International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 11 | Nov - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Vehicle Parking Management System

Jitendra A Gaikwad
Department of Instrumentation Engineering
Vishwakarma Institute of Technology
Pune, India
Email- Jitendra.gaikwad@vit.edu

Utkarsha B Lipte
Department of Instrumentation Engineering
Vishwakarma Institute of Technology
Pune, India
Email - utkarsha0107lipte@gmail.com

***Abstract*-Due to the increasing population in cities, there is an exponential rise in the number of vehicles which is leading to major problems leading to poor traffic management and congestion. Another major problem faced by the vehicle owners is the availability of parking space. The idea of Smart Cities is slowly gaining pace with the ever increasing technologies. A parking management system automates a car parking system. It optimizes parking space and make processes efficient. It gives real-time car parking information such as vehicle & slot counts, available slots display, reserved parking, pay-and-park options, easy payments, reports, and a host of other features. 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.**
It enables the time management and control of vehicles by using parking number.

I.INTRODUCTION

Vehicle Parking Management system is a web-based technology that will manage the records of the incoming and outgoing vehicles in an parking house. It's an easy for Admin to retrieve the data if the vehicle has been visited through number he can get that data. Vehicle parking management system is an automatic system which delivers data processing in very high speed in systematic manner.
In VPMS we use PHP and MySQL database. This is the project which keeps records of the vehicle which is going to park in the parking area. VPMS has two module admin and user.

1. ADMIN
1.Dashboard: In this section, admin can briefly view the number of vehicle entries in a particular period.
2.Category: In this section, admin can manage category (add/update/delete).
3.Add Vehicle: In this section, admin add vehicle which is going to park.
4.Manage Vehicle: In this section, admin can manage incoming and outgoing vehicle and admin can also add parking charges and his/her remarks.
5.Reports: In this section admin can generate vehicle entries reports between two dates.
6.Search: In this section, admin can search a particular vehicle by parking number.
Admin can also update his profile, change the password and recover the password.

2. USERS
1.Dashboard: It is welcome page for an users.
2.View Vehicle: In this section, users view the details of vehicle parking which is parked by him/her.
Users can also update his profile, change the password and recover the password.

3.Purpose:
The purpose of developing vehicle parking management system is to computerized the tradition way of parking. Another purpose for developing this application is to generate the report automatically.

4.Scope:
In the modern age. Many people have vehicles. Vehicle is now a basic need. Every place is under the process of urbanization. There are many corporate offices and shopping centers 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. With the help of this system we can deliver a good service to customer who wants to park their vehicle into the any organization's premises.

II. Analysis and Design

1. Analysis

In present all visitors parking work done on the paper. The whole year visitor parking record is stored in the registers. We can't generate reports as per our requirements because it take more time to calculate the visitors parking report.

Disadvantage of present system:

1.Not user friendly: The present system not user friendly because data is not stored in structure and proper format.

2.Manual Control: All report calculation is done manually so there is a chance of error.

3.Lots of paper work: Visitors maintain in the register so lots of paper require storing details.

2. Design Introduction

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in

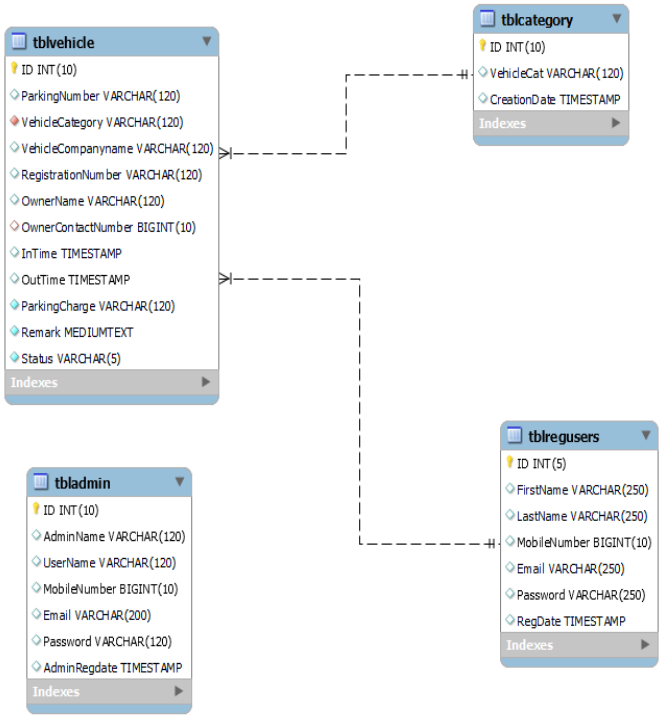
sufficient detail to permit its physical realization. Ones the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

III. Diagram

1.Class Diagram



Dig. Class Diagram

2.ER Diagram

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

1.It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.

