A Project Report on

Vehicle Parking Management

By

Aashutosh Rajendra Pawar

MCA – I, SEM – I

2023-24

To
Savitribai Phule Pune University,Pune

In Partial Fulfillment of the Degree of
Master in Computer Application (M. C. A.)

Under The Guidance Of
Prof. Smita Chavan

Suryadatta Group of Institutes, Pune
Suryadatta Institute of Business Management and Technology(SIBMT)

CERTIFICATE

This is to certify that Mr. / Ms. Aashutosh Rajendra Pawar, have successfully completed project work entitled **“Vehicle Parking Management”** in partial fulfillment of MCA – I Semester-I program for the year A.Y. 2023-24 . He / She has worked under our guidance and direction.

Prof.(Smita Chavan)
(Project Guide)

Dr.Manisha Kumbhar
Professor & Director SIBMT

Examiner 1

Examiner 2

Date :
Place : Pune

Acknowledgment

I am the student of MCA first year. Here by I express my thanks to our project guide for allowing us to do the project on “**Vehicle Parking Management”** This project work has been the most exciting part of our learning experience which would be an asset for our future carrier. I would especially like to thank our guide and mentor Prof. Smita Chavan , who constantly guided us in developing, pushing us to search for more answers to her numerous questions. Also I would like to thank our project coordinators Dr. Rupali Dahake & Prof. Apurva Patil, for their support. As a building block of MCA Department, I thank Professor & Director of SIBMT, Dr. Manisha Kumbhar , for her continuous support and help. I am grateful to many classmates who contributed their suggestions. Their hard work and examples push us to limits of our capability and encourage us daily.

Thank You

Aashutosh Rajendra Pawar

Index

Chapter 1 : Introduction

Chapter 1 : Introduction

- Parking management system for managing the records of the records of the incoming and outgoing vehicles in a parking lot its an easy for Admin to retrieve the data if the vehicle has been visited through number.

- Now days in many public places such as college, malls, problem of vehicle parking. The vehicle parking area has many lanes/slots for car parking. Instead of vehicle caught in towing the vehicle can park on safe and security with low cost.

- Parking control system has been provided in such a way that it is filled with many secure services such as, bill payment mechanism, time and attendance tracking, car/bike counting system etc. These features are hereby very necessary nowadays to secure your vehicle and also to evaluate the fee structure for every vehicle's entry and exit.

- The objective of this project is to build a Vehicle Parking management system that enables the time management and control of vehicles using computerized system. The system that will track the entry and exit of vehicles, maintain a listing of vehicles within the parking lot, and determine if the parking lot is full or not.

1.1 Existing System

The existing system provide the basic functionalities which are to be fulfilled by the owner. There is no intelligence of the software in such cases. In the existing system all the customer details, product and order details are maintained manually by the owner.

There are many loopholes when we look at the security of the system. These are the main disadvantages of the existing systems that are overcome in the proposed model.

Key points of existing system:

- The existing system only provides text-based interface with manual reporting, which is not user- friendly.
- System works in manual format so it takes a lot of time for the implementation of the work
- As working is in manual way, so fetching records is slow.

Data can be lost, redundant and even mismatch as details are preserved on paper.

1.2 - Need for New System

One needs a system that provides the functionality necessary to run the services in an easy and convenient way for the users, and delivers a competitive advantage. The system they have in place today may not be meeting the needs or may not be able to support the growth and innovation that are part of their parking management system success.

Following are some reasons to take a serious look at replacing their current traditional system with the one that is able to support the shop's need:

- By computerized system the owner just has to click or be sometimes type the record whenever needed, as the software will provides friendly screen so he will get all the information quite easily whenever required.

- The admin can easily modify and edit the data as per customer details and proceed further with the help of the software

- Keeping the records manually can introduce many data errors and hence introducing a computerized system will help reducing errors in the data.

➢ **Problem with existing system :**

- Limited Parking Spaces: In urban areas, finding a suitable parking spot can be a significant challenge due to the high demand for limited spaces.

- Congested Parking Lots: Parking lots, especially in shopping centers or events, can become crowded and chaotic, making it difficult to find an available space.

- Parallel Parking Difficulty: Parallel parking can be daunting for many drivers, especially in busy streets or tight spaces. It requires skill and precision.

- Lack of Adequate Signage: Poor signage or unclear parking regulations can lead to confusion and result in parking violations or towing.

- Size of Parking Spaces: Some parking spaces may be too small for larger vehicles like trucks or SUVs, making it challenging to park without the risk of damage.

- Illegal Parking: Illegally parked vehicles, such as those blocking driveways or fire hydrants, can create problems for both the owner of the vehicle and other road users.

- Parking Fees and Restrictions: Many parking areas have fees and time restrictions, causing inconvenience for those who need to park for an extended period or who are on a tight budget.

- Limited Accessibility: Lack of accessible parking spaces for people with disabilities can pose a significant problem, making it challenging for them to access buildings and services.

- Lack of Parking Infrastructure: In some areas, the absence of proper parking infrastructure contributes to the challenges people face in finding suitable parking spaces.

- Risk of Theft or Vandalism: In poorly lit or isolated parking areas, there is an increased risk of vehicle theft or vandalism, leading to concerns for the safety of the parked vehicle.

- Traffic Congestion: The process of finding parking, especially in busy areas, can contribute to traffic congestion as drivers circle around in search of a space.

1.3 Operating Environment Hardware and Software

Hardware Requirements:

- **Server:**
    - Description: The system might utilize a centralized server for hosting the MySQL database.
    - Requirements: A server with sufficient processing power, memory, and storage to handle database operations.

- **Client Devices:**
    - Description: Devices running the Java desktop application.
    - Requirements: Standard desktop computers or laptops capable of running Java applications, with sufficient memory and processing power.

- **Software Requirements:**

- **Operating System (Server and Client):**
    - Description: The underlying software on both the server and client machines.
    - Requirements:

    Server: Windows Server.
    Client: Windows, macOS, or Linux for desktop application support.

- **Java Development Kit (JDK):**
    - Description: Software development kit for building Java applications.
    - Requirements: The appropriate version of JDK installed on both the server and client machines.

- **IDE for Java Development:**
    - Description: Integrated Development Environment for building Java applications.
    - Requirements: IDEs like NetBeans, Eclipse, or IntelliJ IDEA for developing the desktop application.
- **MySQL Database Server:**
    - Description: Database management system for storing blood bank information.
    - Requirements: MySQL Server installed and configured on the server.
- **Java Database Connectivity (JDBC) Driver:**
    - Description: Software for connecting Java applications to the MySQL database.
    - Requirements: MySQL Connector/J or another JDBC driver compatible with MySQL.
- **Java Desktop Application Framework/Library:**
    - Description: Framework or library for building Java desktop applications.
    - Requirements: JavaFX, Swing, or another suitable library for creating the graphical user interface (GUI).

Chapter 2 : Proposed System

2.1 Proposed System(Introduction of System)

A Vehicle Parking Management System is designed to efficiently manage and optimize parking spaces, providing a seamless experience for both parking operators and users. Here's a proposed system with key features:

- **User Registration and Authentication:**
  - Users register their vehicles and personal information.
- **Parking Space Availability:**
  - Real-time monitoring of parking spaces to determine availability.
  - Display available spaces through digital displays at the entrance.
- **Reservation System:**
  - Allow users to reserve parking spaces in advance.
  - Set a time limit for reservation to ensure fairness.
- **Parking Guidance System:**
  - Use sensors and digital signage to guide drivers to available spaces.
- **Security Features:**
  - Implement surveillance cameras for security.
  - Integrate an alarm system to deter unauthorized access.
- **Scalability:**
  - Design the system to be easily scalable to accommodate future expansions.

This proposed system aims to enhance the overall parking experience by leveraging technology to streamline processes, improve efficiency, and provide a user-friendly interface for both operators and users.

2.2 Module Specification

A module specification outlines the individual components or modules within a system and describes their functionality and interactions. For a Vehicle Parking Management System, various modules can be identified to ensure effective management and operation. Below is a suggested list of modules along with their scope:

- **User Management Module:**
  - User registration and authentication.
  - Profile management.
  - Access control and permission settings.
- **Parking Space Availability Module:**
  - Real-time monitoring of parking spaces.
  - Detection of occupied and vacant spaces.
- **Parking Guidance Module:**
  - Real-time display of available parking spaces.
  - Navigation to the selected parking space.
  - Dynamic signage for different parking zones.
- **Data Analytics Module:**
  - Analysis of parking usage data.
  - Identification of peak hours.

These modules collectively form a comprehensive Vehicle Parking Management System, addressing various aspects of user interaction, system functionality, security, and maintenance. Each module should be designed to work seamlessly with others to ensure the overall efficiency of the system.

2.3 Objectives of System

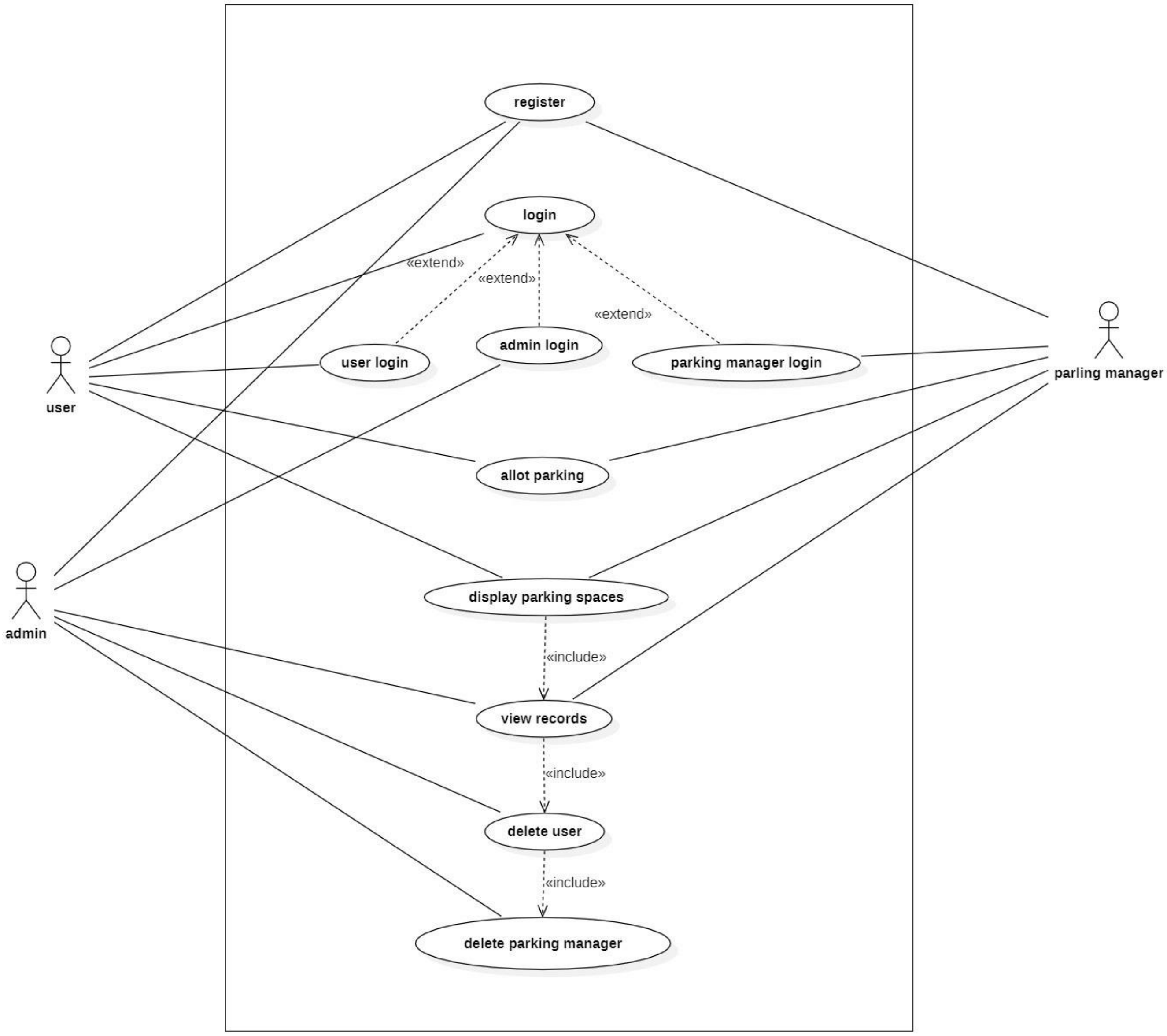
The objectives of a Vehicle Parking Management System are to streamline the parking process, enhance user experience, optimize space utilization, and ensure the efficient operation of parking facilities. Here are the key objectives:

