

Mohammad Amdadul Huq Teto

CAR RENTAL WEBSITE

Thesis

CENTRIA UNIVERSITY OF APPLIED SCIENCES

Information Technology

June 2020

ABSTRACT

Centria University of Applied Sciences	Date June 2020	Author Mohammad Amdadul Huq Teto
Degree programme Information Technology		
Name of thesis CAR RENTAL WEBSITE		
Instructor Kauko Kolehmainen	Pages 65	
Supervisor Kauko Kolehmainen		
<p>The main purpose of this thesis is to create a website for car renting. This website helps the owner to maintain the car business and the user to book and inquiry about new cars. Both the user and the admin have different rights from their perspectives as well as different credentials.</p> <p>Each prosperous task needs the correct arrangement. This initiative gains experience at the proper management level. Some portion of the theory portrays present-day web innovation on the best way to utilize web components in the web programming language. Another segment portrays the way toward actualizing web improvement with functional information. The use of HTML, CSS, JavaScript, and bootstrap is discussed in user interface processing, while PHP and MySQL are developed on the back. Using all these tools and technology required website can be built.</p> <p>As a result of the thesis, the whole webpage is a proper case of a business website. This can be a guarantee to fabricate a vehicle rental website.</p>		

<p>Key words CSS, HTML, JavaScript, PHP, XAMPP, MYSQL</p>
--

CONCEPT DEFINITIONS

XAMP	Cross-Platform(X), Apache(A), MySQL(M), PHP(P), Perl(P)
HTM	Hypertext Markup Language
CSS	Cascading Style Sheet
WWW	World Wide Web
ERD	entity relationship diagram

ABSTRACT
CONCEPT DEFINITIONS
CONTENTS

1 INTRODUCTION.....	1
2 TOOLS AND TECHNOLOGIES	2
2.1 Frontend Web Development	3
2.1.1 Hypertext Markup Language	5
2.1.2 Cascading Style Sheet	6
2.1.3 Bootstrap.....	7
2.1.4 JavaScript	8
2.2 Backend Web Development	9
2.2.1 PHP.....	11
2.2.2 ASP.NET	11
2.2.3 Node.js	11
2.3 Databases	12
2.3.1 MySQL	12
2.3.2 NoSQL	12
2.4 Content Management System	13
2.4.1 WordPress.....	13
2.4.2 Joomla	13
2.4.3 Drupal.....	14
2.5 WEB SERVER	14
2.5.1 Apache.....	14
2.5.2 Nginx.....	15
2.5.3 XAMPP	15
3 WEBSITE DEVELOPMENT PROCESS	16
3.1 Phases	16
3.2 User Requirements.....	17
3.3 Design	17
3.4 Implementation	19
3.5 Publishing.....	20
4 CAR RENTAL PROJECT.....	21
4.1 User Requirements.....	21
4.1.1 Hardware Requirement.....	23
4.1.2 Software Requirement	23
4.2 Webpage Design	23
4.2.1 Entity Relationship Diagrams	25
4.2.2 Usecase Diagrams.....	26
4.2.3 Activity Diagrams.....	27
4.3 Database Design	29
4.4 Implementation	30
4.4.1 Homepage and sub-pages	32
4.4.2 Login and Registration page	35
4.4.3 Admin Panel	40
4.5 Publishing.....	43

5 CONCLUSION44

REFERENCES.....45

APPENDICES

1 INTRODUCTION

The world is changing in every moment. Everything is changing rapidly in the activities of daily life. Technology and its development is transforming into a new idea and innovation. The best part is that with the help of the internet everything is so easy to get. Website is a part of this modern change. Most of the enterprises in the world are using the websites for fulfilling their goals. Individuals build website with their requirements and goals. This virtual net feels very real for communication too. Whatever the requirement is, what level of database and security needs and what the size of the company is the web development core is the same. However, a medium level of skills is needed for small and medium-sized databased website. Nowadays it is mandatory to have an official website for every company. It is important not only for selling products but also for the future of the company.

