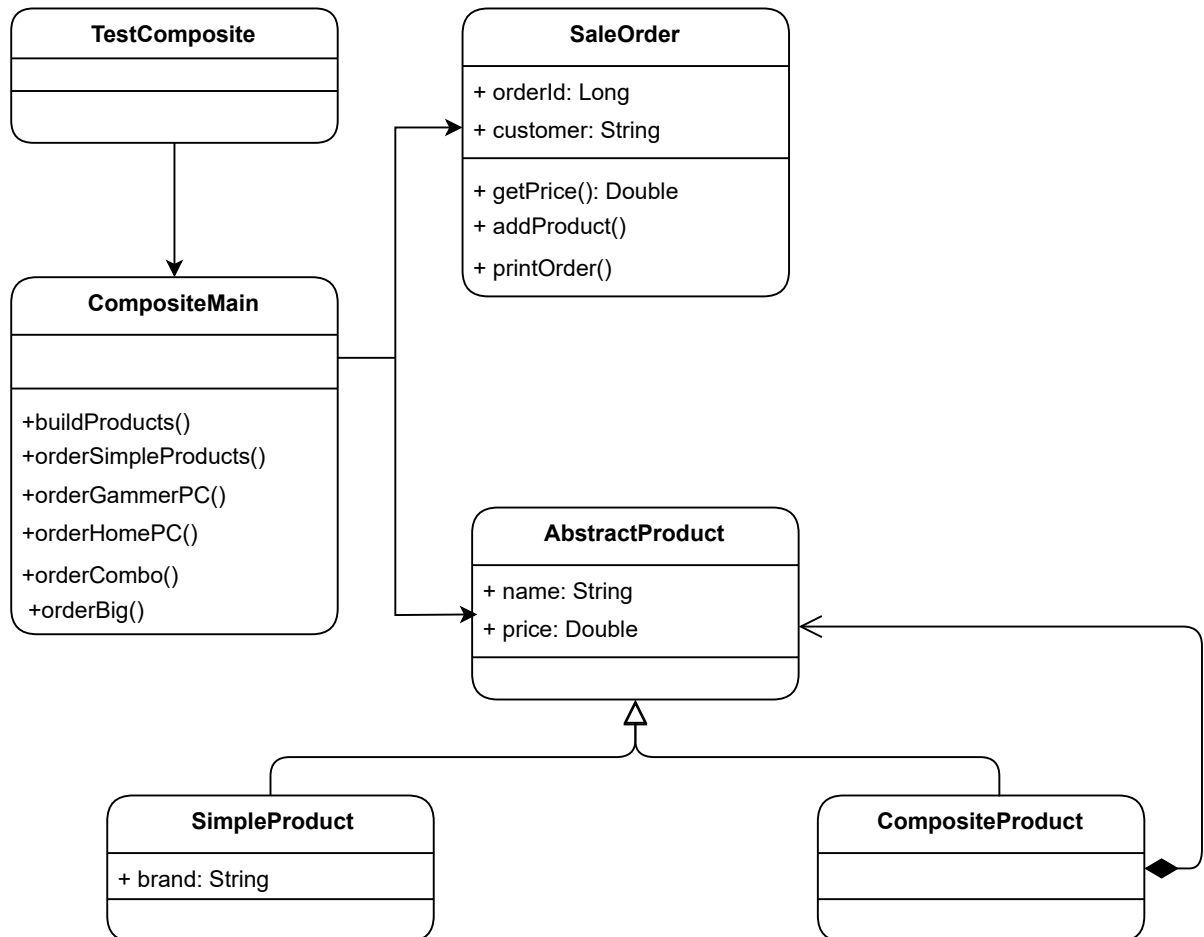
Patrón de diseño Adapter - Implementación



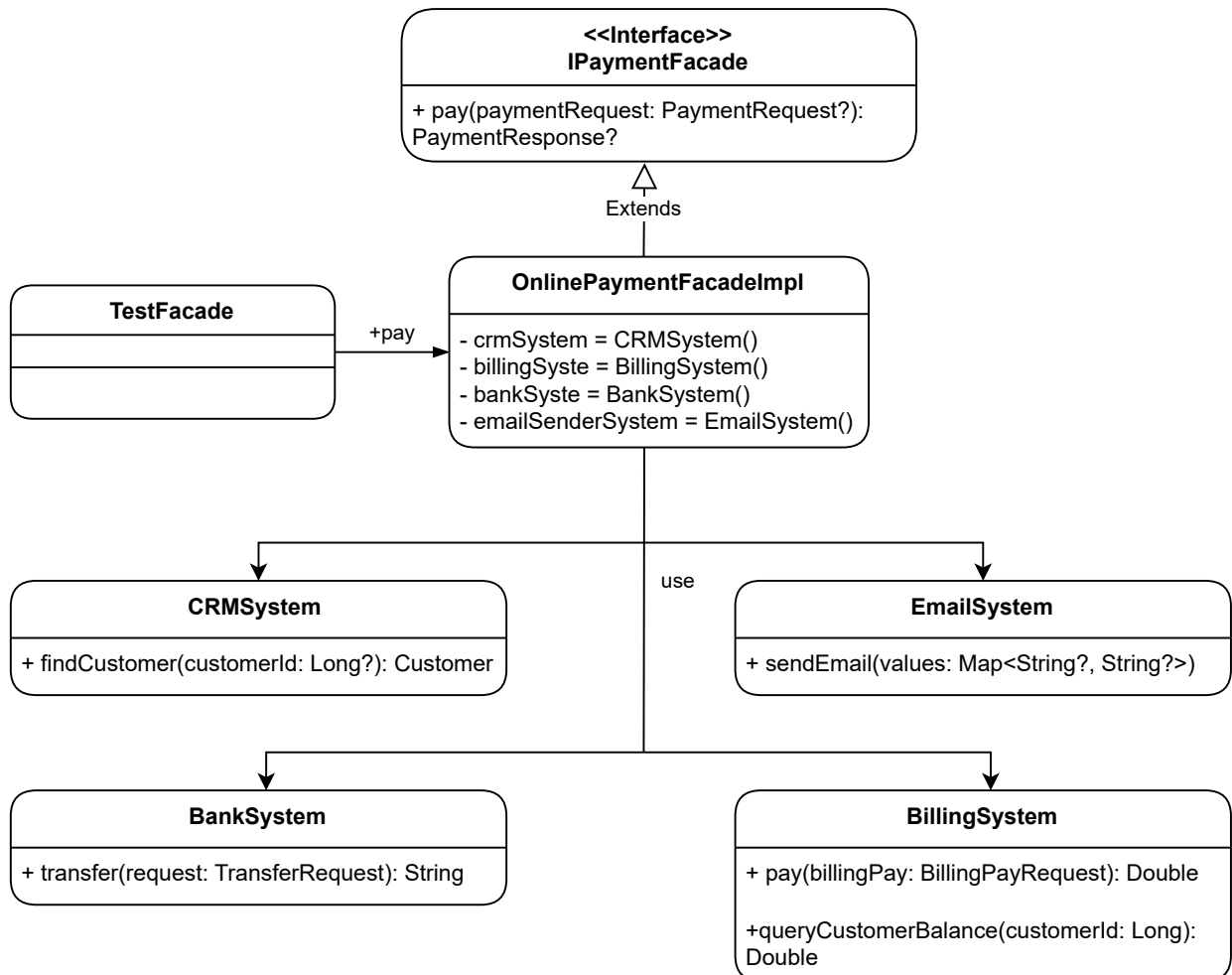
Patrón de diseño Bridge - Implementación