- **Optimize Space Utilization:**
  - Maximize the use of available parking spaces to reduce congestion.
  - Efficiently allocate parking spots based on demand.
- **Enhancing Security:**
  - Ensure the safety of both vehicles and pedestrians within the parking facility.
- **Improving User Experience:**
  - Provide real-time information about available parking spaces through digital displays.
  - Offer convenient payment options and user-friendly interfaces for a positive experience.

By achieving these objectives, a Vehicle Parking Management System can contribute to a more organized, efficient, and user-friendly parking environment.

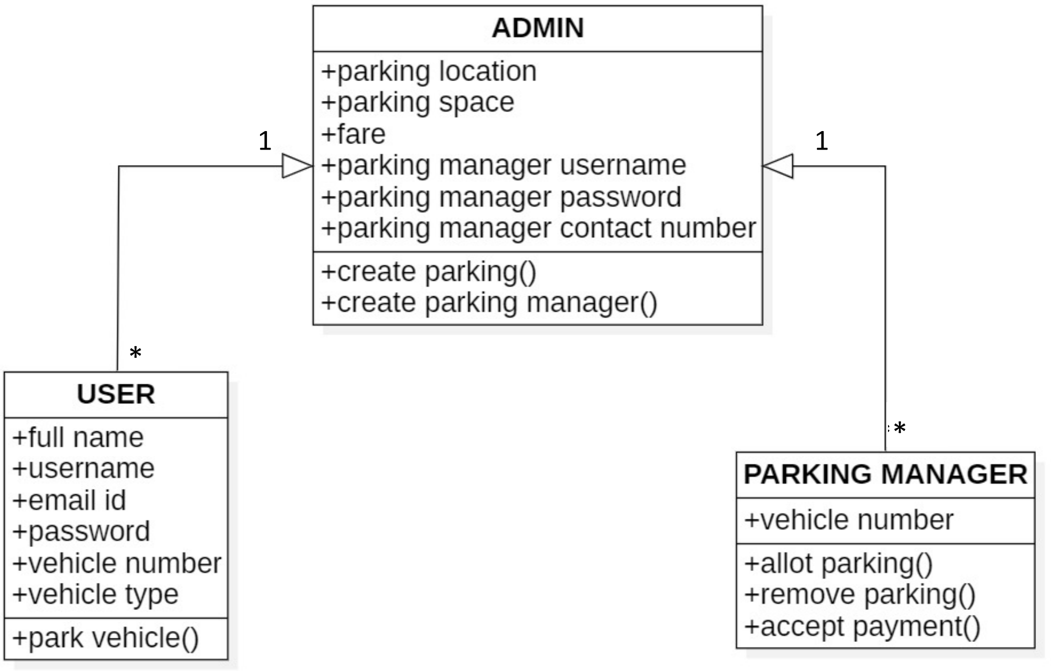
Chapter 3 : Analysis & Design

System Use Case :

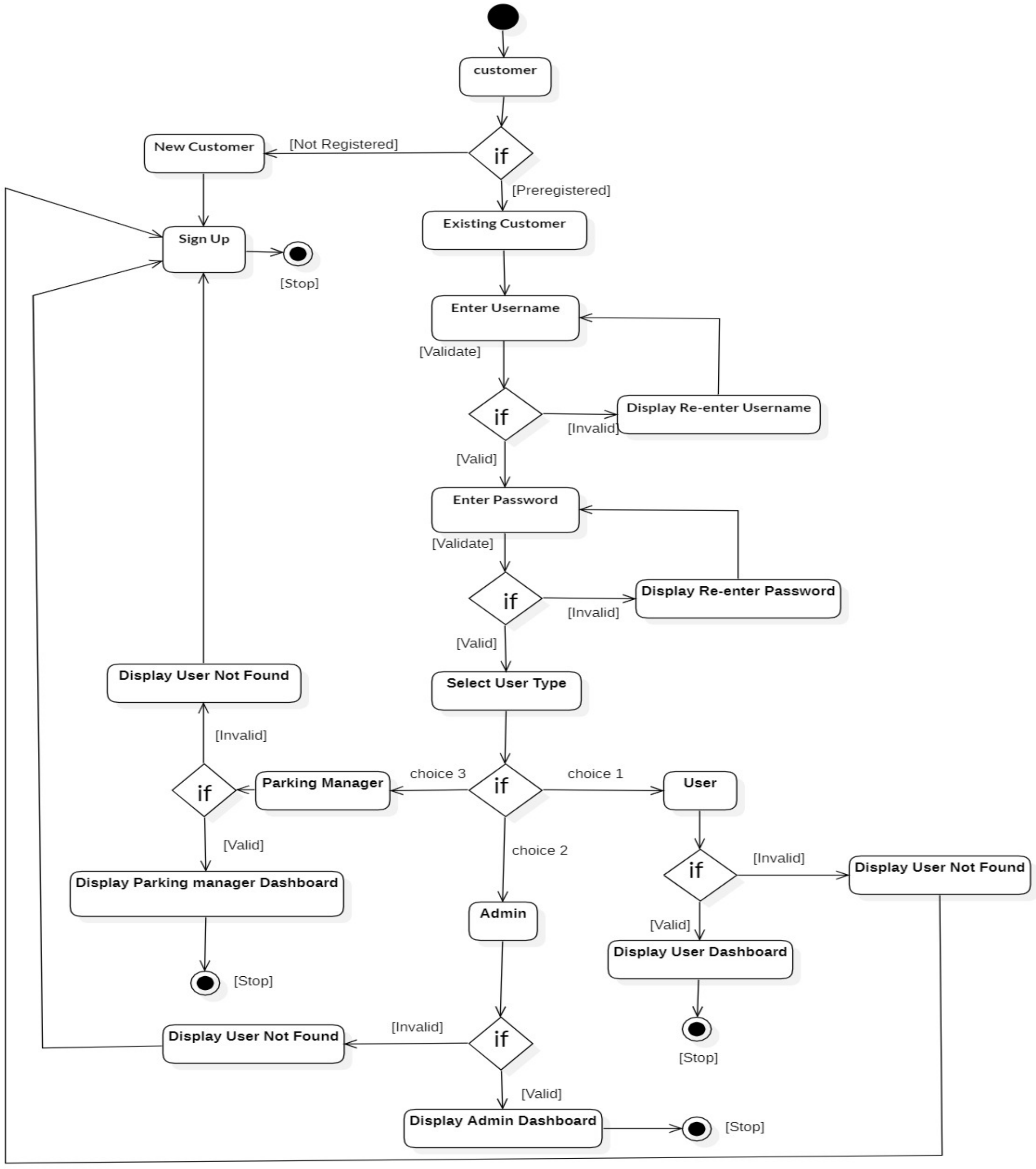


Class Diagram :

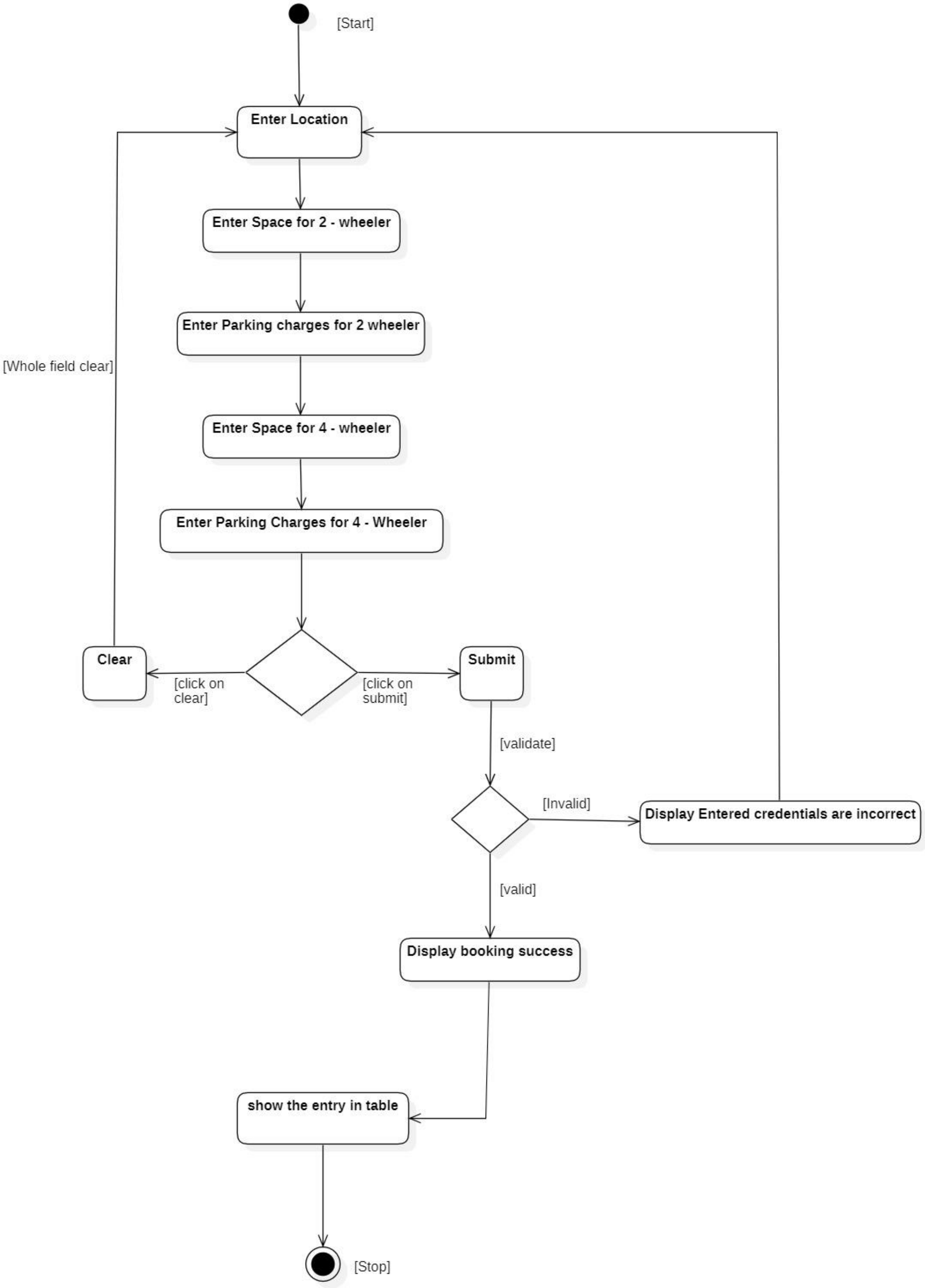
Class Diagram for vehicle parking Manager



