**APPENDIX 1**

**VEHICLE PARKING MANAGEMENT**

**SYSTEM**

**A MINI-PROJECT REPORT**

***Submitted by***

**717822I213 GOKUL M**

**717822I240 PREM KUMAR G**

**717822I246 ROHITH KUMAR M**

***in partial fulfilment for the award of the degree***

***of***

# BACHELOR OF TECHNOLOGY

***in***

**ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**

**KARPAGAM COLLEGE OF ENGINEERING COIMBATORE – 641 032**

**ANNA UNIVERSITY: CHENNAI 600 025**

**MAY 2024**

**APPENDIX 2**

**KARPAGAM COLLEGE OF ENGINEERING COIMBATORE – 641 032**

**ANNA UNIVERSITY: CHENNAI 600 025**

# BONAFIDE CERTIFICATE

Certified that this mini project report **“VEHICLE PARKING MANAGEMENT SYSTEM”** is the bonafide work of **“GOKUL M, PREM KUMAR G, ROHITH KUMAR M”** who carried out the mini project work under my supervision.

Dr. S. MANIKANDAN M.Tech,Ph.D Ms. U. Gayathri M.E.,

**HEAD OF THE DEPARTMENT SUPERVISOR**

ARTIFICIAL INTELLIGENCE & DATA SCIENCE ARTIFICIAL INTELLIGENCE & DATA SCIENCE

KARPAGAM COLLEGE OF ENGINEERING, KARPAGAM COLLEGE OF ENGINEERING, COIMBATORE – 641032 COIMBATORE - 641032

Certified that the candidate was examined during the viva voce examinations held on

Signature of the Internal Examiner with date Signature of External Examiner with date

**DECLARATION**

We hereby declare that this Mini Project report entitled “**Vehicle Parking Managemant system**” submitted by us for the degree of **B. Tech Artificial Intelligence & Data Science at Karpagam College of Engineering**, **Coimbatore** is the record of original work done by me under the guidance and supervision of

**Ms.Gayathri.U ME.**,**Assitant professor** at the Department of Artificial Intelligence & Data Science, Karpagam College of Engineering, Coimbatore – 641032 and has not formed the basis for the award of any degree, diploma or titles in this institution or any other Institution of higher learning.

|  |  |  |
| --- | --- | --- |
| Date | : | **Name and Signature of the Candidate(s)** |
| Place | : Coimbatore | **GOKUL M – 717822I213** |
|  |  | **PREM KUMAR G – 717822I240** |
|  |  | **ROHITH KUMAR M– 717822I246** |
|  |  |  |

## ACKNOWLEDGEMENT

### First and foremost praises and thanks to the almighty for her showers and blessings throughout our mini project work to complete it successfully.

We extend our gratitude to the Management of Karpagam College of Engineering, Coimbatore for the excellent infrastructure and support facilities to undergo the mini- project work.

We are very grateful to **Dr. V. KUMAR CHINNAIYAN M.E.**,**Ph.D.,** the Principal and **DR. S. MANIKANDAN M.Tech., Ph.D.**, Head of the department **Artificial Intelligence & Data Science** for providing the facilities, support and permission to carry out our /my mini project work at our esteemed institution.

We record my sincere gratitude to our Mini project Coordinator **Ms.Gayathri.U ME.**,**Assitant professor** for giving input, and encouragement for continuous improvement during the process and to complete this mini-project work.

We would like to express our sincere gratitude to our Supervisor **Ms.Gayathri.U ME.**,**Assitant professor** Head of the department for the continuous support for our UG study, for his motivation and adequate guidance which helped us to achieve success in all our accomplishments and to complete this mini project work.

We also thank all the teaching faculty members and non-teaching Staff members of the Department of **Artificial Intelligence & Data Science,** Karpagam College of Engineering, Coimbatore for their kindness and support.

We would like to thank **our parents, family members and friends** who sacrificed their time and energy to complete the mini-project work successfully.

**GOKUL M, PREM KUMAR G, ROHITH KUMAR M**

## ABSTRACT

The Online vehicle Parking Management system is designed from day to day working of parking management system. The project provides online platform to accomplish day to day vehicle management data. The 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 that delivers data processing in very high speed in systematic manner

**APPENDIX 3**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE** |

NO. DECLARATION iii

[ACKNOWLEDGEMENT](#_bookmark0) iv

[ABSTRACT v](#_bookmark1)

[LIST OF FIGURES i](#_bookmark2)x

LIST OF ABBREVIATIONS x

1. INTRODUCTION
   1. [Mini Project Overview 1](#_TOC_250021)
   2. [Scope 6](#_TOC_250019)
   3. [Problem Statement 7](#_TOC_250018)
   4. [Web Technologes 7](#_TOC_250017)
   5. Database Management System 7
2. LITERATURE REVIEW
   1. [Overall Description 8](#_TOC_250016)
   2. [Specific Requirement 9](#_TOC_250015)
3. DETAILED DESIGN
   1. [System Design 11](#_TOC_250013)
   2. [Entity Relationship Diagram 12](#_TOC_250012)
   3. [Flow Diagram 12](#_TOC_250011)
   4. Description Tables 12
4. IMPLEMENTATION
   1. [Modules And Their Roles 17](#_TOC_250006)
5. TESTING
   1. [Software Testing 19](#_TOC_250004)
   2. Module Testing And Integration 20
6. OUTPUTS
   1. [Login Page 21](#_TOC_250003)
   2. Dashboard 21
   3. Vehicle Category 21
   4. Vehicle Details 21
   5. Vehicle OUT Details 21
   6. Parking Rece
7. CONCLUSION
   1. Conclusion 22
8. PROGRAM
   1. [parking\_db.sql 23](#_TOC_250002)