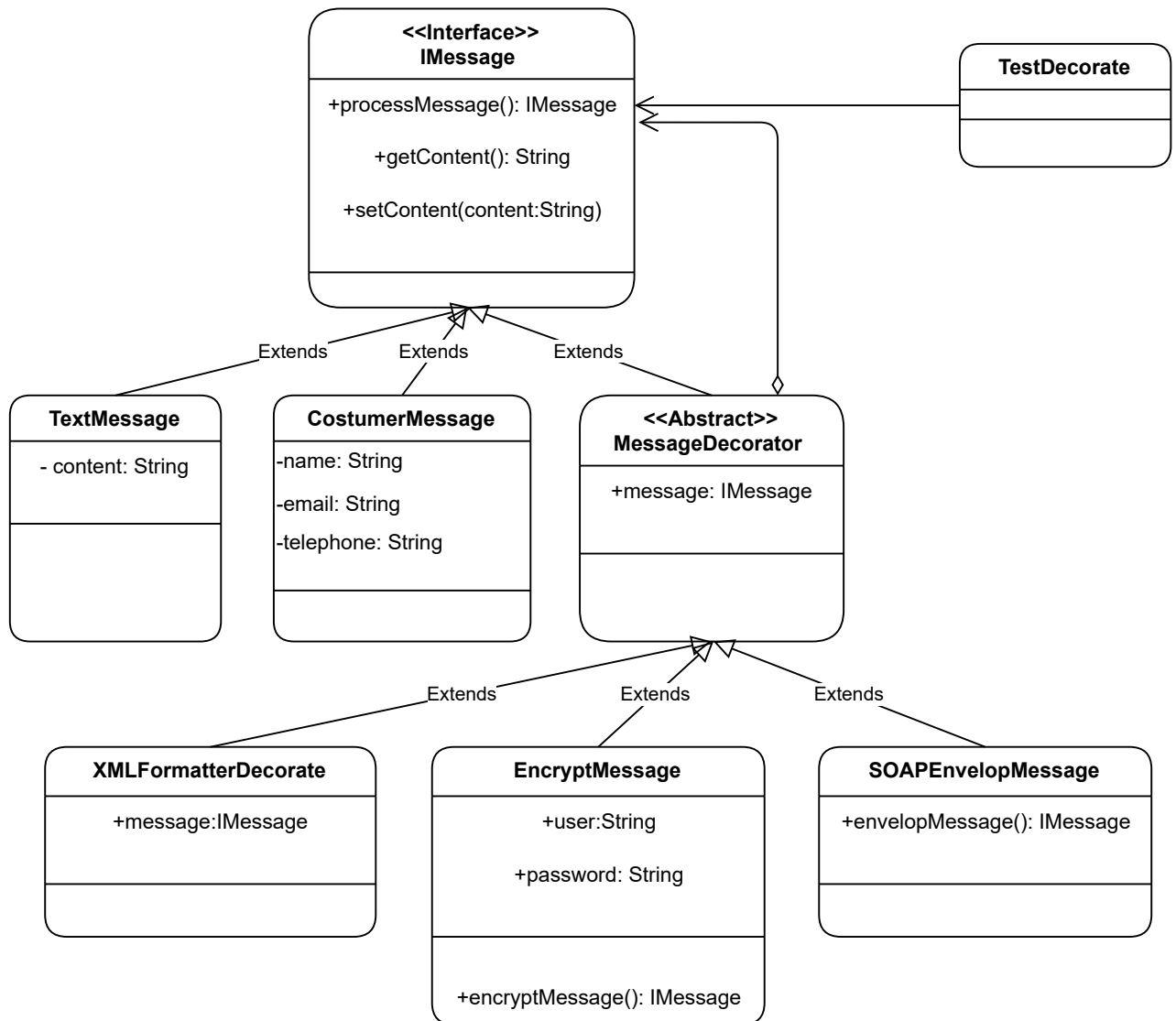
Patrón de diseño Composite - Implementación



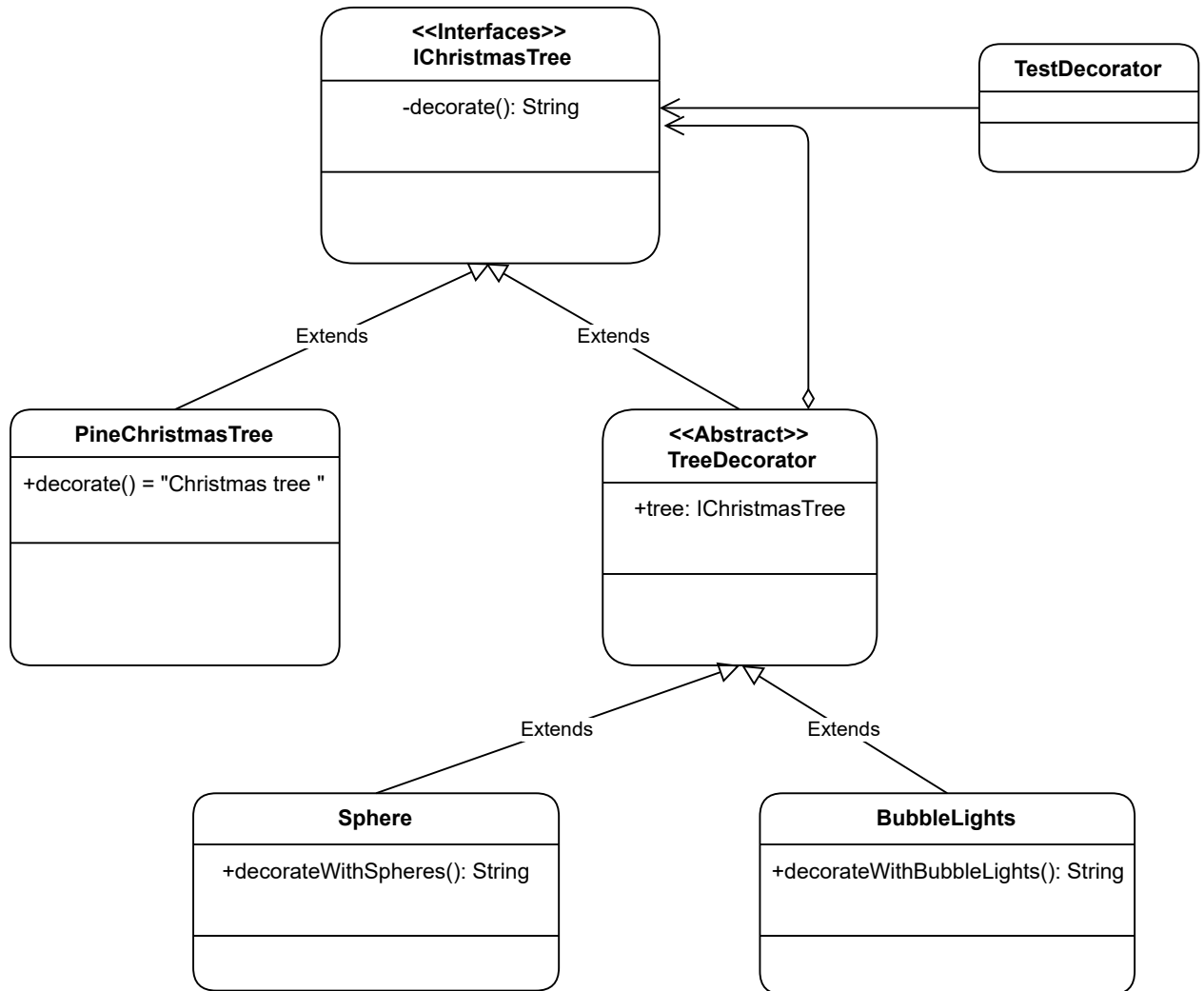
Patrón de diseño Facade - Implementación



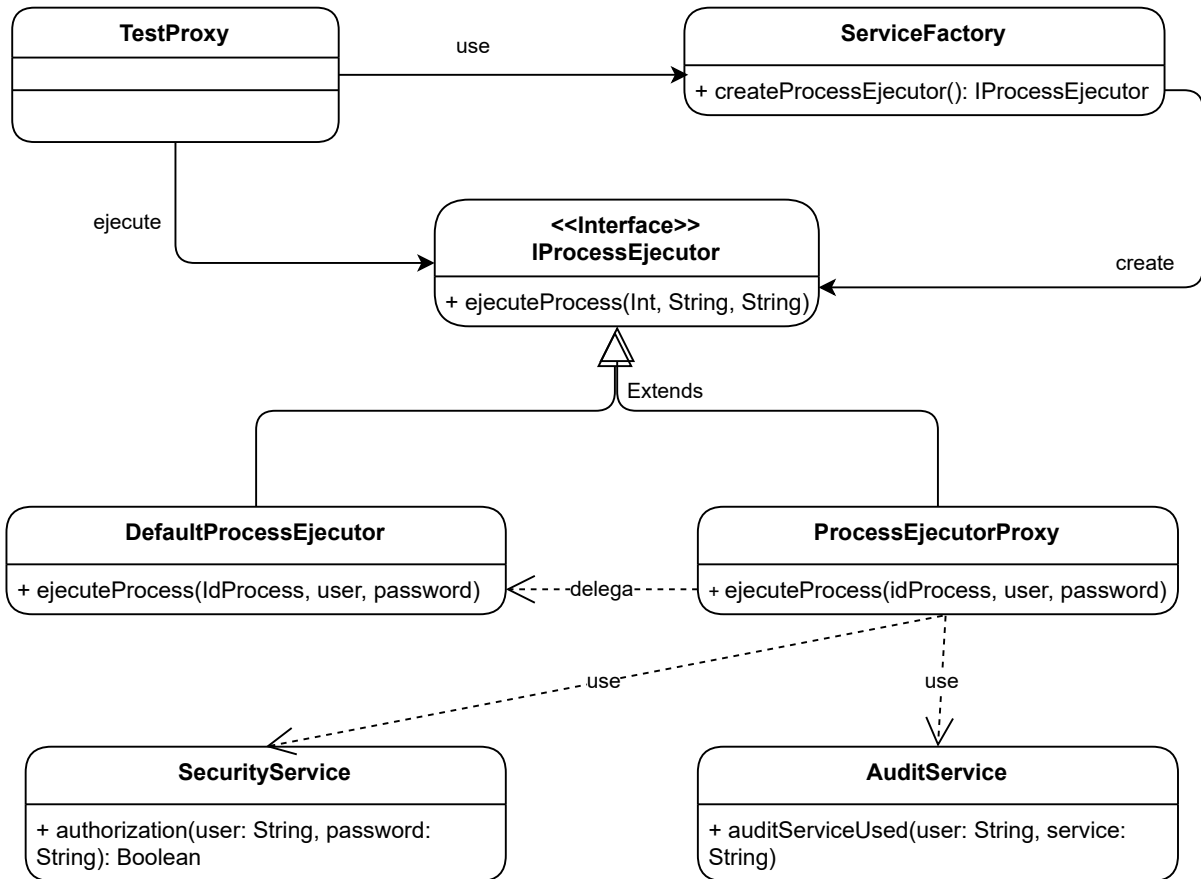
Patrón de diseño Decorator - Implementación