2.It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.

3.In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

-ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used; among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin.

-The symbols used for the basic ER constructs are:

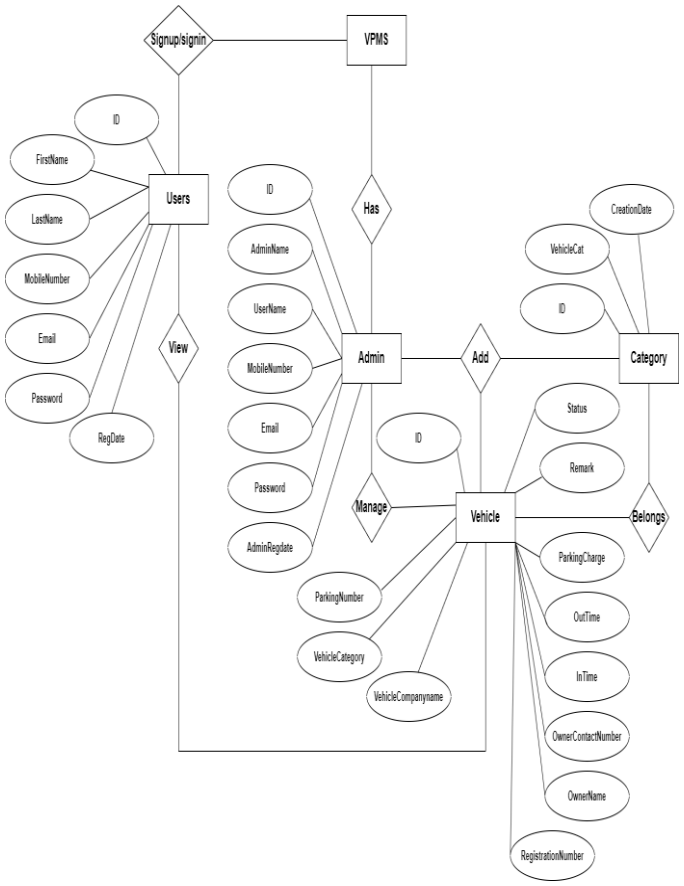
1.Entities are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.

2.Relationships are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs

3.Attributes, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.

4.Cardinality of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.

5.Existence is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional.



Dig. ER Diagram

3.Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.
It shows how data enters and leaves the system, what changes the information, and where data is stored.
The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

-The following observations about DFDs are essential:

1.All names should be unique. This makes it easier to refer to elements in the DFD.

2.Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.

3.Suppress logical decisions. If we ever have the urge to draw a diamond-shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.

4.Do not become bogged down with details. Defer error conditions and error handling until the end of the analysis.

-Standard symbols for DFDs are derived from the electric circuit diagram analysis and are shown in fig:

Symbol	Name	Function
	Data flow	Used to Connect Processes to each , other , to sources or Sinks; te arrow head indicates direction of data flow.
	Process	Perfroms Some transformation of Input data to yield output data.
	Source of Sink (External Entity)	A Source of System inputs or Sink of System outputs.
	Data Store	A repository of data; the arrow heads indicate net inputs and net outputs to store.

Symbols for Data Flow Diagrams

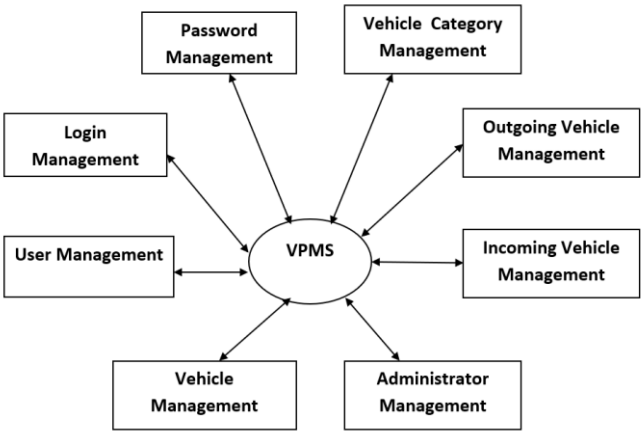
1.Data Flow: From dig. A curved line shows the flow of data into or out of a process or data store.

2.Circle: From dig. A circle (bubble) shows a process that transforms data inputs into data outputs.

