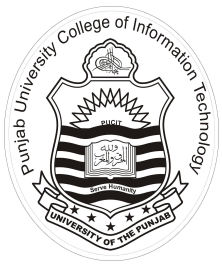


PUCIT
Punjab University College of Information Technology

Second Deliverable for Object Oriented Approach

Version 1.0

Group Members:

1. **Ashhad Haseen (BCSF19A514)**
2. **Muhammad Shaban (BSEF20M526)**
3. **Muhammad Osama(BCSF19A542)**

Project Supervisor: Dr. Muhammad Idrees

Table of Contents

1. Introduction

Customized Transport System is a platform that helps users personalize their travel by allowing them to pick travelers along the way according to their mutual destination, choose the gender they want to sit with and calculate the distributed fare that comes after reaching their destination.

Chapter Number 2: Use Case Modelling

2.1 Use Case Descriptions

2.1.1 Sign up

Use Case Name	UC_Sign Up
Actors	Admin, User(Passenger, Driver)
Description	Users(Passenger, Driver) and Admins will have to register first in order to use the app.
Pre-Condition	Internet must be connected and the device's android version must be greater than 7(Naugat).
Post-Condition	Users(Passenger, Driver) will be registered. Now user can interact with the complete application.
Basic Flow	App asks the user to signup. App asks the Users(Driver, Passenger) to enter their details. The Users(Driver, Passenger) now has access to the app's functionality.
Alternative Flow	Device is not connected to the internet. The user provides invalid information or has not entered anything.

2.1.2 Log in.

Use Case Name	UC_Login
Actors	Admin, User(Passenger, Driver)
Description	Users(Passenger, Driver) and Admins will receive a prompt to login their credentials.
Pre-Condition	Users(Passenger, Driver) and Admins must be registered first in order to login. Users(Passenger, Driver) and Admins must have an internet connection.
Post-Condition	Users(Passenger, Driver) will be registered. Now user can interact with the complete application.
Basic Flow	App prompts the User/Admin for login. User/Admin enters his username and password. App authenticates the login. User/Admin gains access to the application dashboard.
Alternative Flow	Device is not connected to the internet. Invalid user credentials.

2.1.3 Passenger's Pickup_Location

Use Case Name	UC_Pickup_Location
Actors	User(Passenger)
Description	Users(Passenger) will receive a prompt to enter their pickup location.
Pre-Condition	Users(Passenger) must have an internet connection. Users(Passenger) must be logged into app.
Post-Condition	Users(Passenger) pickup location will be set. Now user can proceed forward to set destination.
Basic Flow	User(Passenger) receives a prompt to enter their pickup location. User(Passenger) can now proceed to select their destination.
Alternative Flow	Device is not connected to the internet.

2.1.4 Select PassengerRequestToDriver

Use Case Name	UC_PassengerRequestToDriver
Actors	User(Driver)
Description	Users(Driver) will receive a prompt from the passenger
Pre-Condition	Users(Driver) must have an internet connection. Users(Driver) must be logged into app.
Post-Condition	Users(Driver) can select or reject passenger's travel request. Now User(Driver) can proceed forward to their passenger's pickup location.
Basic Flow	User(Driver) logs in. User(Driver) receives a prompt to select or reject the passenger's travel request. User(Driver) can now proceed to their passenger's pickup location.
Alternative Flow	Device is not connected to the internet.

2.1.5 Passenger's Destination

Use Case Name	UC_Destination
Actors	User(Passenger)
Description	Users(Passenger) will receive a prompt to enter their destination point.
Pre-Condition	Users(Passenger) must have an internet connection. Users(Passenger) must be logged into app.
Post-Condition	Users(Passenger) destination point will be set. Now user can proceed forward to select their ride.
Basic Flow	User(Passenger) receives a prompt to enter their destination point. User(Passenger) can now proceed to select their ride.
Alternative Flow	Device is not connected to the internet.

2.1.6 Select Available_Rides

Use Case Name	UC_SelectAvailableRides
Actors	User(Passenger)
Description	Users(Passenger) will receive a prompt to see all available rides.
Pre-Condition	Users(Passenger) must have an internet connection. Users(Passenger) must be logged into app.
Post-Condition	Users(Passenger) selected ride will be set. Now user can proceed forward to select their suitable ride with a suitable fare.
Basic Flow	User(Passenger) receives a prompt to enter their desired ride. User(Passenger) can now proceed to select available seats.
Alternative Flow	Device is not connected to the internet.

2.1.7 Select PassengerGender

Use Case Name	UC_SelectPassengerGender
Actors	User(Passenger)
Description	Users(Passenger) will receive a prompt to select the suitable passenger they want to share the car with.
Pre-Condition	Users(Passenger) must have an internet connection. Users(Passenger) must be logged into app.
Post-Condition	Users(Passenger) selected passenger gender will be set. Now user can proceed forward to select their destination.
Basic Flow	User(Passenger) receives a prompt to enter their desired ride. User(Passenger) can now proceed to select their driver.
Alternative Flow	Device is not connected to the internet.

2.1.8 Select Logout

Use Case Name	UC_SelectLogout
Actors	User(Passenger, Driver) and Admin.
Description	User(Passenger, Driver) and Admin can log out with a button.
Pre-Condition	User(Passenger, Driver) and Admin must have an internet connection. User(Passenger, Driver) and Adminmust be logged into app.
Post-Condition	User(Passenger, Driver) and Admin will return to their login screen. Now user can log in (if registered) or sign up for a new account.
Basic Flow	User(Passenger, Driver) and Admin logs out. User(Passenger, Driver) and Admin can login again or create a new account.
Alternative Flow	Device is not connected to the internet. User is already logged out.