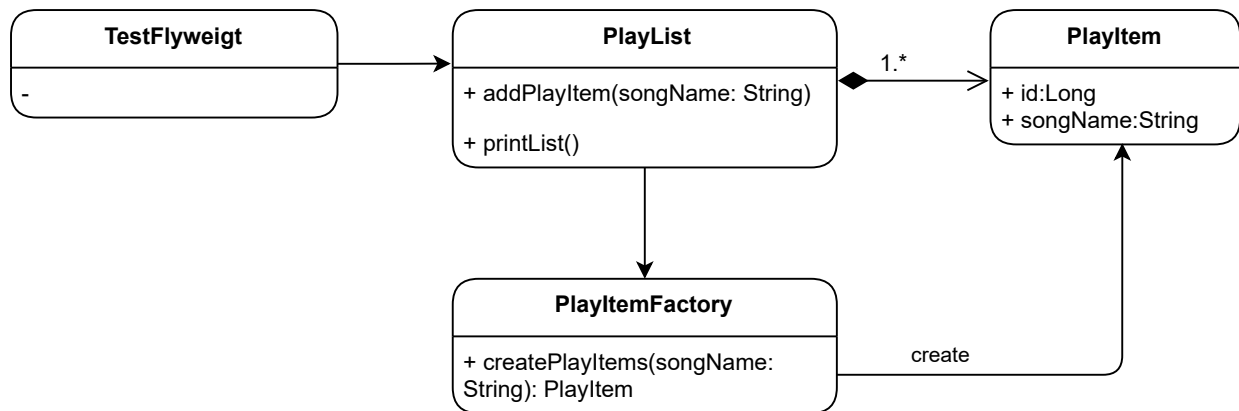
Patrón de diseño Decorator - Implementación
Versión 2 Árbol de Navidad



Patrón de diseño Proxy - Implementación

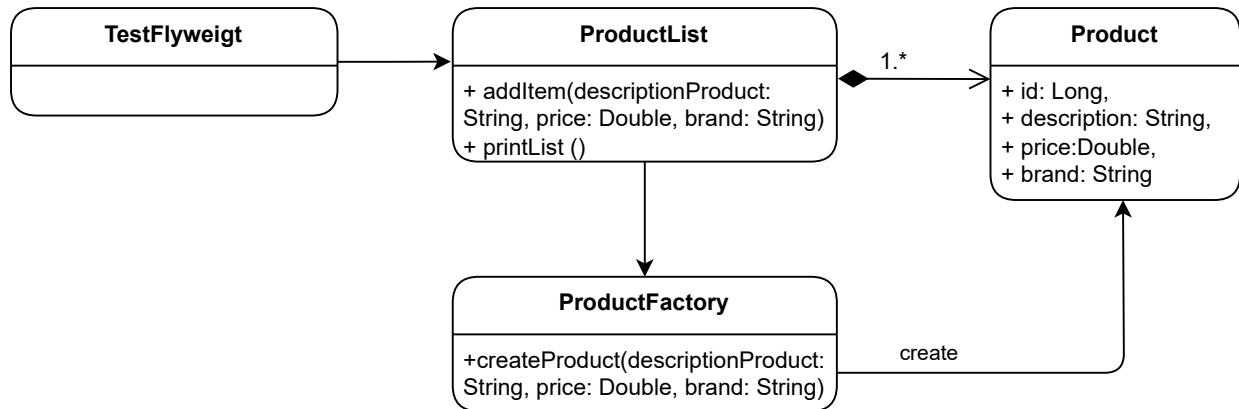


Patrón de diseño Flyweight - Implementación



Patrón de diseño Flyweight - Implementación

Versión 02



Enlace de los patrones en GitHub.

https://github.com/EstherAtiny/Backend_PD



TECNOLÓGICO NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO DEL VALLE DE OAXACA

DEPARTAMENTO DE: CIENCIAS ECONÓMICO – ADMINISTRATIVA

PROGRAMA EDUCATIVO: INGENIERÍA INFORMÁTICA

ASIGNATURA: DESARROLLO BACK-END

UNIDAD 3. PATRONES DE COMPORTAMIENTO

ACTIVIDAD 3.2 DIAGRAMA DE CLASES

DOCENTE: CARDOSO JIMÉNEZ AMBROSIO

ESTUDIANTE: ESTHER MEJÍA ANTONIO

OCTAVO SEMESTRE GRUPO: I8A

CICLO: ENERO - JUNIO DE 2023

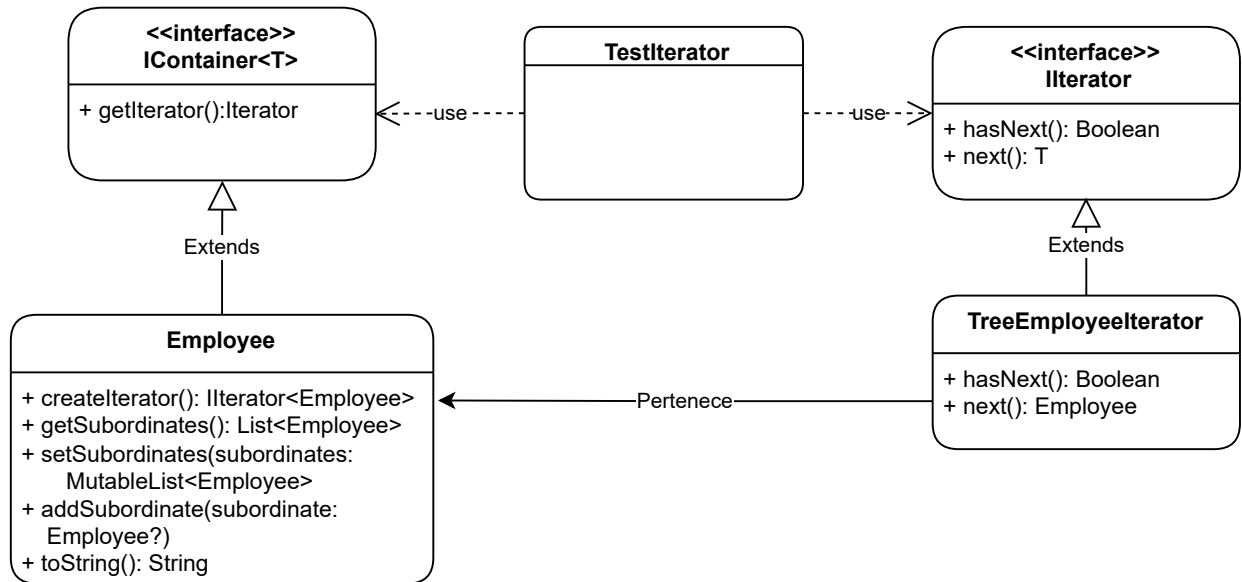
EX HACIENDA DE NAZARENO, XOXOCOTLÁN, OAXACA.