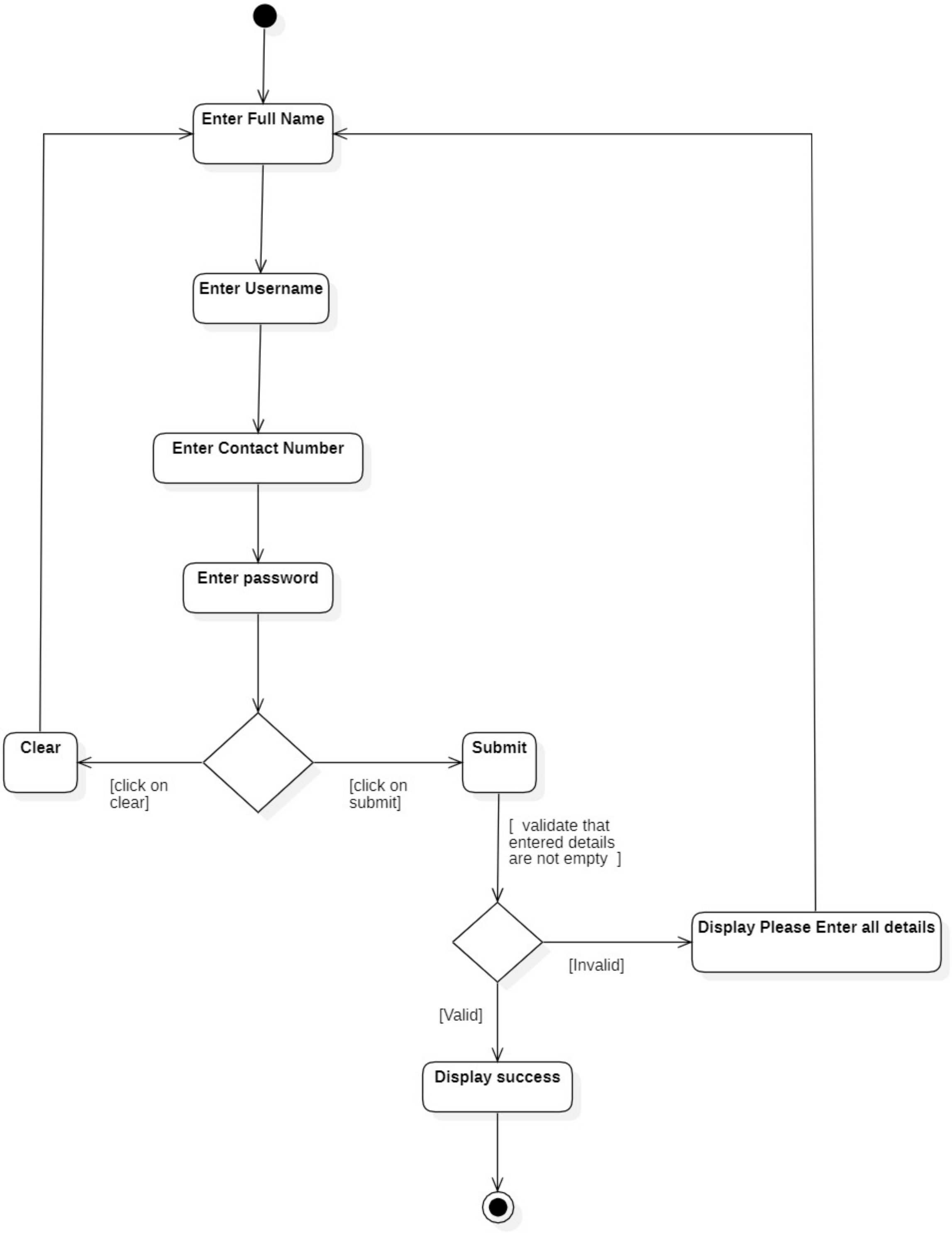
Activity Diagram - Login Process



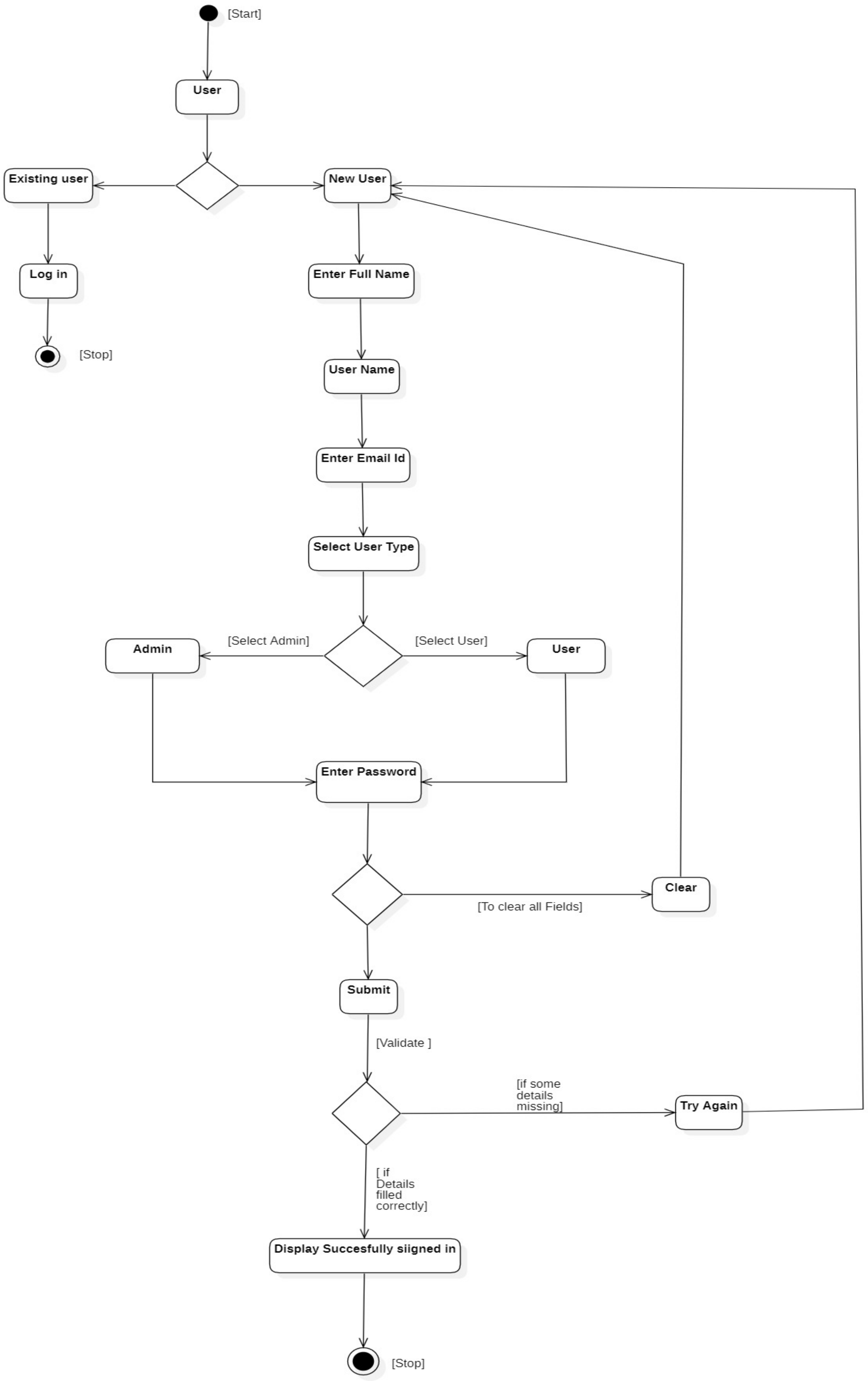
Admin Dashboard - add space


[Start]
Enter Location
Enter Space for 2 - wheeler
Enter Parking charges for 2 wheeler
[Whole field clear]
Enter Space for 4 - wheeler
Enter Parking Charges for 4 - Wheeler
Clear
[click on clear]
[click on submit]
Submit
[validate]
[Invalid]
Display Entered credentials are incorrect
[valid]
Display booking success
show the entry in table
[Stop]

