

Comsats university Islamabad, dhamtur campus



**Assignment:1**

**Name: Naseem shahid**

**Number: SP23-BSE-076**

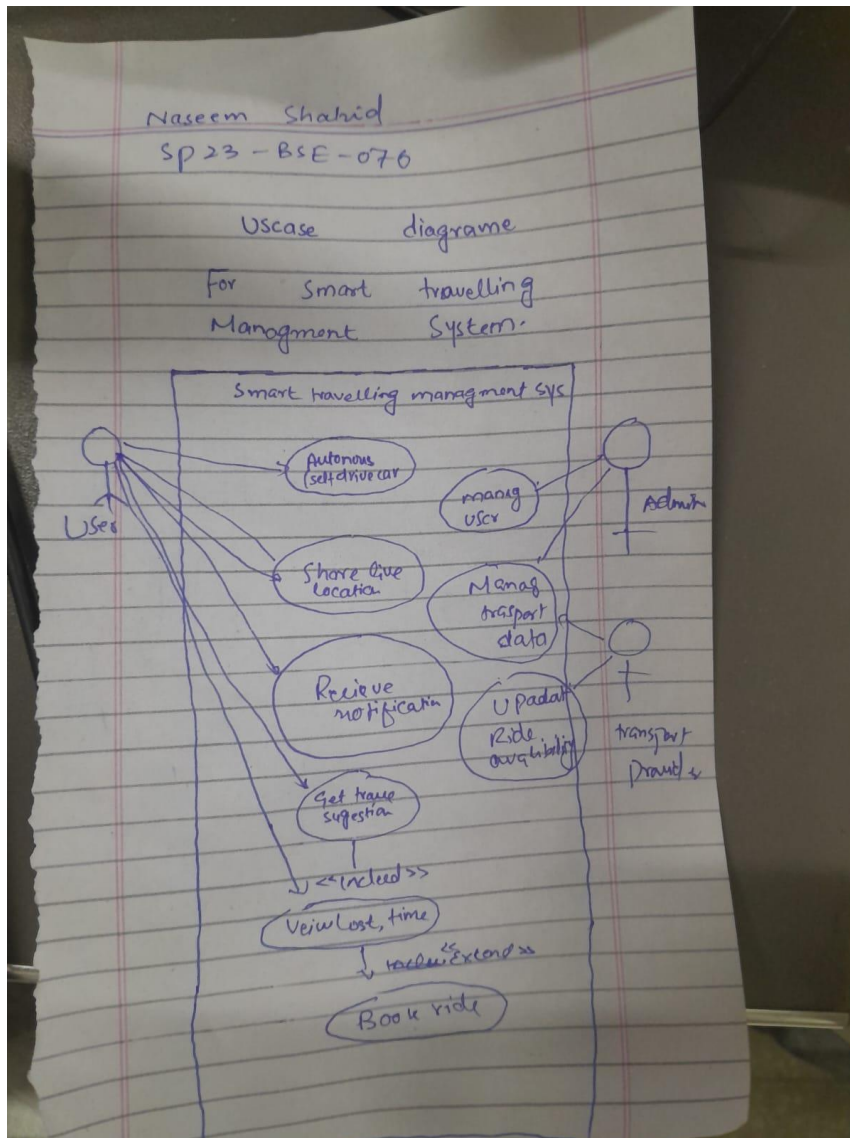
**Course: SDA**

**Teacher: Sir mukhtiar zamin**

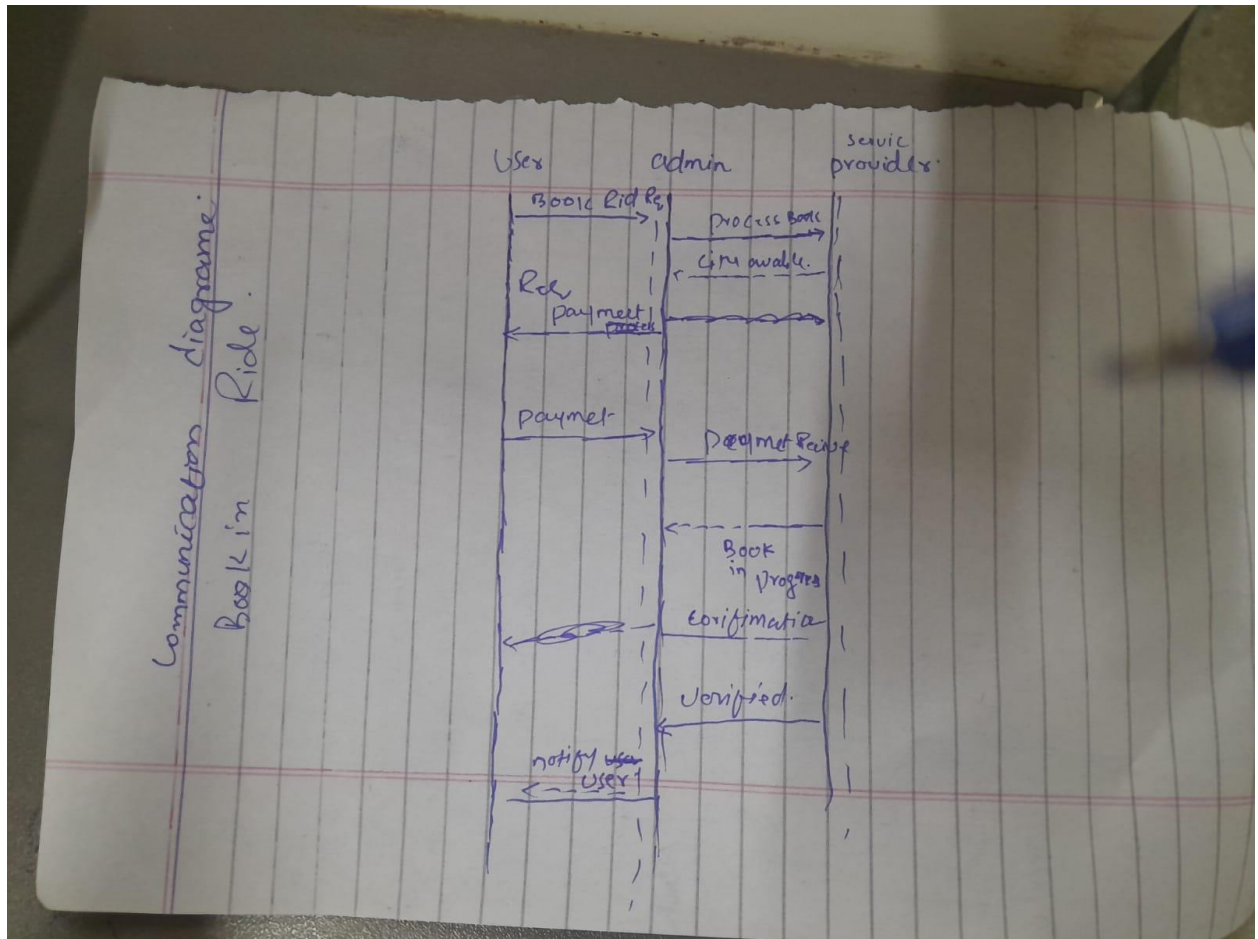
**Date: 22/05/2025**

## Case study: Smart travelling management system

Use case diagram:



Communication diagram:  
Use case: booking ride



## **Pattern and principle:**

### **1. Principles Demonstrated**

a) Single Responsibility Principle (SRP) Each component (App, Booking System, Payment System) has a clear, distinct responsibility:

App handles user interaction.

Booking System manages booking logic.

Payment System processes payments.

This separation simplifies maintenance and improves modularity.

b) Separation of Concerns

Different concerns (UI, business logic, payment) are handled by different components.

This helps isolate changes (e.g., payment gateway changes affect only Payment System).

c) Loose Coupling

Components communicate via well-defined messages, not tightly integrated.

The Booking System doesn't directly manage UI or payment internals;

it just sends/receives requests and responses.

### **2. Design Patterns Illustrated**

a) Model-View-Controller (MVC) (Partial)

The App acts as the View/Controller, interacting with the user and passing requests.

The Booking System represents the Model where the core business logic runs.

This pattern organizes the flow between UI and business logic clearly.

b) Facade Pattern

The Booking System acts like a Facade that provides a simple interface to complex subsystems (payment, ride availability).

It hides complexity and coordinates multiple operations.

c) Observer Pattern (Implied for Notifications)

Although not explicitly shown in the simple diagram, the notification process usually follows the Observer pattern where the user is notified of changes asynchronously.