**Advance Software Engineering (CS 5910)**

**Project Title: CARPOOL ARRANGEMENT SYSTEM**

**Design Document**

Instructor: Dr. Yui Man Lui

Team C:

Burra Rajesh Reddy (700654107)

Yellapu Baldev (700657957)

1. Overview of Document

This document describes the design of carpool arrangement system. This document includes the class diagram of the entire system, interactive diagram, CRC diagram and Deployment diagram. The class diagram depicts the classes involved in the system and the relationship among them. The interactive diagram depicts the activities that take place in the system between the classes or components of the system. The CRC diagram shows the responsibilities of each class corresponding to the system. The deployment diagram depicts the overall structure of the communication that takes place in the application. At last, the document describes the milestones and responsibilities of teammates in handling the application. Tool used to draw elements of design diagrams is Astah Community.

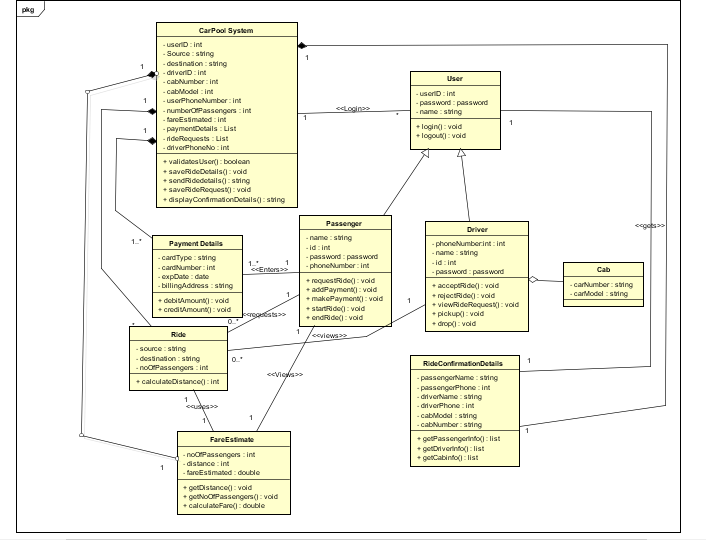
This application allows the passenger to request for a ride. And the driver accepts the request and checks for other requests till the capacity of the cab is reached. The applications display the estimated amount for the ride. In this system, the passenger shares the ride with other passengers requesting the ride in the same direction. The passenger confirms the ride by making the payment and application send the ride confirmation details. Once the ride is confirmed, the cab driver picks the passenger from the requested pickup location and drops at the destination. The application also provides the option to rate the driver. The above workflow is depicted in this document in the form of various diagrams.

2. Elements of Design

2.1. Complete Class Diagram

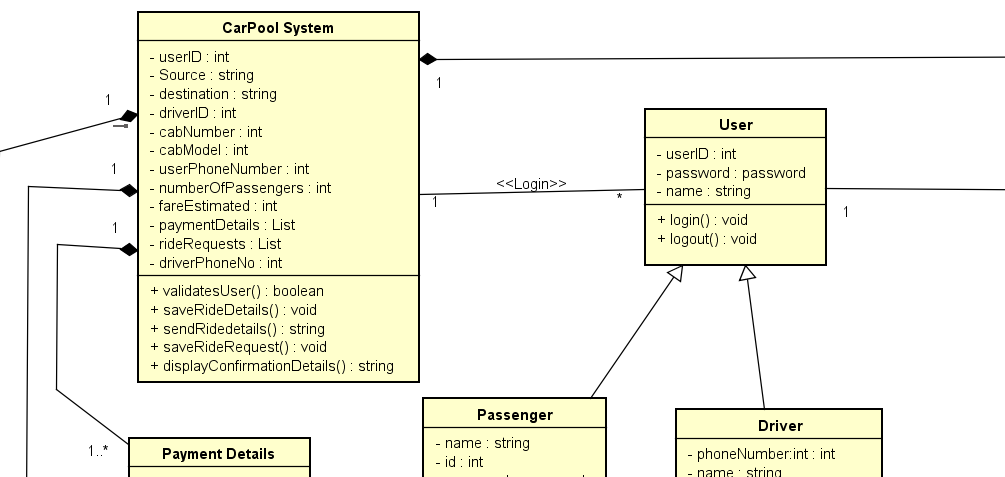
The SOLID principle that used here is dependency inversion principle where the major working of the system depends on the low-level classes and at one stage single responsibility principle is used to do some functionality (fare estimate and Ride Confirmation Details) to reduce the burden of classes.