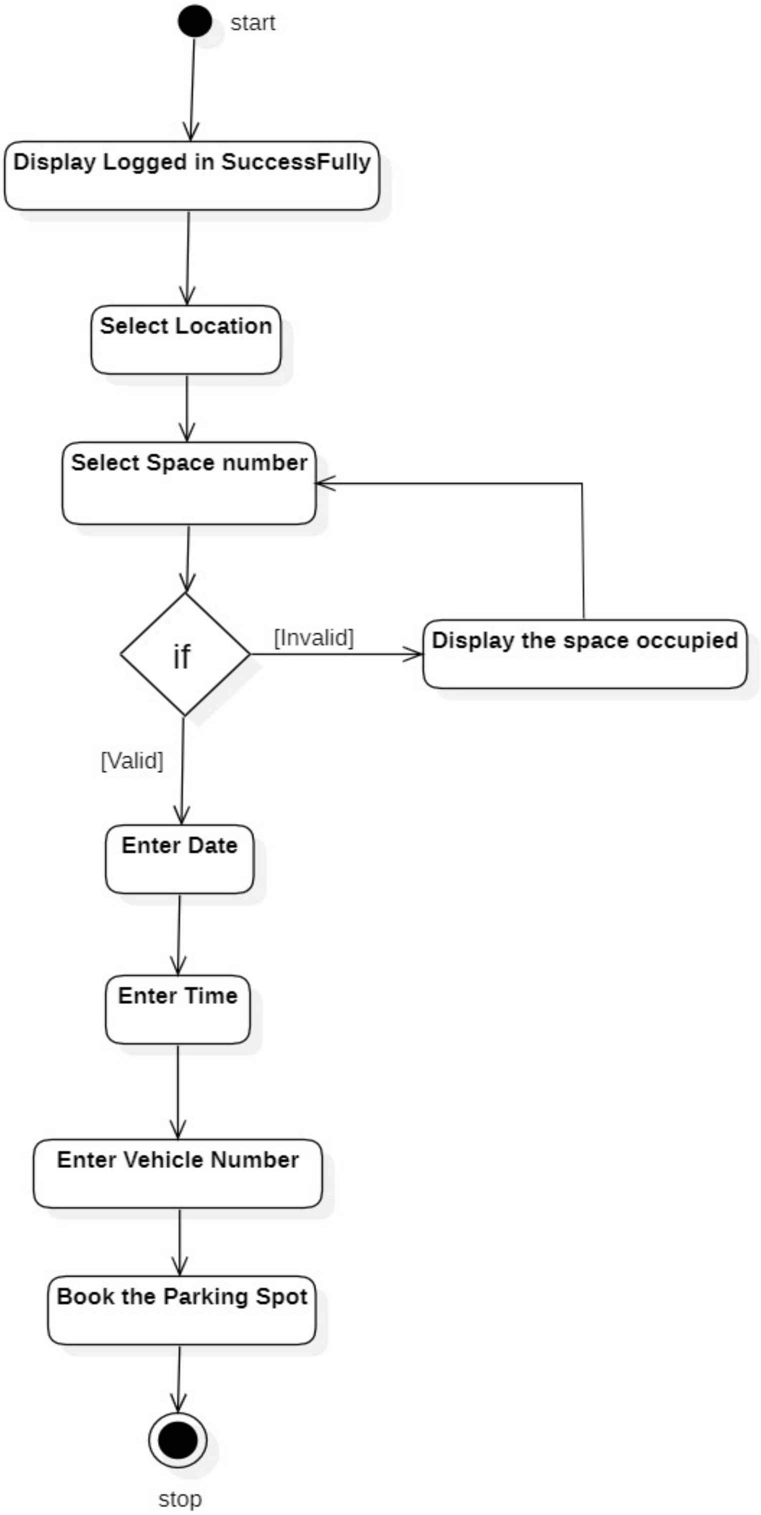
Admin Dashboard - new parking manager



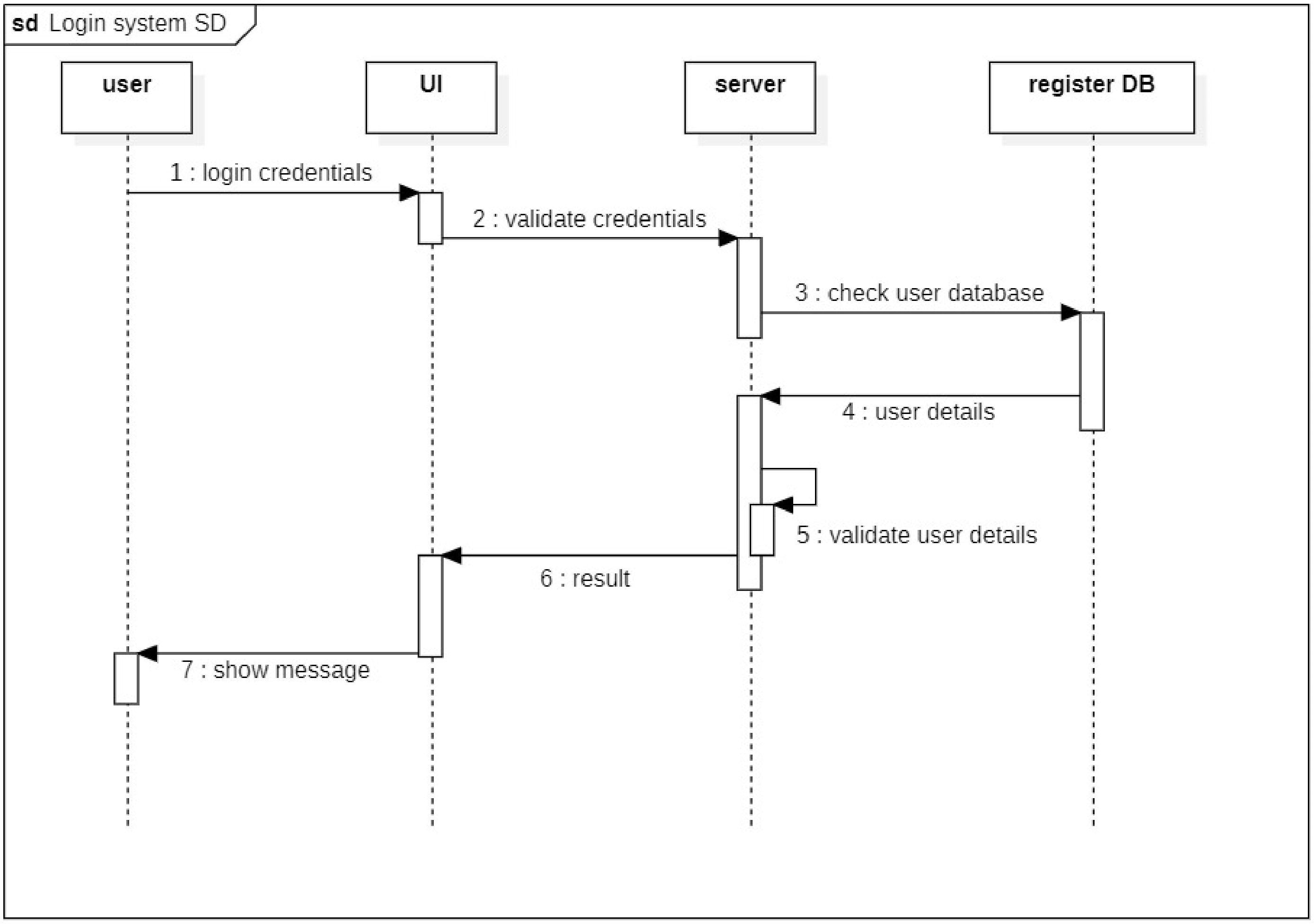
Signup process



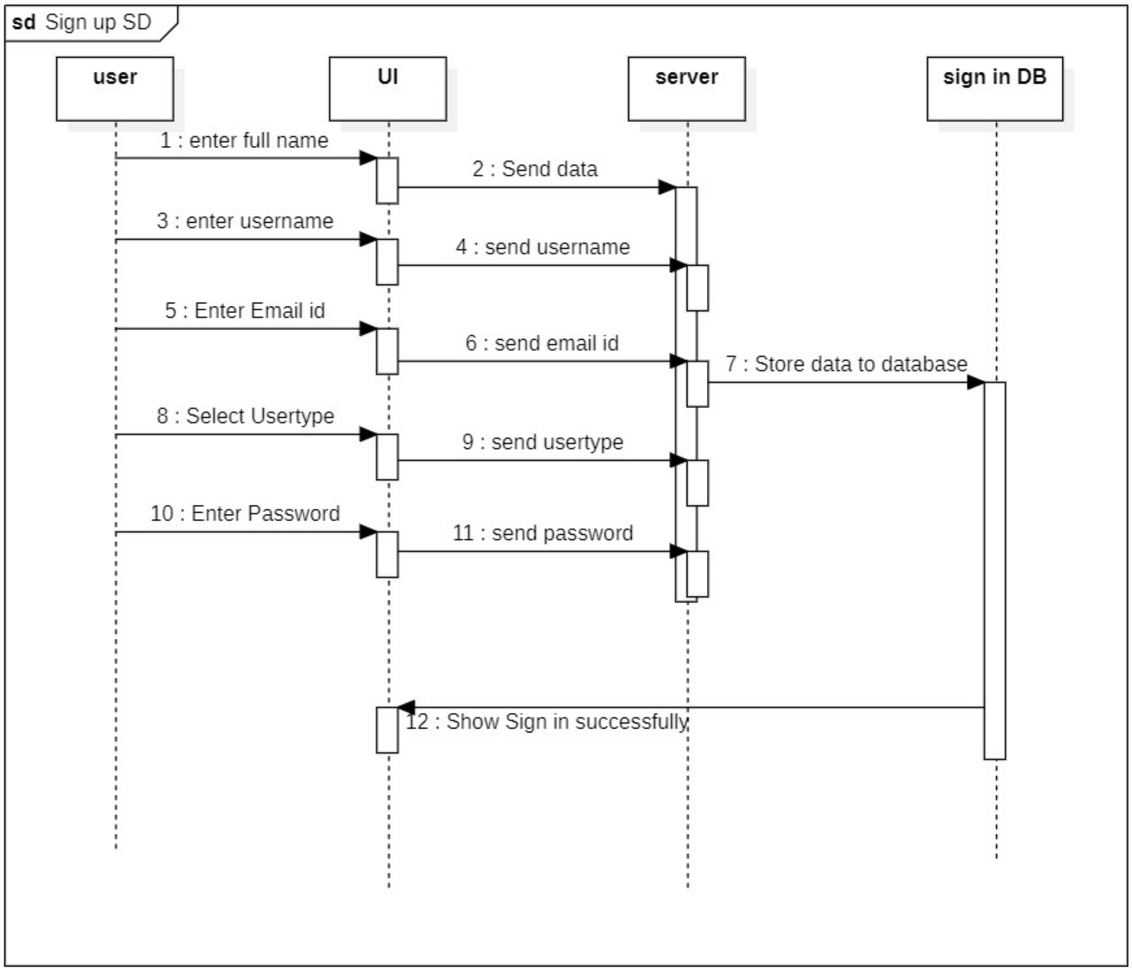
User Dashboard Activity Diagram



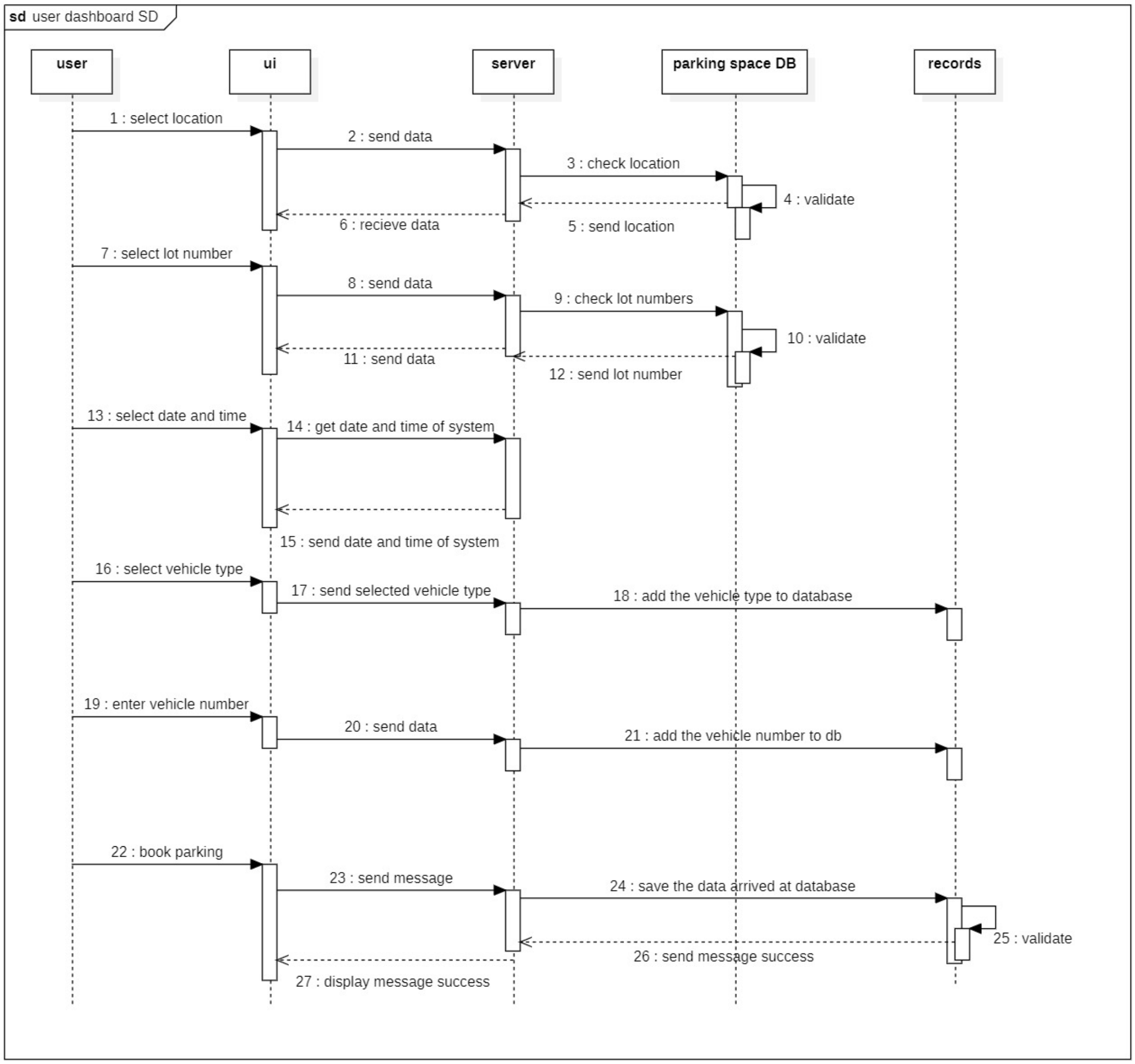
Log in System Sequence Diagram:



