

Design Patterns

1. Observer Pattern

Pattern Type: Behavioral

Participants:

- Subject: Notification System
- Observer: User
- Concrete Observer: Passenger, Driver

Explanation:

The Observer pattern allows an object (User) to be notified automatically of state changes in another object (Notification System). This is ideal for handling events like route updates or ride status.

Example:

- User receives alerts from the Notification System.
- Notification System initiates a notification when a route is announced.

2. Mediator Pattern

Pattern Type: Behavioral

Participants:

- Mediator: Taxi Request System
- Colleague: Passenger, Driver

Explanation:

The Mediator pattern centralizes complex communication between objects. Instead of Passenger directly interacting with Driver, requests are handled through the Taxi Request System.

Example:

- Taxi Request System acts as an intermediary between Passenger and Driver.
- Passenger makes a request for pickup to a driver, but the Taxi request system acts as the middleman for this request.