The Design Pattern used in the system is Abstract Factory design pattern in which objects are created accordingly based on the utility of a class. Abstract design pattern delegates the responsibilities of object instantiation to another object via composition.

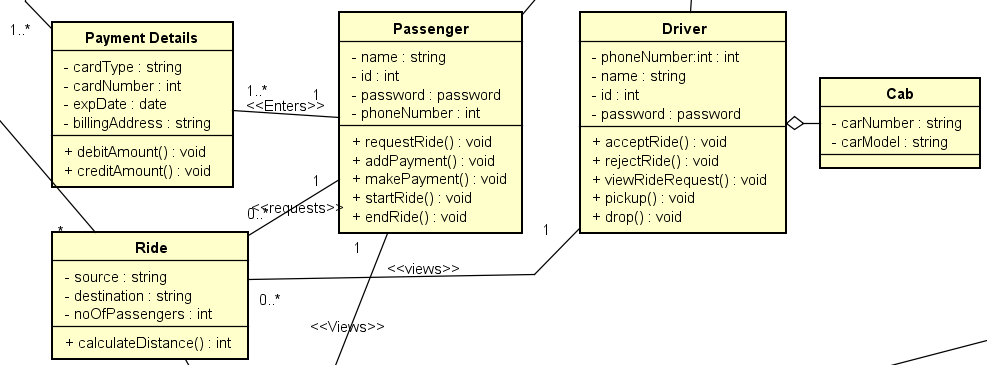


The above diagram is split into three parts for better clarity.