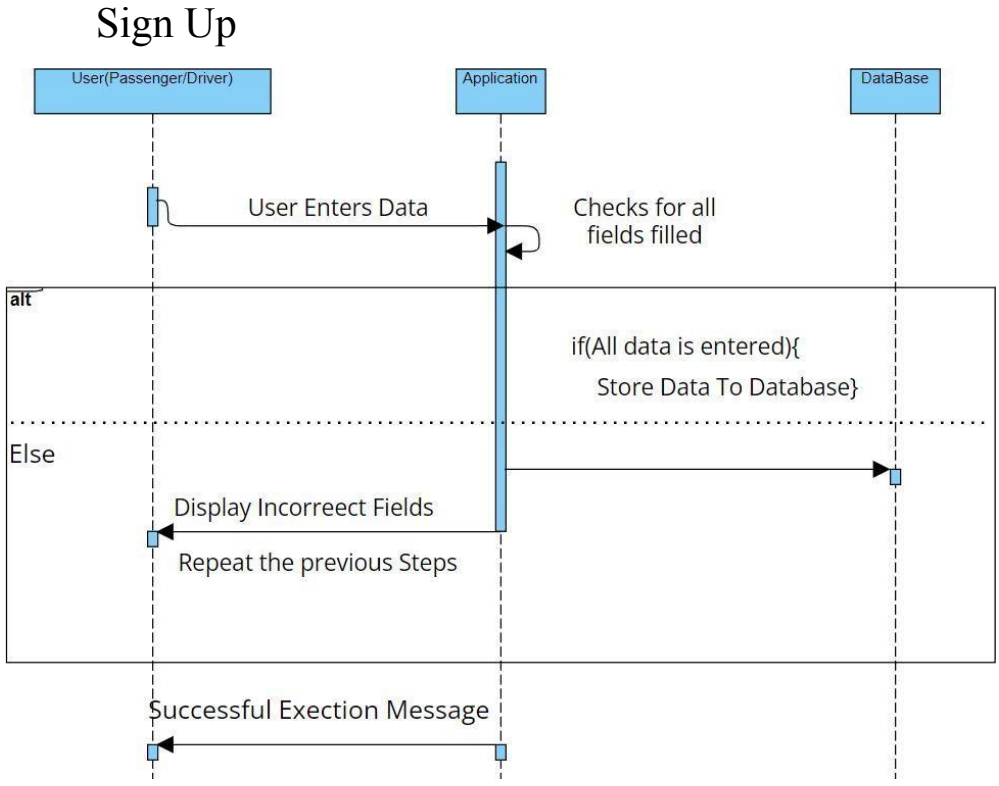
2.2 Design Mapping

Design mapping is a crucial aspect of documentation that plays a pivotal role in conveying complex information in a visually comprehensible manner. It involves the creation of visual representations, often in the form of diagrams or charts, to illustrate the architecture, structure, and relationships within a design. These visual aids serve as a roadmap for understanding the overall concept and intricacies of a system or project.

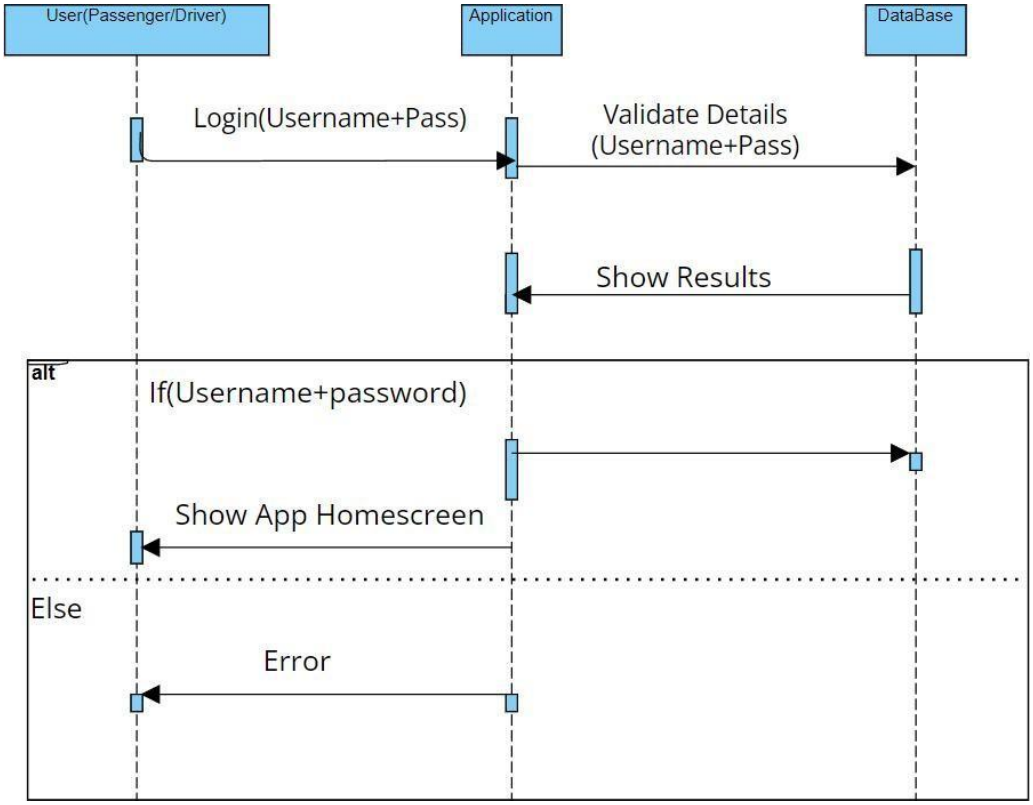
One of the primary benefits of design mapping is its ability to enhance clarity and communication. By condensing intricate details into a visual format, it becomes easier for both technical and non-technical stakeholders to grasp the key components and their interconnections. Design maps can range from flowcharts depicting process workflows to entity-relationship diagrams illustrating database structures. This versatility allows for the effective communication of diverse design aspects, fostering a shared understanding among team members and stakeholders.