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure no** | **List of Figures** | **Page.no** |

* 1. HTML5 APIs and related technologies 14
  2. Taxonomy and status of CSS3 modules 15
  3. Parking management system web architecture 18
  4. Flow diagram of parking management system 18

3.4.1 Admin Table 18

* + 1. Vehicle Category Table 18
    2. Vehicle details Table 18
  1. Login Page 20
  2. Dashboard 21
  3. Vehicle Category
  4. Vehicle Details
  5. Vehicle Out Details
     1. Parking Ticket
     2. Parking Receipt

6.7 Vehicle Search

## LIST OF ABBREVIATION & SYMBOLS

1. **HTML** HyperText Markup Language
2. **CSS** Cascading Style Sheets
3. **JS** JavaScript
4. **MYSQL** MyStructured Query Language
5. **PHP** Hypertext Preprocessor

### CHAPTER 1 INTRODUCTION

### Mini project Overview

Vehicle Parking management system is a tool that is used to maintain vehicle parking details. The system makes use of a single centralized database to maintain records of all the customers who park their vehicle. Authentication process is implemented to allow only the Admin. Vehicle-in data, Vehicle-out data, history (of customers) credentials are all stored in database. All the processes are carried out through an interactive interface on a web platform.

Having a centralized database for all vehicle of the same group helps in maintaining the records and makes administrator job much easier. It also helps in comparing different attributes of the group.

### Objective:

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 is adopted 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.

### 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. 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.

* 1. **PROBLEM STATEMENT**

“To handle and maintain the details of vehicle in any parking places.”

The aim of the project is to show the real-world implementation of Database ManagementSystem developed using technologies such as HTML5, CSS, JavaScript, PHP. The tool isdeployed on web for the admin to access and maintain details of Vehicle parking. It can beused across different operating systems and solely depends on the type of web browser andversion of MySQL used, as the project is developed for web platform. But the deployedversion only depends on the user’s browser and its version

* 1. **WEB TECHNOLOGES**

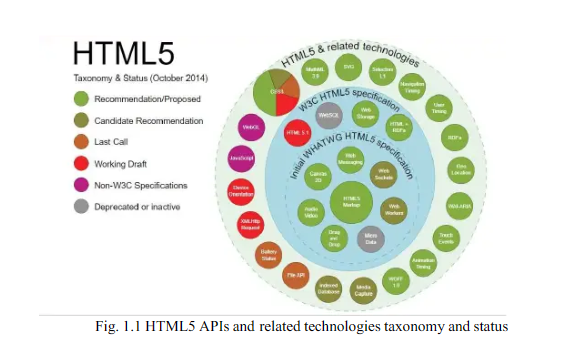
Web technology refers to the means by which computers communicate with each otherusing markup languages and multimedia packages. It gives us a way to interact with hostedinformation, like websites. Web technology involves the use of hypertext markup language(HTML) and cascading style sheets (CSS). There are many other technologies that areavailable that helps us to create website best suited for our needs. We will learn more aboutthese technologies in the following sub sections

**1.5.1 HTML**

HTML5 is a markup language used for structuring and presenting content on the WorldWide Web. It is the fifth and current major version of the HTML standard.

It was published in October 2014 by the World Wide Web Consortium (W3C) toimprove the language with support for the latest multimedia, while keeping it both easilyreadable by humans and consistently understood by computers and devices such as web browsers, parsers, etc. HTML5 is intended to subsume not only HTML 4, but alsoXHTML 1 and DOM Level 2 HTML.

HTML5 includes detailed processing models to encourage more interoperableimplementations; it extends, improves and rationalizes the markup available for documents,and introduces markup and application programming interfaces (APIs) for complex webapplications. For the same reasons, HTML5 is also a candidate for cross- platform mobileapplications, because it includes features designed with low-powered devices in mind.

****

**1.5.2 CSS**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstonetechnology of the World Wide Web, alongside HTML and JavaScript.

CSS was first proposed by Håkon Wium Lie on October 10, 1994. At the time, Liewas working with Tim Berners-Lee at CERN. Several other style sheet languages for theweb were proposed around the same time, and discussions on public mailing lists and insideWorld Wide Web Consortium resulted in the first W3C CSS Recommendation (CSS1) being released in 1996. In particular, Bert Bos proposal was influential; he became co-author of CSS1 and is regarded as co-creator of CSS.

Style sheets have existed in one form or another since the beginnings of StandardGeneralized Markup Language (SGML) in the 1980s, and CSS was developed to providestyle sheets for the web. One requirement for a web style sheet language was for style sheetsto come from different sources on the web. Therefore, existing style sheet languages likeDSSSL and FOSI were not suitable. CSS, on the other hand, let a document's style beinfluenced by multiple style sheets by way of "cascading" styles.