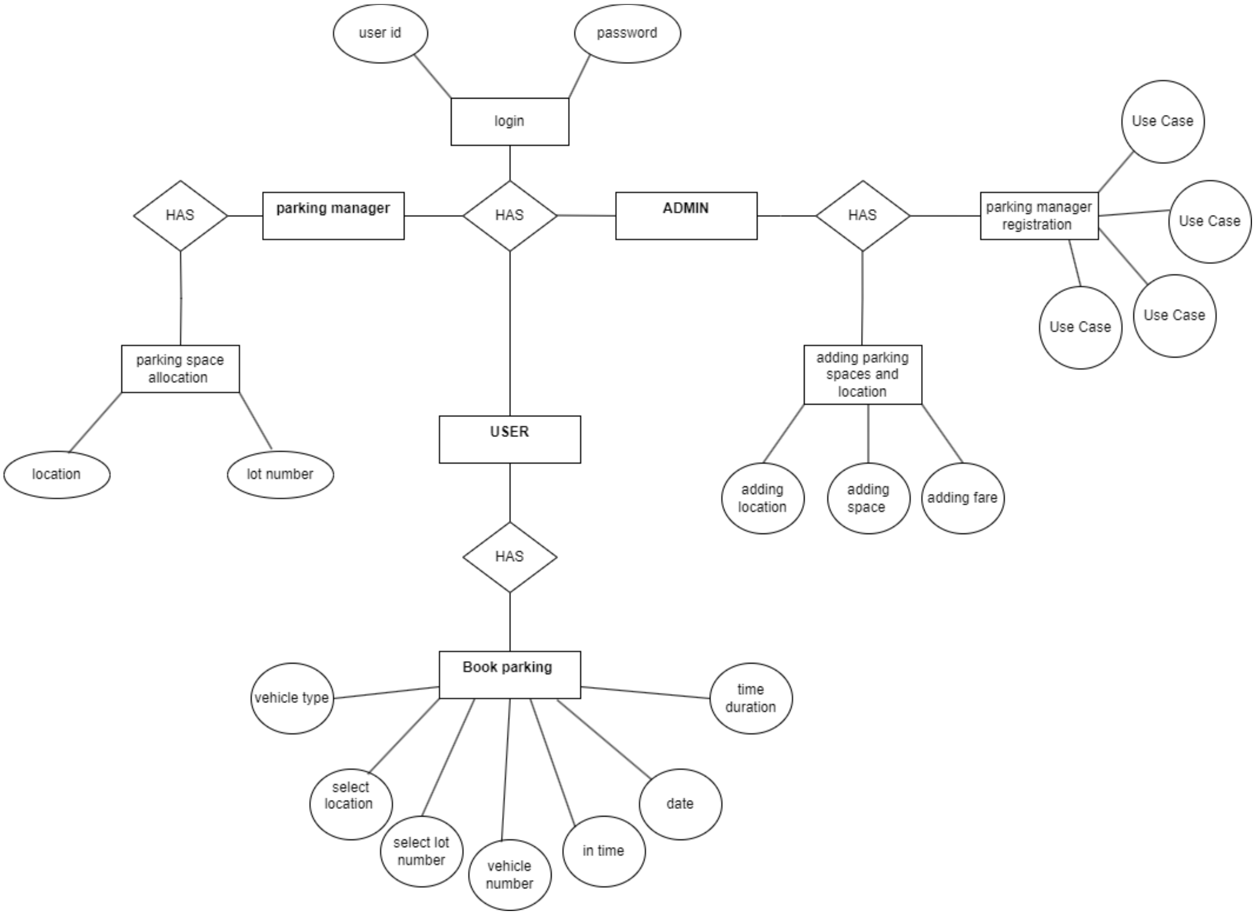
Sign up Sequence Diagram:



User dashboard sequence Diagram:



Entity Relationship diagram :



Database Design / Data Dictionary:

User Admin Registration:-

Attribute	Datatype	Constraints	Description
Full name	Varchar(20)	Not Null	To store name
User Id	Varchar(18)	Primary key	To store id
Vehicle Type	Varchar(15)	Not Null	To store vehicle type
Vehicle No	Varchar(10)	Not Null	To store vehicle no
EmailId	Varchar(25)	Not Null	To store email id
Contact No	Varchar(10)	Not Null	To store contact no
Password	Varchar(12)	Not Null	To store password

Parking Manager:-

Attribute	Datatype	Constraints	Description
ManagerName	Varchar(20)	NotNull	To store manager name
User ID	Varchar(20)	Primary key	To Store manager id
Usertype	Varchar(20)	Not Null	To Store user type
Contact Number	Varchar(10)	Not Null	To store manager contact
Password	Varchar(12)	NotNull	To store password

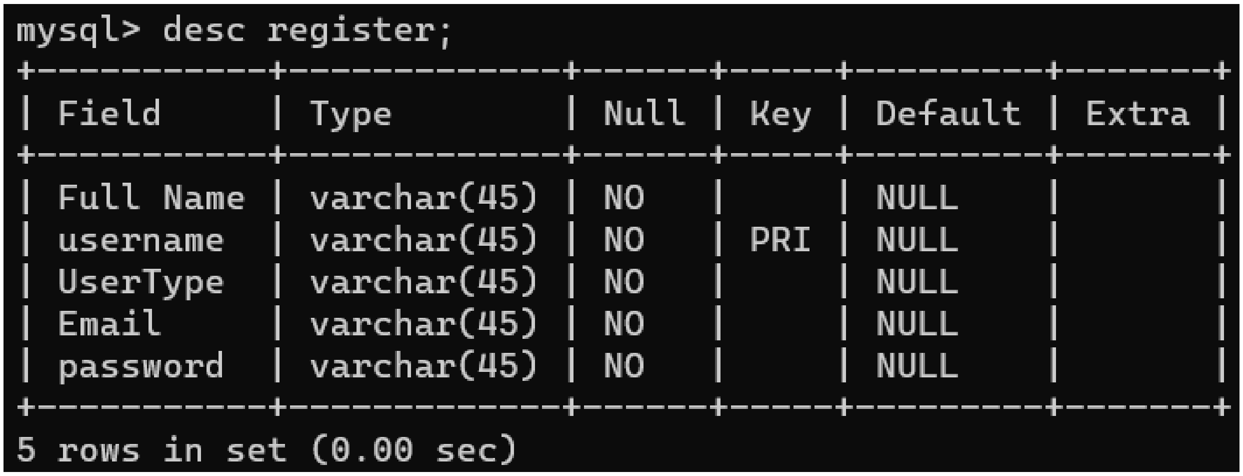
User Records:-

Attribute	Datatype	Constraints	Description
Location	Varchar(45)	Primary Key	To store Location
Location Number	Int	Not Null	To location number
Date	Varchar(45)	Not Null	To store date
In-Time	Varchar(45)	Not Null	To store time
Time Duration	Varchar(45)	Not Null	To store time duration
Vehicle Type	Varchar(45)	Not Null	To store vehicle type
Vehicle No	Varchar(45)	Not Null	To store vehicle no

Parking Spaces: -

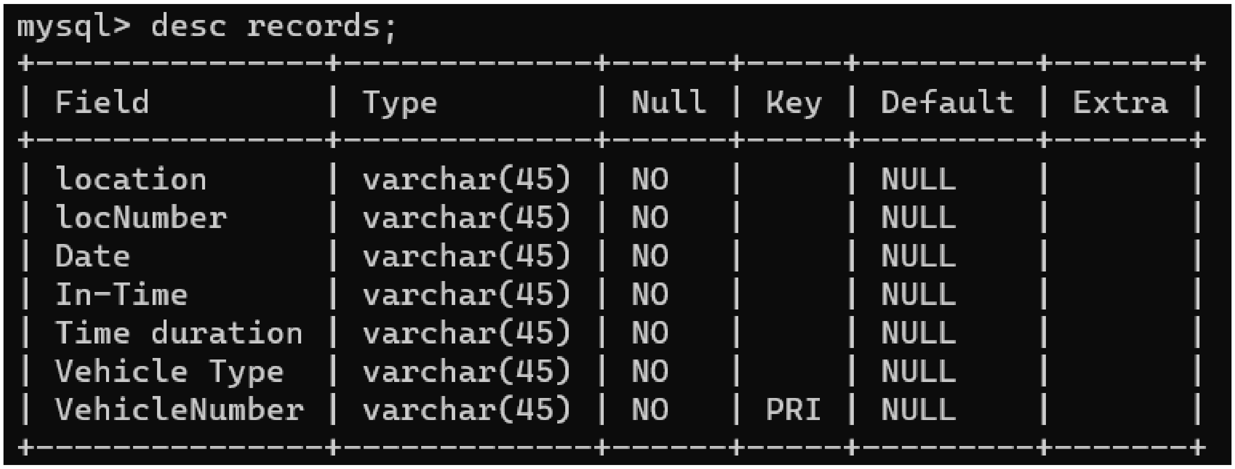
Attribute	Datatype	Constraints	Description
Location	Varchar(45)	Not Null	To store location
2-Wheeler space	Int	Not Null	To store vehicle space
2-wheeler parking fare	Int	Not Null	To store space
4-wheeler space	Int	Not Null	To store vehicle space
4-wheeler parking fare	Int	Not Null	To store space

Register Table :



