1. Abstract

Vehicle Parking Management System maintains a good record of vehicles check in and checkout time. Both two wheeler & four wheeler can be managed by this system and have different pricing system. Vehicle parking management system that enables the time management and control of vehicles by using parking number. The system that will track the entry and exit of vehicles, maintain a listing of vehicle within the parking lot, and determine the parking and it will also determine the cost of parking of vehicle.

2. INTRODUCTION

2.1 Motivation

Now days in many public places such as malls, multiplex system, hospitals, offices, market areas there is a crucial problem of vehicle parking. The vehicle parking area hasmany lanes/slots for car parking. So to park a vehicle one has to look for all the lanes.Moreover this involves a lot of manual labour and investment. Instead of vehicle caughtin towing the vehicle can park on safe and security with low cost.

2.2 Problem Statement

Parking control system has been generated in such a way that it is filled with manysecure devices such as, parking control gates, toll gates, time and attendance machine,car counting system etc. These features are hereby very necessary nowadays to secureyour car and also to evaluate the fee structure for every vehicles entry and exit

2.3 Purpose/Objective and Goals

We can park our vehicle in our own slot by paying

- Because of that there is no towing problems,
- And our vehicle has been parked as a secure condition.
- There is no risk for vehicle owner for parking the car.
- In case of any damages and problem of vehicle that will claim by parking management
- As the world is facing many threads daily, robberies are done easily with no track to trace, bomb blasts occur with the use of vehicle, so if a proper system isadopted each and every record can be saved and anyone can be track easily therefore mainly is to make a better and fast software, most important user-friendly
- Maintain records in short time of period.
- Determines the parking area is full or not.
- Enhances the visitor's experience

.

2.4 Literature survey

Vehicle Parking Management System has become a necessity in order to create simple and accessible way to support today system. The internet has dramatically changed the role of

Internet today (Cassidy 2002:1). Internet is the tool or vehicle for many applications, as well as to maintain registration for government, companies, and many events. This is happened as result of the simplicity of internet access in many parts of the world.

An Vehicle Parking Management System is systems that maintained the registration flow for the user and provide extensive capability for the administrator to maintain the content, report, and ability to add, update, or delete the content of a system. Currently there are many applications that have the ability to manage registration online. Some of them are very simple, and more complicated that use current technology. Almost all web-based programming language support the capability to make online registration, such as PHP or .NET provide many options to build intelligence course registration and management system. A good system must be able to provide sufficient information and services needed by user as well as delivering extensive report to the administrator

2.5 Project scope and limitations

Scope:

In the modern age. Many people have vehicles. Vehicle is now a basic need. Everyplace is under the process of urbanization. There are many corporate offices and shopping centres etc. There are many recreational places where people used to go for refreshment. So, all these places need a parking space where people can park their vehicles safely and easily. Every parking area needs a system that records the detail of vehicles to give the facility. These systems might be computerized or non-computerized. With the help of computerized system we can deliver a good service to customer who
wants to park their vehicle into the any organization's premises.
Vehicle parking management system is an automatic system which delivers data processing in very high speed in systematic manner. Parking is a growing need of the time. Development of this system is very useful in this area of field. We can sell this system to any organization. By using our system they can maintain records very easily. Our system covers the every area of parking management. In coming future there will be excessive need of Vehicle parking management system

Limitations:

- Now a days in parking like valet parking they maintain just with the tokens and they have records the vehicle details in books so that during some critical situations like police enquiry of terrorist car or vehicle referrer that case it is difficult to find the details of particular vehicle but in this case is easy to find in 1to 2 seconds

- By parking the vehicle in public place the vehicle can be claimed by towing person but in this case there is no towing problems and no need to give fine for anything we can park our vehicle with securely

.

3. SYSTEM ANALYSIS

3.1 Existing System

An old vehicle parking management system typically includes physical equipment such as parking meters, pay stations, and gate barriers, as well as an administrative system for collecting and managing payment and enforcement data. These systems may not have the capabilities of modern systems such as real-time occupancy tracking, dynamic pricing, and mobile payment options. They also may not be integrated with other smart city infrastructure or technologies.

3.2 Scope of Existing System

The scope of an old vehicle parking management system would typically include the following:

1. Providing a way for drivers to pay for parking at designated spots, such as through the use of parking meters or pay stations.
2. Enforcing parking regulations, such as time limits and parking restrictions, through the use of parking enforcement officers and ticketing systems.
3. Collecting and managing data on parking usage, such as occupancy rates and revenue, through the use of administrative tools and reporting systems.
4. Maintaining and repairing parking equipment and infrastructure, such as replacing broken meters and repainting parking spaces.
5. Providing customer service and assistance to drivers, such as helping with parking-related questions and addressing complaints.

It might not have the capability of real-time occupancy tracking, dynamic pricing, and mobile payment options. Also, it may not be integrated with other smart city infrastructure or technologies

.

3.3 Project Perspective and Features

an online vehicle parking management system could have the following features:

1. Real-time occupancy tracking: The system would be able to provide real-time information on the number of available parking spaces in a given area, as well as the location of available spots.
2. Mobile payment options: Drivers would be able to pay for parking through their mobile devices, using options such as credit/debit cards, digital wallets, and parking apps.

3. Dynamic pricing: The system would be able to adjust parking prices based on demand, time of day, and other factors.
4. Automatic enforcement: The system would be able to automatically issue parking violations and fines based on data collected from sensors and cameras.
5. Analytics and reporting: The system would be able to provide detailed data and insights on parking usage, occupancy rates, revenue, and other metrics.
6. Integration with other systems: The system would be able to integrate with other smart city infrastructure and technologies, such as traffic management systems, public transportation systems, and navigation apps.

3.4 Stakeholder

he stakeholders of an online vehicle parking management system would include:

1. Drivers: The primary users of the system, who would be able to pay for parking, locate available spots, and receive parking violations and fines through the system.
2. Parking operators: The individuals or organizations responsible for managing and maintaining the parking facilities, who would be able to manage and enforce parking regulations, collect and manage revenue, and access analytics and reporting through the system.
3. City/municipal government: The government entity responsible for overseeing parking regulations and enforcement within the jurisdiction, who would be able to access data and insights on parking usage and revenue through the system.
4. Smart city/IT department: The department or organization responsible for managing and integrating the system with other smart city infrastructure and technologies, such as traffic management systems, public transportation systems, and navigation apps.
5. Technology vendors: The companies or organizations responsible for developing and maintaining the system, who would be able to provide customer service and technical support to the other stakeholders.

3.5 Requirement Analysis

For software requirement

- Admin panel, user panel
- Time amenities

- Impressibility
- Easily suitable for any device
- Login System for Admin panel, user panel

An admin can login directly the system but user cannot login directly. The User can firstly sign in with name, username and password, after sign in user can login using their username and password.

3.5.1 Function Requirements:

It deals with the functionalities required from the system which are follows:

- The website will help the colleges/organizations/companies to conduct
- their user registration.
- Only authorized person can access related details.
- Organizations can change their information regarding themselves.
- user can logins through USERNAME and PASSSWORD.
- Administrator will be response for updating the site.

3.5.2 Performance Requirements:

a) Speed and Latency Requirements:

The system is required a fair amount of speed especially while browsing through the catalogue and presenting different possibilities for the schedule. The outcomes of the product are not directly influenced by its speed, because all the operations are linked to each other and one operation cannot be computed before the one causing it.

b) Reliability and Availability Requirements:

The reliability of the system is directly linked to the level of update of the documents to which it is correlated, such as the catalogue or the students' database.

c) Robustness or Fault-Tolerance Requirements:

When the system is disconnected or frozen due to over access at the same time, it should save all the process of the users have made up to the point of abnormal happenings. When the users log in with the same username and password, all the work should be provided.

d) Capacity Requirements

The system should be able to manage all the information incoming from the database.

3.5.3 Security Requirements

a. Access Requirements:

Everyone (stakeholders and guests) can have access to the system. Every student must have secure and private access to his/her data. . All these accesses require identification through username and password.

b. Integrity Requirements:

Data integrity should be assured by limiting access to the database and by appropriate synchronization, and back-up functionalities.

4. System Design

4.1 Design Constraints

MySQL Data Tables:

Admin Table:(Table name is admin)

This store admin personal and login details.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(10)			No	*None*		AUTO_INCREMENT
2	AdminName	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
3	UserName	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
4	MobileNumber	bigint(10)			Yes	*NULL*		
5	Email	varchar(200)	latin1_swedish_ci		Yes	*NULL*		
6	Password	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
7	AdminRegdate	timestamp			Yes	current_timestamp()		

Category Table(Table name is tblcategory

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(10)			No	*None*		AUTO_INCREMENT
2	VehicleCat	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
3	CreationDate	timestamp			Yes	current_timestamp()		

Vehicle Table: (Table name is vehicle)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(10)			No	*None*		AUTO_INCREMENT
2	ParkingNumber	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
3	VehicleCategory	varchar(120)	latin1_swedish_ci		No	*None*		
4	VehicleCompanyname	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
5	RegistrationNumber	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
6	OwnerName	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
7	OwnerContactNumber	bigint(10)			Yes	*NULL*		
8	InTime	timestamp			Yes	current_timestamp()		
9	OutTime	timestamp			Yes	*NULL*		ON UPDATE CURRENT_TIMESTAMP()
10	ParkingCharge	varchar(120)	latin1_swedish_ci		No	*None*		
11	Remark	mediumtext	latin1_swedish_ci		No			
12	Status	varchar(5)	latin1_swedish_ci		No	*None*		

This store the vehicle details and admin remark

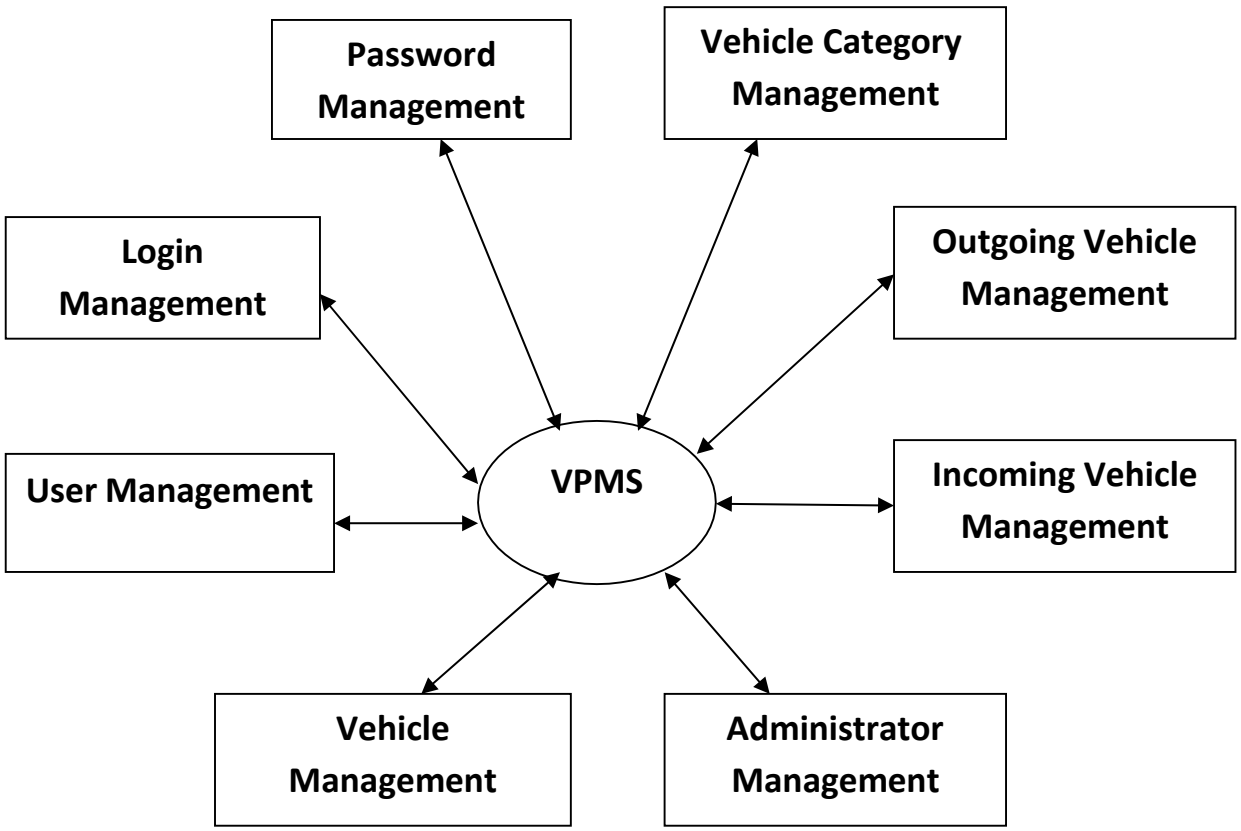
User Table: (Table name is tblregusers)