CSS is designed to enable the separation of presentation and content, includinglayout, color’s, and fonts. This separation can improve content accessibility, provide moreflexibility and control in the specification of presentation characteristics, enable multipleweb pages to share formatting by specifying the relevant CSS in a separate .css file, andreduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the samemarkup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices.CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name cascading comes from the specified priority scheme to determine whichstyle rule applies if more than one rule matches a particular element. This cascading priorityscheme is predictable

Some of the notable advantages are as follows:

 Separation of content from presentation

CSS facilitates publication of content in multiple presentation formats based onnominal parameters. Nominal parameters include explicit user preferences, differentweb browsers, the type of device being used to view the content (a desktop computeror mobile Internet device), the geographic location of the user and many othervariables.

Site-wide consistency

When CSS is used effectively, in terms of inheritance and "cascading", a global stylesheet can be used to affect and style elements site-wide. If the situation arises that thestyling of the elements should be changed or adjusted, these changes can be made byediting rules in the global style sheet. Before CSS, this sort of maintenance was moredifficult, expensive and time-consuming.

Bandwidth

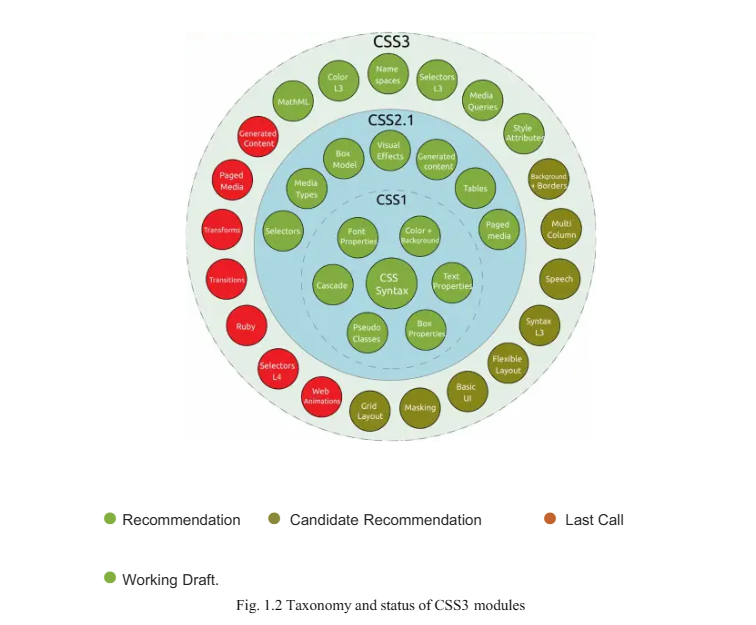
A stylesheet, internal or external, specifies the style once for a range of HTMLelements selected by ‘class’ type or relationship to others. This is much more efficientthan repeating style information inline for each occurrence of the element. Anexternal stylesheet is usually stored in the browser cache,and can therefore be usedon multiple pages without being reloaded, further reducing data transfer over anetwork.

Page reformatting

With a simple change of one line, a different style sheet can be used for the same page. This has advantages for accessibility, as well as providing the ability to tailor a page or site to different target devices. Furthermore, devices not able to understandthe styling still display the content

Accessibility

Without CSS, web designers must typically lay out their pages with techniques suchas HTML tables that hinder accessibility for vision-impaired users.

****

**1.5.3 JAVA SCRIPT**

JavaScript, often abbreviated as JS, is a high-level, interpreted programming language. Itis a language which is also characterized as dynamic, weakly typed, prototype-based andmulti-paradigm.

Alongside HTML and CSS, JavaScript is one of the three core technologies of theWorld Wide Web. JavaScript enables interactive web pages and thus is an essential part ofweb applications. The vast majority of websites use it, and all major web browsers have adedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, andimperative (including object-oriented and prototype-based) programming styles. It has anAPI for working with text, arrays, dates, regular expressions, and basic manipulation of theDOM, but the language itself does not include any I/O, such as networking, storage, orgraphics facilities, relying for these upon the host environment in which it is embedded.

Initially only implemented client-side in web browsers, JavaScript engines are nowembedded in many other types of host software, including server-side in web servers anddatabases, and in non-web programs such as word processors and PDF software, and inruntime environments that make JavaScript available for writing mobile and desktopapplications, including desktop widgets.Some of the notable advantages are as follows:

Speed

Being client-side, JavaScript is very fast because any code functions can be runimmediately instead of having to contact the server and wait for an answer

Simplicity JavaScript is relatively simple to learn and implement.

Versatility

JavaScript plays nicely with other languages and can be used in a huge variety ofapplications. Unlike PHP or SSI scripts, JavaScript can be inserted into any web pageregardless of the file extension. JavaScript can also be used inside scripts written inother languages such as Perl and PHP.

**1.5.4 PHP**

PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for Web development, and also used as a general-purpose programming language. It was originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge. The reason behind the popularity of PHP is its several advantages. PHP is most suited for the purpose of web development. The advantages of PHP are discussed briefly below:

Cross Plat form

All the PHP based applications can run on various types of platforms. PHP is supported by majority of Operating Systems, some of which includes Solaris, UNIX ,Windows and Linux. The mentioned platforms can be used to write codes in PHP and also view web pages or run the PHP based applications.