This thesis is about creating a dynamic website for renting a car for various purposes. Part two describes the theoretical background of the web project such as web development tools, applications, social media parts as well as why the company should use the website. Chapter three explains the website development process. Chapter four represents the projected website and its summary. Moreover, it explains the whole website, testing, and implementation. Conclusions come respectively with the author knowledge from this thesis.

2 TOOLS AND TECHNOLOGIES

Technology is changing rapidly. People are accustomed to typing code to build systems and applications. So designers need to be able to master IT skills. However, advanced technology makes everything easier. Anyone with high IT skills can create a variety of tools to allow ordinary people to build their technology and applications like technical staff. Technology makes the way of building a website easier too. There are two ways in which a person or company can build a website. One is the traditional method. It is based on language planning. It takes time and it takes a team to solve it. It is very important to have the necessary skills for building a website. Using traditional methods to build website requires strong skills in the use of web programming languages and sometimes even work experience. The second one depends on the development platform. With these features, everyone can create their website online and publish it instantly. If the user does not know how to do it, these workshops offer a variety of templates for their users to choose from. Users can add or delete articles, photos, videos, and even make a own site by himself or herself. This method to create a website does not require users to have very powerful knowledge. However, user having skills, he or she will be able to create a website. (w3schools, 2020.)

There are various tools and technologies to create innovations. Today, people are looking at different technologies for innovation. The use of technology has made human life easier. Information transfer and accessibility is easier and more convenient because of the variety of tools and technologies used on the Internet. It makes life easier to access information in the short term. So many different tools and technologies are used to make the website. (Horton & Horton, 2003.) HTML, CSS, BOOTSTRAP, JQUERY, JAVASCRIPT are used as the frontend followed by PHP, and MYSQL are used as backend for making this website dynamically. XAMP, Bracket, and draw.io are used as necessary tools to create this website.

2.1 Frontend Web Development

Frontend web development is used to send queries, requests and receive data from backend systems (techopedia). It refers to the Hypertext Markup Language (HTML), Cascading Style Sheet (CSS), and JavaScript sections. Frontend web development is called the client end-user development system. Frontend development allows the developer to customize the visual elements of a website or an application. The end users can view this encoding (edx). Figure 1 illustrates the wooden structure of an entire front-end programming language with specialized tools and technologies to create the first part of web development. Today people are adopting modern technology. People are living with problems without the use of technology and programmers help to eliminate this problem with the design of new technologies. Besides, they are creating and modernizing present day innovation with the assistance of data innovation skills and web developers are building dynamic websites. (Duckett, 2014.)

Figure 1 demonstrates the front-end structures of a website.