This table stores the details of registered users

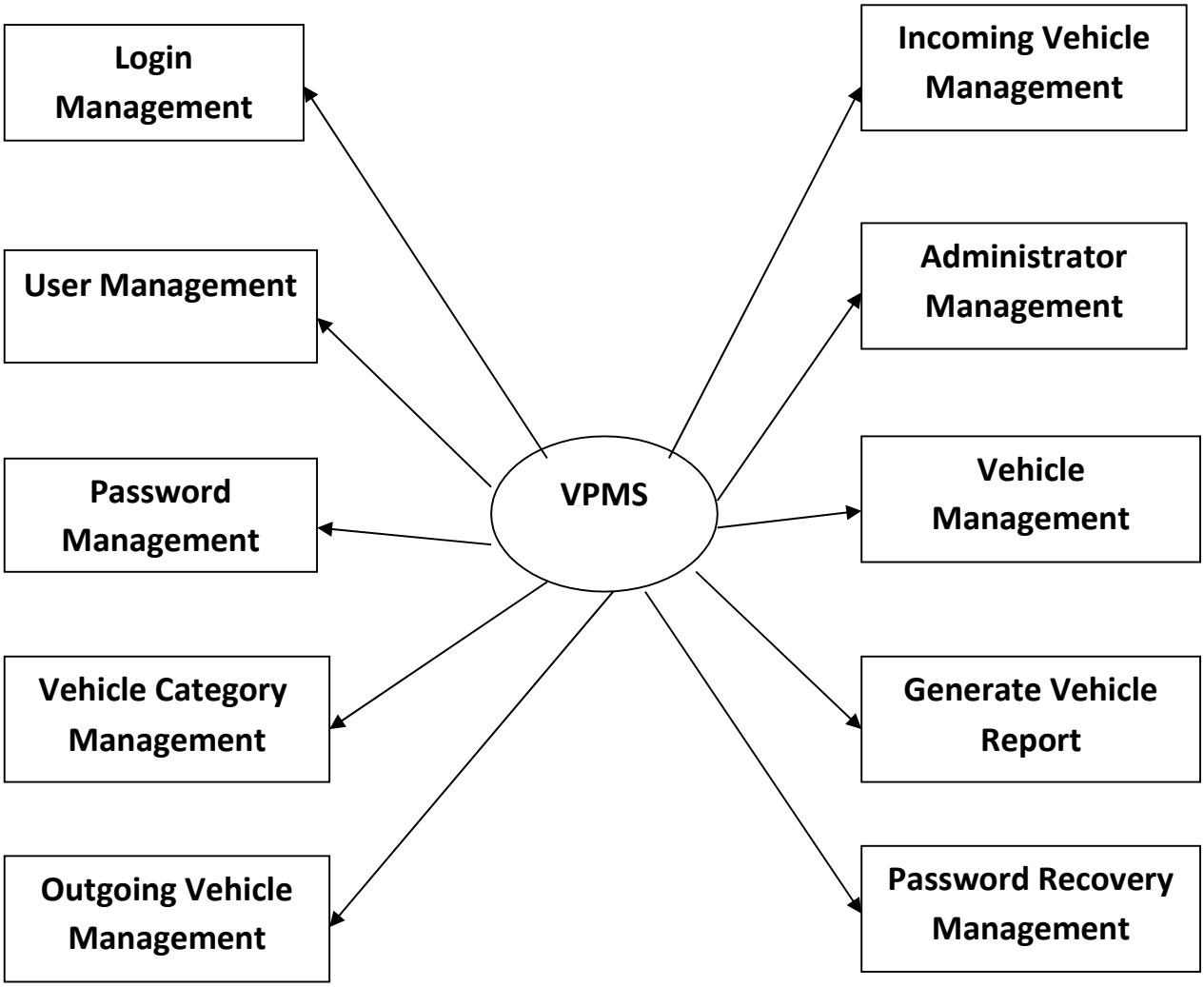
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(5)			No	*None*		AUTO_INCREMENT
2	FirstName	varchar(250)	latin1_swedish_ci		Yes	*NULL*		
3	LastName	varchar(250)	latin1_swedish_ci		Yes	*NULL*		
4	MobileNumber	bigint(10)			Yes	*NULL*		
5	Email	varchar(250)	latin1_swedish_ci		Yes	*NULL*		
6	Password	varchar(250)	latin1_swedish_ci		Yes	*NULL*		
7	RegDate	timestamp			Yes	current_timestamp()		