PHP easily interfaces with MySQL and Apache both. An effortless integration of PHP can be done with various other technologies like Java and there is no requirement of re-development. Therefore, saving both time and money, giving it an important advantage.

Easy database connection

A programming language like PHP is widely used on the internet and needs to connect to the database very often. Therefore, having a feature that could help PHP to connect to database easily is mandatory. Several websites such as the e-commerce websites, require good database management system.

PHP has a built-in module that helps it in connecting with database easily. Therefore, PHP has a great demand in the field of web development where a data driven website needs to be developed. PHP significantly reduces the time needed in developing the web application that needs an efficient database management system.

Easy to use

PHP is widely used because it is easy to use. In contrast with other programming languages that are complex, PHP is simple, fluent, clean and organized, hence it is a boon for the new users. PHP has a well-organized syntax which is logical at the same time .PHP does not require any intensive studying or manual to use it. Command functions of PHP are easily understood as the user can easily figure out from the name of the commands itself what it does. A person who is new to PHP can still code because the syntax is somewhat similar to C.A person who is new to PHP can still code because the syntax is somewhat similar to C. Hence, if a person who knows C can easily code in PHP. Hence, it is easier to create and optimize the application using PHP.

Speed is the primary need of web development. There are people who face the challenge of slow internet connection and slow data speed. Furthermore, a fast loading website is always preferred by people across the globe. When compared toother programming languages, PHP is found to be the fastest programming language.

In normal circumstances, it takes a lot of time to connect to the database, when you attempt to fetch certain data from the database. It takes a lot of time in connecting to the database, then executing the statement and finally getting the data. PHP performs these set of tasks faster than other scripting languages. PHP is faster in both connecting to the database and in using other important applications.

The high speed of PHP gives it an advantage over other scripting languages and gives it an application in important administrations such as the server administration and mail functionalities.

Open source

One of the important advantages of PHP is that it is Open Source. Therefore, PHP is readily available and is entirely free. In contrast to other scripting languages used for web development which requires the user to pay for the support files, PHP is open to everyone, anytime and anywhere.

A beginner in PHP need not worry about the support as PHP is maintained and developed by a large group of PHP developers which helps in creating support community of PHP that helps people in PHP implementation and manipulation.

* 1. **DATABASE MANAGEMENT SYSTEM**

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified -- and the database schema, which defines the database’s logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform administration procedures. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

**1.6.1 ADVANTAGE FOR DBMS**

Central storage and management of data within the DBMS provides:

•Data abstraction and independence

•Data security

•A locking mechanism for concurrent access

•An efficient handler to balance the needs of multiple applications using the same data

•The ability to swiftly recover from crashes and errors, including re startability and recoverability

•Robust data integrity capabilities

•Logging and auditing of activity

•Simple access using a standard application programming interface (API)

•Uniform administration procedures for data

**1.5 SQL**

SQL (Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS).SQL offers two main advantages: first, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify how to reach are cord, e.g. with or without an index.

Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language, data manipulation language, and data control language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control.

SQL became a standard of the American National Standards Institute (ANSI) in1986, and of the International Organization for Standardization (ISO) in 1987. Since then, the standard has been revised to include a larger set of features. Despite the existence of such standards, most SQL code is not completely portable among different database systems without adjustments.

### CHAPTER 2

**REQUIREMENTS SPECIFICATION**

A computerized way of handling information and Vehicle details is efficient, organized and time saving, compared to a manual way of doing so and having a single database for multiple Parking Places makes maintenance much easier. This is done through a database driven web application whose requirements are mentioned in this section.

**2.1 OVERALL DESCRIPTION**

A reliable and scalable database driven web application with security features, that is easy to use and maintain is the requisite and is developed on Windows 10 OS.

The purpose of developing this application is to have the stats and data of all the Vehicle of a single brand in one centralized database. This will help keep a record of all the regular customers. All this is implemented in a single web application and used across different Parking places.

**2.2 SPECIFIC REQUIREMENT**

The specific requirements of the Vehicle Parking Management System are stated as follows:

**2.2.1 SOFTWARE REQUIREMENT:**

Web Browser – Google Chrome 62.0.3202.89 (stable)

Editor – Microsoft Visual Studio Code v1.29

 XAMPP v7.2.8

XAMPP control panel v3.2.2

phpMyAdmin v4.8.2

PHP v7.2.8

Apache v2.4.34

Operating System – Windows 8 or later

Database Support – MariaDB v10.1.34

Client API library version - mysqlnd 5.0.12-dev – 20150407

**2.2.2 HARDWARE REQUIREMENT:**

Processor –Intel Pentium 4 or higher

RAM – 2 GB (4 GB Recommended)

HDD – 4 GB

Monitor – VGA of 1024x768 screen resolution

Keyboard and mouse

**2.2.3 TECHNOLOGY STACK:**

 HTML provides a means to structure text based information in a document. It allows users to produce web pages that include text, graphics and hyperlinks.

 JavaScript is a scripting language which supports the development of both client and server applications. It is preferred at client side to write programs that can be executed by a web browser within the context of a web page.