Furthermore, design mapping serves as a valuable reference point throughout the development lifecycle. It aids in identifying potential bottlenecks, dependencies, and areas for optimization. As the documentation evolves, design maps can be updated to reflect changes, ensuring that the documentation remains a dynamic and accurate representation of the project's design. In essence, design mapping is an indispensable tool for enhancing collaboration, facilitating comprehension, and maintaining the integrity of design documentation. Here’s the sequence diagrams of our use-cases for the design mapping of our project documentation:

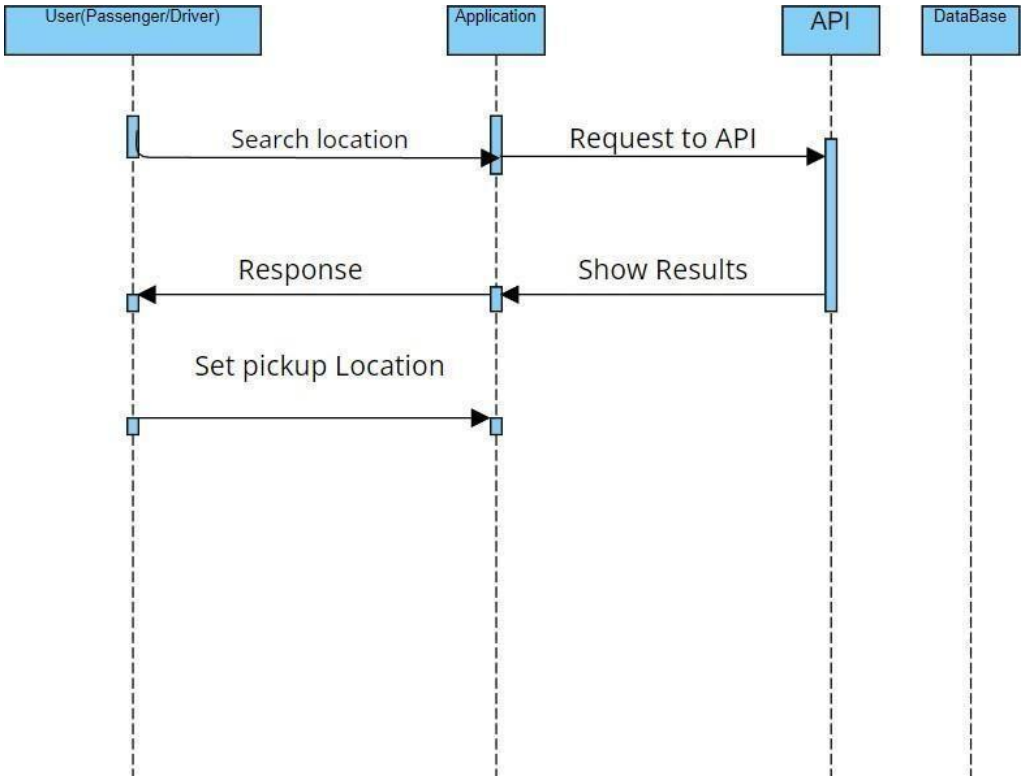
2.3. Sequence Diagrams



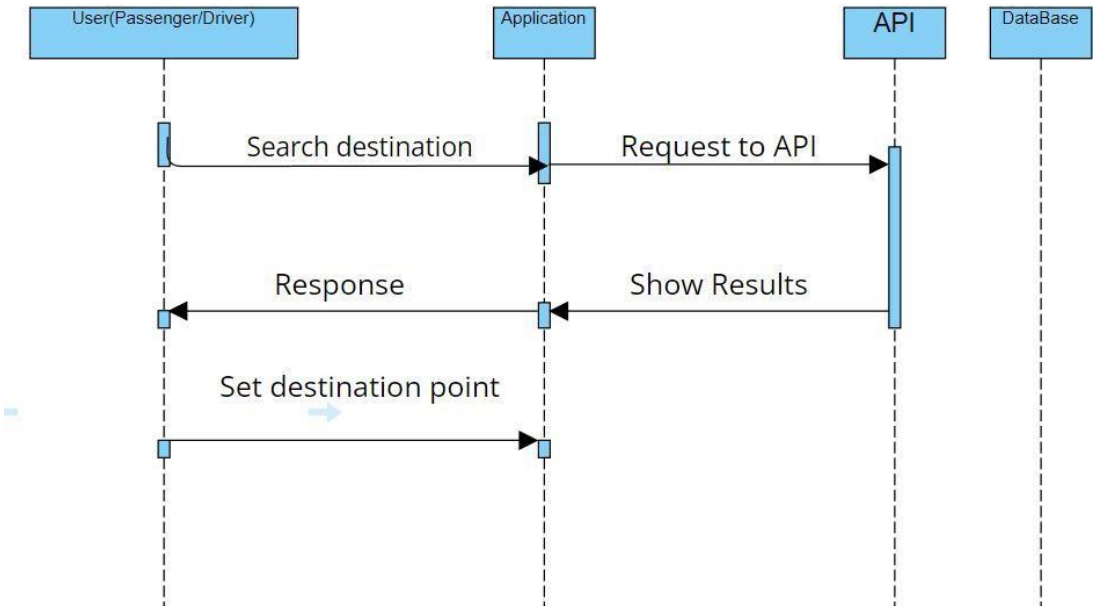
2.3.1 Log in


User(Passenger/Driver)
Application
DataBase
Login(Username+Pass)
Validate Details
(Username+Pass)
Show Results
alt
If(Username+password)
Show App Homescreen
Else
Error