4.2 System Model: DFD (Data Flow Diagram)

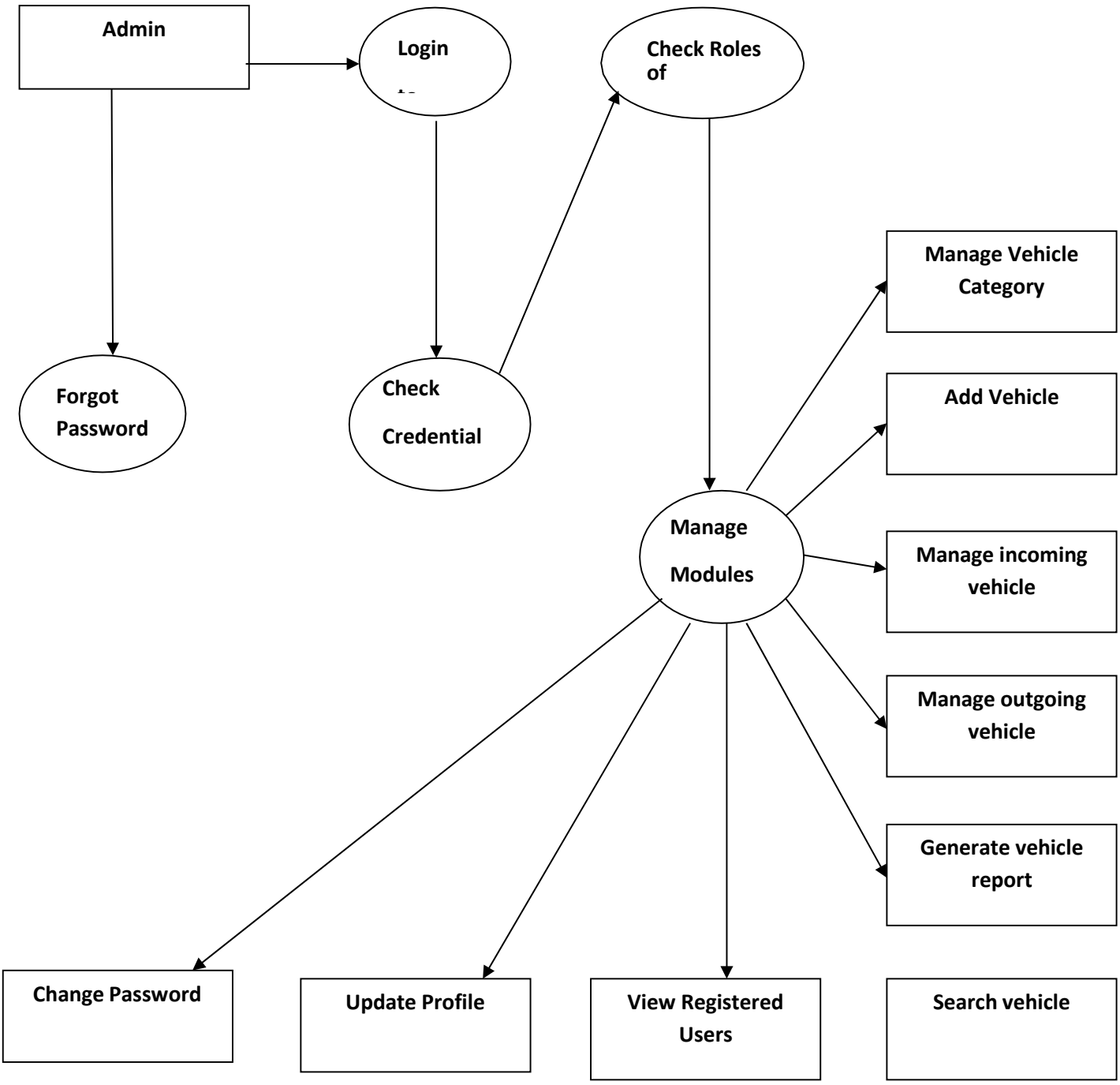
Zero Level DFD



First Level DFD

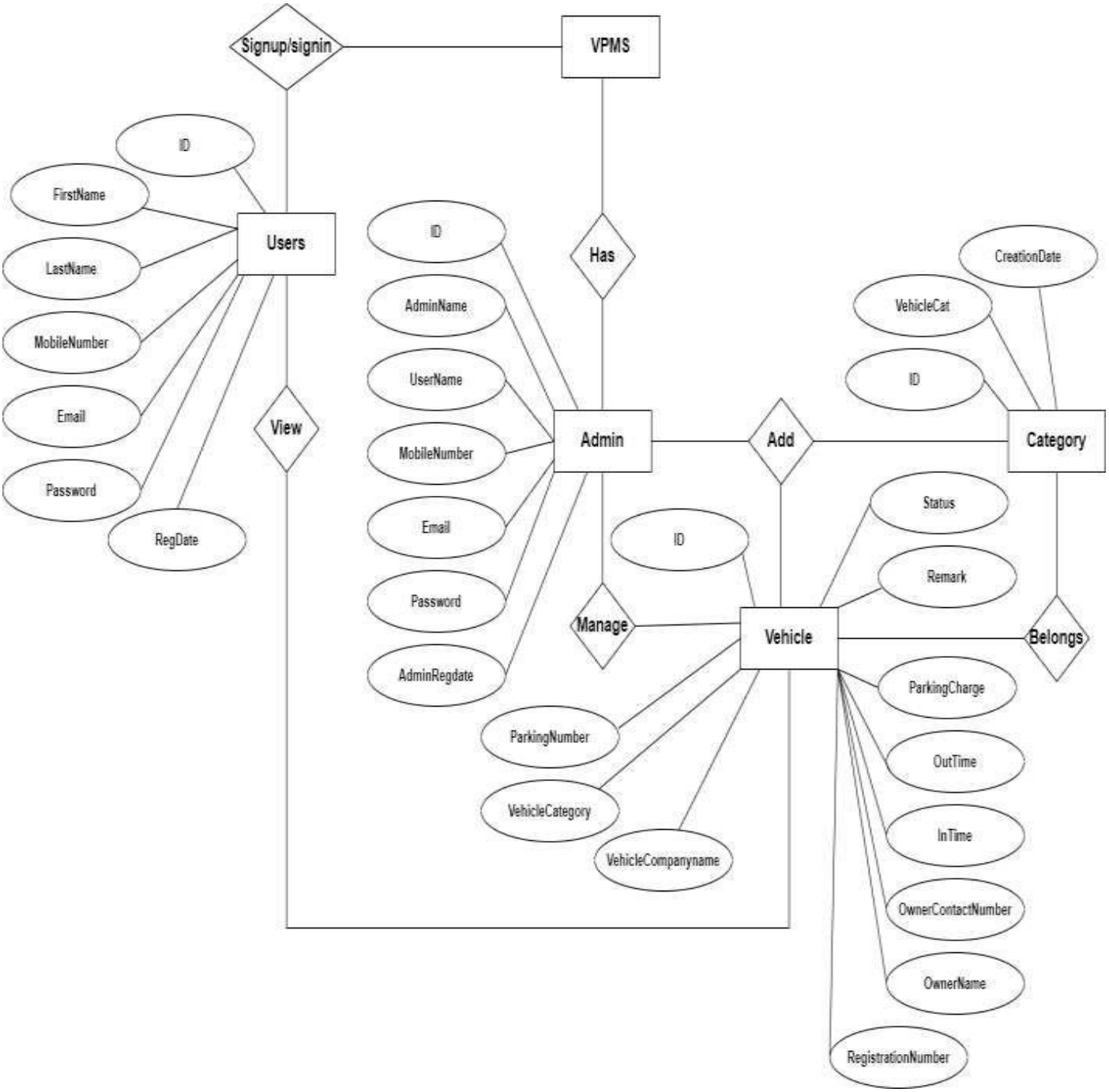


Second Level DFD



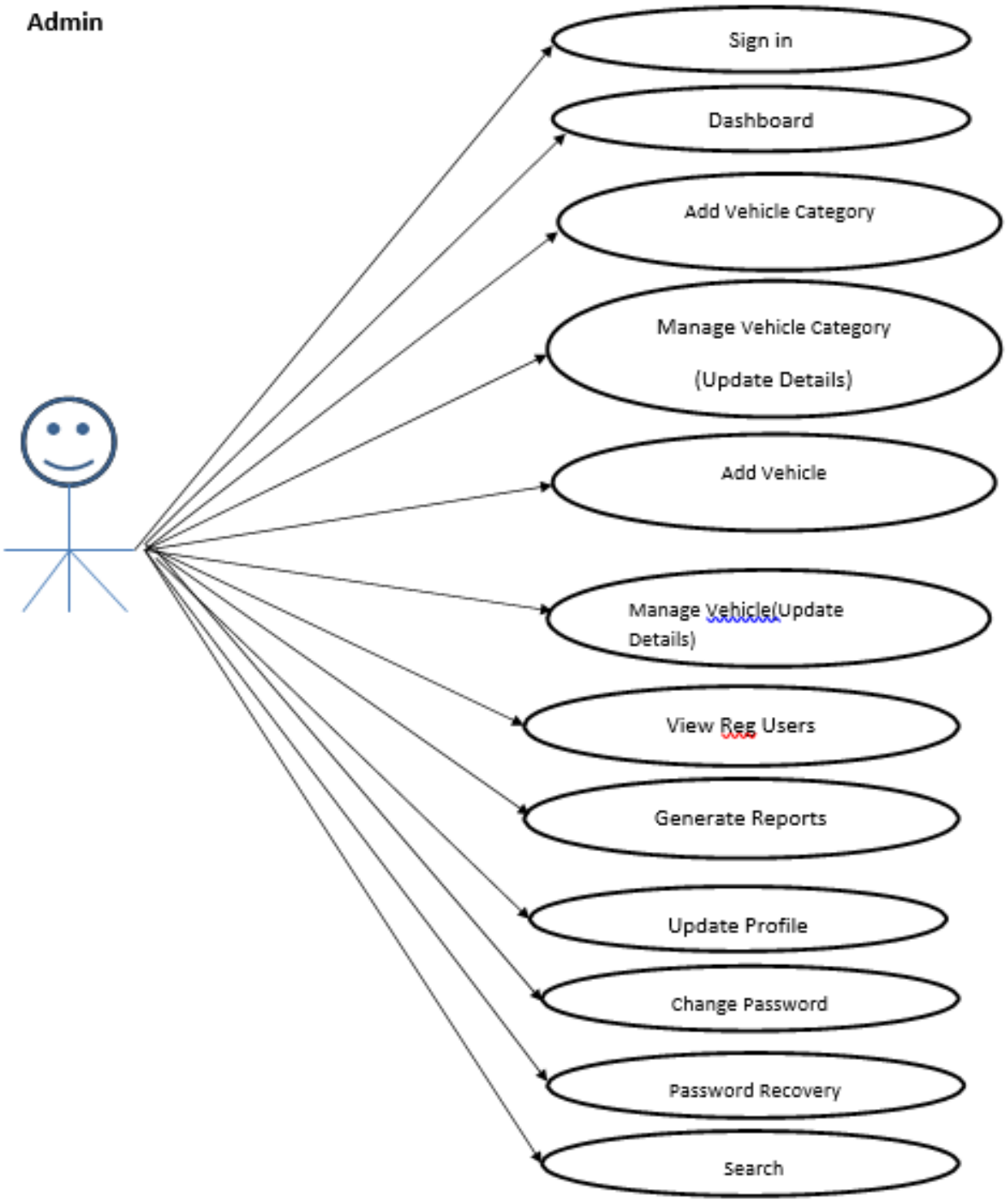
UML

Entity Relationship Diagram

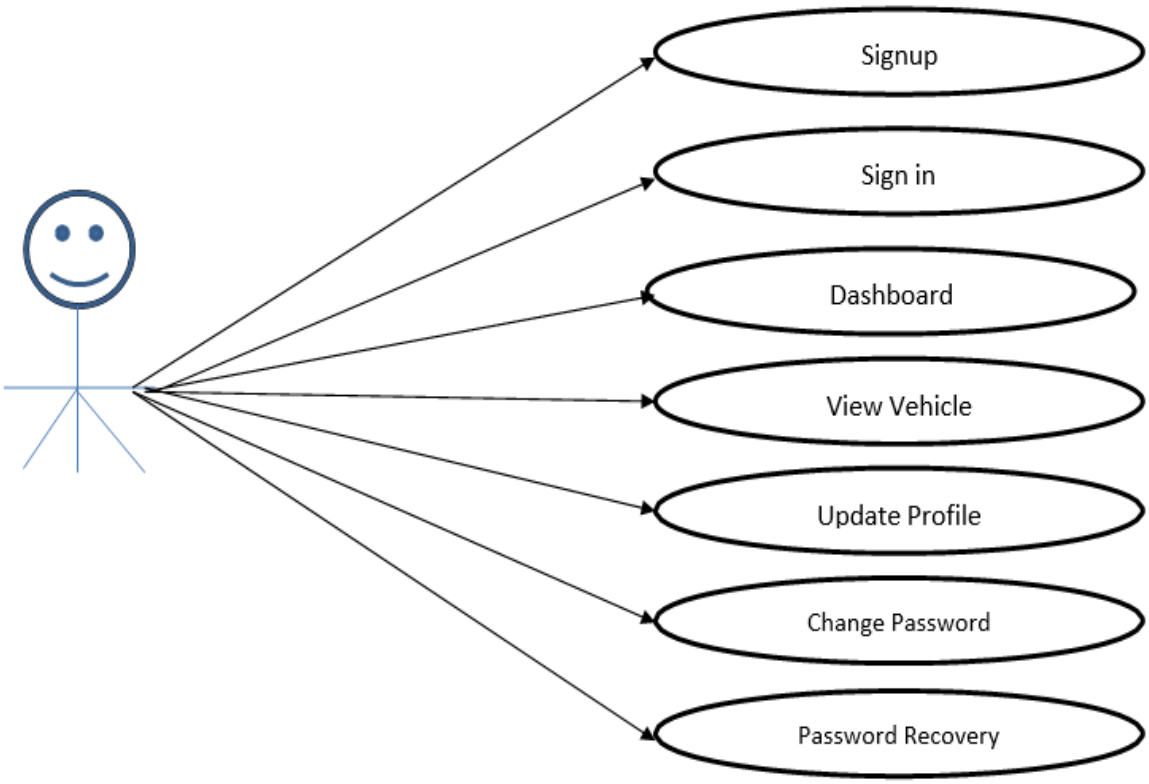


Use Case Diagrams:

Admin

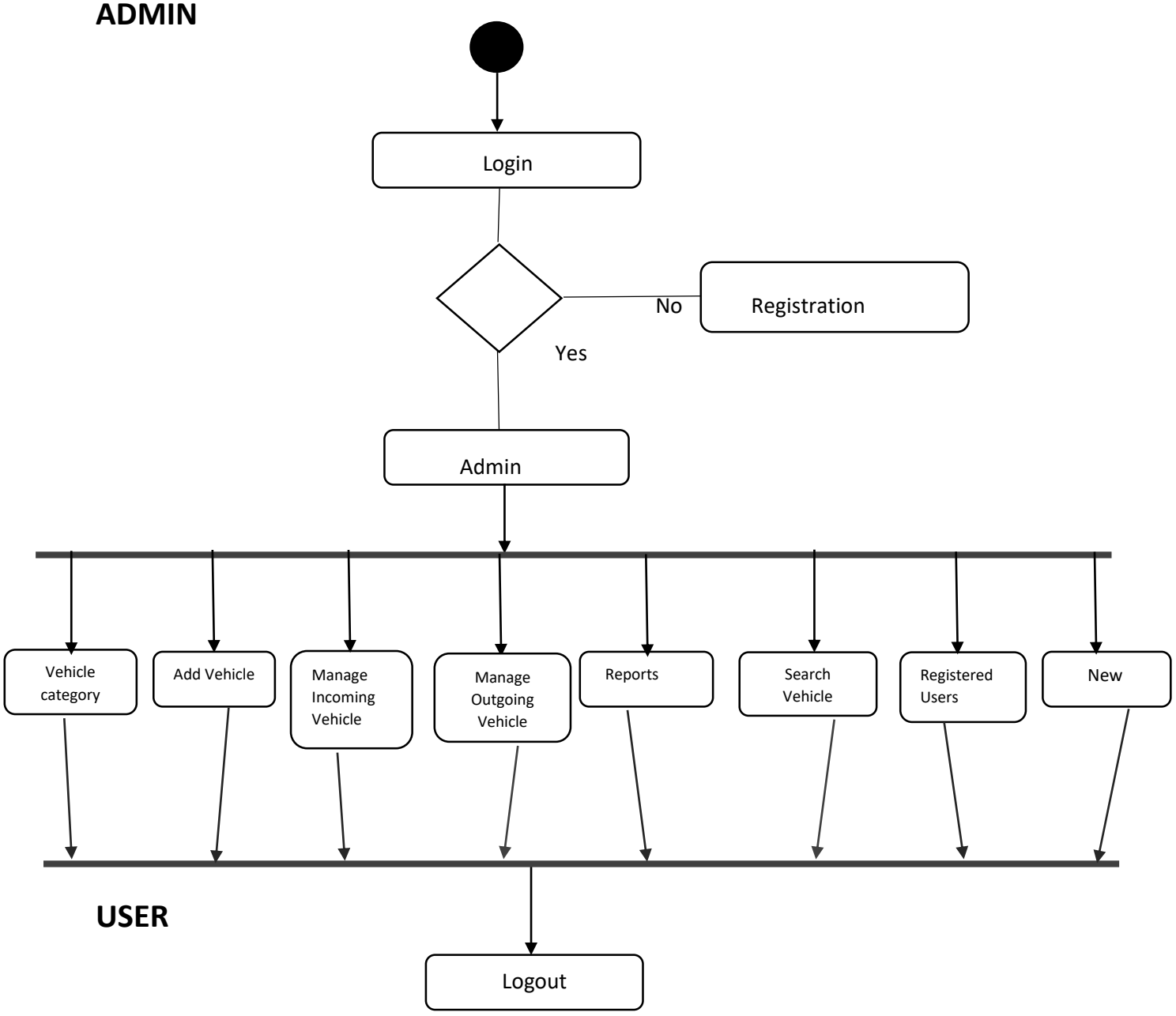


Users

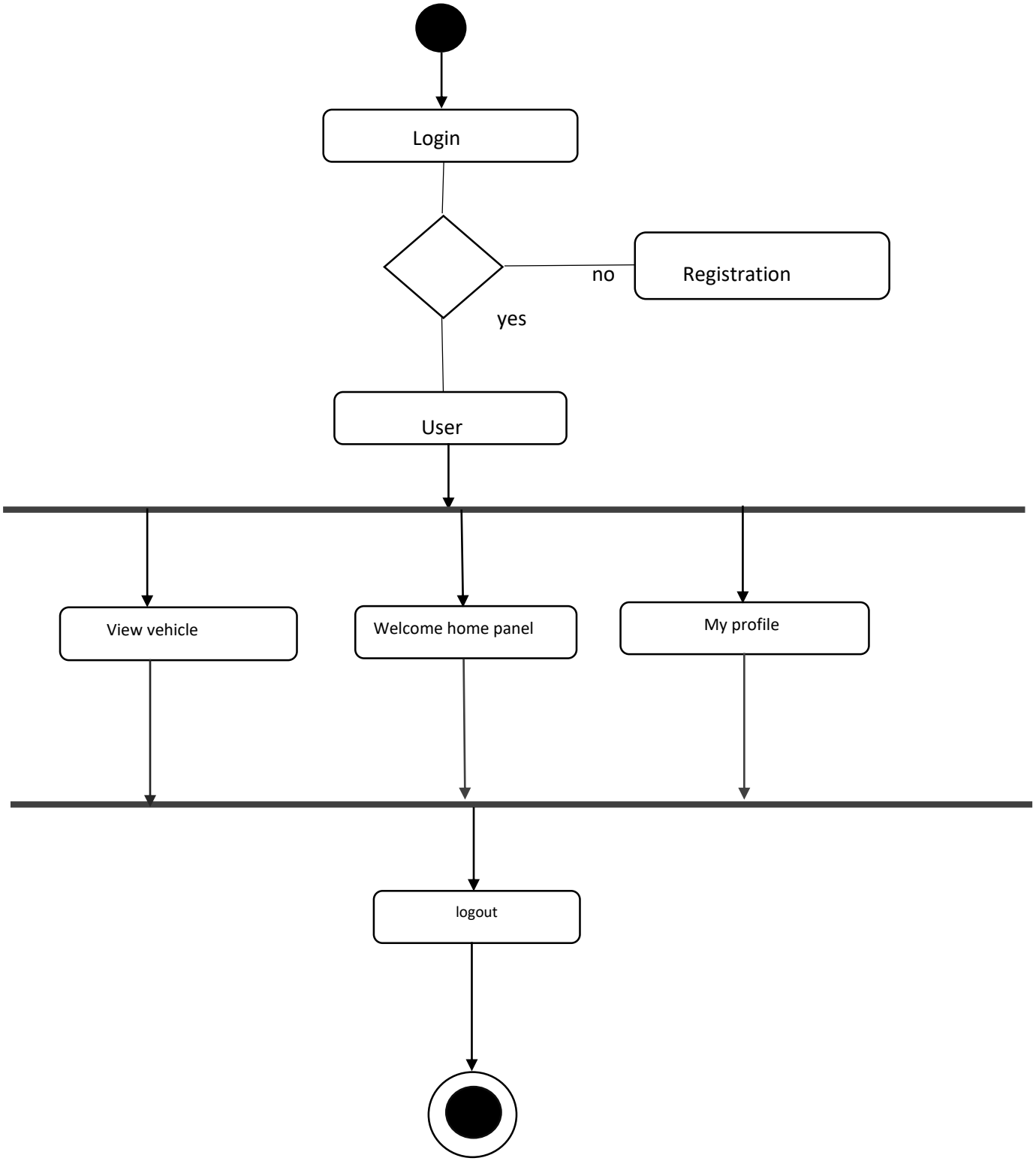

Signup
Sign in
Dashboard
View Vehicle
Update Profile
Change Password
Password Recovery