02 DE JUNIO DEL 2023.

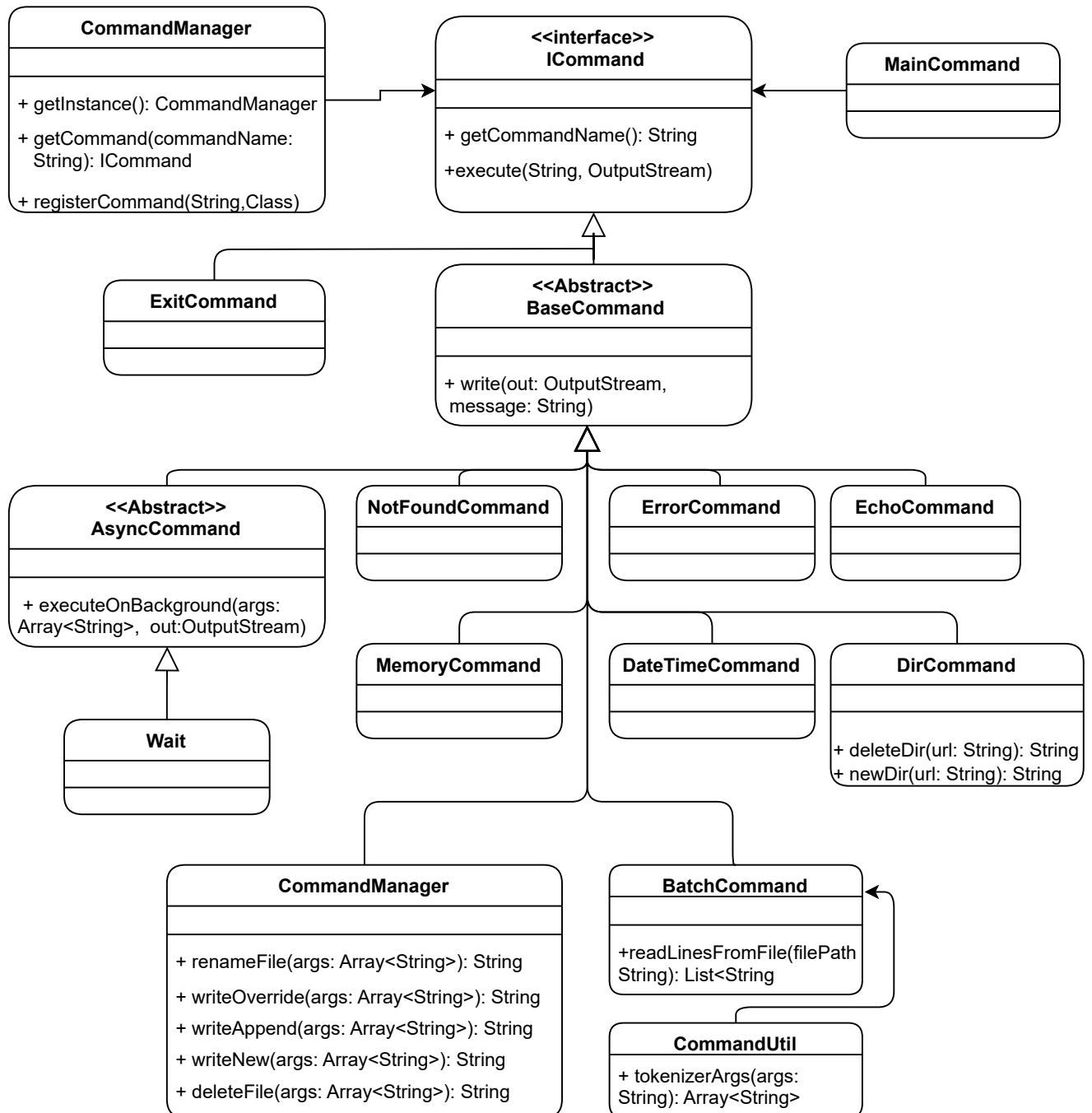
CONTENIDO

- 3.1 Iterator
- 3.2 Command
- 3.3 Observer
- 3.4 Template method
- 3.5 Strategy
- 3.6 Chain of Responsibility
- 3.7 Interpreter
- 3.8 Mediator
- 3.9 Memento
- 3.10 Null object
- 3.11 State
- 3.12 Visitor

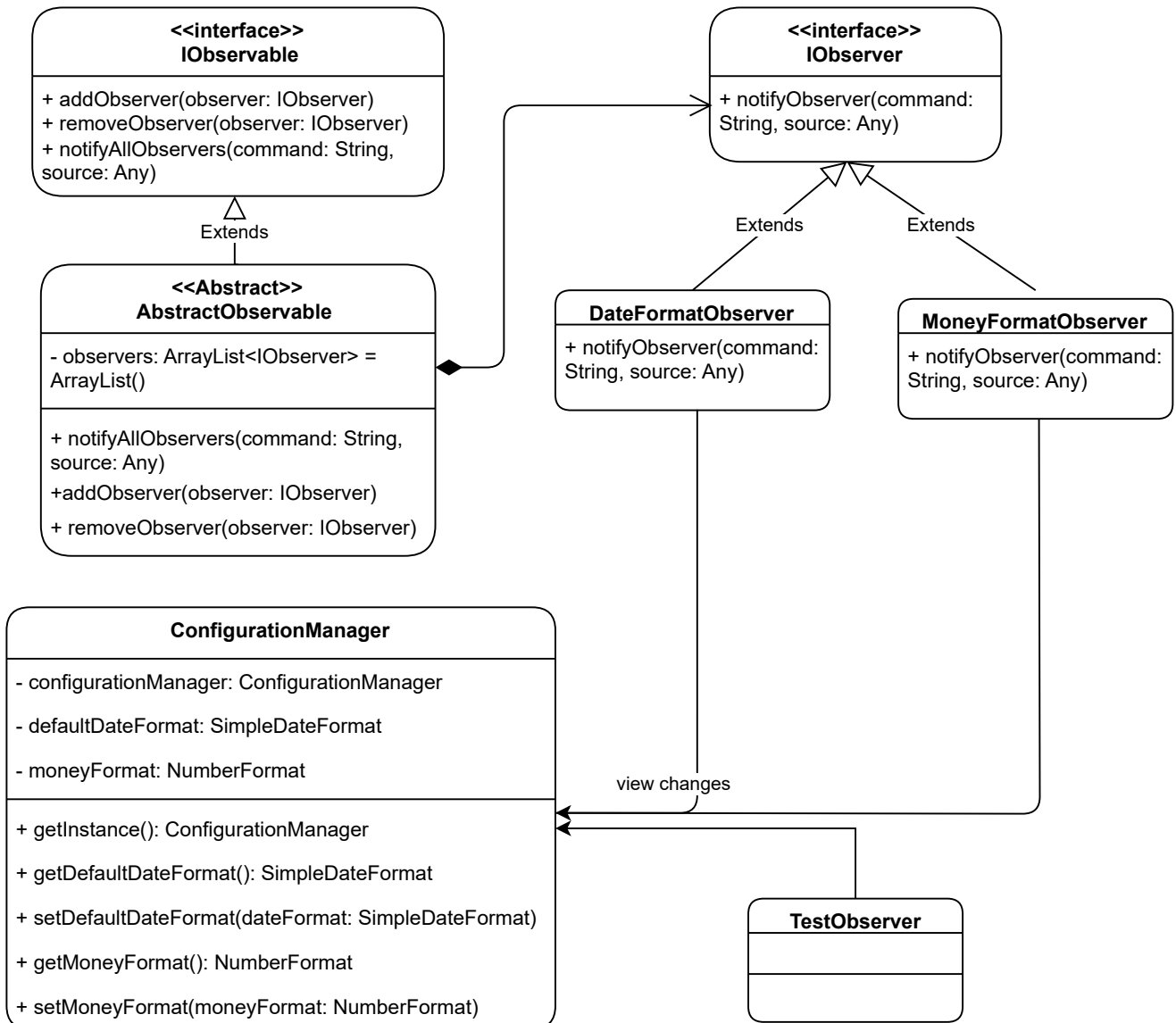
Patrón de diseño Iterator - Implementación



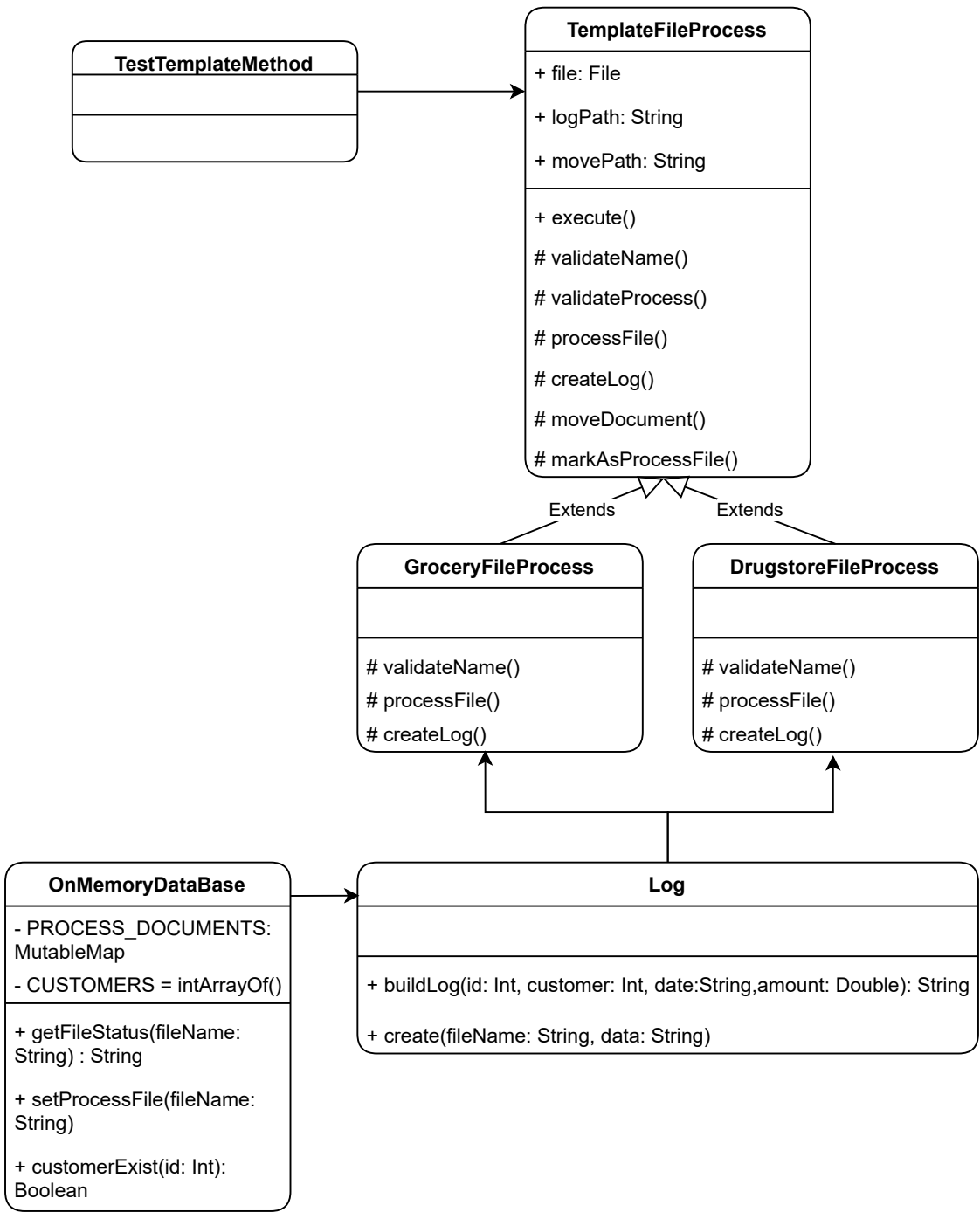
Patrón de diseño Command - Implementación



