

Patrones de diseño

Strategy

Estudiantes:
Victoria Orozco. B45038
Kenia Moya. B34760

Tipo:

De comportamiento.

Propósito:

Se puede utilizar para mantener un equilibrio entre la flexibilidad y complejidad del programa.

Maneja los diversos comportamientos de un objeto con diferentes estrategias, para evitar la sobrecarga de código.

¿Cuándo aplicar Strategy?

- ❖ Cuando se tienen diferentes formas de realizar una acción.
- ❖ Cuando se desconoce el método a usar hasta que el programa está en ejecución.
- ❖ Para tener un programa mantenible.
- ❖ Si se desean añadir nuevos comportamientos o acciones a los métodos.

Actores de implementación:

StrategyClient : Clase que usa diferentes estrategias para ciertas tareas. Contiene una referencia a la instancia **Strategy**.

Strategy : Interfaz que define todos los métodos disponibles para **StrategyClient**.

ConcreteStrategy : Clase que implementa a **Strategy** utilizando un conjunto específico de reglas para cada método.

Estructura de Strategy