Activity Diagram

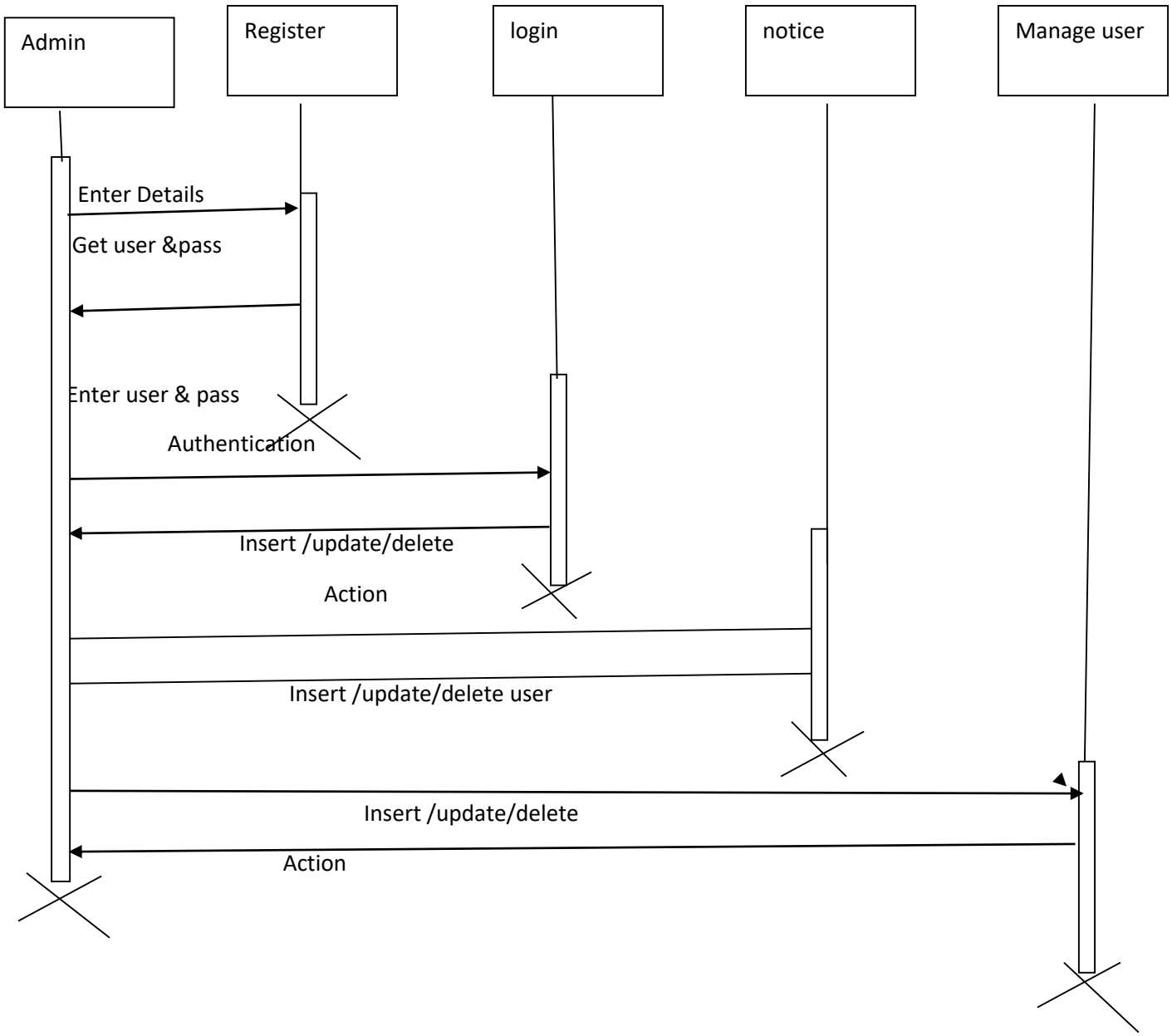
ADMIN



USER

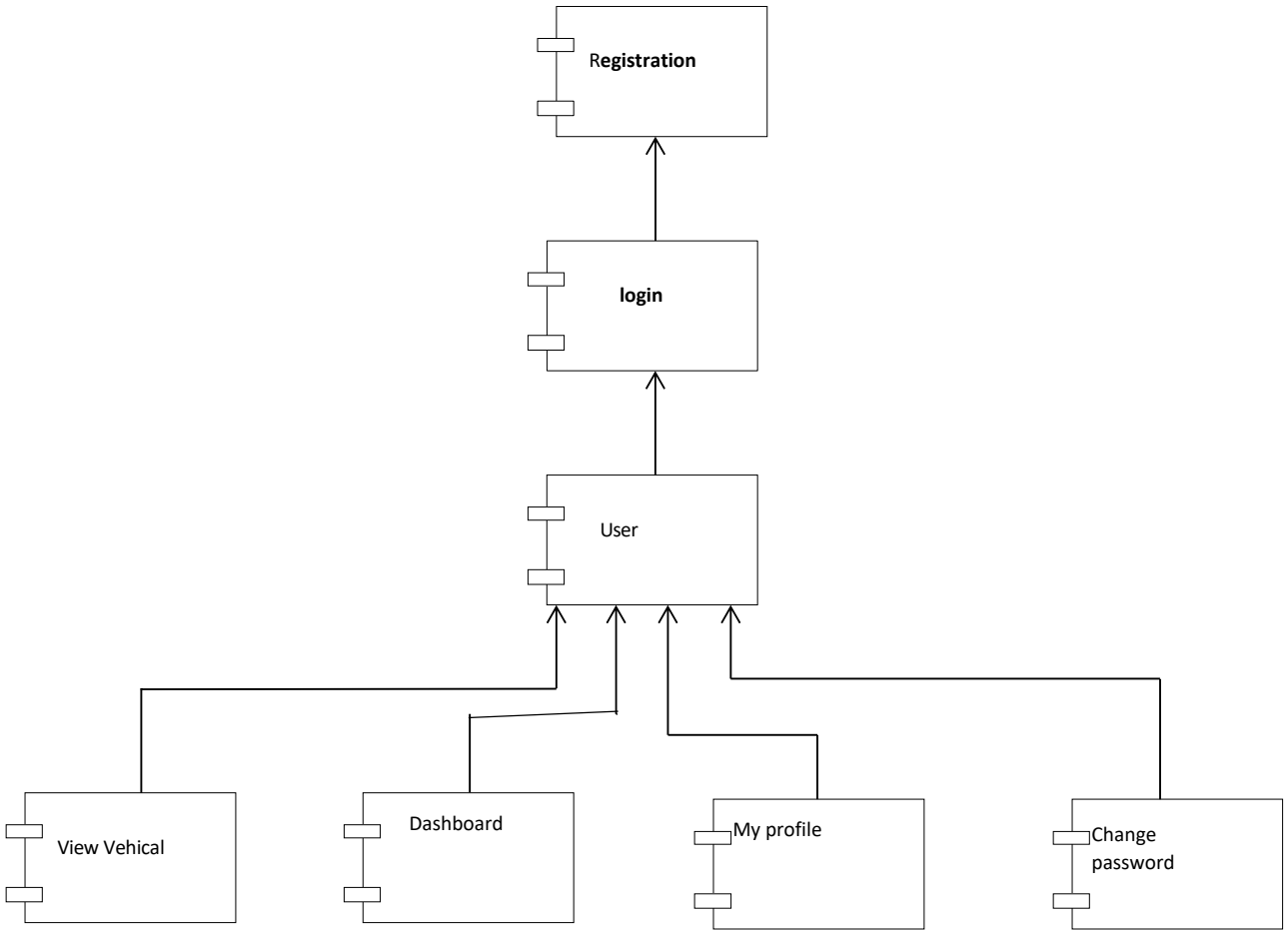

Login
no
Registration
yes
User
View vehicle
Welcome home panel
My profile
logout

Sequence diagram:

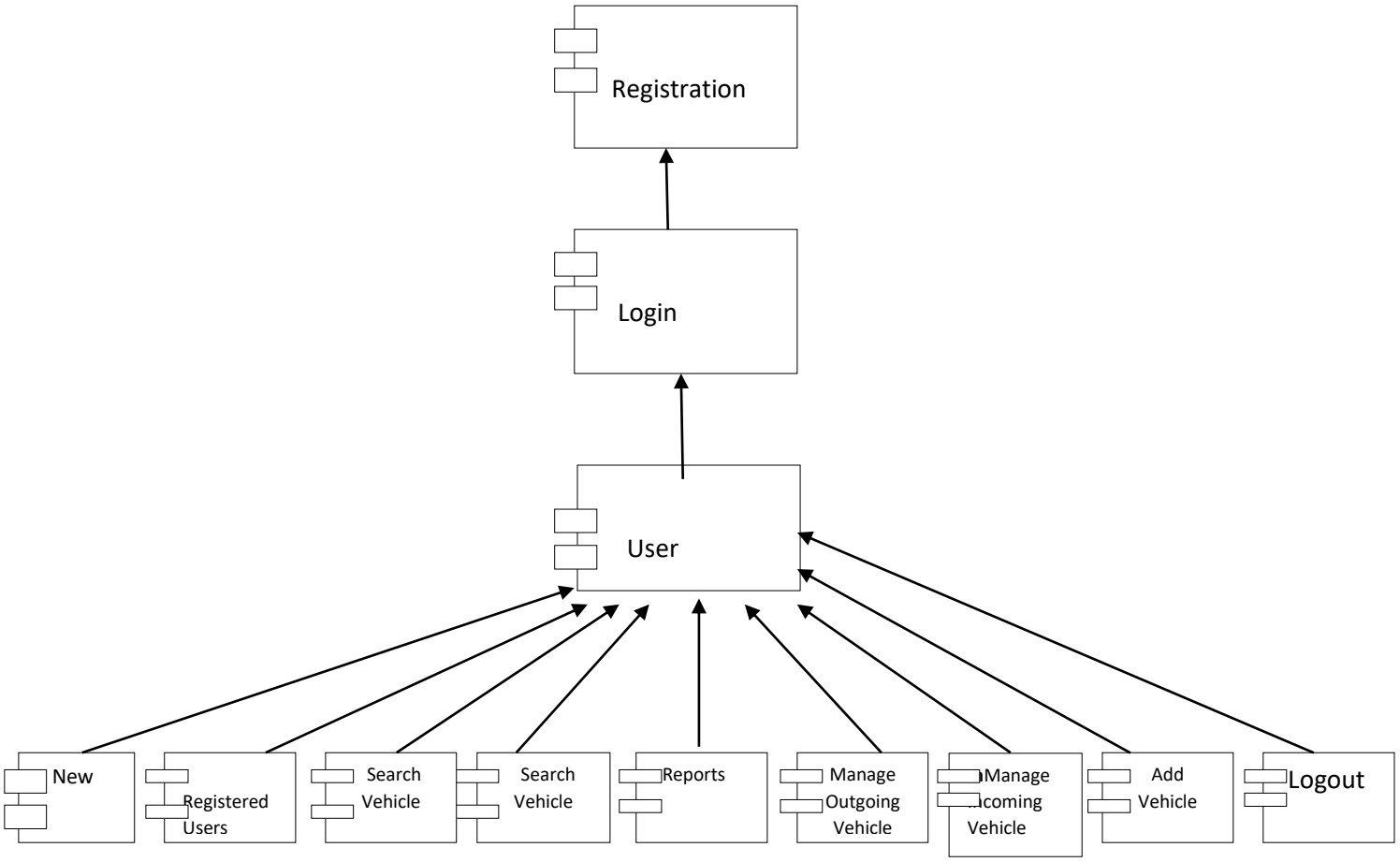


COMPONENT DIAGRAM

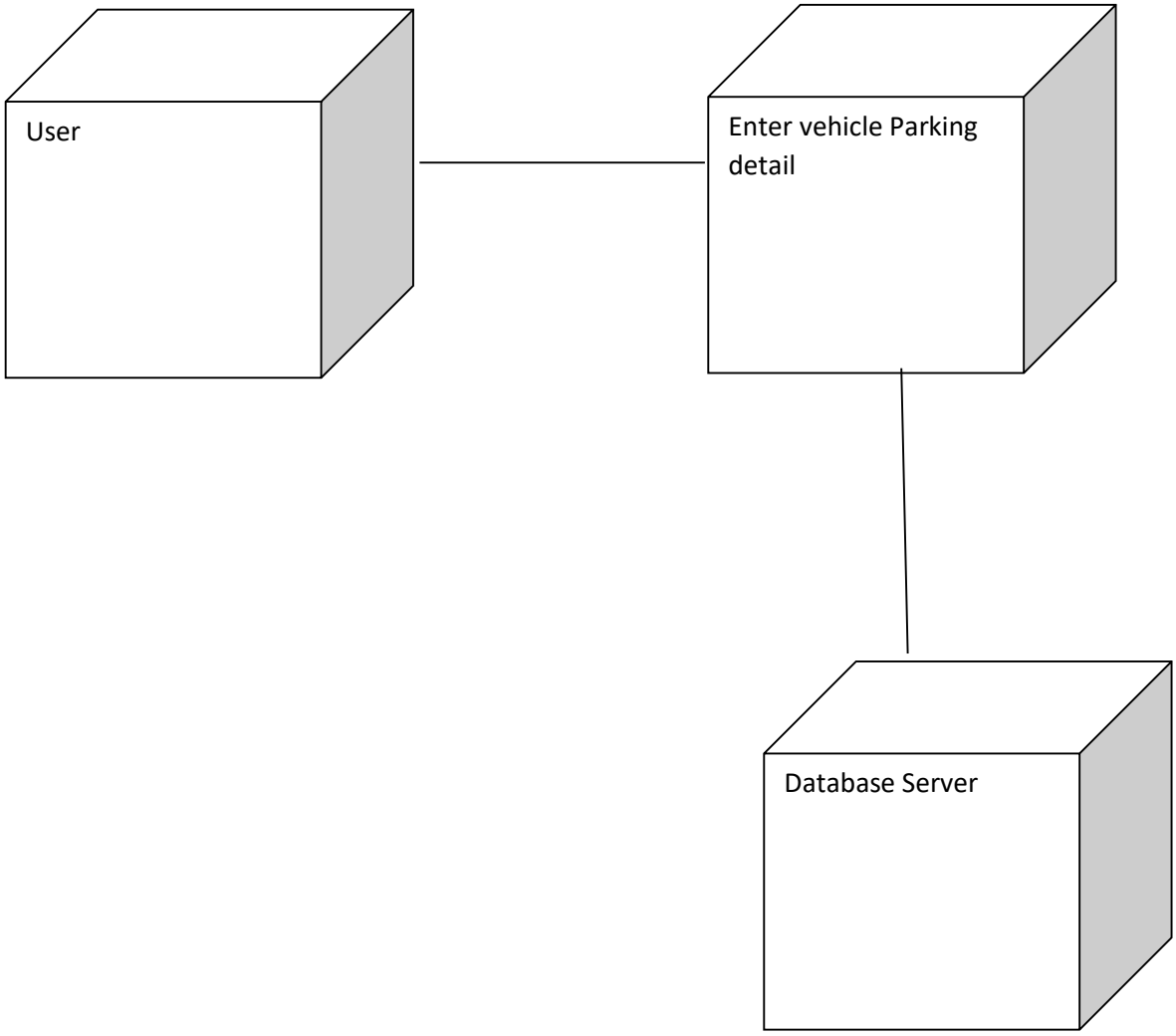
- **USER**



ADMIN


Registration
Login
User
New
Registered Users
Search Vehicle
Search Vehicle
Reports
Manage Outgoing Vehicle
Manage Vehicle
Add Vehicle
Logout

DEPLOYMENT DIAGRAM



4.4 User Interface

The user interface is a very important part of the system. This helps every member of the system to interact with each detail properly. As we have shown several use cases in this system. We have developed these interfaces to interact with the system.

