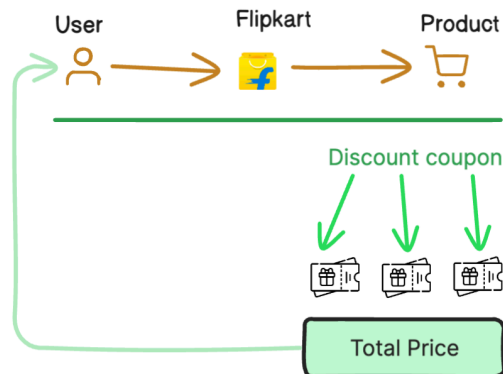


24 Build Discount Coupon Engine

Functional requirements:

- We can add new coupons at runtime.
- Both cart level and product level discounts.
- Supports multiple coupon types (seasonal discount, banking discount, loyalty discount).
- Supports both flat and percentage discounts.
- One coupon can / can't be applied on the top of the another coupon.
- Thread safe.¹

Happy Flow:



High-Level Architecture:

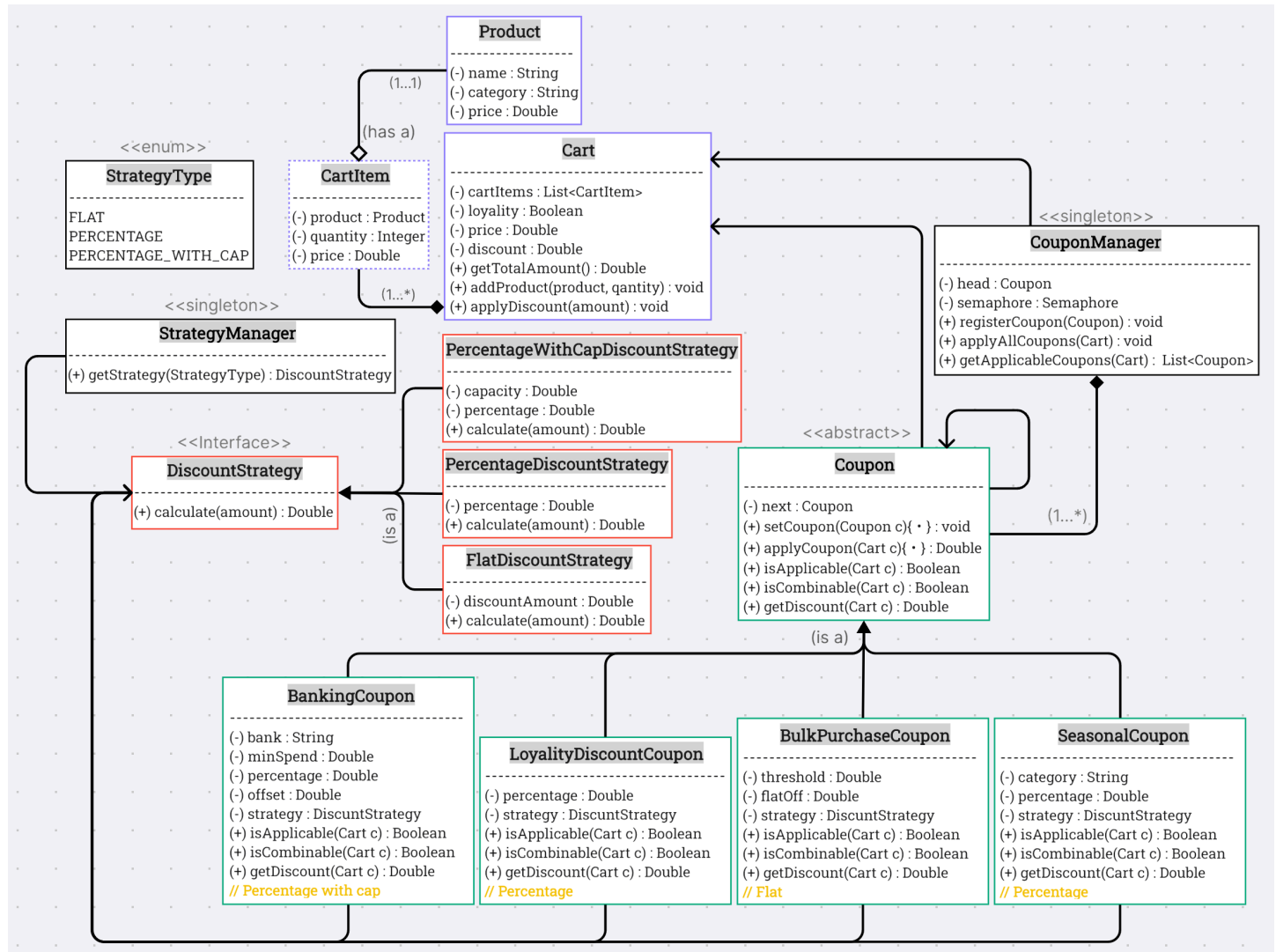
- The system is centered around three main abstractions:
 - **Cart domain model** – Represents products, cart items, and cart state.
 - **Coupon hierarchy** – Encapsulates business rules and eligibility logic.
 - **Discount strategy layer** – Encapsulates how the discount amount is calculated.
- The **CouponManager** acts as a centralized, thread-safe coordinator for registering and applying coupons.
- The **StrategyManager** works as a factory/registry for discount strategies.
- Coupons are linked together using a **Chain of Responsibility**, allowing the system to apply multiple coupons sequentially while respecting combinability rules.

Design Patterns Used

- **Strategy Pattern**
Encapsulates discount calculation logic and makes it interchangeable.
- **Factory Pattern**
StrategyManager creates and provides appropriate strategy instances.
- **Singleton Pattern**
Ensures only one instance of CouponManager and StrategyManager exists.
- **Chain of Responsibility Pattern**
Coupons are linked and processed sequentially, supporting multiple coupon application and combinability rules.
- **Composition**
Coupons contain strategies, and carts contain cart items.

24 Build Discount Coupon Engine

UML diagram:



Code Link: <https://github.com/sibasundarj8/-System-Design--LLD-/tree/main/Projects/Discount%20Coupon%20Engine>