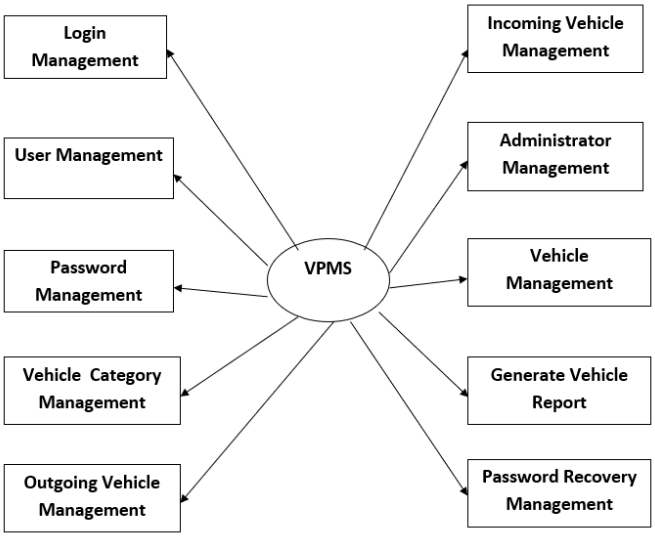
3.Source or Sink: From dig. Source or Sink is an external entity and acts as a source of system inputs or sink of system outputs.

4.Data Store: From dig. A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements.

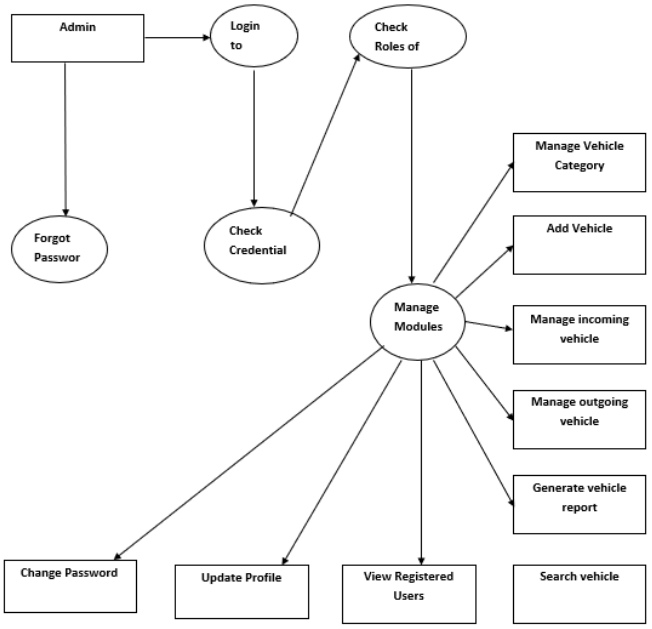
1. Dig. Zero Level DFD



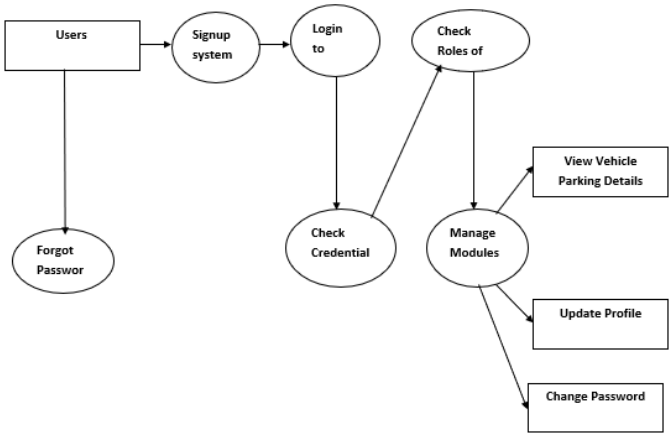
2. Dig. First Level DFD



3.Dig.Second Level DFD



4.Dig. Third Level DFD



IV. Server Side

1.APACHE

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

The Apache HTTP Server ("httpd") was launched in 1995 and it has been the most popular web server on the Internet since April 1996. It has celebrated its 20th birthday as a project in February 2015.

2.PHP

PHP stands for PHP: Hypertext Preprocessor. PHP is a server-side scripting language. It is a widely-used, open-source scripting language designed specifically for web development but also used as a general-purpose programming language. PHP scripts are executed on the server, and the result is sent to the client's web browser as plain HTML. PHP can interact with a wide range of databases, such as MySQL, PostgreSQL, Oracle, and

Microsoft SQL Server.