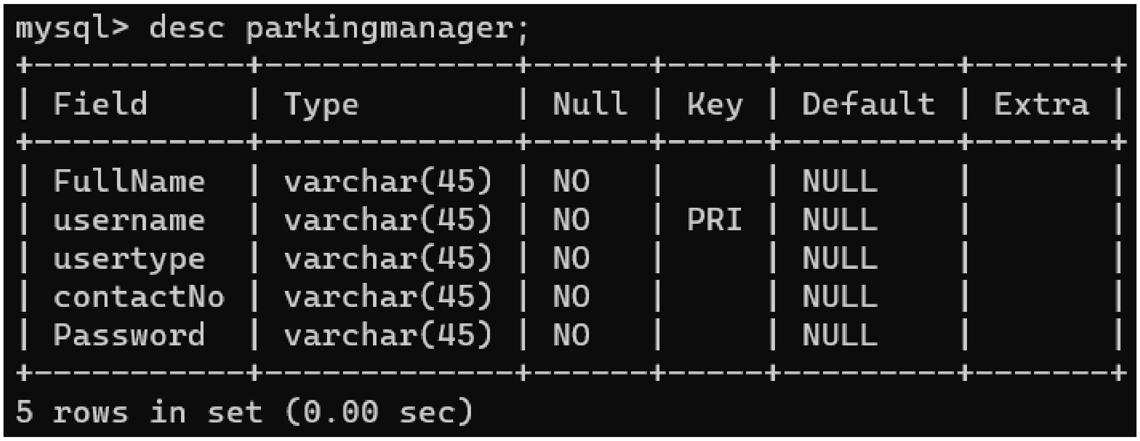
```
mysql> desc register;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| Full Name | varchar(45) | NO   |     | NULL    |       |
| username  | varchar(45) | NO   | PRI | NULL    |       |
| UserType  | varchar(45) | NO   |     | NULL    |       |
| Email     | varchar(45) | NO   |     | NULL    |       |
| password  | varchar(45) | NO   |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Records :



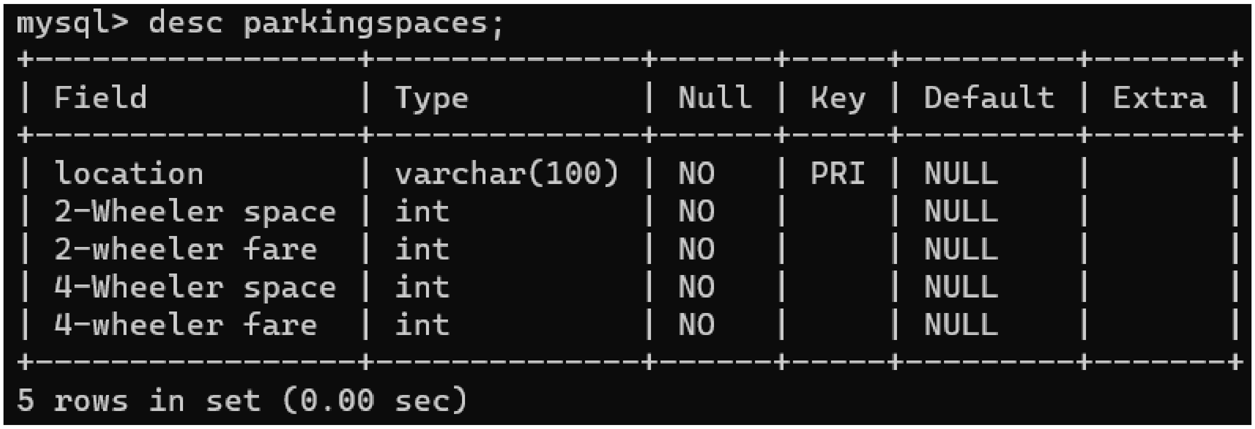
```
mysql> desc records;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| location      | varchar(45) | NO   |     | NULL    |       |
| locNumber     | varchar(45) | NO   |     | NULL    |       |
| Date          | varchar(45) | NO   |     | NULL    |       |
| In-Time       | varchar(45) | NO   |     | NULL    |       |
| Time duration | varchar(45) | NO   |     | NULL    |       |
| Vehicle Type  | varchar(45) | NO   |     | NULL    |       |
| VehicleNumber | varchar(45) | NO   | PRI | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
```

Parking Manager :



```
mysql> desc parkingmanager;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| FullName  | varchar(45) | NO   |     | NULL    |       |
| username  | varchar(45) | NO   | PRI | NULL    |       |
| usertype  | varchar(45) | NO   |     | NULL    |       |
| contactNo | varchar(45) | NO   |     | NULL    |       |
| Password  | varchar(45) | NO   |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Parking Spaces :