A GUI (graphical user interface) is a system of interactive visual components for computer software. A GUI displays objects that convey information, and represent actions that can be taken by the user. The objects change color, size, or visibility when the user interacts with them.

5.IMPLEMENTATIONS DETAILS

This section describes the different technologies used for the entire development process of the Front-end as well as the Back-end development of the application.

5.1 SOFTWARE AND HARDWARE SPECIFICATIONS

SOFTWARE REQUIREMENT

- operating system - Windows XP/Windows
- Language - PHP
- Server-side Script - JavaScript, Html
- Server - XAMP
- Database - MYSQL
- Browser - Google Chrome

HARDWARE REQUIREMENTS

- Device name - LAPTOP-RSS7D5AA
- Processor - AMD Ryzen 3 3250U with Radeon Graphics 2.60 GHz
- Installed RAM - 4.00 GB (3.45 GB usable)
- System type - 64-bit operating system, x64-based processor

6: REPORT TESTING

6.1: Black-Box Testing

Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings.

This method of test can be applied virtually to every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher-level testing, but can also dominate unit testing as well.

6.2 Test Procedures

Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. The tester is aware of what the software is supposed to do but is not aware of how it does it. For instance, the tester is aware that a particular input returns a certain, invariable output but is not aware of how the software produces the output in the first place.

6.1.3 Test Cases

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output, often with the help of an oracle or a previous result that is known to be good, without any knowledge of the test object's internal structure.

5.2: White-Box Testing

White-box testing (also known as clear box testing, glass box testing, transparent box testing, and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (i.e., black-box testing). In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g., in-circuit testing (ICT). White-box testing can be applied at the unit, integration and system levels of the software testing process. Although traditional testers tended to think of white-box testing as being done at the unit level, it is used for integration and system testing more frequently today. It can test paths within a unit,

paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it has the potential to miss unimplemented parts of the specification or missing requirements.

5.5.1 Levels

Unit Testing:

Each module is considered independently. It focuses on each unit of software as

implemented in the source code. It is white box testing.

Validation Testing:

Validation testing was per formed to ensure that all the functional and performance requirements are met.

Steps

1. Integration of all the modules/forms in the system.

2. Preparation of the possible test data with all the validation checks.

3. Preparation of the test cases.

4. Actual testing done manually.

5. Recoding of all the reproduced errors.

6. Modifications done for the errors found during testing.

Test Case-1

TEST NO: 1

TEST TYPE: Unit Testing

INPUT: Username and Password

OBJECTIVE: Checking Password Security

EXPECTED OUTPUT: Invalid Username or Password

ACTUAL OUTPUT: : Login successful

RESULT: Access to only Authorized Users

Test Case- 2

TEST NO: 2

TEST TYPE: Unit Testing

INPUT: Pincode

OBJECTIVE: Validating User

EXPECTED OUTPUT: Invalid Pincode

ACTUAL OUTPUT: : Pincode Verification Successfully

RESULT: Access to only Authorized Users

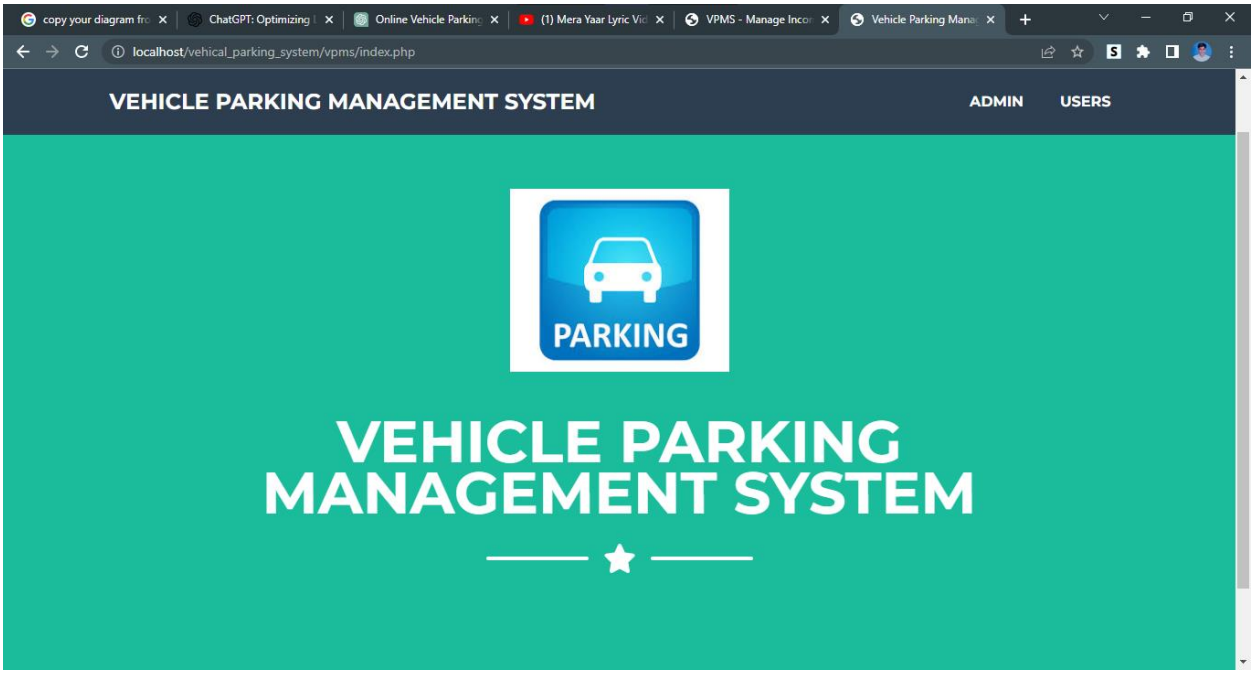
5.5.2 Procedures

White-box testing's basic procedures involves the tester having a deep level of understanding of the source code being tested. The programmer must have a deep understanding of the application to know what kinds of test cases to create so that every visible path is exercised for testing. Once the source code is understood then the source code can be analysed for test cases to be created. These are the three basic steps that white-box testing takes in order to create test cases:

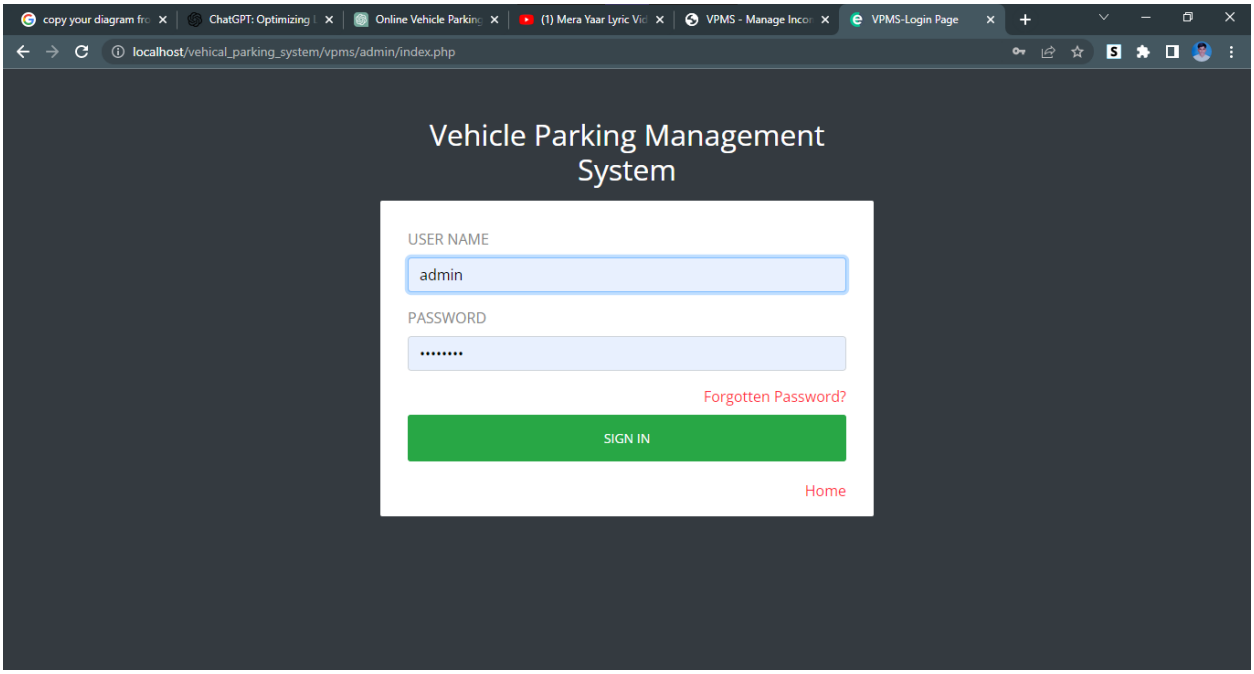
- Input involves different types of requirements, functional specifications, detailed designing of documents, proper source code, security specifications. This is the preparation stage of white-box testing to layout all of the basic information.
- Processing involves performing risk analysis to guide whole testing process, proper test plan, execute test cases and communicate results. This is the phase of building test cases to make sure they thoroughly test the application the given results are recorded accordingly.
- Output involves preparing final report that encompasses all of the above preparations and results.