3. MYSQL

MySQL is a widely used relational database management system (RDBMS). MySQL is free and open-source. MySQL is ideal for both small and large applications.

V. Implementation and System Testing

After all phase have been perfectly done, the system will be implemented to the server and the system can be used.

System Testing: The goal of the system testing process was to determine all faults in our project .The program was subjected to a set of test inputs and many explanations were made and based on these explanations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing.

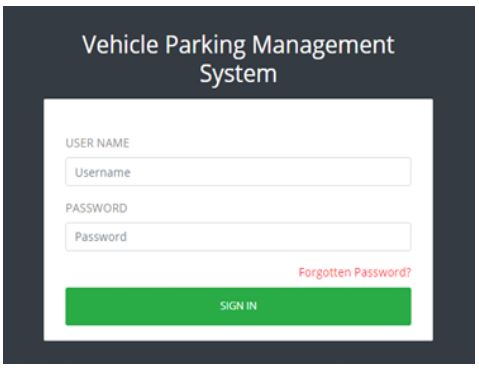
1. Unit testing: Unit testing is commenced when a unit has been created and effectively reviewed .In order to test a single module we need to provide a complete environment i.e. besides the section we would require.

2.Integration testing: In the Integration testing we test various combination of the project module by providing the input.The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module.

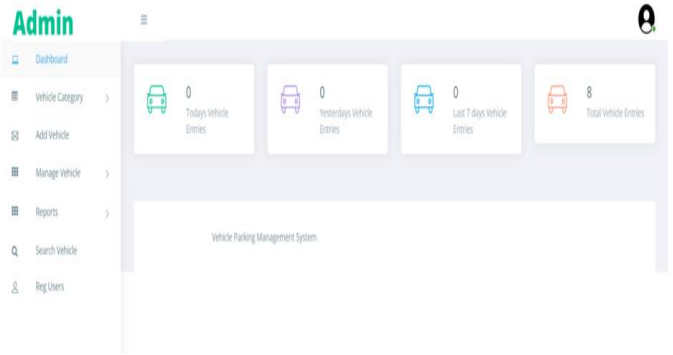
VI. Evaluation

1.Home Page

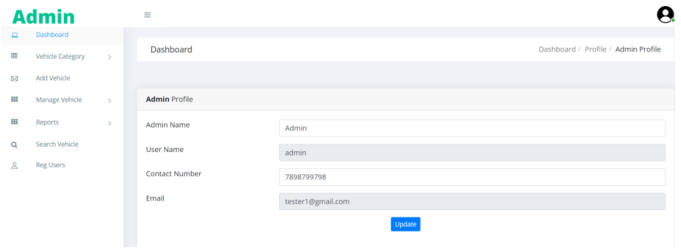
2.Admin Login Page



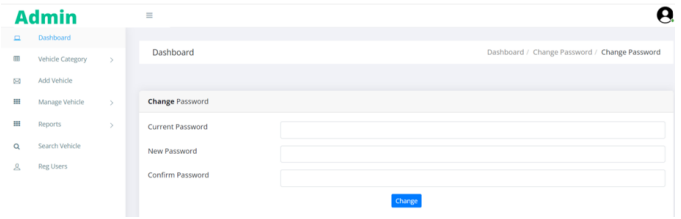
3.Dashboard



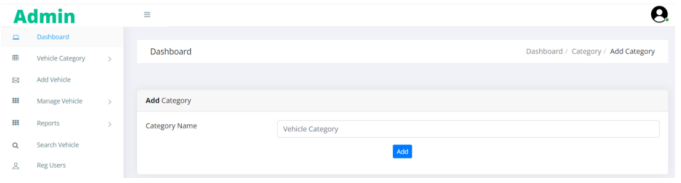
4.Profile



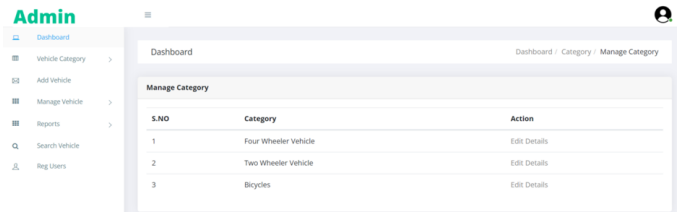
5.Change Password



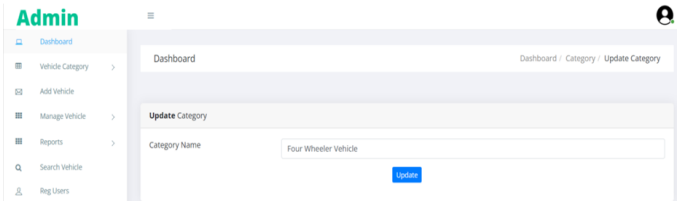
6.Add Category



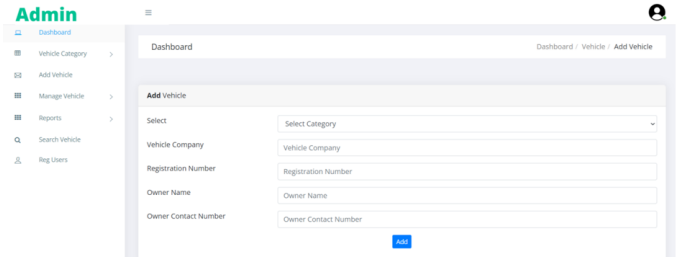
7.Managed Category



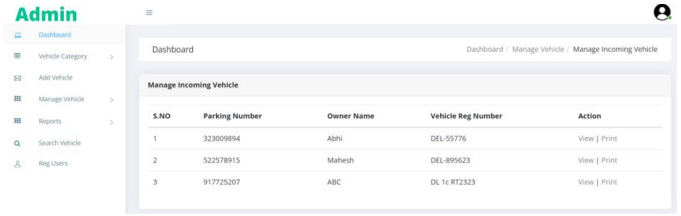
8.Update Category



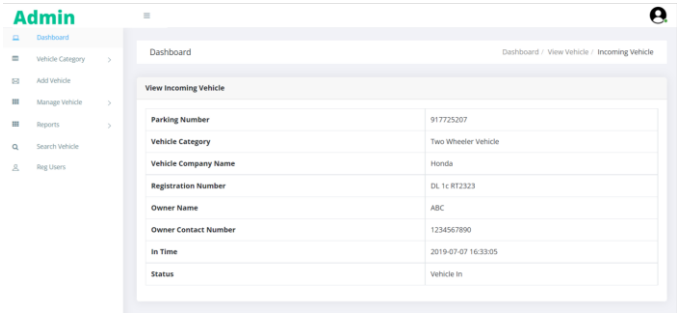
9.Add Vehicle



10.Managed Incoming Vehicle



11.View Incoming Vehicle

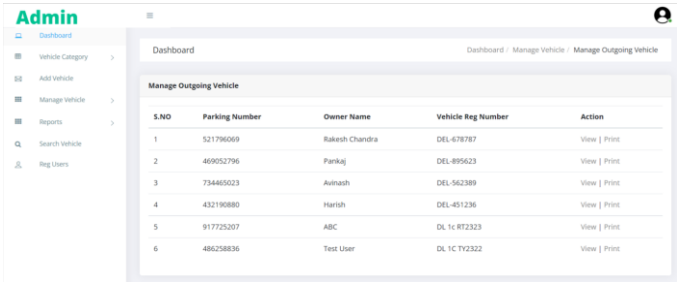


12.Parking Receipt

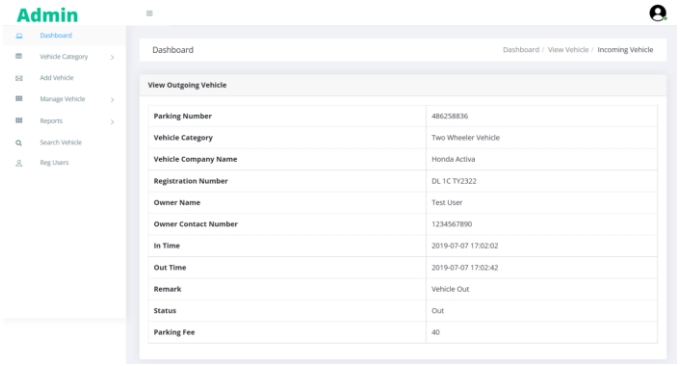
Vehicle Parking receipt

Parking Number	323009894	Vehicle Category	Two Wheeler Vehicle
Vehicle Company Name	Activa	Registration Number	DEL-55776
Owner Name	Abhi	Owner Contact Number	4654654654
In Time	2019-07-06 14:28:38	Status	Incoming Vehicle