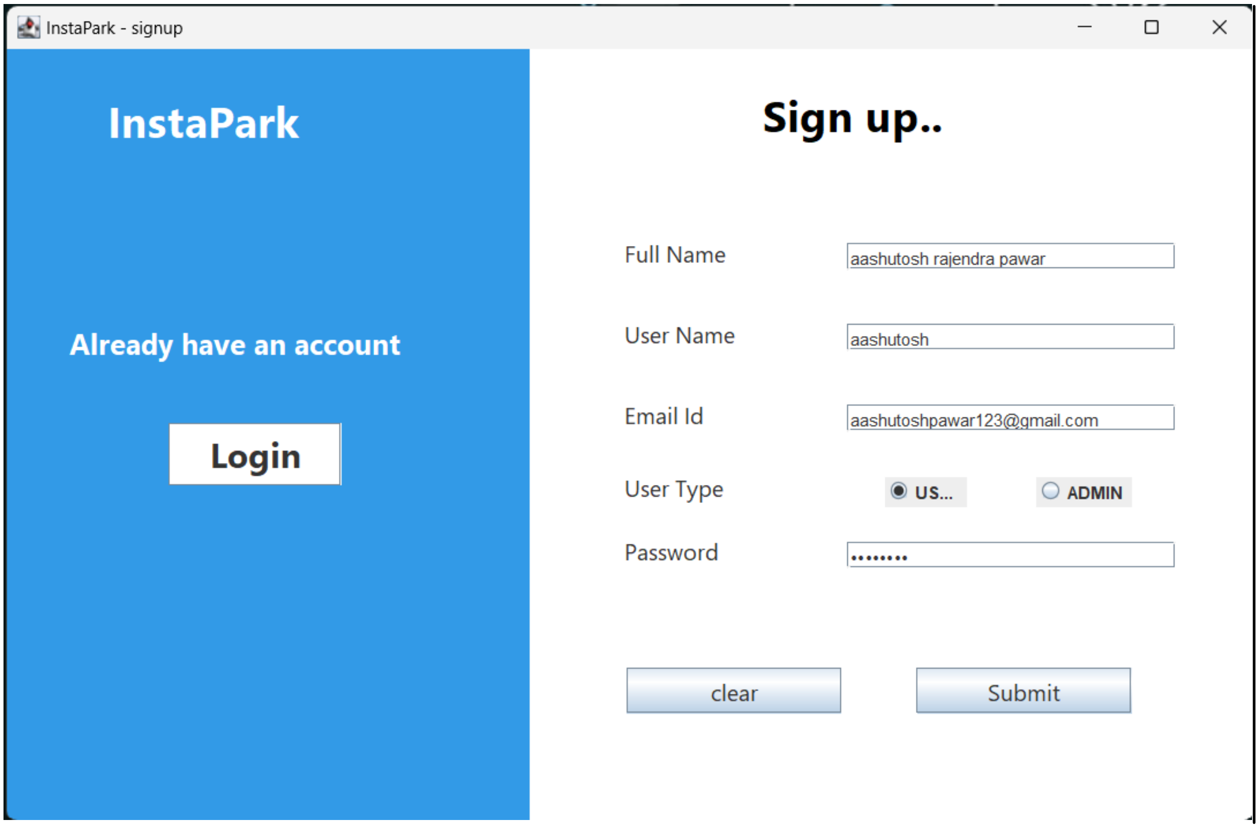


```
mysql> desc parkingspaces;
+-----------------+--------------+------+-----+---------+-------+
| Field           | Type         | Null | Key | Default | Extra |
+-----------------+--------------+------+-----+---------+-------+
| location        | varchar(100) | NO   | PRI | NULL    |       |
| 2-Wheeler space | int          | NO   |     | NULL    |       |
| 2-wheeler fare  | int          | NO   |     | NULL    |       |
| 4-Wheeler space | int          | NO   |     | NULL    |       |
| 4-wheeler fare  | int          | NO   |     | NULL    |       |
+-----------------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Chapter 4: User Manual

4.1 User Interface Screens (Input)

Sign up page



Full Name: aashutosh rajendra pawar

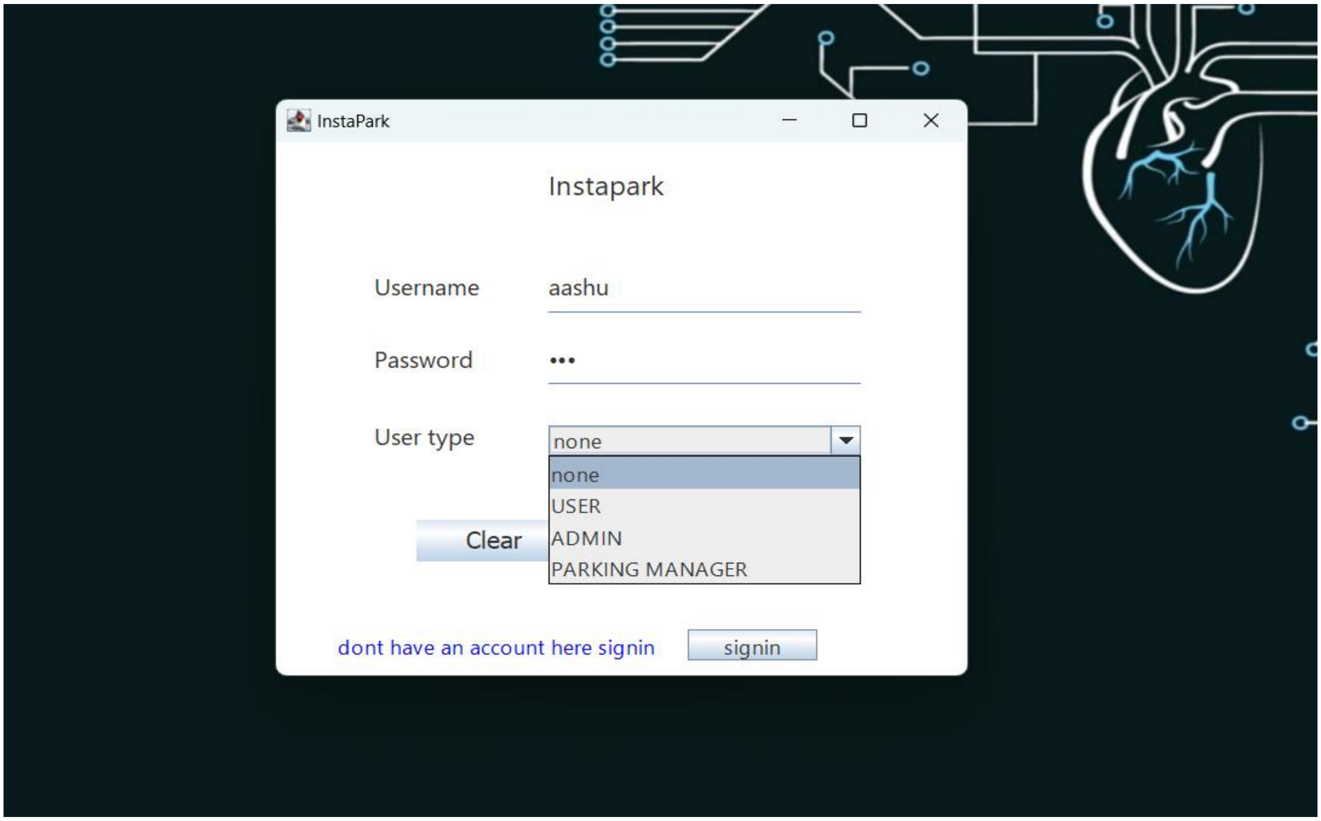
User Name: aashutosh

Email Id: aashutoshpawar123@gmail.com

User Type: ● US... ○ ADMIN

Password: ••••••••

Login Page:



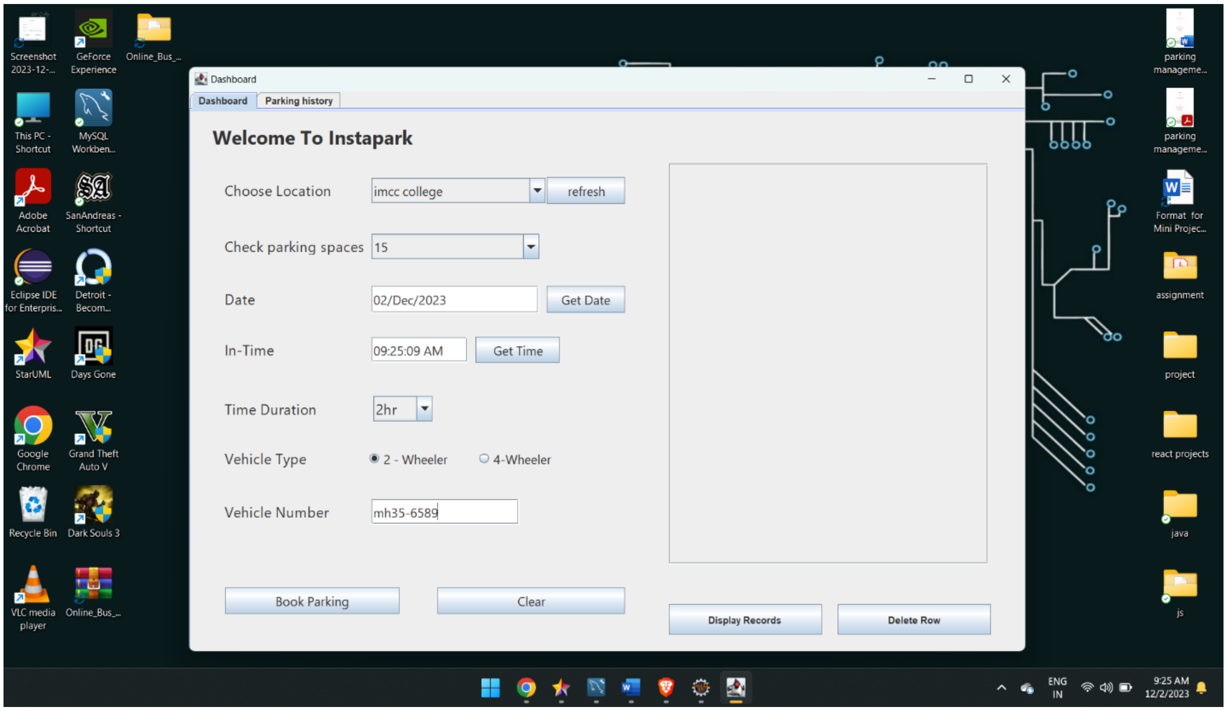
Instapark

Username aashu

Password •••

User type none

User Dashboard:



Welcome To Instapark

Choose Location: imcc college

Check parking spaces: 15

Date: 02/Dec/2023

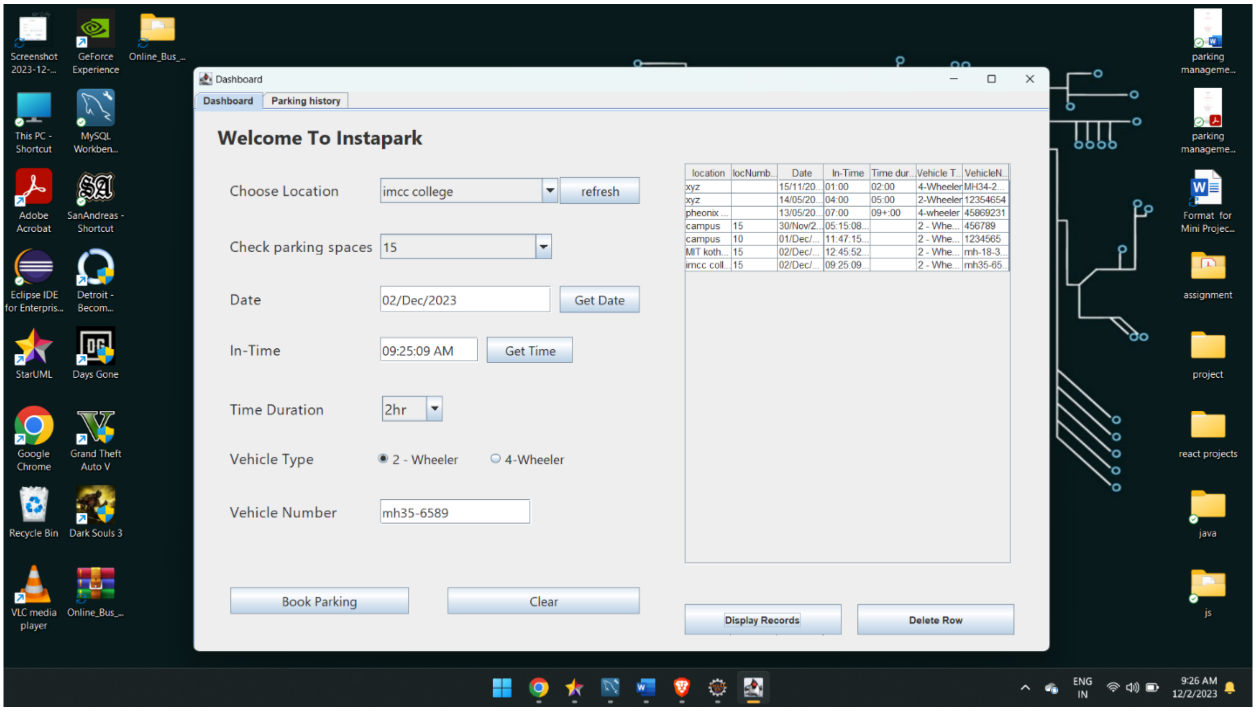
In-Time: 09:25:09 AM

Time Duration: 2hr

Vehicle Type: ● 2 - Wheeler ○ 4-Wheeler

Vehicle Number: mh35-6589

4.2 Output Screen With User Dashboard: -
database table successfully fetched the data from MySQL database



Welcome To Instapark

Choose Location: imcc college

Check parking spaces: 15

Date: 02/Dec/2023

In-Time: 09:25:09 AM

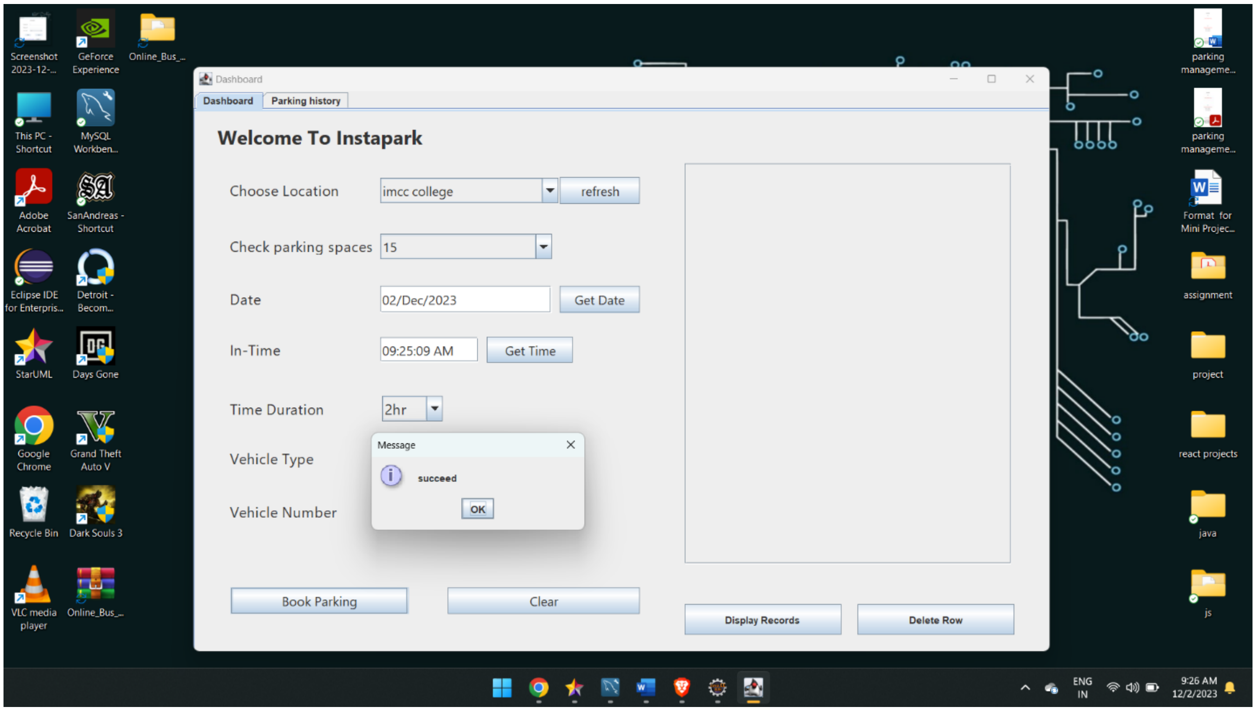
Time Duration: 2hr

Vehicle Type: ● 2 - Wheeler ○ 4-Wheeler

Vehicle Number: mh35-6589

location	locNumb...	Date	In-Time	Time dur...	Vehicle T...	VehicleN...
xyz		15/11/20...	01:00	02:00	4-Wheeler	MH34-2...
xyz		14/05/20...	04:00	05:00	2-Wheeler	12354654
pheonix ...		13/05/20...	07:00	09+:00	4-wheeler	45869231
campus	15	30/Nov/2...	05:15:08...		2 - Whe...	456789
campus	10	01/Dec/...	11:47:15...		2 - Whe...	1234565
MIT koth...	15	02/Dec/...	12:45:52...		2 - Whe...	mh-18-3...
imcc coll...	15	02/Dec/...	09:25:09...		2 - Whe...	mh35-65...

User Dashboard parking success :



Welcome To Instapark

Choose Location: imcc college

Check parking spaces: 15

Date: 02/Dec/2023

In-Time: 09:25:09 AM

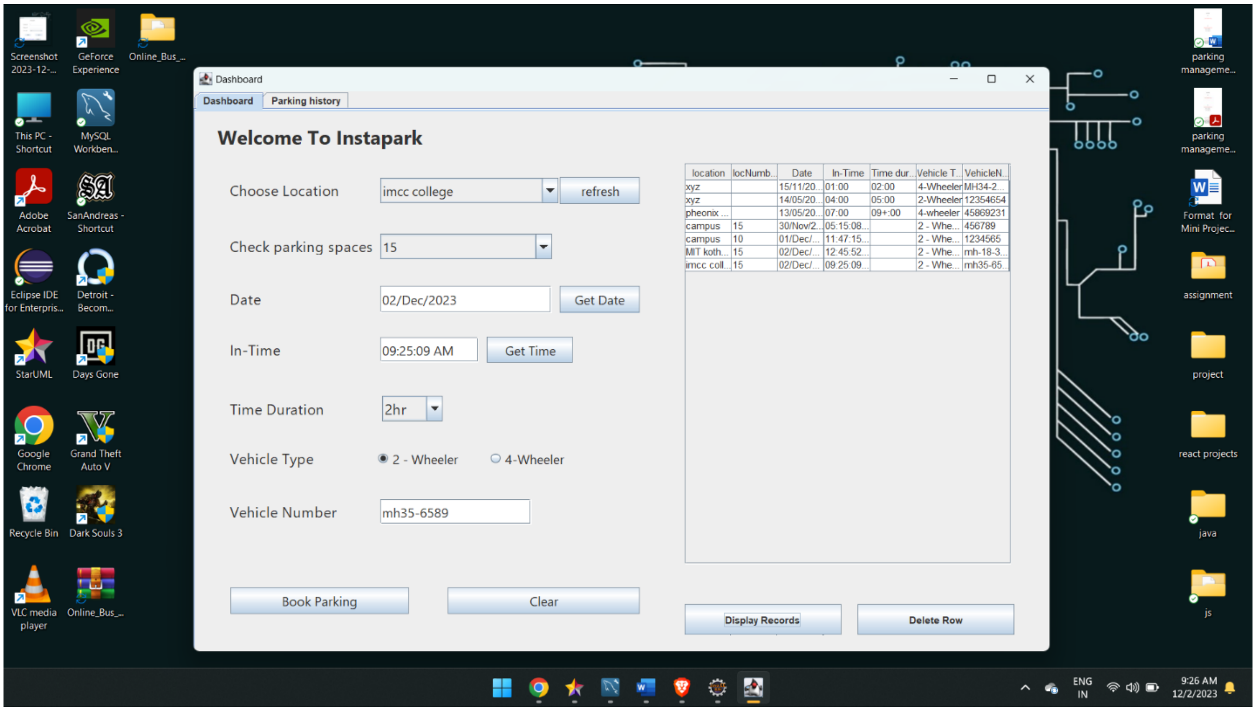
Time Duration: 2hr

Vehicle Type

Vehicle Number

succeed

4.3 Data report :



Welcome To Instapark

Choose Location: imcc college

Check parking spaces: 15

Date: 02/Dec/2023

In-Time: 09:25:09 AM

Time Duration: 2hr

Vehicle Type: ● 2 - Wheeler ○ 4-Wheeler

Vehicle Number: mh35-6589

location	locNumb...	Date	In-Time	Time dur...	Vehicle T...	VehicleN...
xyz		15/11/20...	01:00	02:00	4-Wheeler	MH34-2...
xyz		14/05/20...	04:00	05:00	2-Wheeler	12354654
pheonix ...		13/05/20...	07:00	09+:00	4-wheeler	45869231
campus	15	30/Nov/2...	05:15:08...		2 - Whe...	456789
campus	10	01/Dec/...	11:47:15...		2 - Whe...	1234565
MIT koth...	15	02/Dec/...	12:45:52...		2 - Whe...	mh-18-3...
imcc coll...	15	02/Dec/...	09:25:09...		2 - Whe...	mh35-65...

4.4 Test Procedure and cases :

1. **User Interface Testing:**

- **Test Procedures:**
  - Verify Login Functionality:
  - Enter valid credentials and ensure the system allows access.
  - Attempt to log in with invalid credentials and verify appropriate error messages.

- **Test Reservation Process :**
  - Navigate through the reservation process and ensure users can reserve parking spaces successfully.
  - Check for proper validation of user inputs during reservation.

- **Check Search Functionality:**
  - Perform searches for available parking spaces based on different criteria (location, time, type) and verify accurate results.

- **Test Entry and Exit Processes:**
  - Simulate vehicle entry and exit to validate the proper functioning of entry and exit modules.

- **Authentication and Authorization:**
  - Verify that only authorized users can access the system.
  - Test different user roles and their respective permissions.

4.5 Limitations and Bibliography

Limitations of the Vehicle Parking Management :

1 Data Accuracy:

- Dependency on accurate donor information.
- Potential discrepancies in manual data entry.

2 Technology Constraints:

- Reliance on stable internet connectivity.
- Compatibility issues with outdated hardware or software.

3 Privacy Concerns:

- Ensuring donor and recipient confidentiality.
- Addressing potential security breaches.

4 Staff Training:

- Adequate training required for staff to use the system efficiently.
- Resistance to change among existing staff members.

5 Financial Constraints:

- Initial setup costs and ongoing maintenance expenses.
- Budget constraints affecting system upgrades.

6 User Accessibility:

- Ensuring the system is user-friendly for all user members.

4.6 Sample Code :