 CSS (Cascading Style Sheets) is a style sheet language used for describing the presentation of a document written in a markup language.

 SQL is the language used to manipulate relational databases. It is tied closely with the relational model. It is issued for the purpose of data definition and data manipulation.

 PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for Web development, and also used as a general-purpose programming language.

### CHAPTER 3

### DETAILED DESIGN

### 3.1 SYSTEM DESIGN:

### PHP is written as standard text files with the (.php) extension. PHP files are often saved within a folder in a web server's public directory (or a web root directory). On most systems this will either be named public or public\_html. For example, if a file was saved as index. php in a web root directory, a user could access it by typing http://www.example.org orhttp://www.example.org/index.php

### 

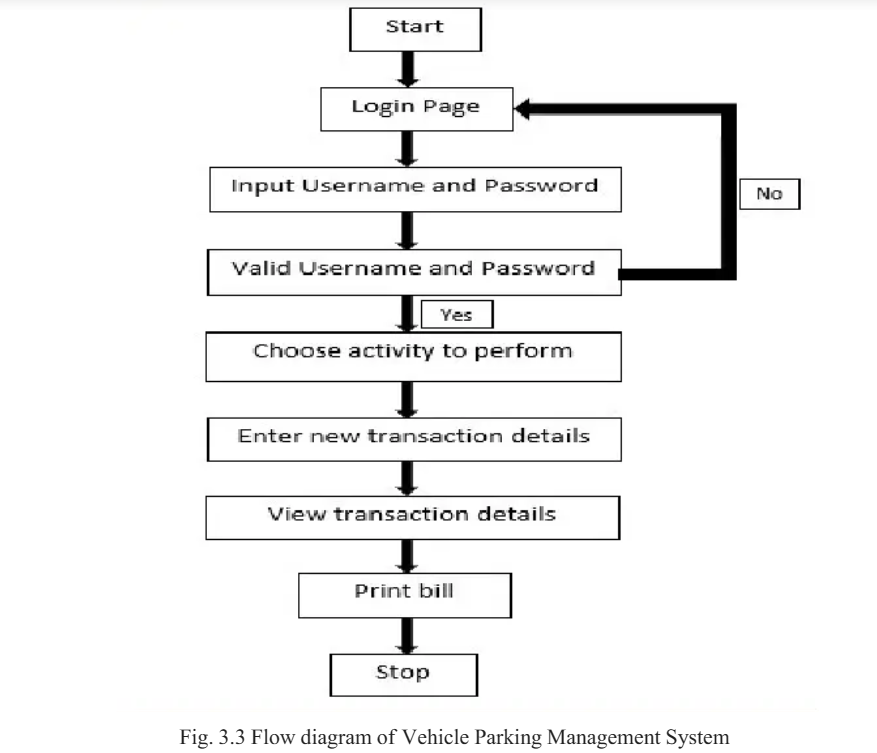
### So what exactly is happening when a user types in the URL http://example.org?When a user types in http://example.org in a Web client (a browser, for instance), the clientissues a GET request to the server (let's assume that we are both using Apache). WhenApache gets this request, it looks for a file named index.php (or index.html, remember thedirectory indexes from earlier?). If a file named index.php is found, Apache essentially says"Hey, this is a PHP file because it has the .php extension. I am going to give this to the PHPinterpreter".After Apache decides that is a PHP file, it gives it to the PHP interpreter. WhenPHP receives the file, it reads through it and executes any PHP code it can find. After it is done with the file, the PHP interpreter gives the output of the code, if any, back to Apache. When Apache gets the output back from PHP, it sends that output back to a browser which renders it to the screen

**3.2 ENTITY RELATIONSHIP DIAGRAM**

An entity–relationship model is usually the result of systematic analysis to define and describe what is important to process in an area of a business. An E-R model does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models.

**3.3FLOW DIAGRAM**

A data flow diagram is a graphical representation of the "flow" of data through information system, modeling its process aspects.

****

**3.4 DESCRIPTION OF TABLES**

The database consists of three tables:

**Table 1: Admin Table**

ID – admin id

AdminName - holds the name of the admin

UserName - admin’s username for login

MobileNumber - mobile number of admin

Email - email-id of the admin

Password - admin’s password used for login

AdminRegDate - date of when admin has been added

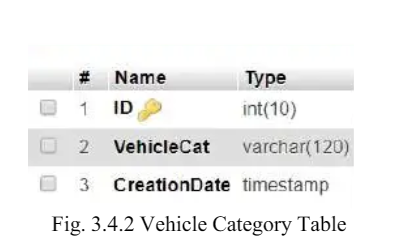
****

**Table 2: Vehicle Category Table**

ID – creates an ID for the vehicle

VehicleCat – holds all the details of different categories of vehicles

CreationDate – holds the details when a category of vehicle was created

****

**Table 3: Vehicle Details Table**

ID – vehicle id

Parking Number – a number provided for each vehicle parked

Vehicle Category – category of vehicle being parked

Vehicle Company Name – Company/ brand of vehicle being parked

Registration Number – vehicle number

Owner Name – name of the vehicle owner

Owner Contact Number – mobile number of the vehicle owner

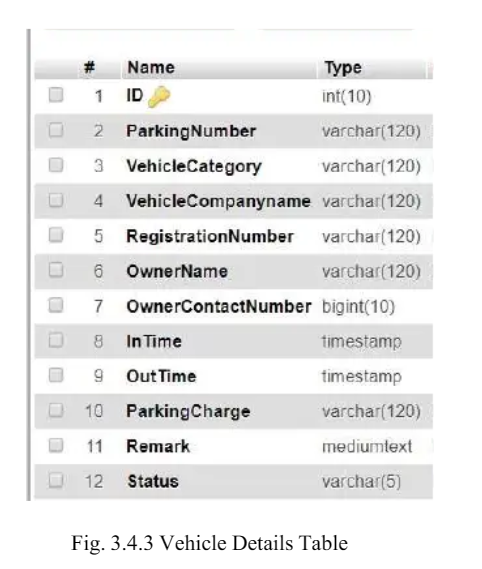
InTime – time when vehicle is entered

OutTime – time when vehicle has made an exit

Parking Charge – the parking fee collected for the vehicle

Remark – notes related to vehicle

Status – tells if vehicle is parked or exited

****

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 MODUES AND THEIR ROLES**

Website is developed using the following modules. Each module has its own functionality and contributes as a feature for the website.

**Code Snippet – print\_receipt.php**

<?php include 'db\_connect.php' ?>

<?php

$qry = $conn->query("SELECT p.\*,c.name as cname,l.location as lname FROM parked\_list p inner join category c on c.id = p.category\_id inner join parking\_locations l on l.id = p.location\_id where p.id= ".$\_GET['id']);

foreach($qry->fetch\_assoc() as $k => $v){

    $$k = $v;

}

$in\_qry = $conn->query("SELECT \* FROM parking\_movement where pl\_id = $id and status = 1");

$in\_timstamp = $in\_qry->num\_rows > 0 ? date("M d, Y h:i A",strtotime($in\_qry->fetch\_array()['created\_timestamp'])) : 'N/A';

$out\_qry = $conn->query("SELECT \* FROM parking\_movement where pl\_id = $id and status = 2");

$out\_timstamp = $out\_qry->num\_rows > 0 ? date("M d, Y h:i A",strtotime($out\_qry->fetch\_array()['created\_timestamp'])) : 'N/A';

?>

                <p><center><b><large>Parking Ticket</large></b></center></p>

                <hr>

                <h4><b>Parking Reference No. : <?php echo $ref\_no ?></b> </h4>

                            <p>Vehicle Parked Area: <b><?php echo $lname ?></b></p>

                <p>Vehicle Category: <b><?php echo $cname ?></b></p>

                <p>Vehicle Owner: <b><?php echo $owner ?></b></p>

                <p>Vehicle Registranion No.: <b><?php echo $vehicle\_registration ?></b></p>

                <p>Vehicle Brand: <b><?php echo $vehicle\_brand ?></b></p>

                <p>Vehicle Description: <b><?php echo !empty($vehicle\_description) ? $vehicle\_description : "No details entered" ?></b></p>

                <p>Vehicle Parked-In Time Stamp: <b><?php echo $in\_timstamp ?></b></p>

**Code Snippet – print\_checkout\_receipt.php**

<?php include 'db\_connect.php' ?>

<?php

date\_default\_timezone\_set('Asia/Manila');

$qry = $conn->query("SELECT p.\*,c.name as cname,c.rate,l.location as lname FROM parked\_list p inner join category c on c.id = p.category\_id inner join parking\_locations l on l.id = p.location\_id where p.id= ".$\_GET['id']);

foreach($qry->fetch\_assoc() as $k => $v){

$$k = $v;

}

$in\_qry = $conn->query("SELECT \* FROM parking\_movement where pl\_id = '".$\_GET['id']."' and status = 1");

$in\_timstamp = $in\_qry->num\_rows > 0 ? date("Y-m-d H:i",strtotime($in\_qry->fetch\_array()['created\_timestamp'])) : 'N/A';

$out\_qry = $conn->query("SELECT \* FROM parking\_movement where pl\_id = $id and status = 2");

$out\_timstamp = $out\_qry->num\_rows > 0 ? date("M d, Y h:i A",strtotime($out\_qry->fetch\_array()['created\_timestamp'])) : 'N/A';

$ocalc = abs(strtotime($out\_timstamp)-strtotime($in\_timstamp));

$ocalc = ($ocalc / (60\*60));

$c = explode('.',$ocalc);

$calc = $c[0];

if(isset($c[1])){

$c[1] = floor(60 \* ('.'.$c[1]));

$calc = $c[1] >= 60 ? ($calc + $c[1]).':00' : $calc.':'.$c[1] ;

}

?>

<style>

.text-right{

text-align: right;

}

th{

text-align: left;

}

</style>

<p><center><b>Parking Receipt</b></center></p>

<table class="table table-bordered" width="100%">

<tr>

<th>Parking Ref. No</th>

<td class="text-right"><?php echo $ref\_no ?></td>

</tr>

<tr>

<th>Chech-In Timestamp</th>

<td class="text-right"><?php echo $in\_timstamp ?></td>

</tr>

<tr>

<th>Chech-Out Timestamp</th>

<td class="text-right"><?php echo $out\_timstamp ?></td>

</tr>

<tr>

<th>Timestamp Difference</th>

<td class="text-right"><?php echo $calc ." (".(number\_format($ocalc,2)).")" ?></td>