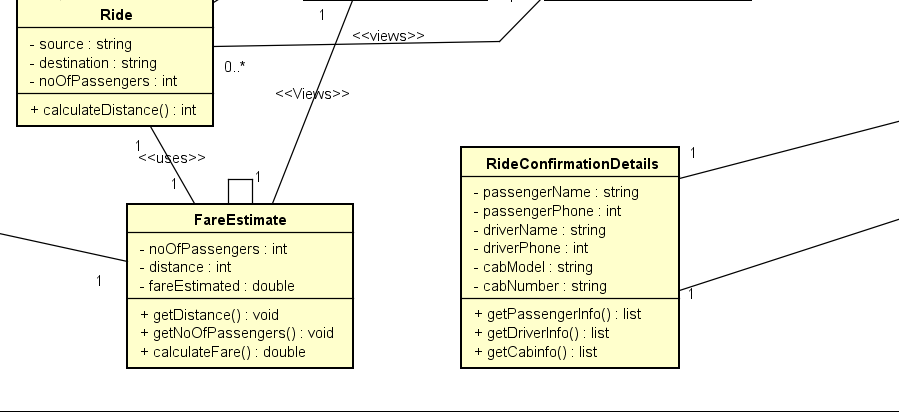
**Part 1:**



**Part 2:**



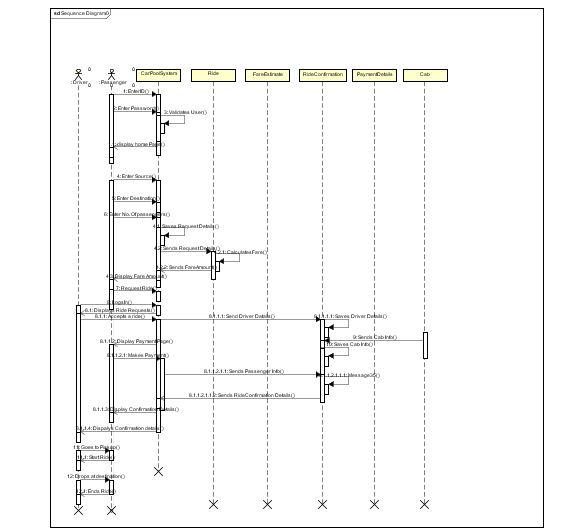
**Part 3:**



**2.2. Interactive Diagram**

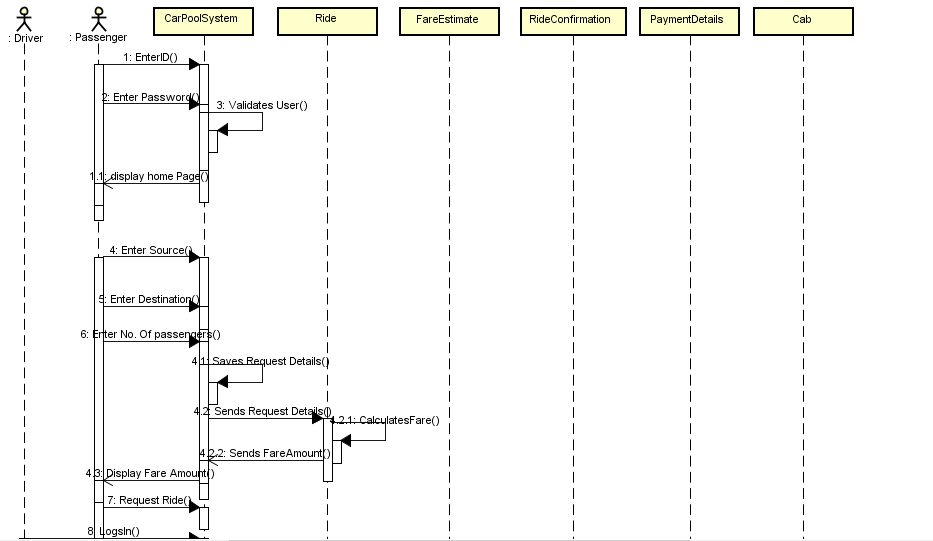
**Sequence Diagram**

The below sequence diagram shows the steps that are performed in using the system, right from the user login, booking a ride, to taking a ride.

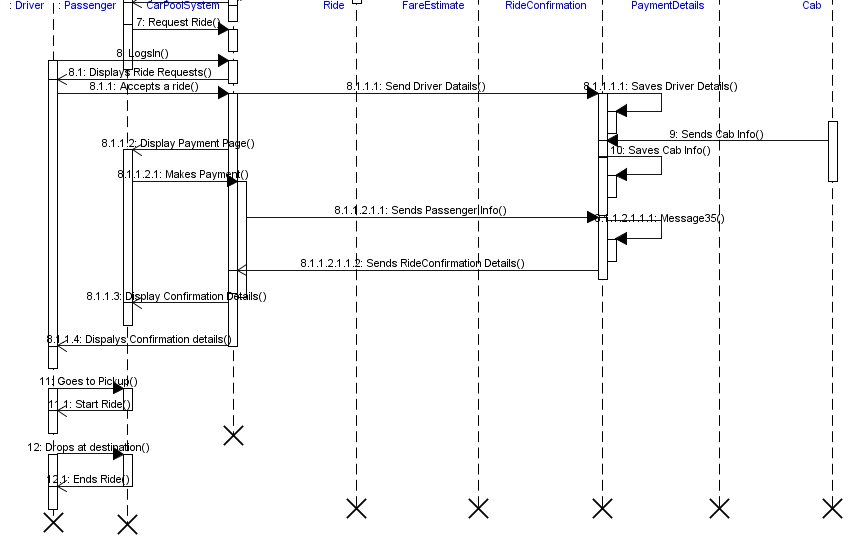


The above diagram is divided into two parts for better clarity.

