

PROBLEM STATEMENT: Design a Food Delivery App (Tomato)

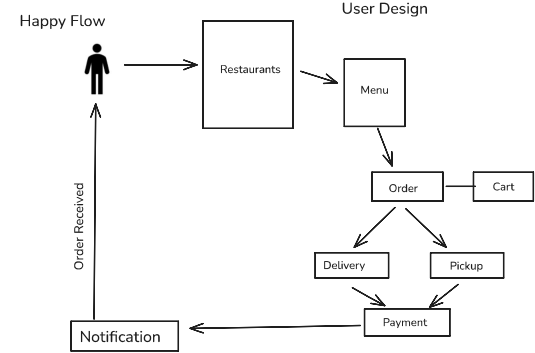
Functional Requirements (Product related questions/ Business-related logic)

- >User can search for Restaurants based on location.
- >User can add items to cart.
- >User can check out by making payments.
- >User should be notified once order placed successfully.

Non-Functional Requirements

- >Each part of design should be scalable and modifiable.

Target: Clean Code, Design Patterns, UML explanation, Relationship b/w multiple objects, working code execution.



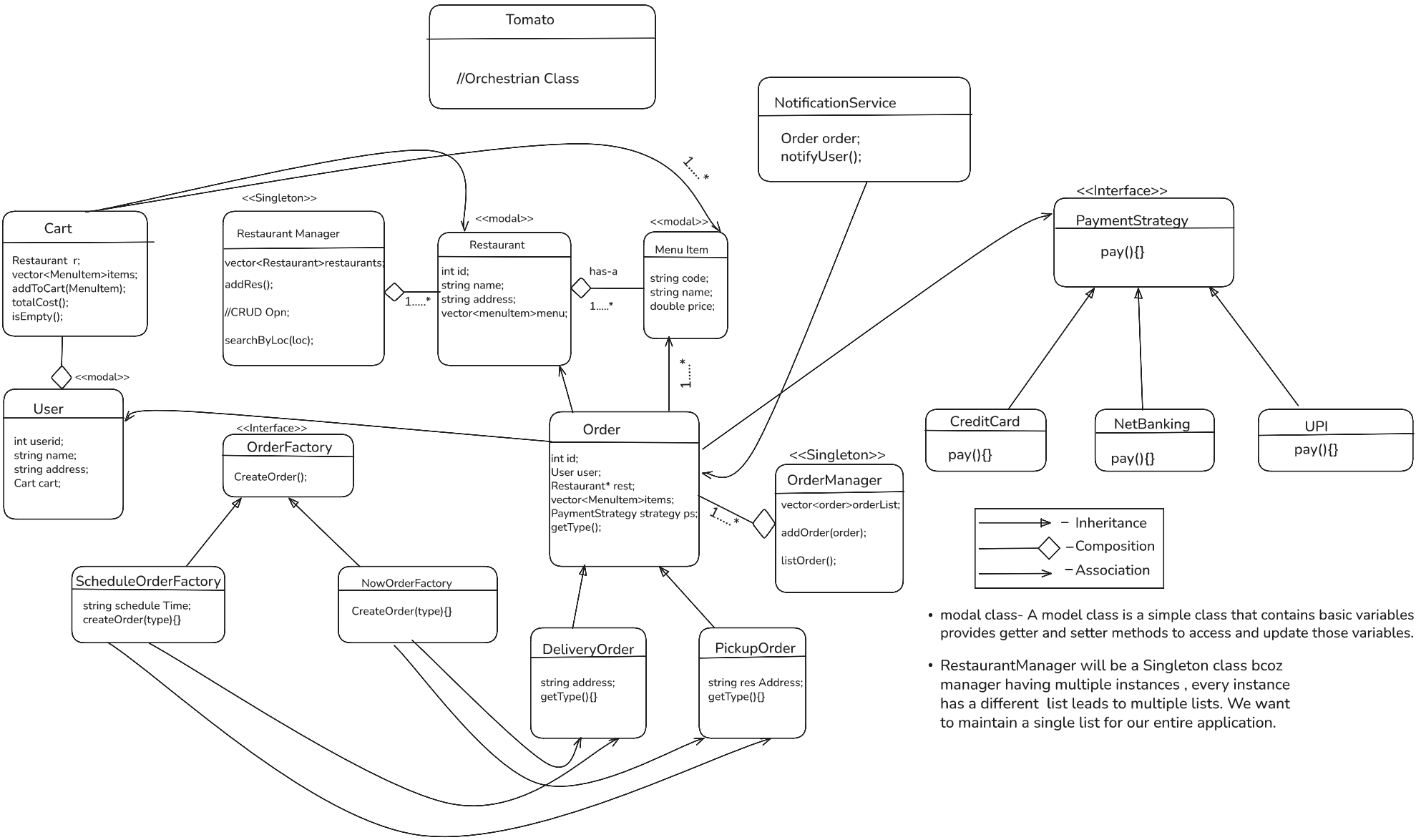
Questions arise

- Payment Service should be done by outside or any third party?
- Online FoodService App into 2: 1.User Centric 2.Delivery Centric

UML Design

LLD Problem can be solved by 2 approaches:
1.BottomUp 2.TopDown

Here we follow bottom-up approach:(smaller objects 1st. dependent objects and then main objects)



- modal class- A modal class is a simple class that contains basic variables and provides getter and setter methods to access and update those variables.
- RestaurantManager will be a Singleton class bcoz manager having multiple instances , every instance has a different list leads to multiple lists. We want to maintain a single list for our entire application.