6. Output Screen

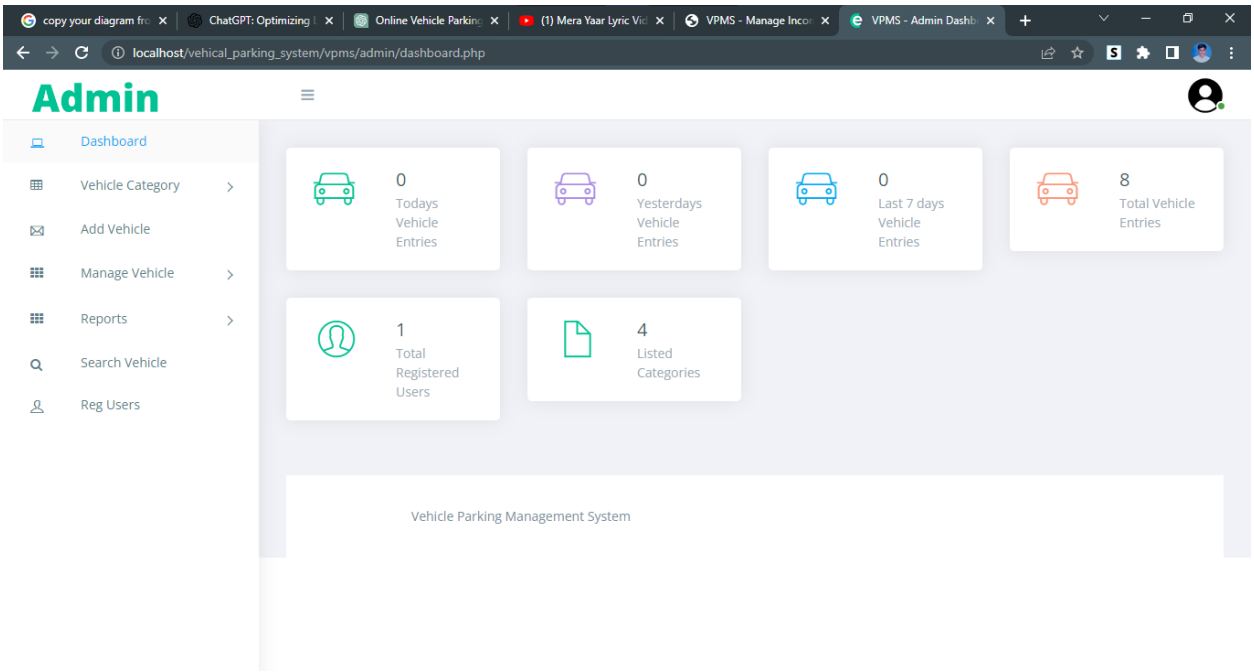
Home screen



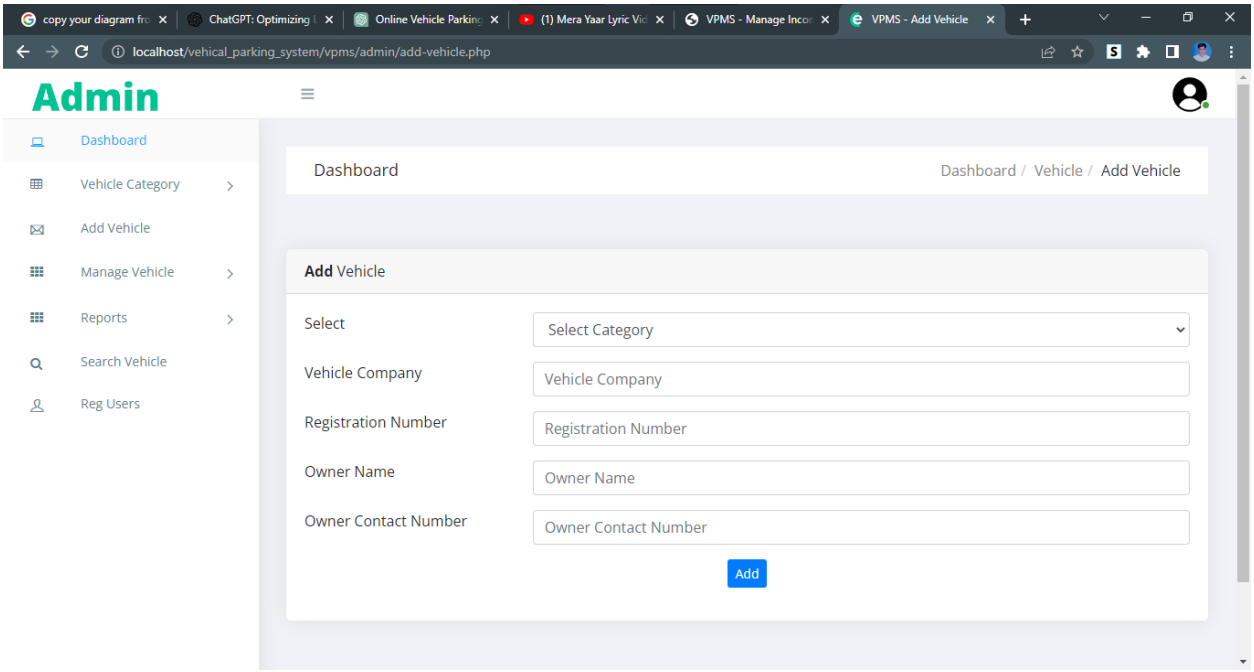
Admin Login Page



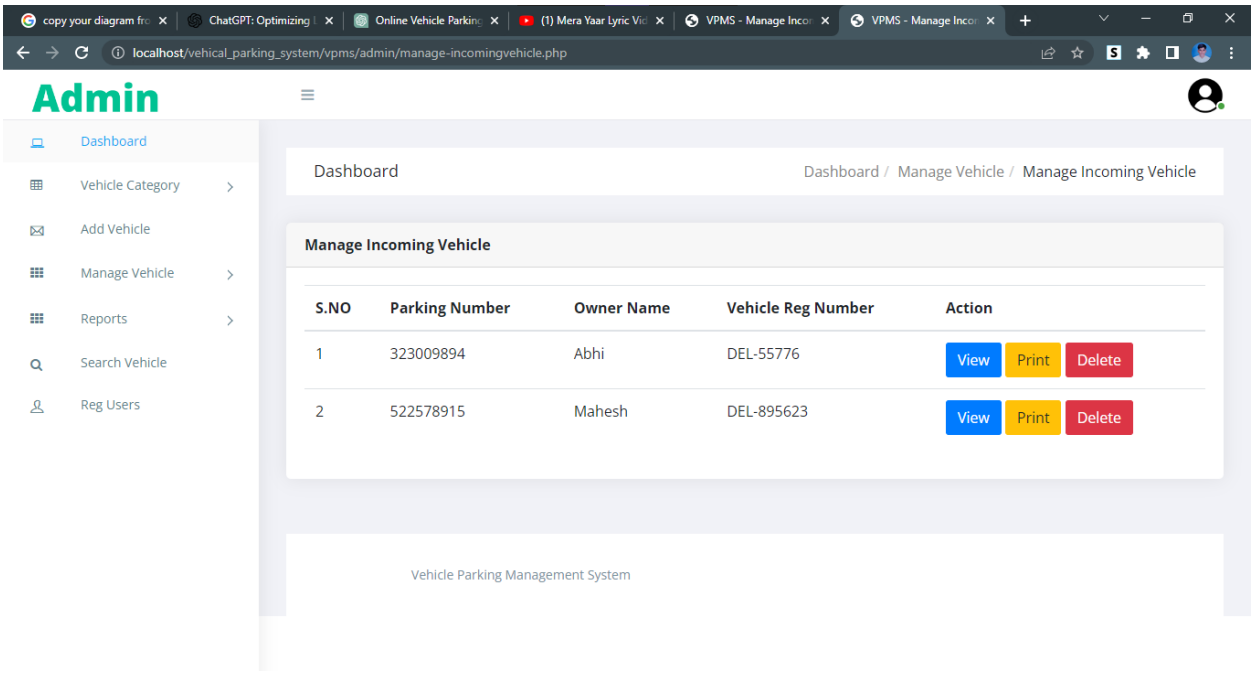
Admin home Screen



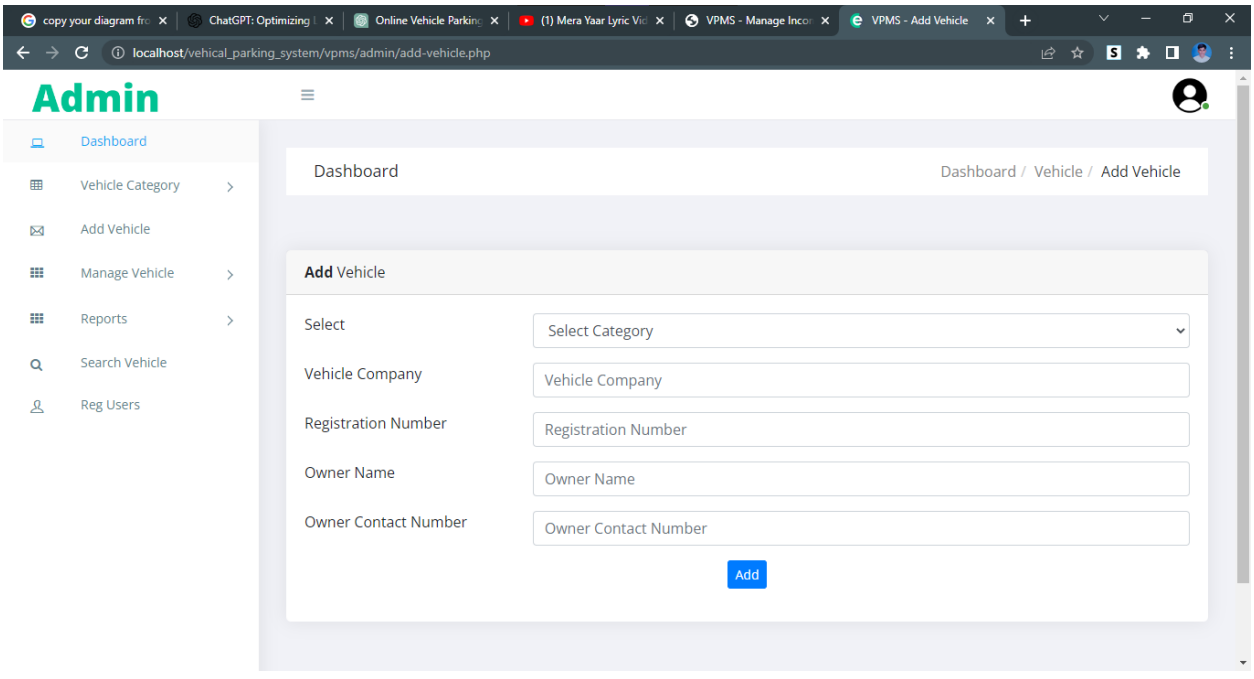
Add vehicle



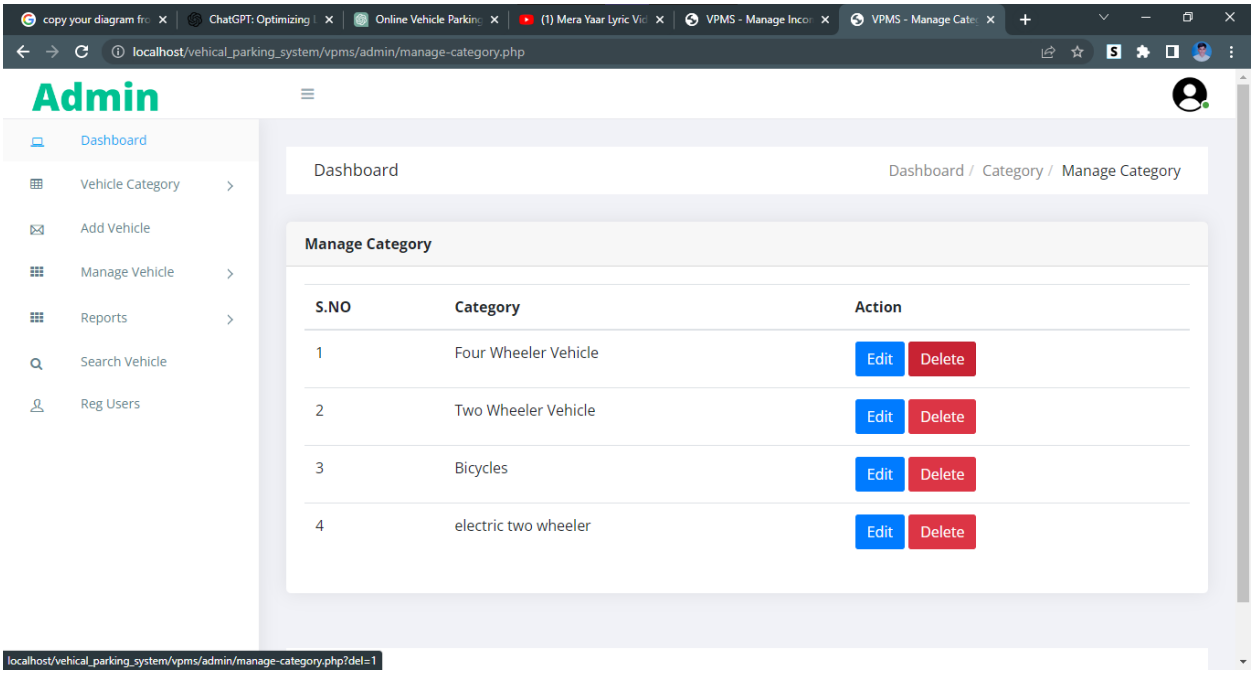
Manage Incoming / outgoing Vehicle



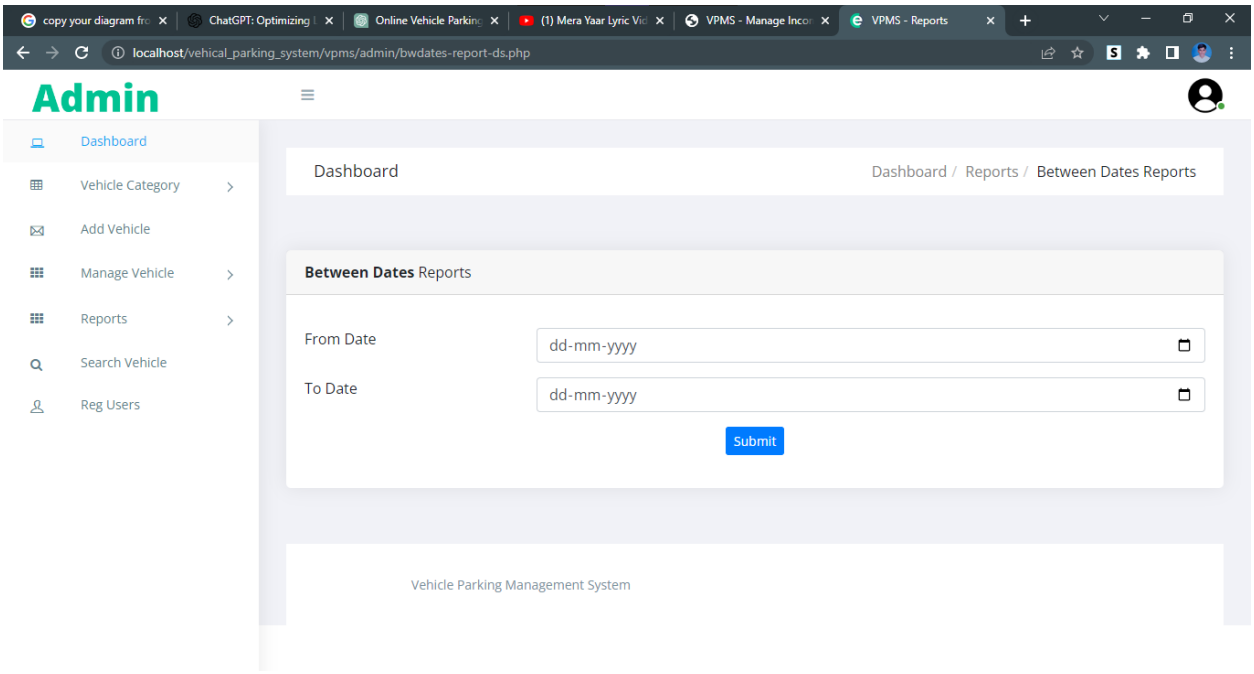
Add Vehicle Category



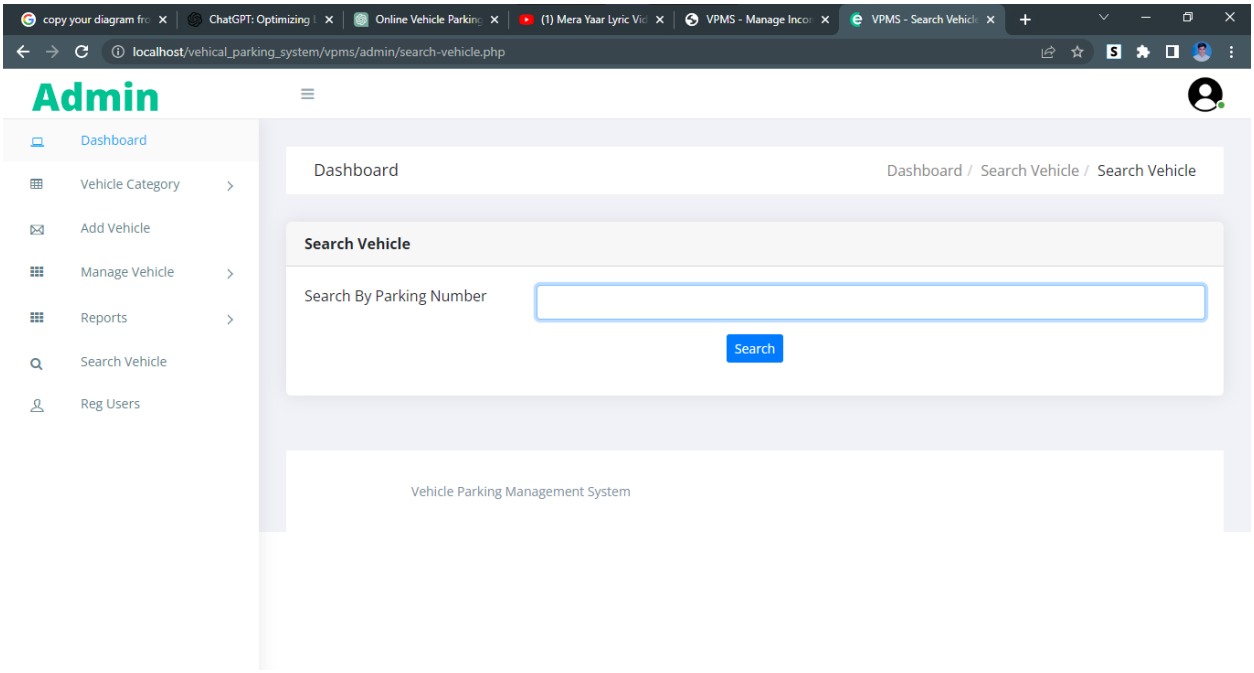
Manage Vehicle Category



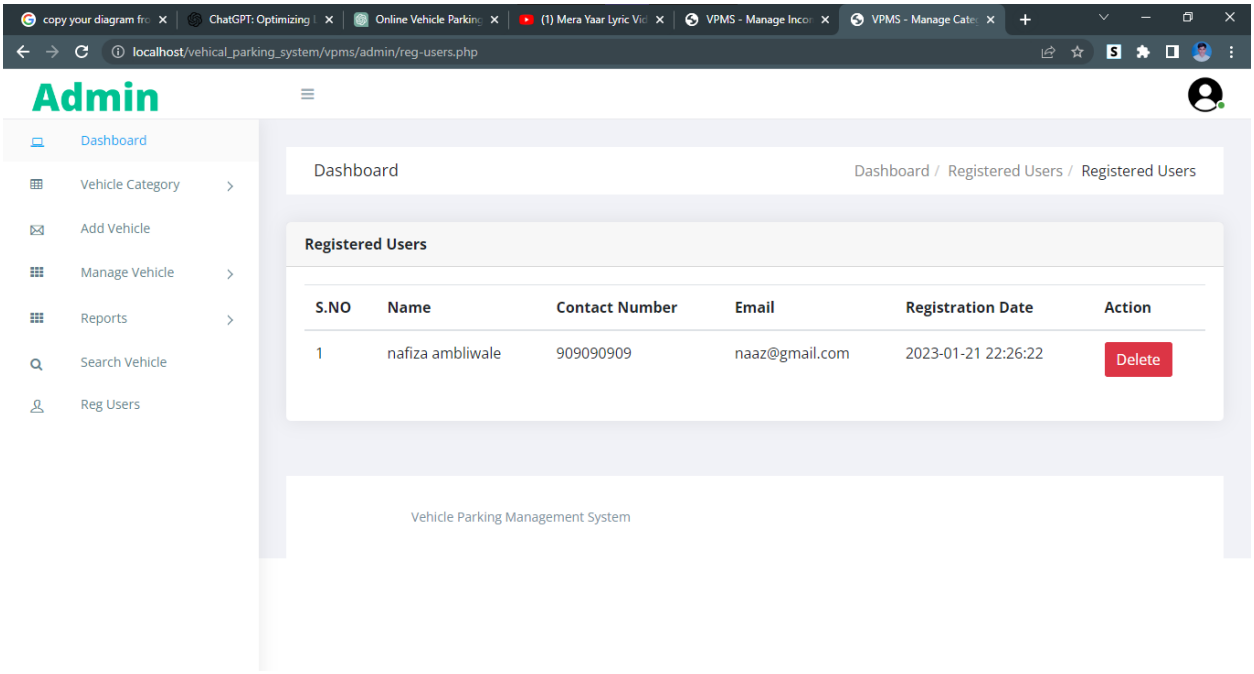
Date Reports



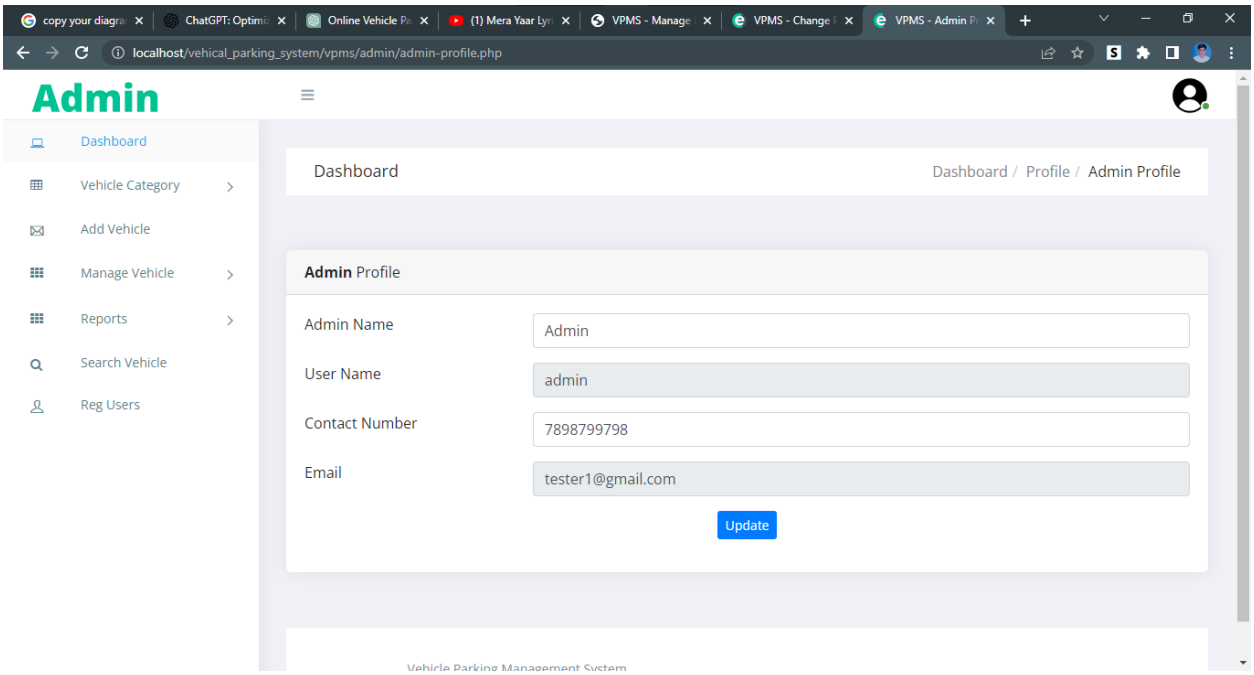
Search Vehicle



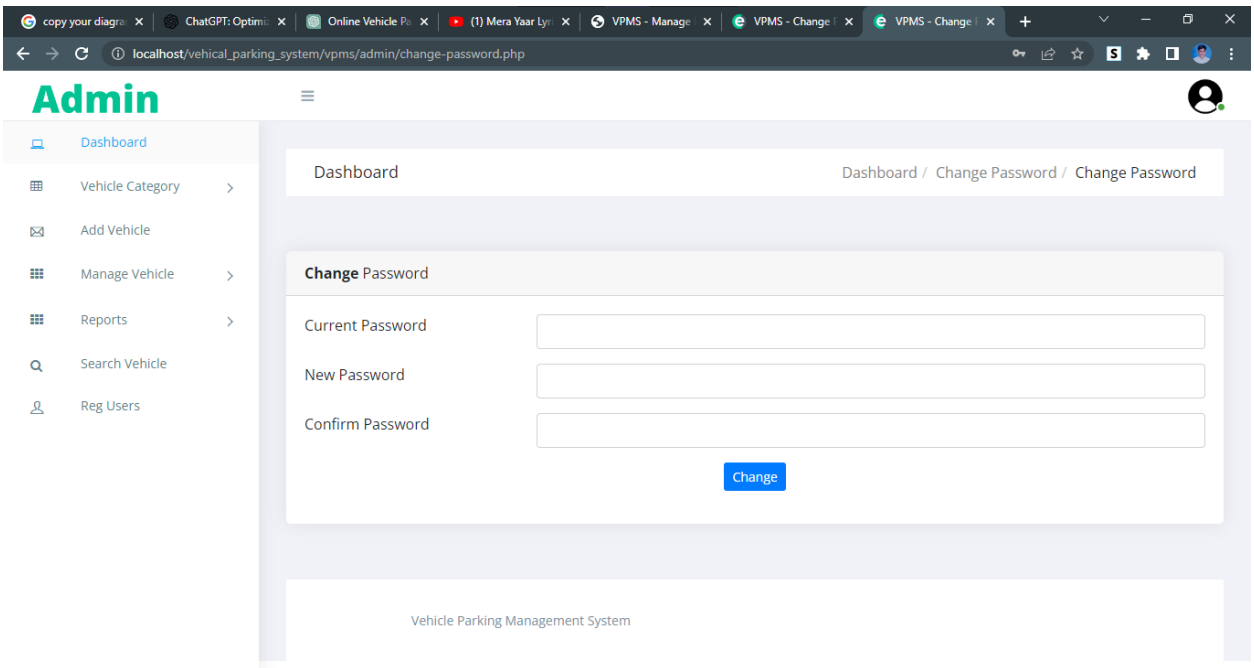
Registered Users



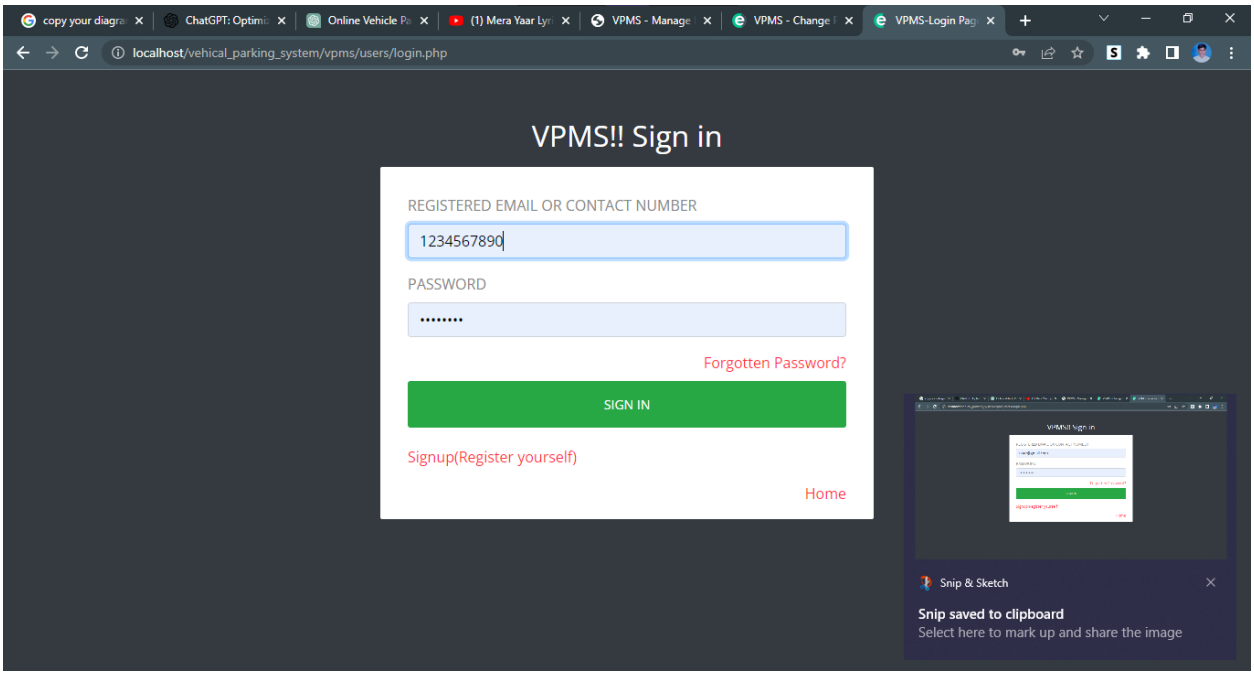
Profile



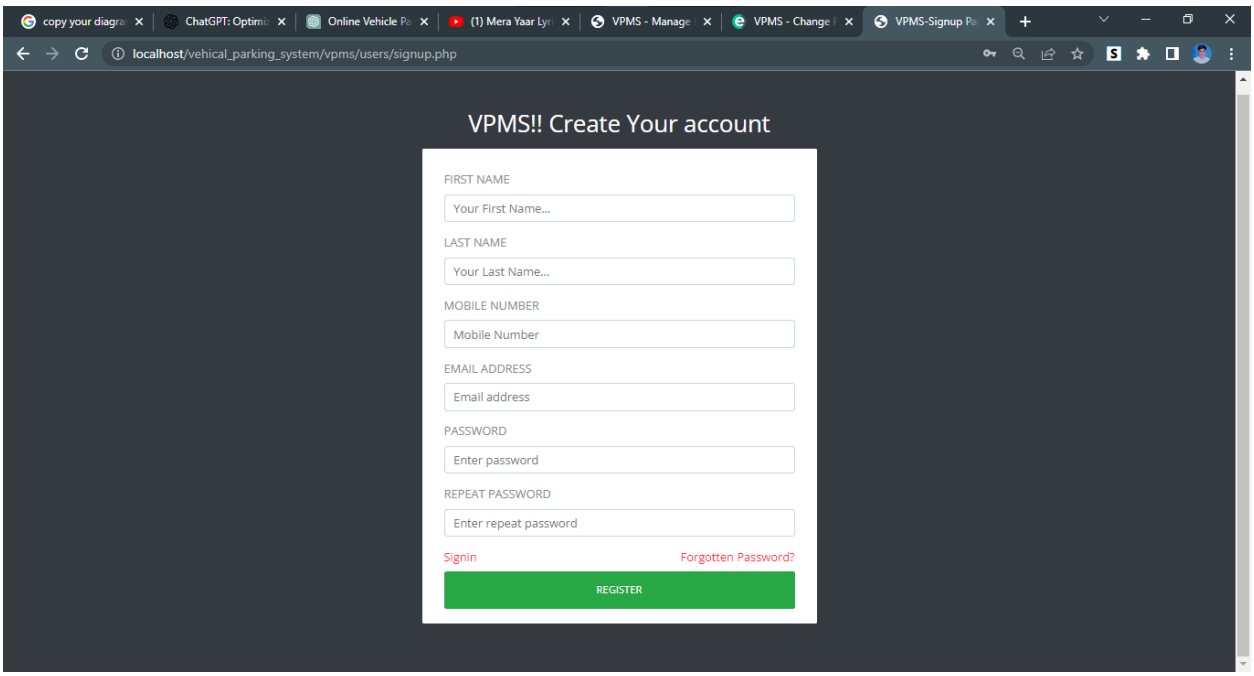
Change password



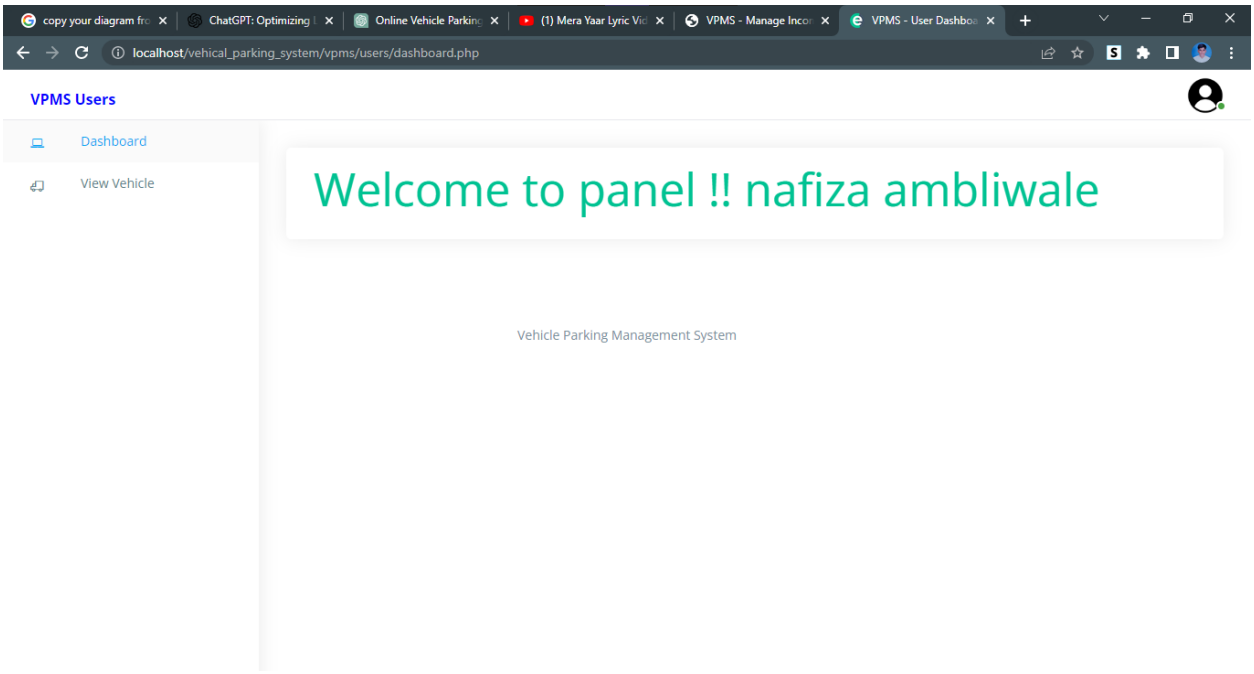
Use login



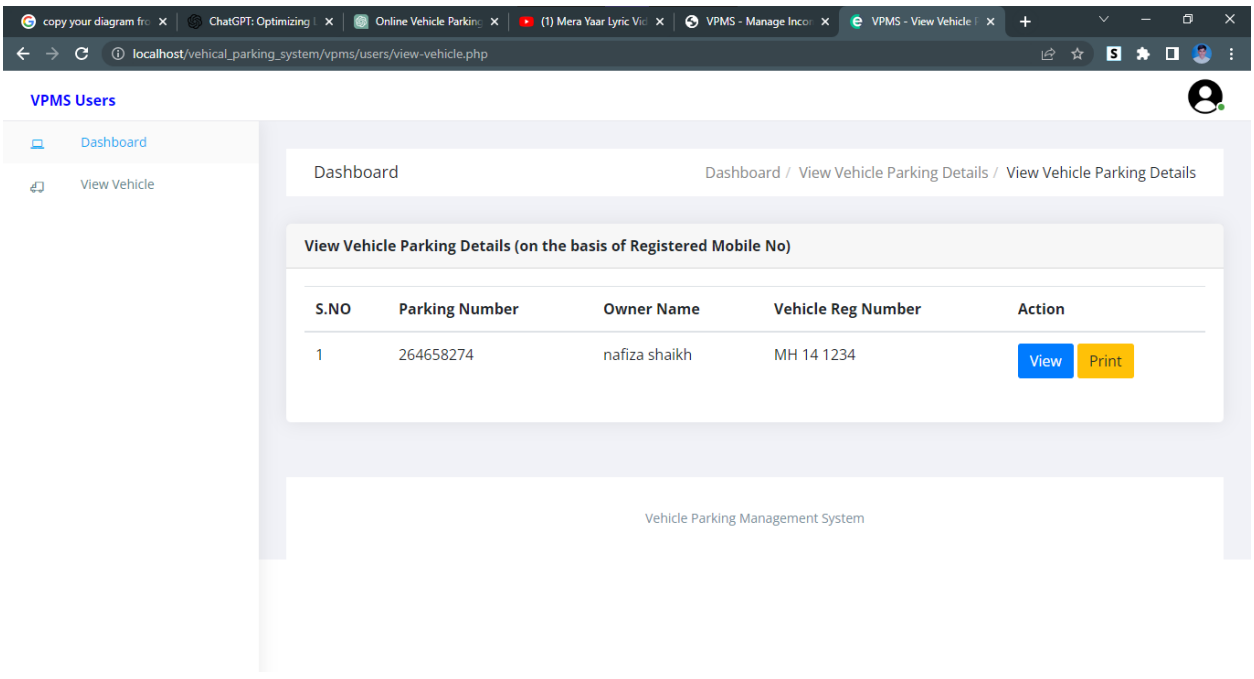
Register new user



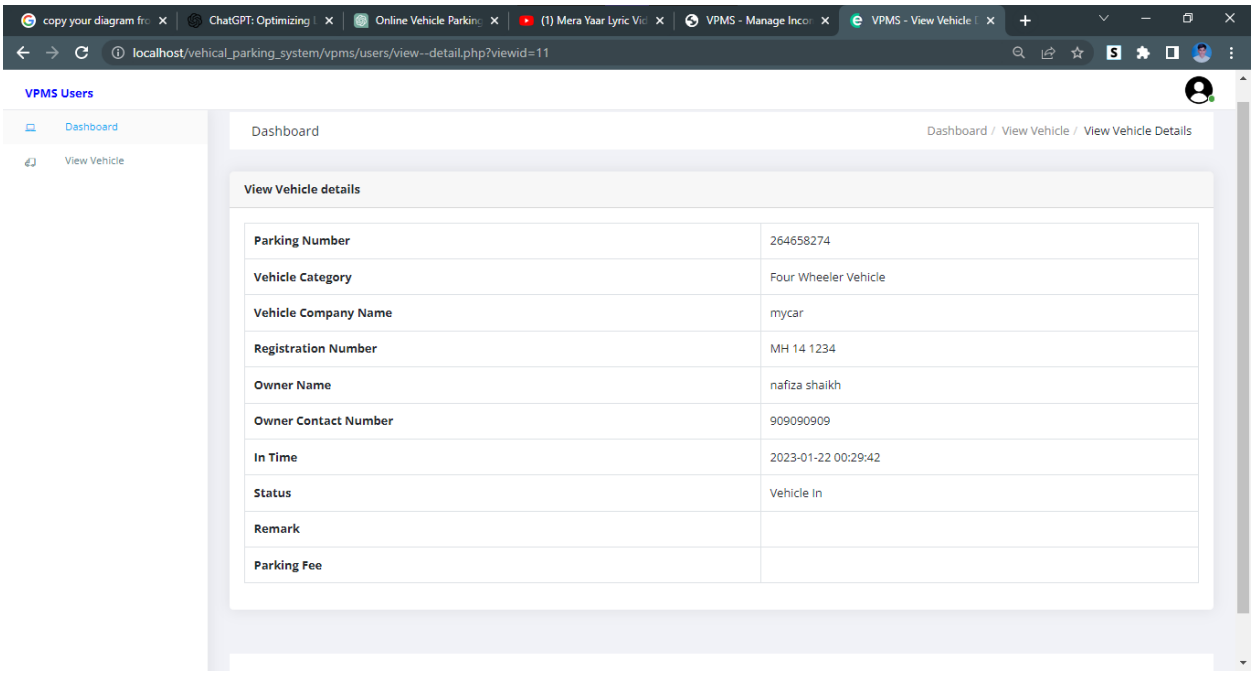
User Dashboard



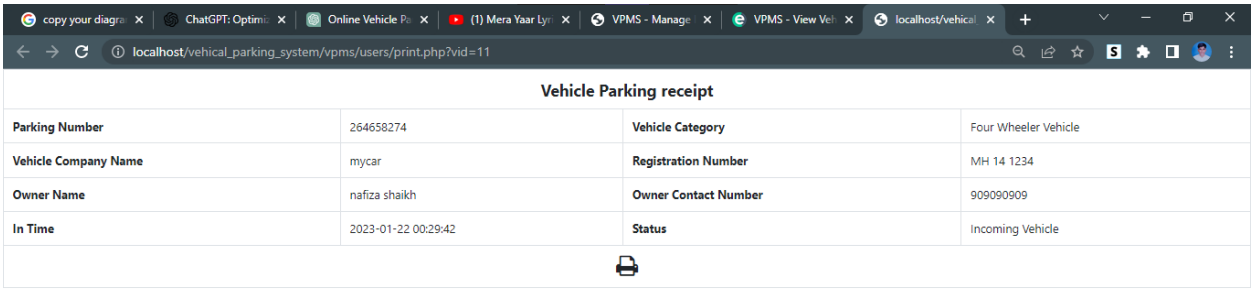
View User Vehicle



View User’s vehicle details