**Part 1:**



**Part 2:**



2.3 Class-Responsibility-Collaboration

This shows the responsibilities of a class and Collaboration a class with other classes to perform the responsibilities.

|  |  |
| --- | --- |
| CarPoolSystem | |
| Responsibilities: | Collaborators: |
| Validates the user  Saves the source, destination and number of passengers  Displays estimated fare amount to the passenger  Saves the payment method entered by the passenger  Completes the transaction via saved payment method  Shows list of ride requests to driver  Sends ride confirmation details to the passenger and driver | Passenger  Driver    Ride    FareEstimate  RideConfirmationDetails  PaymentDetails |

|  |  |
| --- | --- |
| Passenger | |
| Responsibilities: | Collaborators: |
| Login into the system  Enter ride details  Request ride details  Adds payment method  Makes the payment  Takes the ride | CarPoolSystem  Ride  PaymentDetails  Driver |

|  |  |
| --- | --- |
| Driver | |
| Responsibilities: | Collaborators: |
| Login into the system  Accepts a ride from the list of available ride requests  Pickups and drops the passenger | CarPoolSystem  Ride  Passenger |

|  |  |
| --- | --- |
| Ride | |
| Responsibilities: | Collaborators: |
| Gets Source, destination and number of passengers from CarPoolSystem and saves them  Calculates the distance  Sends distance and number of passengers to fare estimate | CarPoolSystem  Passenger  Driver    FareEstimate |

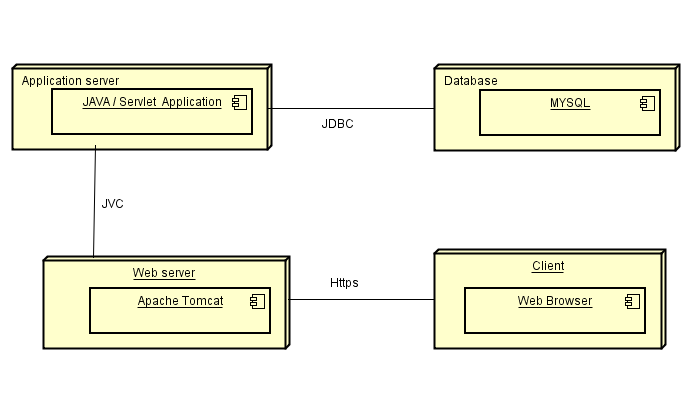
|  |  |
| --- | --- |
| FareEstimate | |
| Responsibilities: | Collaborators: |
| Takes the distance and number of passengers from the Ride and calculates the fare.  Sends estimated fare amount to the car pool system | CarPoolSystem  Ride |

|  |  |
| --- | --- |
| RideConfirmationDetails | |
| Responsibilities: | Collaborators: |
| Gets driver details, passenger details, cab details and save them  Sends the ride confirmation details to the carpool system | CarPoolSystem  Passenger  Driver |

|  |  |
| --- | --- |
| PaymentDetails | |
| Responsibilities: | Collaborators: |
| Adds the payment method  Saves the payment method  Sends the saved payment method to the carpool system | CarPoolSystem  Passenger |

**2.4 Deployment Diagram**

All users can use the system using a web browser. Web browser communicates with the web server that uses apache tomcat, through HTTP protocol. Web server then communicates with the application server that uses JAVA or servlet for communication. Then Application server communicates the MYSQL database for the data.



**3. Schedules and Milestones**

Design Document - 11/04/2018

Implementation - 11/15/2018

User Acceptance Testing - 11/ 17/2018

Deployment - 11/25/2018

Final release with bug fixing - 12/09/2018

**4. Teammate Responsibility**

Rajesh Reddy Burra - developing the front end of the application, implementation, and

documentation

Baldev Yellapu - developing the backend of the application, implementation and

documentation.