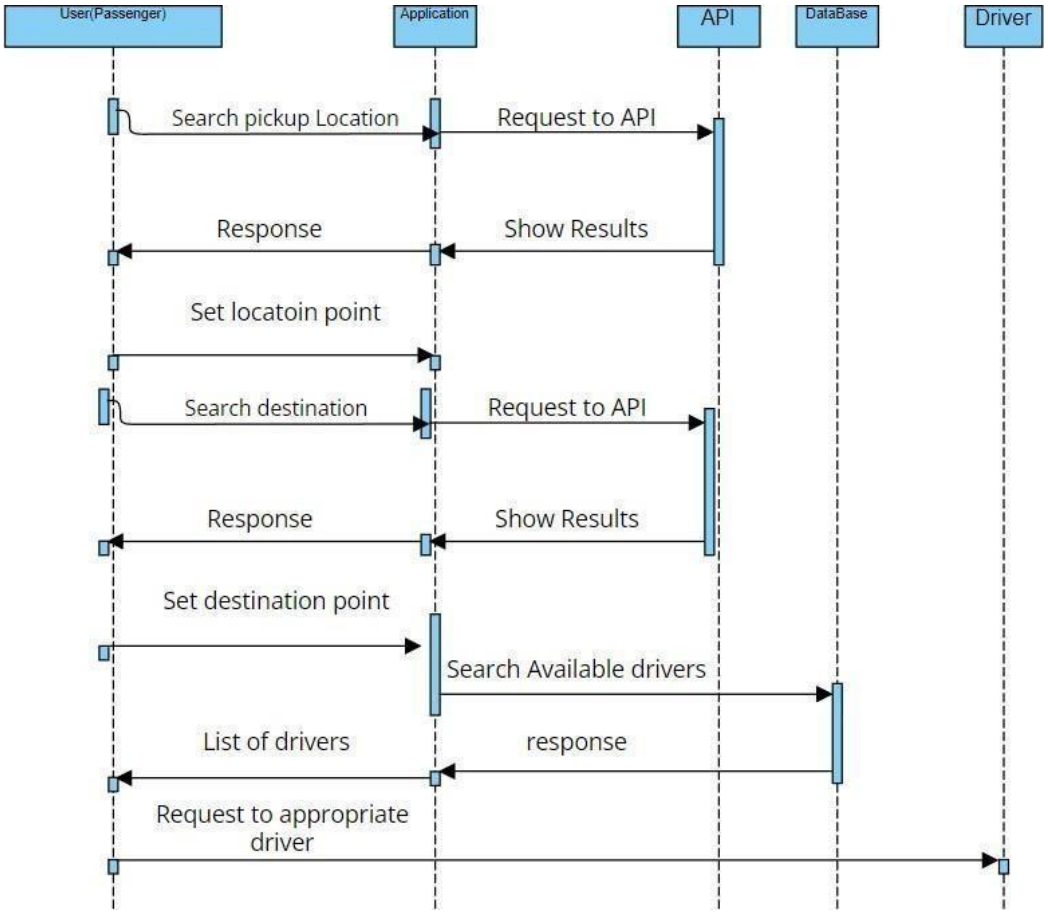
2.3.2 Pickup location



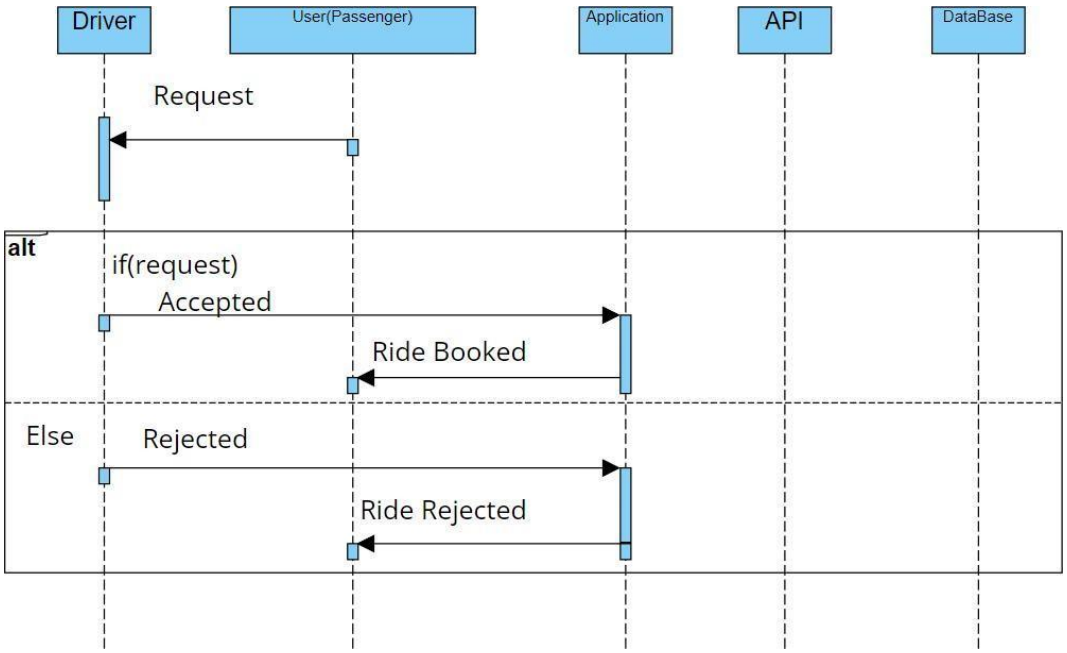
2.3.3 Select Destination



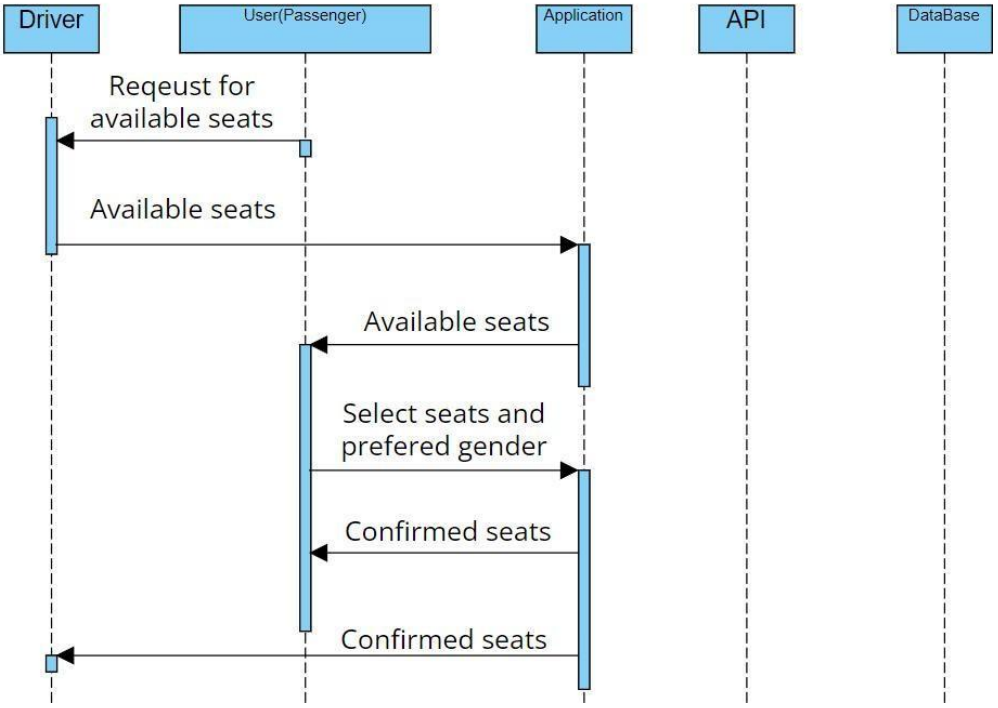
2.3.4 Passenger request to driver



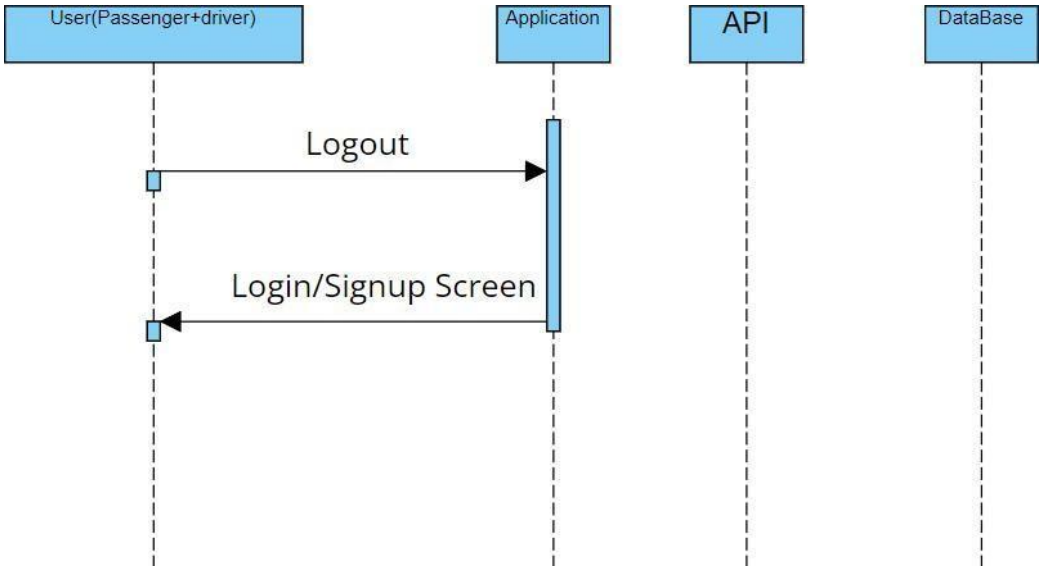
2.3.5 Driver Response to Passenger



2.3.6 Seats availability



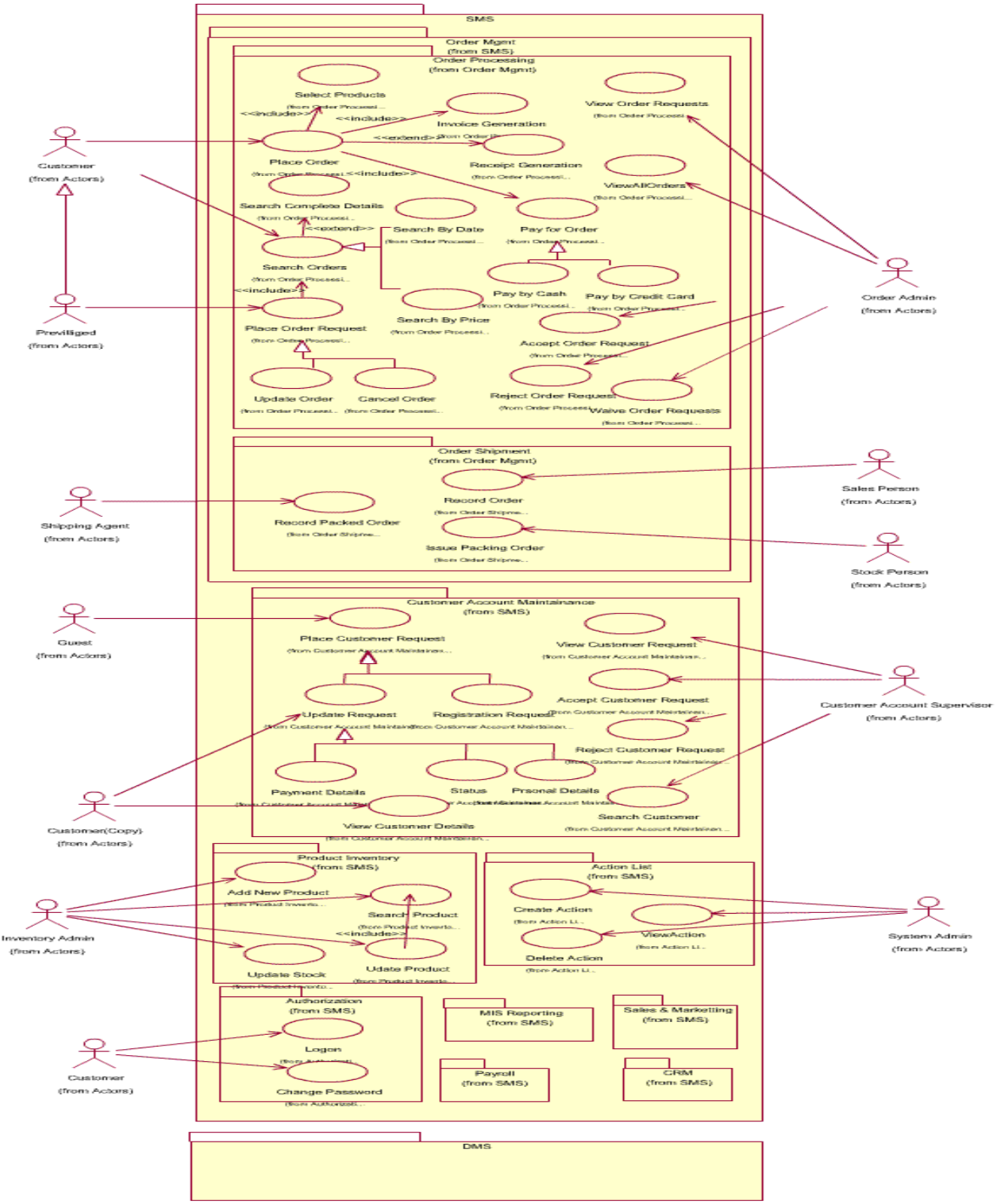
2.3.6 Logout

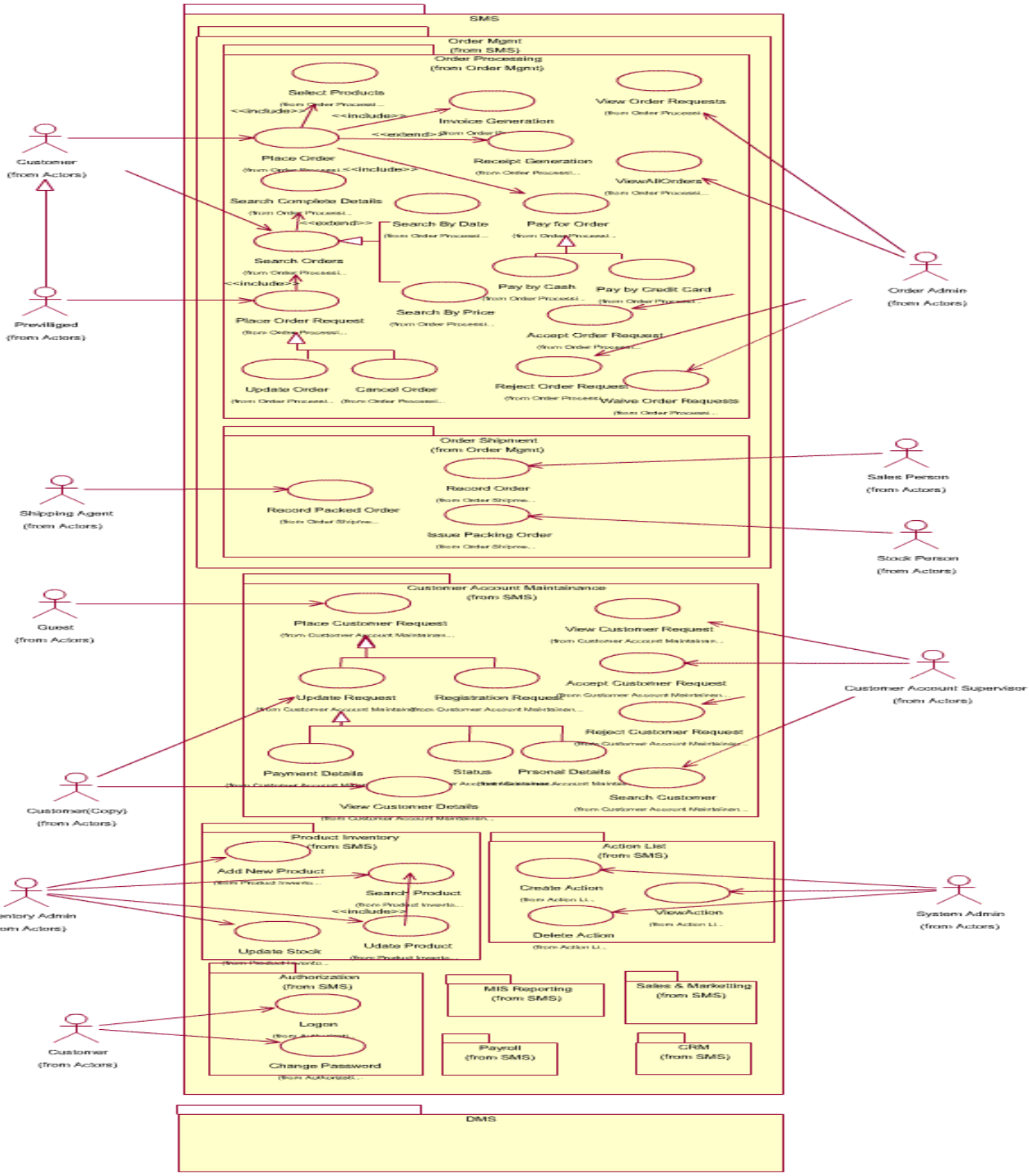


Chapter 3: Refined Use Case Diagram

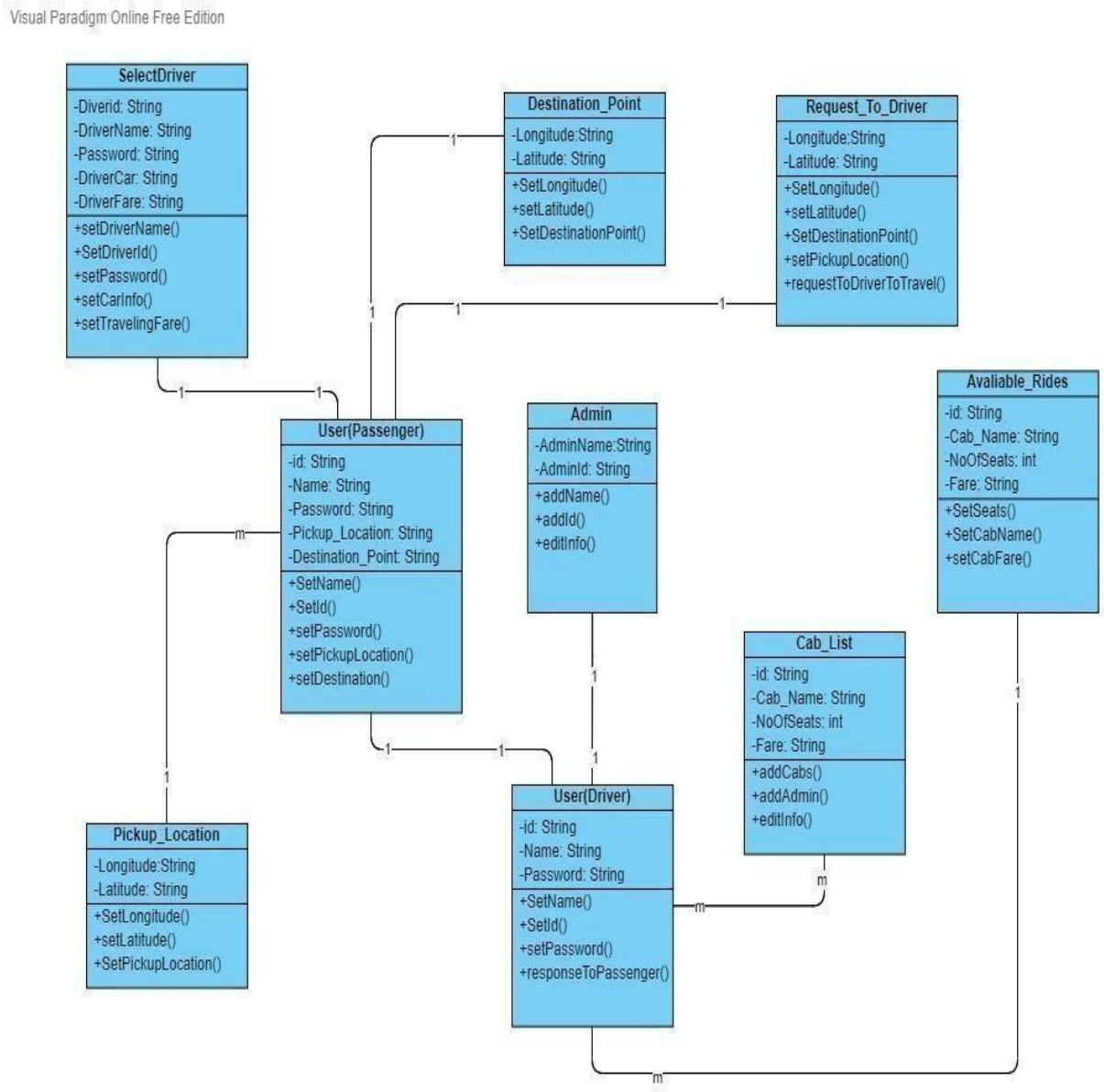
A High-level use case diagram is the explanation of high-level use case diagram. In this diagram, high-level use cases are expanded in a way that exhibits how high-level use cases will reach their functionality. Two types of relationships are used in this diagram. Which are:

- Extend
- Include


SMS
Order Mgmt