Print receipt of parking



7. CONCLUSION

This Application provides a computerized version of Vehicle ParkingManagement System which will benefit the parking premises.

It makes entire process online and can generate reports. It has a facility of staff's login where staff can fill the visitor details and generate report.

The Application was designed in such a way that future changes can be done easily. The following conclusions can be deduced from the development ofthe project.

- Automation of the entire system improves the productivity.
- It provides a friendly graphical user interface which proves to bebetter when compared to the existing system.
- It gives appropriate access to the authorized users depending ontheir permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it isnecessary.

8.FUTURE SCOPE

The future scope of online vehicle parking systems includes a wide range of possibilities, including increased efficiency, automation, and integration with other technologies. Some potential developments include:

- Smart parking: The use of sensors and other technologies to dynamically manage parking spaces, directing drivers to available spots and reducing congestion.
- Predictive parking: Predictive analytics that anticipate parking demand and adjust pricing or availability accordingly.
- Integration with other systems: Integration with traffic management systems, navigation apps, and other technologies to provide drivers with real-time information about parking availability and guide them to open spots.
- Automation: Automated parking systems, such as robotic garages, that can park and retrieve vehicles without human intervention.
- Electric Vehicle charging: integration of EV charging stations with the parking system, allowing the EV owner to reserve a parking spot and charging point in advance

9.BIBLIOGRAPH AND REFERENCE

- https://www.tutorialspoint.com/index.htm
- https://www.javatpoint.com
- https://www.w3schools.com
- https://html.com
- www.google.com
- www.wikipedia.com
- www.studymafia.org