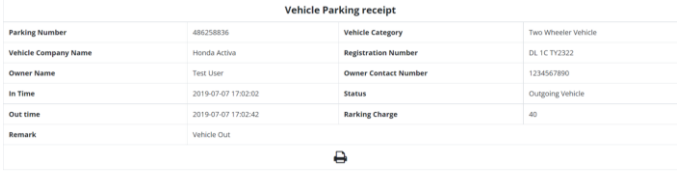
13.Managed Outgoing Vehicle



14.View Outgoing Vehicle



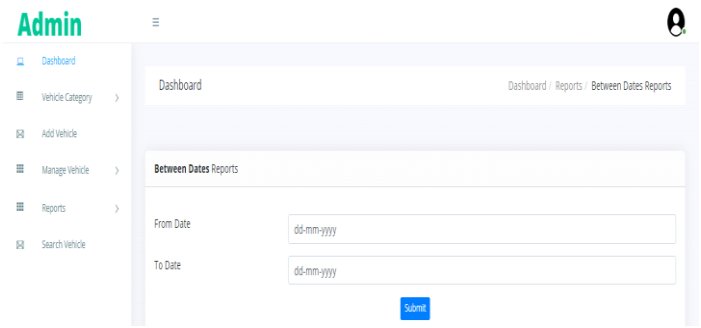
15.Vehicle Parking Receipt



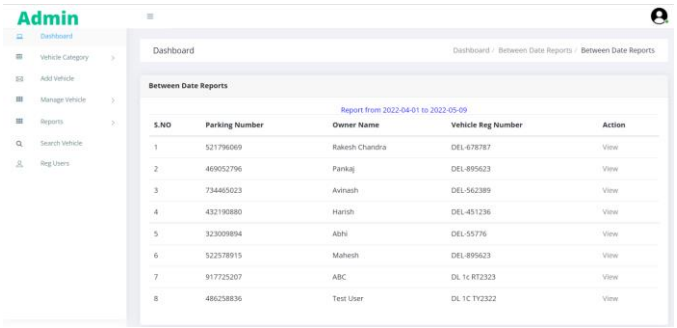
Vehicle Parking receipt

Parking Number	486258836	Vehicle Category	Two Wheeler Vehicle
Vehicle Company Name	Honda Activa	Registration Number	DL 1C TY2322
Owner Name	Test User	Owner Contact Number	1234567890
In Time	2019-07-07 17:02:02	Status	Outgoing Vehicle
Out time	2019-07-07 17:02:42	Rarking Charge	40
Remark	Vehicle Out		