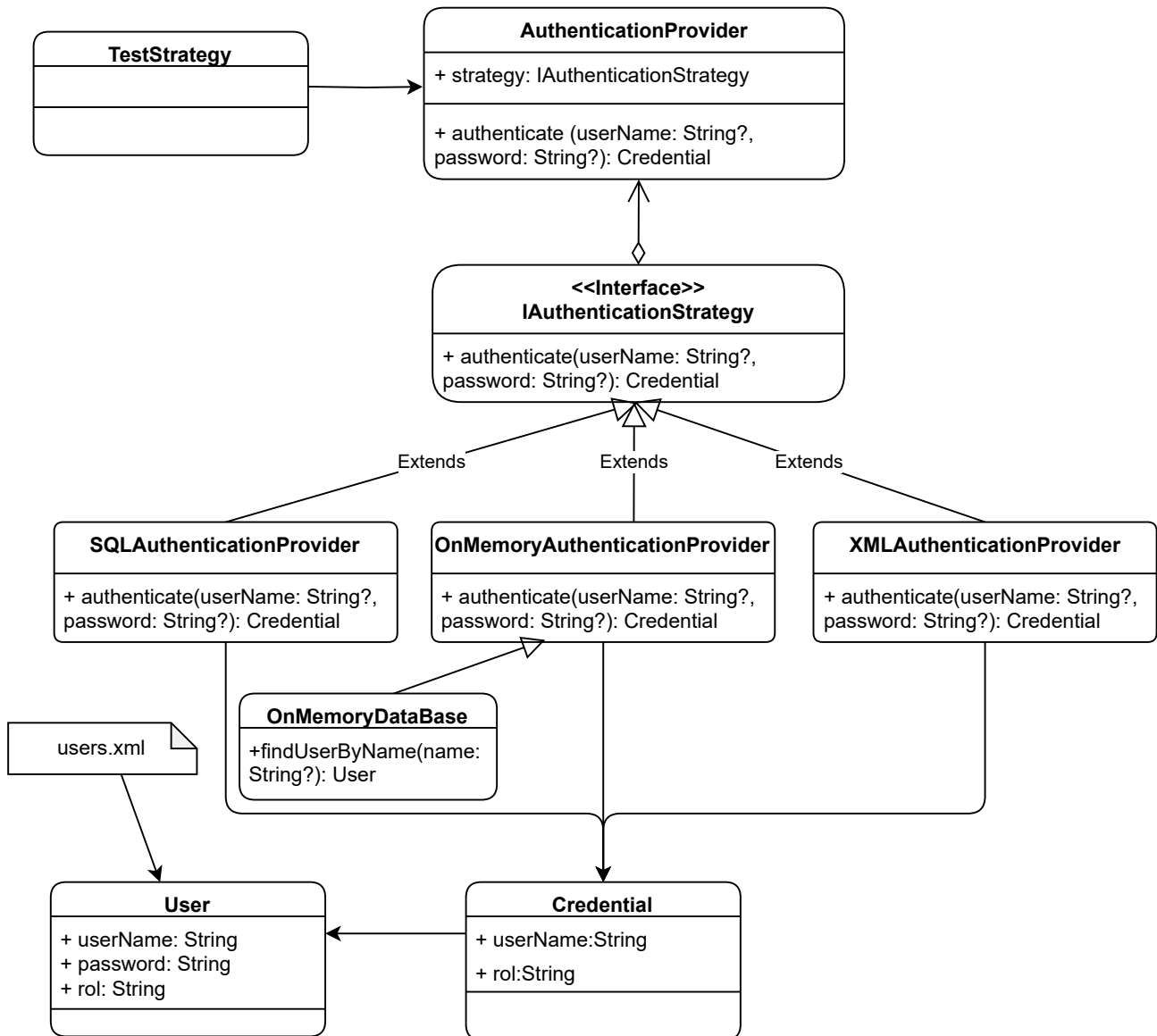
Patrón de diseño Observer - Implementación