</tr>

<tr>

<th>Vehicle Type Hourly Rate</th>

<td class="text-right"><?php echo number\_format($rate,2) ?></td>

</tr>

<tr>

<th>Amount Due</th>

<td class="text-right"><?php echo number\_format($rate \* $ocalc,2) ?></td>

</tr>

<tr>

<th>Amount Tendered</th>

<td class="text-right"><?php echo number\_format($amount\_tendered,2) ?></td>

</tr>

<tr>

<th>Change</th>

<td class="text-right"><?php echo number\_format($amount\_change,2) ?></td>

</tr>

</table>

**CHAPTER** 5

**TESTING**

**5.1 SOFTWARE TESTING**

Testing is the process used to help identify correctness, completeness, security and qualityof developed software. This includes executing a program with the intent of finding errors.It is important to distinguish between faults and failures. Software testing can provide objective, independent information about the quality of software and risk of its failure tousers or sponsors. It can be conducted as soon as executable software (even if partially complete) exists. Most testing occurs after system requirements have been defined and then implemented in testable programs.

**5.2 MODULE TESTING AND INTEGRATION**

Module testing is a process of testing the individual subprograms, subroutines, classes, or procedures in a program. Instead of testing whole software program at once, module testing recommends testing the smaller building blocks of the program. It is largely white box oriented. The objective of doing Module testing is not to demonstrate proper functioning of the module but to demonstrate the presence of an error in the module. Module testing allows implementing of parallelism into the testing process by giving the opportunity to test multiple modules simultaneously.

### CHAPTER 6

### OUTPUTS

### 6.1 LOGIN PAGE

### 

### Fig.6.1 Login Page

### 6.2 DASHBOARD

### 

### Fig.6.2 Dashboard

### 6.3 VEHICLE CATEGORY

### 

### Fig.6.3 Vehicle Category

### 6.4 VEHICLE DETAILS

### 

### Fig.6.4 Vehicle Details

### 6.5 VEHICLE OUT DETAILS

### 

### Fig.6.5 Vehicle Out Details

### 6.6 PARKING RECEIPT

### 

### Fig.6.6.1 Parking Ticket

### 

### Fig.6.6.2 Parking Receipt

### 6.7 VEHICLE SEARCH

### 

### Fig.6.7 Vehicle Search

### CHAPTER 7

### CONCLUSION

The Vehicle Parking Management System provides easier maintenance of customer data when multiple vehicles are parked. It allows simplified operation and is a time saving platform .The application has been completed successfully and tested with suitable testcases. This is developed using HTML5, CSS, JavaScript, PHP and SQL in Windows environment.

The goals achieved by this project are:

1.Easier administration job

2.User friendly tool

3.Single centralized database

4.Efficient management of records

5.Simplification of operations

### CHAPTER 8

**PROGRAM**

**8.1 parking\_db.sql**

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

START TRANSACTION;

SET time\_zone = "+00:00";

CREATE TABLE `category` (

`id` int(30) NOT NULL,

`name` text NOT NULL,

`rate` double NOT NULL

)

ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

INSERT INTO `category` (`id`, `name`, `rate`) VALUES

(1, 'Car', 50),

(2, 'Motorcycle', 35),

(3, 'Sample vehicle', 50),

(4, 'Vehicle type2', 70),

(5, 'BMW', 60),

(6, 'Hammer ', 55);

CREATE TABLE `parked\_list` (

`id` int(30) NOT NULL,

`category\_id` int(30) NOT NULL,

`location\_id` int(30) NOT NULL,

`ref\_no` varchar(100) NOT NULL,

`vehicle\_brand` varchar(200) NOT NULL,

`vehicle\_registration` varchar(15) NOT NULL,

`owner` text NOT NULL,

`vehicle\_description` text NOT NULL,

`status` tinyint(1) NOT NULL DEFAULT 1 COMMENT '1=in, 2= out',

`amount\_due` double NOT NULL,

`amount\_tendered` double NOT NULL,

`amount\_change` double NOT NULL,

`date\_created` datetime NOT NULL DEFAULT current\_timestamp()

)

ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

INSERT INTO `parked\_list` (`id`, `category\_id`, `location\_id`, `ref\_no`, `vehicle\_brand`, `vehicle\_registration`, `owner`, `vehicle\_description`, `status`, `amount\_due`, `amount\_tendered`, `amount\_change`, `date\_created`) VALUES

(3, 1, 1, '5020555486', 'Ford Mustang', 'CDM-0623', 'John Smith', 'Black', 1, 0, 0, 0, '2020-10-02 11:38:57'),

(5, 1, 1, '4970885858', 'Fortuner', 'GCN-1514', 'Claire Blake', 'White', 2, 137.5, 150, 12.5, '2020-10-02 12:09:10'),

(6, 1, 1, '9428140638', 'Sample', 'WER-7894', 'Sample Only', 'Sample', 2, 123.33333333333, 150, 26.67, '2020-10-02 12:09:56'),