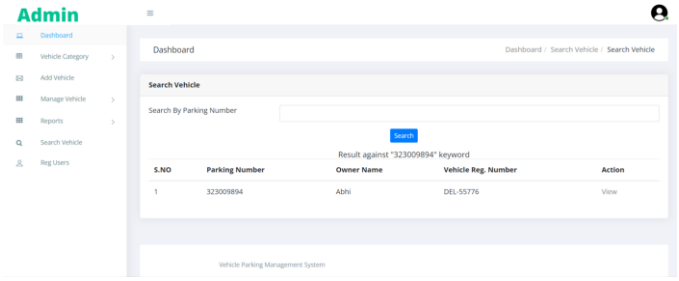
16.Between Dates Report



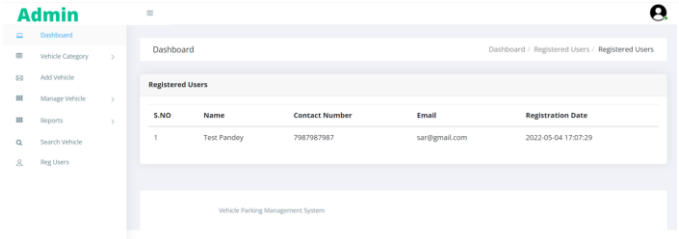
17.View Report



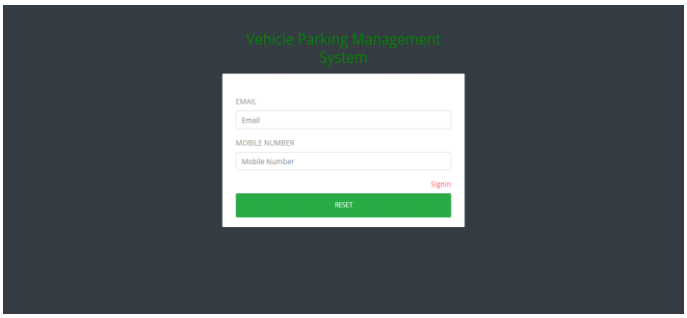
18.Search Vehicle



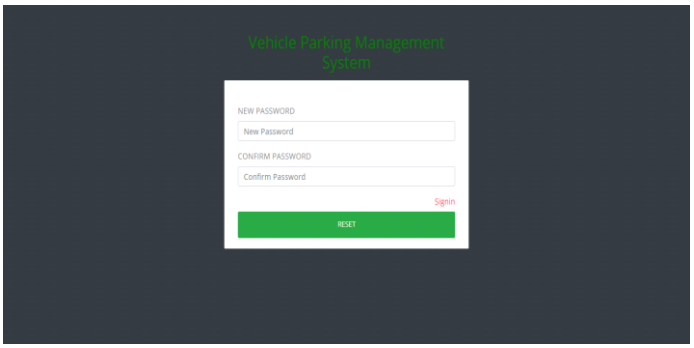
19.View Registered Users



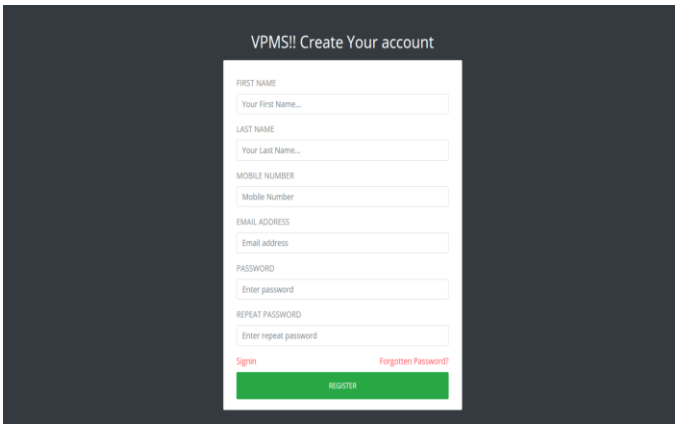
20.Forgot Password



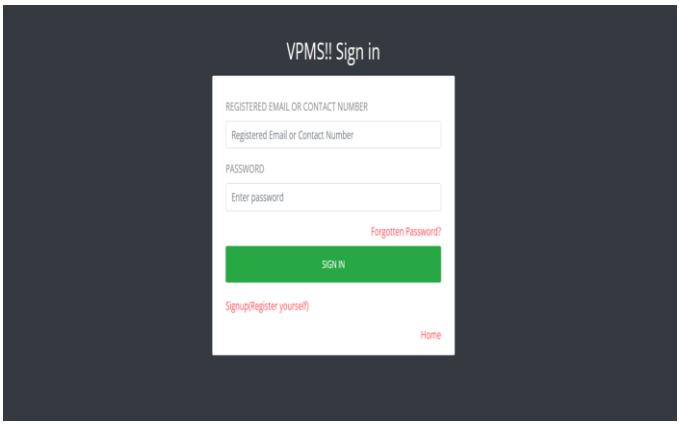
21.Reset Password



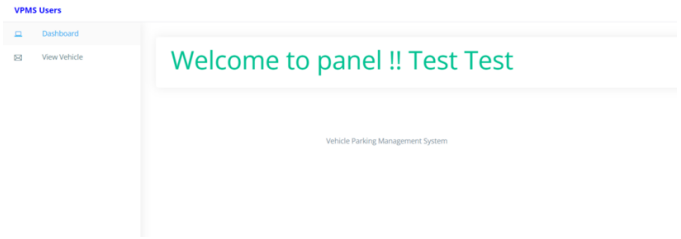
22.User Sign up



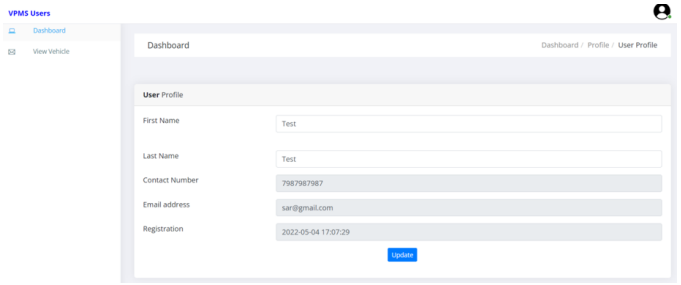
23.Sign in



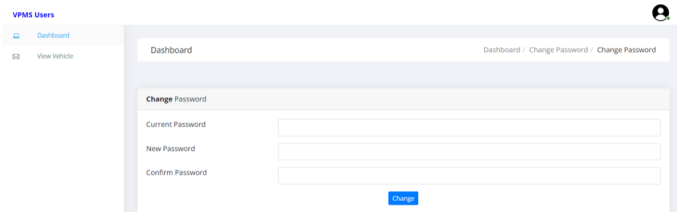
24.Dashboard



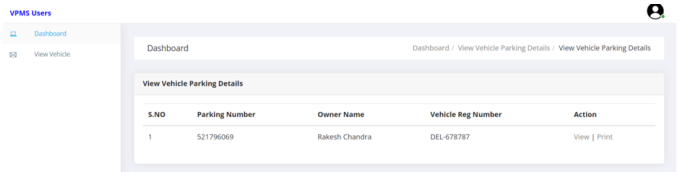
25.Profile



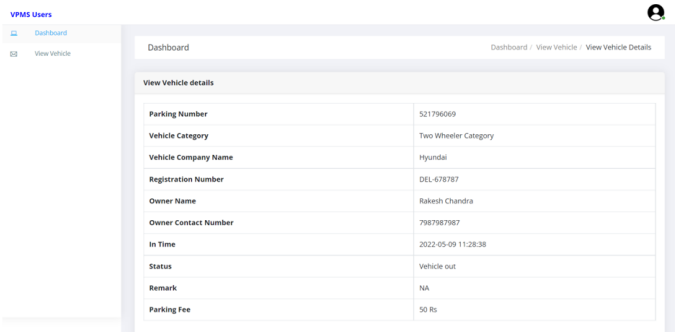
26.Change Password



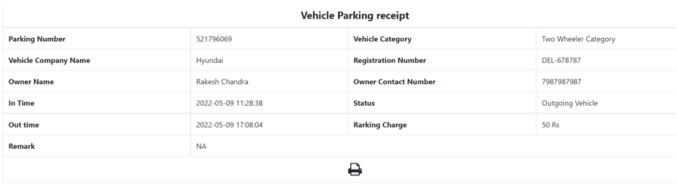
27.View Vehicle



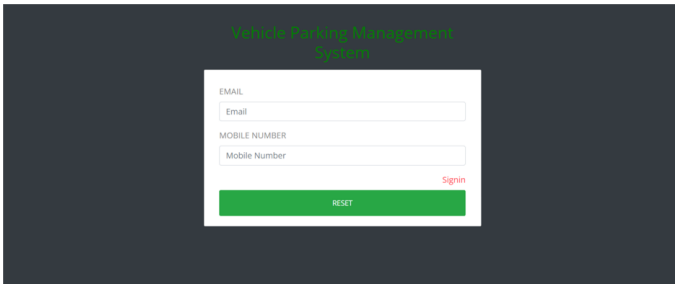
28.View Vehicle in Details



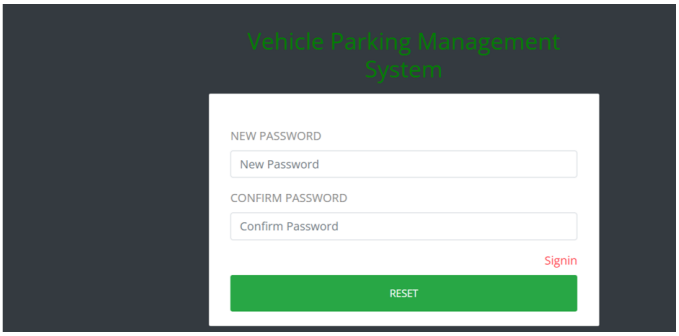
29.View Parking Receipt



30.Forgot Password



31.Reset Password



VII.CONCLUSION

This Application provides a computerized version of Vehicle Parking Management 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 of the project.

1.Automation of the entire system improves the productivity.

2.It provides a friendly graphical user interface which proves to be better when compared to the existing system.
3.It gives appropriate access to the authorized users depending on their permissions.
4.It effectively overcomes the delay in communications.
5.Updating of information becomes so easier.
6.System security, data security and reliability are the striking features.
7.The System has adequate scope for modification in future if it is necessary.

REFERENCES

[1] "Smart Parking Management Systems: A Comprehensive Guide" by [Author Name] - This book provides an overview of smart technologies in parking management, including sensors and IoT applications.
[2] "Intelligent Transportation Systems: Technologies and Applications" Covers various applications of intelligent transportation systems, including advanced parking solutions.
[3] sFor PHP
https://www.w3schools.com/php/default.asp
https://www.sitepoint.com/php/
https://www.php.net/
[4] For MySQL
https://www.mysql.com/
http://www.mysqltutorial.org
[5] For XAMPP
https://www.apachefriends.org/download.html