(from SMS)
Order Processing
(from Order Mgmt)
Select Products
(from Order Processi...
<<include>>
<<include>>
Invoice Generation
(from Order Pr...
<<extend>>
Place Order
(from Order Processi...
<<include>>
Receipt Generation
(from Order Processi...
View Order Requests
(from Order Processi...
ViewAllOrders
(from Order Processi...
Search Complete Details
(from Order Processi...
<<extend>>
Search By Date
(from Order Processi...
Pay for Order
(from Order Processi...
Search Orders
(from Order Processi...
<<include>>
Pay by Cash
(from Order Processi...
Pay by Credit Card
(from Order Processi...
Search By Price
(from Order Processi...
Place Order Request
(from Order Processi...
Accept Order Request
(from Order Processi...
Update Order
(from Order Processi...
Cancel Order
(from Order Processi...
Reject Order Request
(from Order Processi...
Waive Order Requests
(from Order Processi...
Order Shipment
(from Order Mgmt)
Record Order
(from Order Shipme...
Record Packed Order
(from Order Shipme...
Issue Packing Order
(from Order Shipme...
Customer Account Maintainance
(from SMS)
Place Customer Request
(from Customer Account Maintainan...
View Customer Request
(from Customer Account Maintainan...
Update Request
(from Customer Account Maintainan...
Registration Request
(from Customer Account Maintainan...
Accept Customer Request
(from Customer Account Maintainan...
Reject Customer Request
(from Customer Account Maintainan...
Payment Details
Status
Prsonal Details
View Customer Details
(from Customer Account Maintainan...
Search Customer
(from Customer Account Maintainan...
Product Inventory
(from SMS)
Add New Product
(from Product Inventa...
Search Product
(from Product Inventa...
<<include>>
Update Stock
(from Product Inventa...
Udate Product
(from Product Inventa...
Action List
(from SMS)
Create Action
(from Action Li...
ViewAction
(from Action Li...
Delete Action
(from Action Li...
Authorization
(from SMS)
Logon
(from Authorizati...
Change Password
(from Authorizati...
MIS Reporting
(from SMS)
Sales & Marketting
(from SMS)
Payroll
(from SMS)
CRM
(from SMS)
DMS
Customer
(from Actors)
Previliged
(from Actors)
Order Admin
(from Actors)
Sales Person
(from Actors)
Shipping Agent
(from Actors)
Stock Person
(from Actors)
Guest
(from Actors)
Customer Account Supervisor
(from Actors)
Customer(Copy)
(from Actors)
Inventory Admin
(from Actors)
System Admin
(from Actors)
Customer
(from Actors)