Patrón de diseño TemplateMethod - Implementación

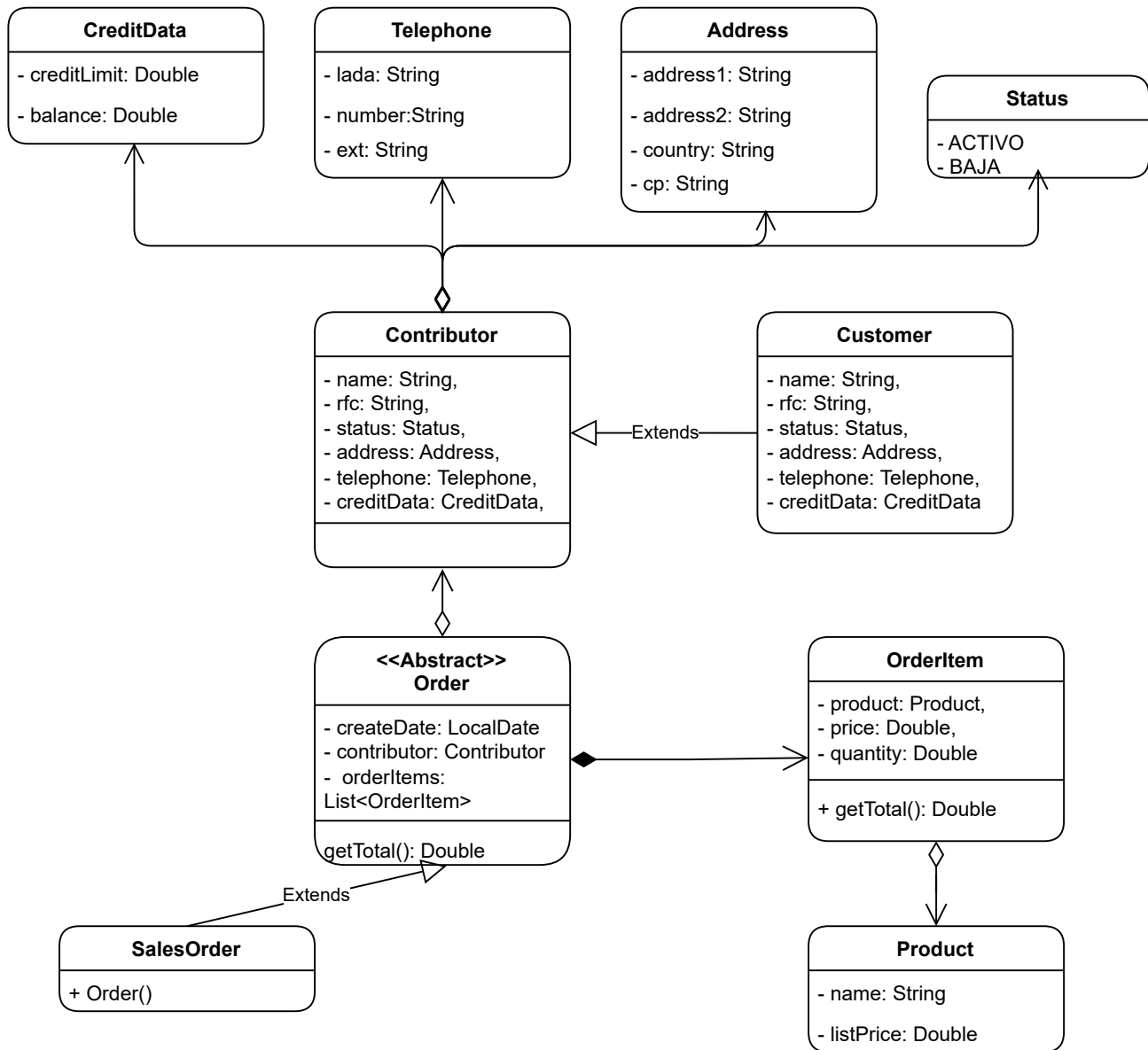


Patrón de diseño Strategy - Implementación

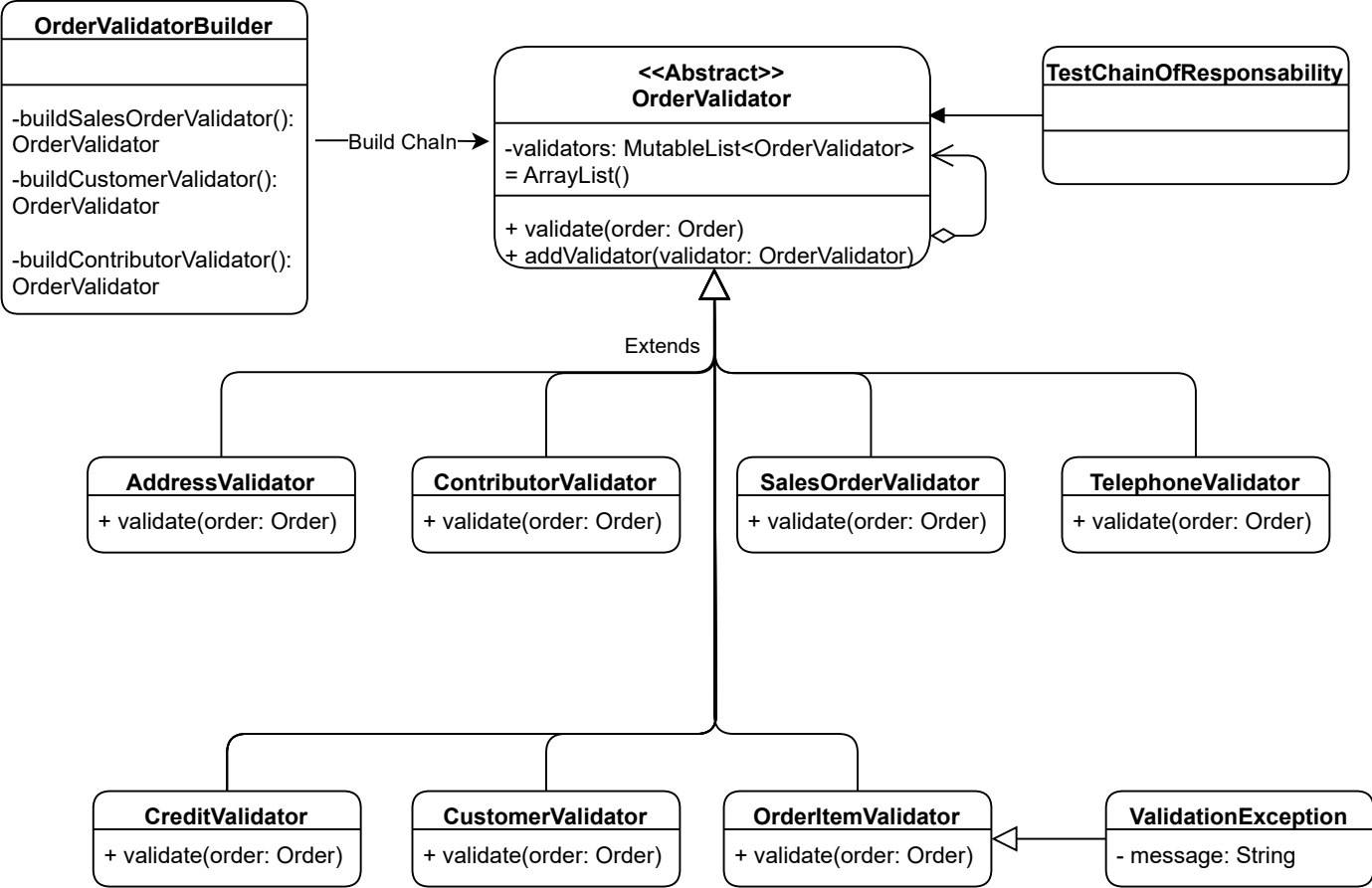


Patrón de diseño Chain Of Responsibility - Implementación