```
package parkingmanagement;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.JSeparator;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.DefaultComboBoxModel;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.awt.event.ActionEvent;
public class mainframe extends JFrame {
    private static final long serialVersionUID = 1L;
    protected static final String userTest = null;
     private JPanel contentPane;
    private JTextField edtUsername;
    private JPasswordField edtPassword;
    public static void main(String[] args) {
      EventQueue.invokeLater(new Runnable() {
         public void run() {
               try {
                  mainframe frame = new mainframe();
                  frame.setVisible(true);
                        } catch (Exception e) {
                        e.printStackTrace();
                        }
               }
```

```
            }
      );
}
public mainframe() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 100, 471, 408);
    contentPane = new JPanel();
    contentPane.setBackground(new Color(255, 255, 255));
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setTitle("InstaPark");
    setContentPane(contentPane);
     contentPane.setLayout(null);
        JLabel lblNewLabel = new JLabel("Instapark ");
        lblNewLabel.setBounds(180, 10, 92, 39);
        lblNewLabel.setFont(new Font("Segoe UI", Font.PLAIN, 19));
      contentPane.add(lblNewLabel);
           JLabel lblNewLabel_1 = new JLabel("Username");
           lblNewLabel_1.setFont(new Font("Segoe UI", Font.PLAIN, 16));
           lblNewLabel_1.setBounds(65, 89, 92, 22);
       contentPane.add(lblNewLabel_1);
             JLabel lblNewLabel_2 = new JLabel("Password");
             lblNewLabel_2.setFont(new Font("Segoe UI", Font.PLAIN, 16));
             lblNewLabel_2.setBounds(65, 139, 92, 22);
           contentPane.add(lblNewLabel_2);
                edtUsername = new JTextField();
                edtUsername.setFont(new Font("Segoe UI", Font.PLAIN, 16));
                edtUsername.setBounds(180, 90, 207, 19);
                contentPane.add(edtUsername);
                edtUsername.setColumns(10);
                edtUsername.setBorder(null);
           JComboBox usertype = new JComboBox();
           usertype.setFont(new Font("Segoe UI", Font.PLAIN, 14));
  usertype.setModel(new DefaultComboBoxModel(new String[] {"none", "USER",
"ADMIN", "PARKING MANAGER"}));
  usertype.setSelectedIndex(0);
  usertype.setBounds(180, 197, 207, 21);
  contentPane.add(usertype);
  edtPassword = new JPasswordField();
  edtPassword.setFont(new Font("Segoe UI", Font.PLAIN, 16));
  edtPassword.setBounds(180, 140, 207, 19);
```

```
    contentPane.add(edtPassword);
    edtPassword.setBorder(null);
    JSeparator separator = new JSeparator();
    separator.setBounds(180, 118, 207, 19);
    contentPane.add(separator);
    JSeparator separator_1 = new JSeparator();
    separator_1.setBounds(180, 168, 207, 19);
    contentPane.add(separator_1);
    JButton clearbtn = new JButton("Clear");
    clearbtn.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e) {
            edtUsername.setText("");
            edtPassword.setText("");
            usertype.setSelectedIndex(0);

        }
    });
    clearbtn.setFont(new Font("Tahoma", Font.PLAIN, 16));
    clearbtn.setBounds(93, 263, 102, 29);
    clearbtn.setBorder(null);
    contentPane.add(clearbtn);
    JButton Loginbtn = new JButton("Login");
    Loginbtn.addActionListener(new ActionListener()
    {                   public void actionPerformed(ActionEvent e)
        {
            try
            {
    Class.forName("com.mysql.cj.jdbc.Driver");
Connection con =
  riverManager.getConnection("jdbc:mysql://localhost:3306/instapark","root","Pass@
  123");
  System.out.print("connection established");
  String username = edtUsername.getText();
  String userTest = username;
  String Password = edtPassword.getText();
  String Usertype = usertype.getSelectedItem().toString();
  System.out.println();
  System.out.print(Usertype);
  Statement stm = con.createStatement();
```

```
   String sql = "select * from register where username = '"+username+"'and Password
   = '"+Password+"'";
   ResultSet rs = stm.executeQuery(sql);
    if(username.equals("")||Password.equals("")||Usertype.equals(""))
    {
    JOptionPane.showMessageDialog(Loginbtn,"wrong credentials please try again");
    edtUsername.setText("");
    edtPassword.setText("");
   usertype.setSelectedIndex(0);
    }else{

if(rs.next())
   {
  String s1 = rs.getString("UserType");

    if(Usertype.equalsIgnoreCase("USER") && s1.equalsIgnoreCase("USER"))
            {
    Dashboard D = new Dashboard();
    D.setVisible(true);
    setVisible(false);
   }
    if(Usertype.equalsIgnoreCase("ADMIN")&& s1.equalsIgnoreCase("ADMIN"))
    {
    Admin_Dashboard AD = new Admin_Dashboard();
    AD.setVisible(true);
    setVisible(false);                                                    }

    if(Usertype.equalsIgnoreCase("PARKING MANAGER")&&
   s1.equalsIgnoreCase("PARKING MANAGER"))
   {
    ParkingManager_Dashboard PD = new ParkingManager_Dashboard();
            PD.setVisible(true);
    setVisible(false);                                                           }}
   else
   {
    JOptionPane.showMessageDialog(rootPane,"entered credentials wrong please try
   again","Error",1);
   }con.close();}}
    catch(Exception e1) {
            System.out.print(e1.getMessage());
            }
        }
    });
    Loginbtn.setFont(new Font("Segoe UI", Font.PLAIN, 16));
    Loginbtn.setBounds(272, 262, 115, 29);
    contentPane.add(Loginbtn);

    JLabel lblNewLabel_3 = new JLabel("User type");
    lblNewLabel_3.setFont(new Font("Segoe UI", Font.PLAIN, 16));
    lblNewLabel_3.setBounds(65, 193, 85, 22);
    contentPane.add(lblNewLabel_3);

    JLabel lblNewLabel_4 = new JLabel("dont have an account here signin");
    lblNewLabel_4.setForeground(new Color(0, 0, 255));
    lblNewLabel_4.setFont(new Font("Segoe UI", Font.PLAIN, 14));
    lblNewLabel_4.setBounds(41, 339, 222, 22);
    contentPane.add(lblNewLabel_4);

    JButton signinbtn = new JButton("signin");
    signinbtn.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            if(e.getSource()==signinbtn) {
                signup s = new signup();
                s.setVisible(true);
                setVisible(false);
            }
        }
    }
);
    signinbtn.setFont(new Font("Segoe UI", Font.PLAIN, 14));
    signinbtn.setBounds(272, 339, 86, 22);
    contentPane.add(signinbtn);
   }
}
```