



UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO

FACULTAD DE INGENIERIA EN CIENCIAS DE LA COMPUTACION Y
TELECOMUNICACIONES



ARQUITECTURA DE SOFTWARE PATRON DE DISEÑO ESTRATEGIA

INTEGRANTE:

- RIVERO GONZALES RONALDO

214021882

DOCENTE:

- ING. VEIZAGA GONZALES JOSUE OBED

SANTA CRUZ – BOLIVIA

Descripción

El patrón Estrategia (Strategy) es un patrón de diseño para el desarrollo de software. Se clasifica como patrón de comportamiento porque determina cómo se debe realizar el intercambio de mensajes entre diferentes objetos para resolver una tarea. El patrón estrategia permite mantener un conjunto de algoritmos de entre los cuales el objeto cliente puede elegir aquel que le conviene e intercambiarlo dinámicamente según sus necesidades.

Participante

Las clases y objetos que participan en este modelo son:

- **Strategy**

Declara una interfaz común para todos los algoritmos compatibles. El **Context** utiliza esta interfaz para llamar al algoritmo definido por un **ConcreteStrategy**

- **ConcreteStrategy**

Implementa el algoritmo utilizando la interfaz de Strategy

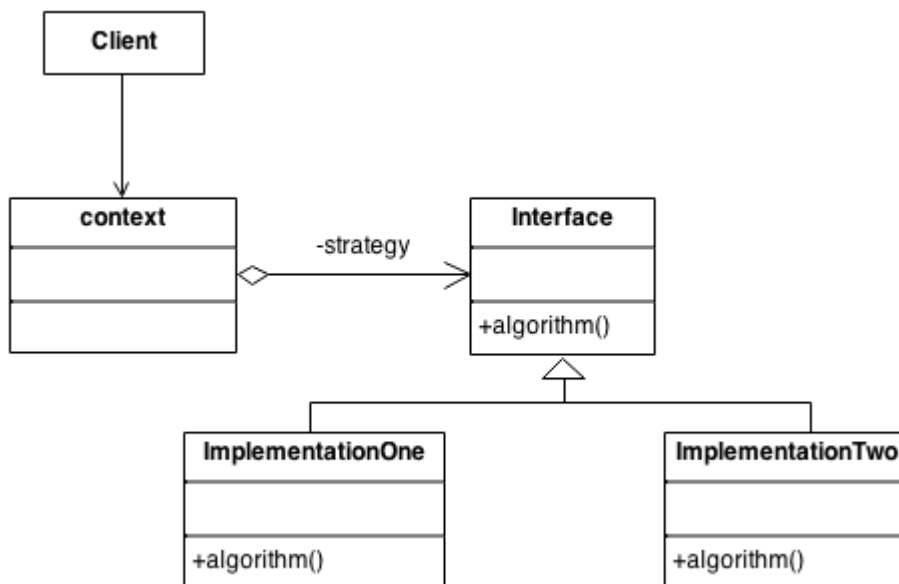
- **Context**

Está configurado con un objeto **ConcreteStrategy**

Mantiene una referencia a un objeto de **Strategy**

Puede definir una interfaz que le permite acceder a sus datos **Strategy**.

Estructura



Ejemplo

Una estrategia define un conjunto de algoritmos que se pueden usar indistintamente. Los modos de transporte a un aeropuerto son un ejemplo de estrategia. Existen varias opciones, como conducir el propio automóvil, tomar un taxi, un servicio de transporte al aeropuerto, un autobús urbano o un servicio de limusina. Para algunos aeropuertos, el metro y los helicópteros también están disponibles como medio de transporte al aeropuerto. Cualquiera de estos modos de transporte llevará al viajero al aeropuerto, y se pueden usar indistintamente. El viajero debe elegir la Estrategia basada en compensaciones entre costo, conveniencia y tiempo.