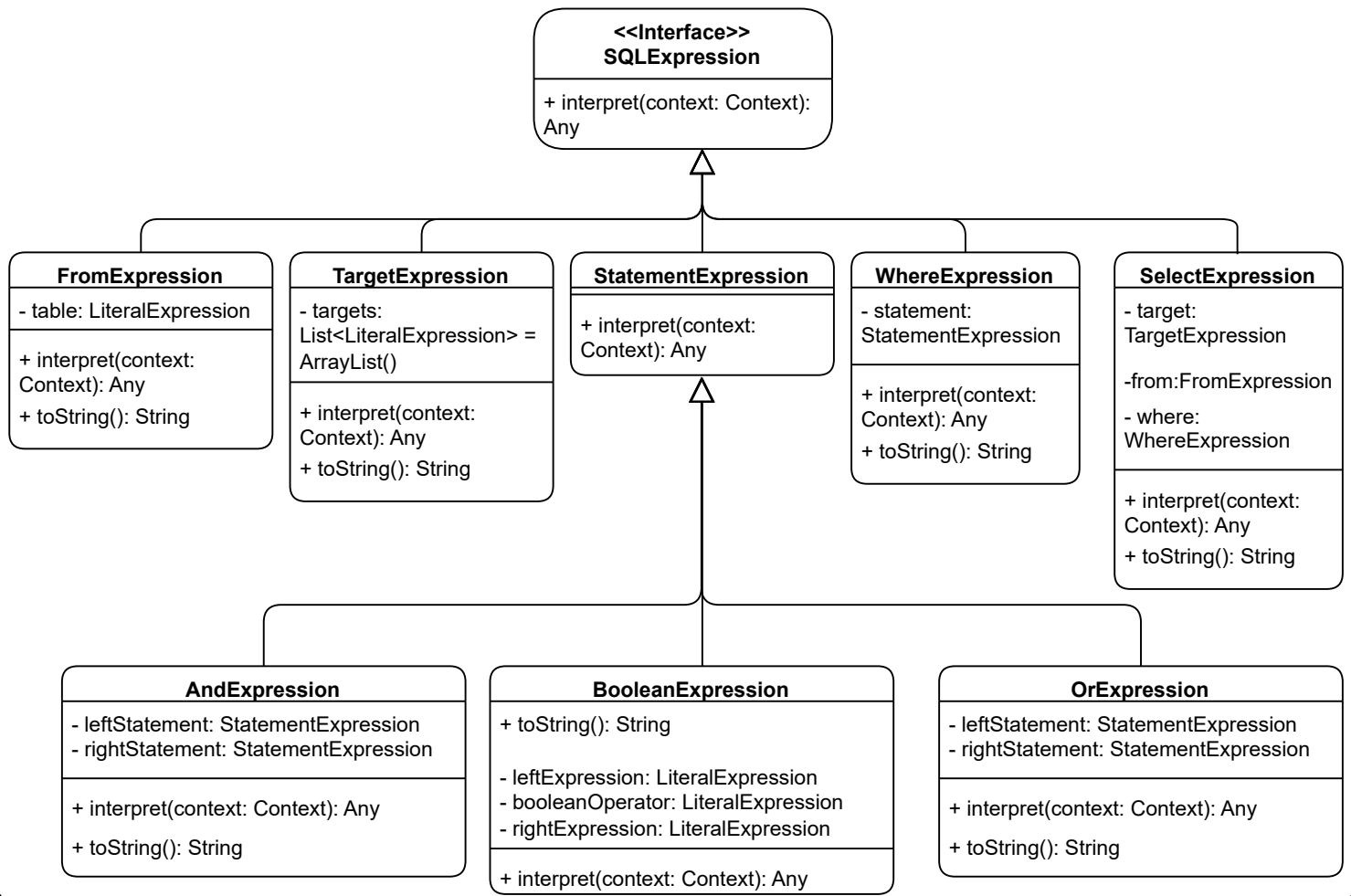
Estructura de la orden



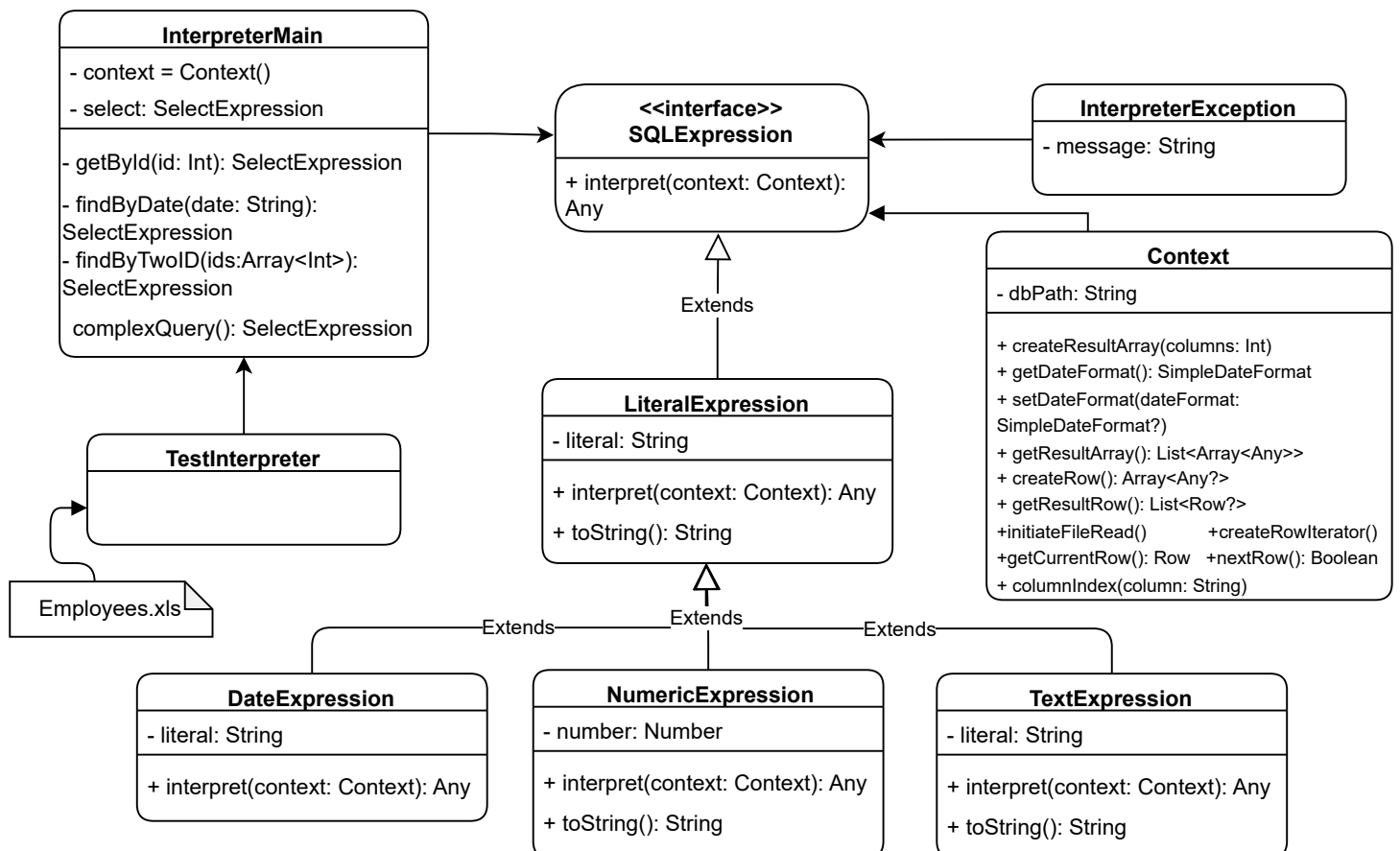
Patrón de diseño Chain Of Responsibility - Implementación



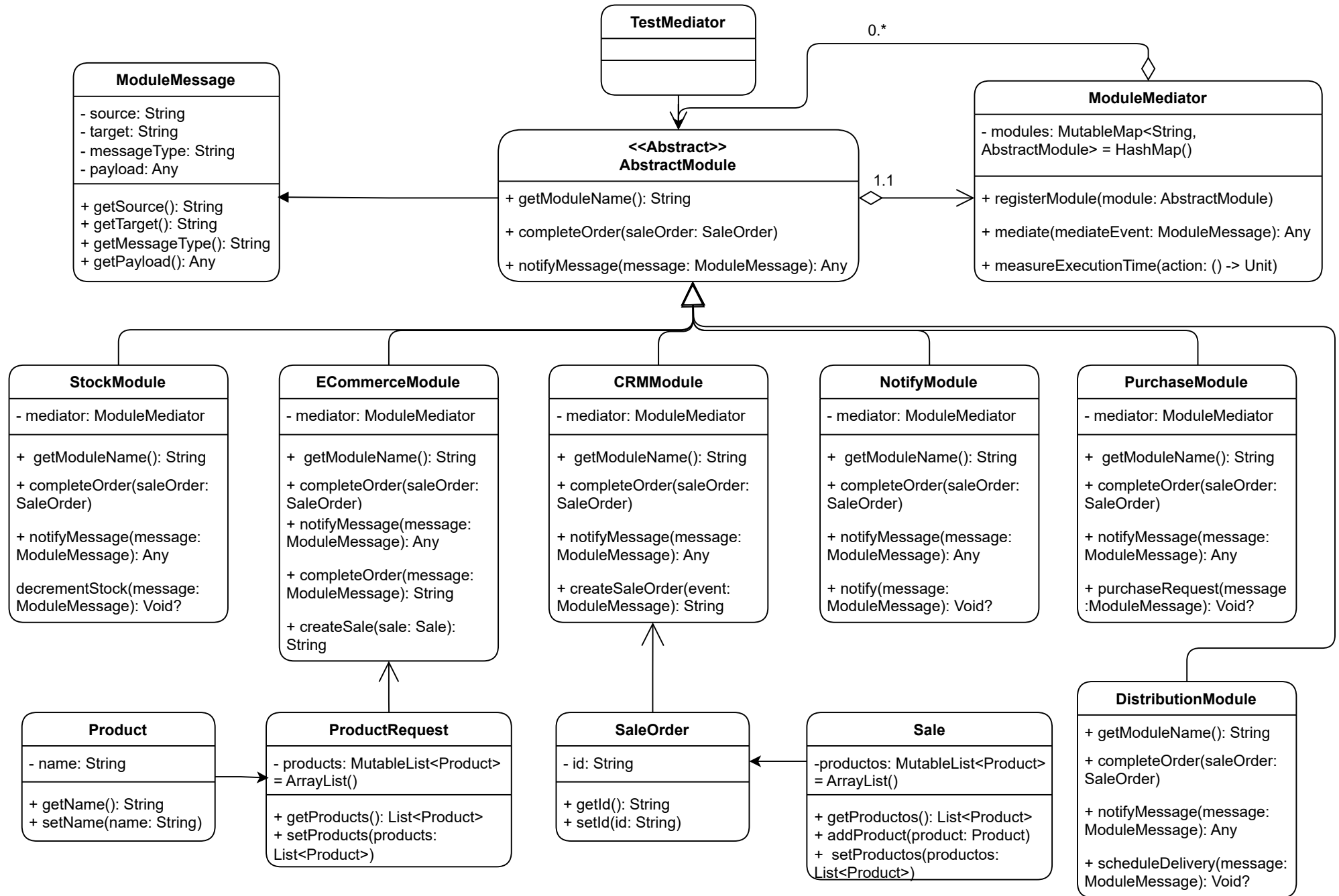
Patrón de diseño Interpreter - Non Terminal Expression