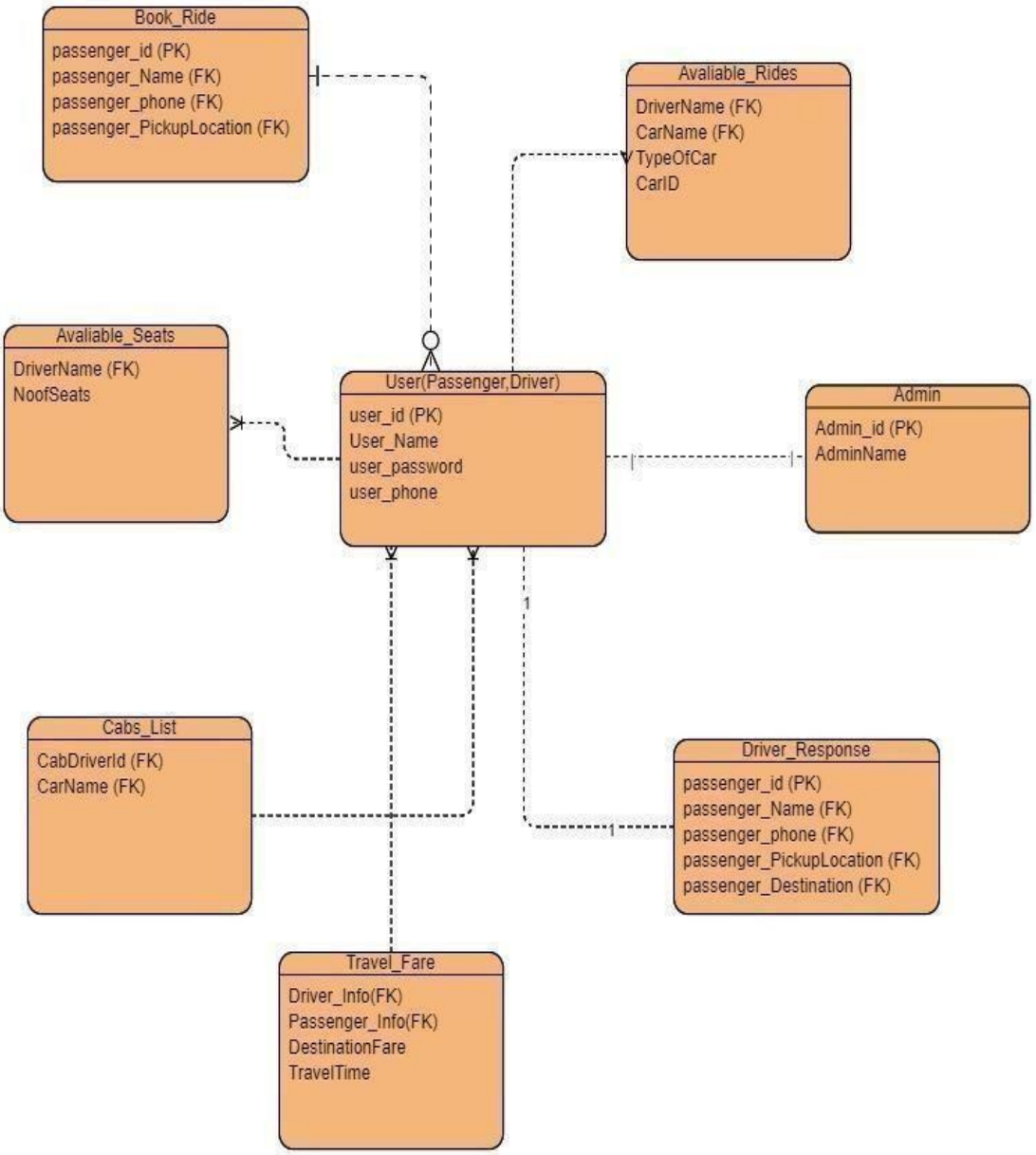

SMS
Order Mgmt
(from SMS)
Order Processing
(from Order Mgmt)
Select Products
<<include>>
Invoice Generation
<<include>>
View Order Requests
<<extend>>
Place Order
Receipt Generation
<<include>>
ViewAllOrders
Search Complete Details
<<extend>>
Search By Date
Pay for Order
Search Orders
<<include>>
Pay by Cash
Pay by Credit Card
Search By Price
Place Order Request
Accept Order Request
Update Order
Cancel Order
Reject Order Request
Waive Order Requests
Order Shipment
(from Order Mgmt)
Record Order
Record Packed Order
Issue Packing Order
Customer Account Maintainance
(from SMS)
Place Customer Request
View Customer Request
Accept Customer Request
Update Request
Registration Request
Reject Customer Request
Payment Details
Status
Personal Details
View Customer Details
Search Customer
Product Inventory
(from SMS)
Add New Product
Search Product
<<include>>
Update Stock
Udate Product
Action List
(from SMS)
Create Action
ViewAction
Delete Action
Authorization
(from SMS)
Logon
Change Password
MIS Reporting
(from SMS)
Sales & Marketting
(from SMS)
Payroll
(from SMS)
CRM
(from SMS)
DMS
Customer
(from Actors)
Previlliged
(from Actors)
Order Admin
(from Actors)
Shipping Agent
(from Actors)
Sales Person
(from Actors)
Stock Person
(from Actors)
Guest
(from Actors)
Customer Account Supervisor
(from Actors)
Customer(Copy)
(from Actors)
Inventory Admin
(from Actors)
System Admin
(from Actors)
Customer
(from Actors)

Chapter 4: Design Class Diagram:



Chapter 5: Data Model:

Data Model Diagram:



Chapter 6: Conclusion

The purpose of this project was to create an application that solves a big problem in the transportation industry – Car pooling. Currently in Pakistan, all the service providers in the same niche are offering their services in an identical way, Uber, Careem, In-Driver, etc have not yet implemented the feature of customizing who they travel with and distributing the share between all parties. Our app will be revolutionary in Pakistan in the sense that it will not only make travel customizable and more convenient for people who prefer to travel with the same gender, but it will also save their costs by dividing the fare between all parties thus making travel more accessible to everyone. In this regard, our ShareFare application will be a better solution compared to the existing solutions and it can make the travel further convenient, cheaper and accessible to the people who use it.

5.1 Future Improvements

We have a few improvements in mind that we have not yet included in the project timeline, but in real life, this app could use some enhancements to make it even better, such as:

Improved User Interface
We can improve the interface in the future to make it look similar to Uber's interface or better.

App Balance
Currently the application will totally rely on the driver receiving the payment and then paying back to the company. In future, we can create a wallet for our app where users can deposit real money by connecting it to their bank accounts, so they can pay directly to the organization.