(7, 2, 2, '4033430792', 'asdasdasd', 'qwa-1234', 'ada asd asd', 'asdasda', 1, 0, 0, 0, '2020-10-02 16:26:27'),

(8, 3, 3, '3599556075', 'Sample', 'GCN-2020', 'Sample Only', 'White ', 2, 3.3333333333333, 50, 46.67, '2020-10-03 08:20:22'),

(9, 4, 4, '4099773928', 'Sample', 'ABC-1234', 'George Wilson', 'Black Vehicle', 2, 1.1666666666667, 5, 3.83, '2020-10-03 08:28:44'),

(10, 3, 3, '4113421689', 'bmw', '16637', 'coding', 'simple area simple area simple area simple area simple area', 1, 0, 0, 0, '2023-09-21 15:43:29'),

(11, 1, 1, '2389248473', 'BMW', '17472', 'Co-founder', 'simple areasimple areasimple areasimple areasimple area', 2, 250.83333333333, 525, 274.17, '2023-09-21 15:46:20'),

(12, 6, 6, '2159762569', 'hmr', '53656', 'hmr', 'simple areasimple areasimple areasimple areasimple areasimple areasimple areasimple area', 2, 275.91666666667, 500, 224.08, '2023-09-21 15:51:57');

CREATE TABLE `parking\_locations` (

`id` int(30) NOT NULL,

`location` text NOT NULL,

`capacity` int(11) NOT NULL,

`category\_id` int(30) NOT NULL

)

ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

INSERT INTO `parking\_locations` (`id`, `location`, `capacity`, `category\_id`) VALUES

(1, 'Car Area 1', 10, 1),

(2, 'Area 1', 30, 2),

(3, 'Sample area', 20, 3),

(4, 'Area Block 23', 10, 4),

(5, 'simple area BMW', 60, 5),

(6, 'simple area 42 hmr', 59, 6);

CREATE TABLE `parking\_movement` (

`id` int(30) NOT NULL,

`pl\_id` int(30) NOT NULL,

`status` tinyint(1) NOT NULL DEFAULT 1 COMMENT '1 = in ,2 = out',

`created\_timestamp` datetime NOT NULL DEFAULT current\_timestamp()

)

ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

INSERT INTO `parking\_movement` (`id`, `pl\_id`, `status`, `created\_timestamp`) VALUES

(1, 2, 1, '2020-10-02 11:13:19'),

(2, 3, 1, '2020-10-02 11:31:41'),

(3, 4, 1, '2020-10-02 11:39:37'),

(4, 5, 1, '2020-10-02 12:09:10'),

(5, 6, 1, '2020-10-02 12:09:56'),

(6, 6, 2, '2020-10-02 14:37:00'),

(7, 5, 2, '2020-10-02 14:54:00'),

(8, 7, 1, '2020-10-02 16:26:27'),

(9, 8, 1, '2020-10-03 08:20:22'),

(11, 8, 2, '2020-10-03 08:24:00'),

(12, 9, 1, '2020-10-03 08:28:44'),

(13, 9, 2, '2020-10-03 08:29:00'),

(14, 10, 1, '2023-09-21 15:43:29'),

(15, 11, 1, '2023-09-21 15:46:20'),

(16, 11, 2, '2023-09-21 20:47:00'),

(17, 12, 1, '2023-09-21 15:51:57'),

(18, 12, 2, '2023-09-21 20:52:00');

CREATE TABLE `users` (

`id` int(30) NOT NULL,

`name` text NOT NULL,

`type` tinyint(4) NOT NULL DEFAULT 2 COMMENT '1 = Admin, 2= staff',

`username` varchar(100) NOT NULL,

`password` text NOT NULL

)

ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

INSERT INTO `users` (`id`, `name`, `type`, `username`, `password`) VALUES

(1, 'Administrator', 1, 'admin', '0192023a7bbd73250516f069df18b500'),

(6, 'member', 2, 'staff', '0192023a7bbd73250516f069df18b500');

ALTER TABLE `category`

ADD PRIMARY KEY (`id`);

ALTER TABLE `parked\_list`

ADD PRIMARY KEY (`id`);

ALTER TABLE `parking\_locations`

ADD PRIMARY KEY (`id`);

ALTER TABLE `parking\_movement`

ADD PRIMARY KEY (`id`);

ALTER TABLE `users`

ADD PRIMARY KEY (`id`);

ALTER TABLE `category`

MODIFY `id` int(30) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=7;

ALTER TABLE `parked\_list`

MODIFY `id` int(30) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=13;

ALTER TABLE `parking\_locations`

MODIFY `id` int(30) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=7;

ALTER TABLE `parking\_movement`

MODIFY `id` int(30) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=19;

ALTER TABLE `users`

MODIFY `id` int(30) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=7;

COMMIT;

**REFERENCES**

[1]Randy Connolly, Ricardo Hoar “Fundamentals of Web Development”, 1St Edition,

[2]Robin Nixon “Learning PHP, MySQL, CSS “,4th Edition, O’Reily Publications,2015

[3]https://www.mayflower.com : Content and look and feel.

[4]https://www.lowys.com : Content and look and feel.

[5]https://www.w3schools.com : For Styles and other references.

[6]https://www.tutorialspoint.com : For Syntax and Styles and other reference