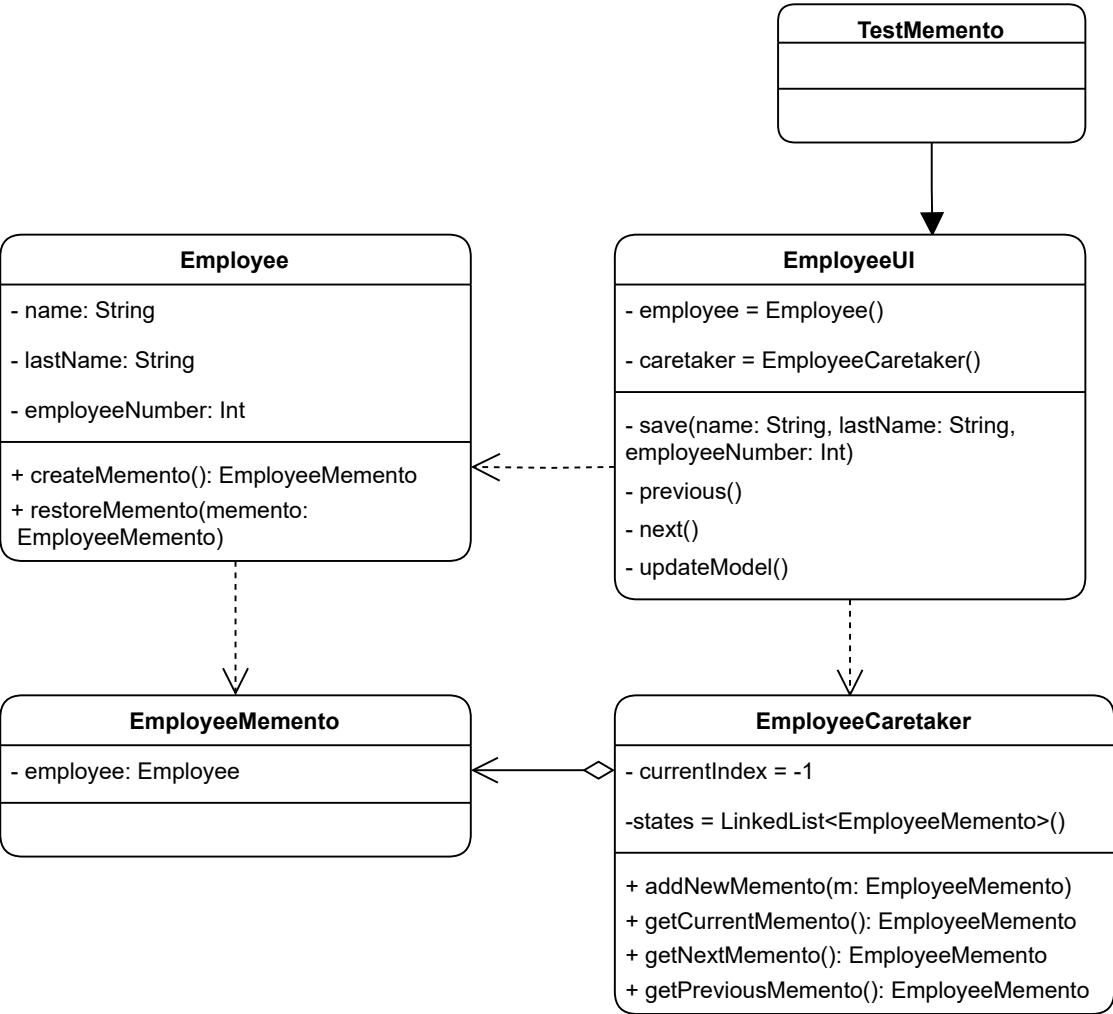
Patrón de diseño Interpreter - Terminal Expression



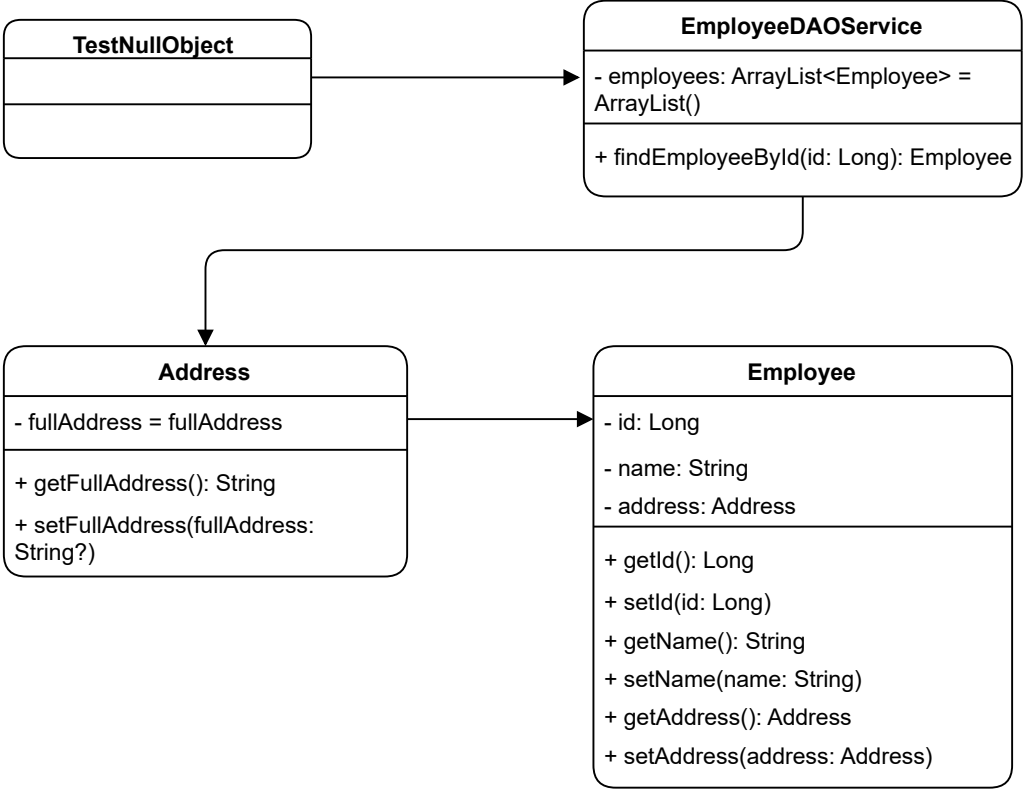
Patrón de diseño Mediator - Implementación



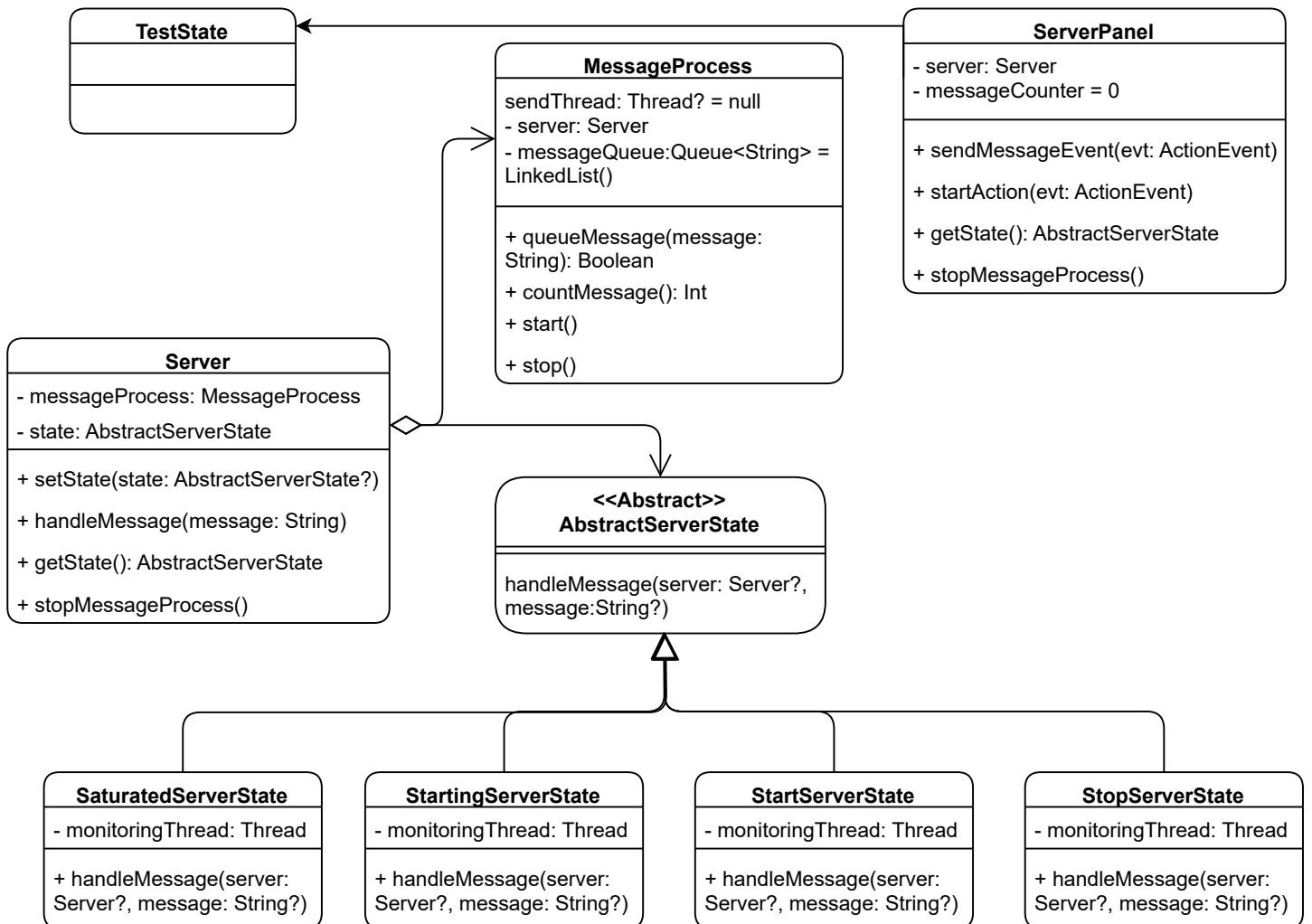
Patrón de diseño Memento - Implementación



Patrón de diseño NullObject - Implementación



Patrón de diseño State - Implementación



Patrón de diseño visitor - Implementación