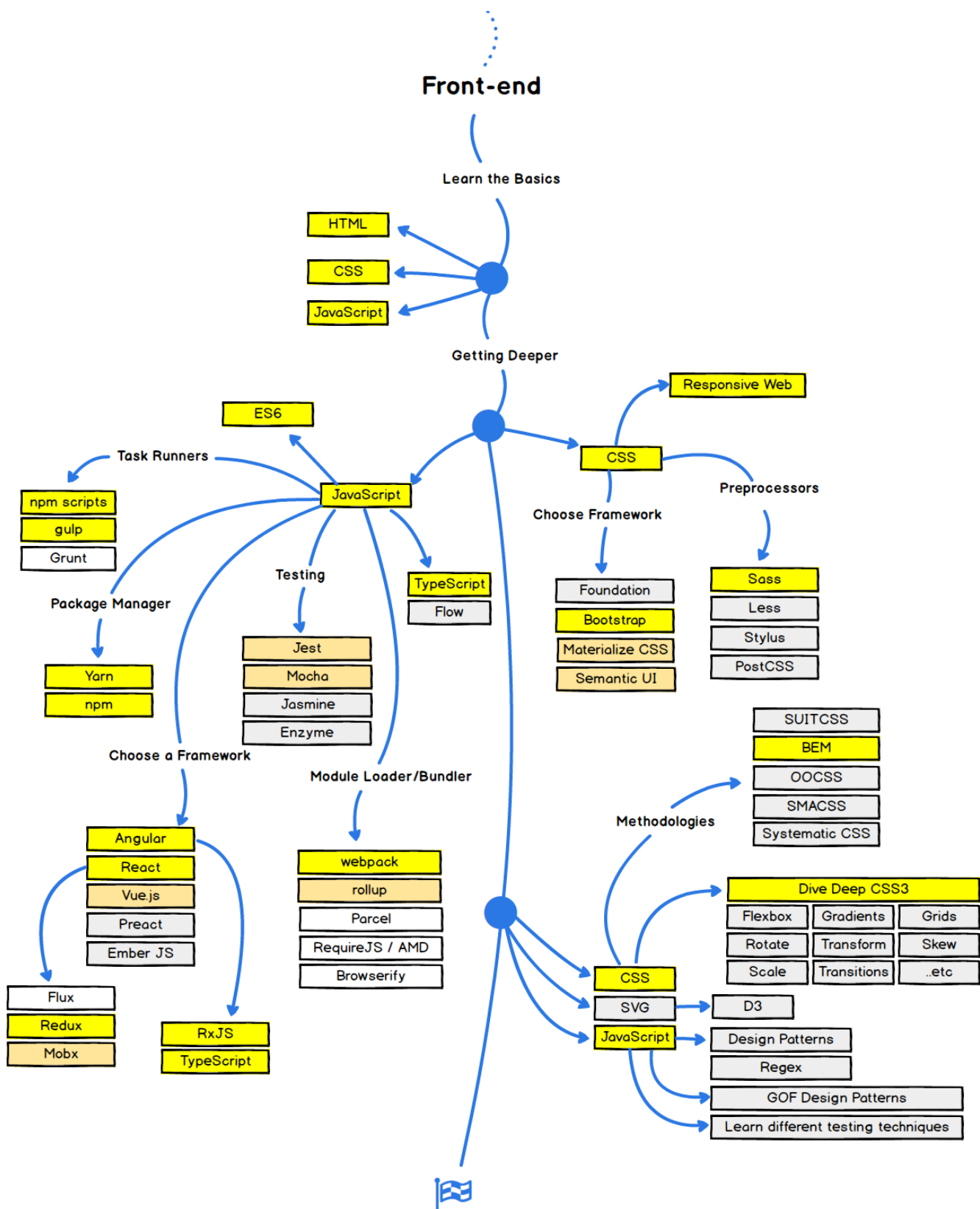


FIGURE 1. Tree-structure of frontend development (codeburst.io, 2020).

2.1.1 Hypertext Markup Language

Hyper markup Language refers to HTML in modern web development. HTML and HTML5 are from the same root but they are different versions. HTML5 is the latest version of HTML. HTML was introduced by the name of Web Application 1.0. The very first version was invented by WHATWG. Later on, it was known as W3C. Where WHATWG stands for Web Hypertext Application Technology Working Group and W3C was the abbreviation of world Web Consortium. It was founded by Opera, Mozilla, and Apple Inc. HTML is the core of the modern website. The platforms used to encode HTML can be a simple text editor with the filename * .HTML. It contains different marking labels. The <Head> tag has a <title> tag and a title, meaning that JavaScript and <style> are related to web page design called CSS. Media and tables are creating to <body> (w3schools, 2020.). The page title must be in the title tag to be displayed in the browser tab (w3schools 2020). HTML is not displayed in the browser, but the content in HTML will be displayed in the browser the same way as embedded in HTML. Figure 2 below shows the structure of the HTML page, showing only the white background in the browser and showing only the components inside the body markup in the browser. (w3schools, 2020.)

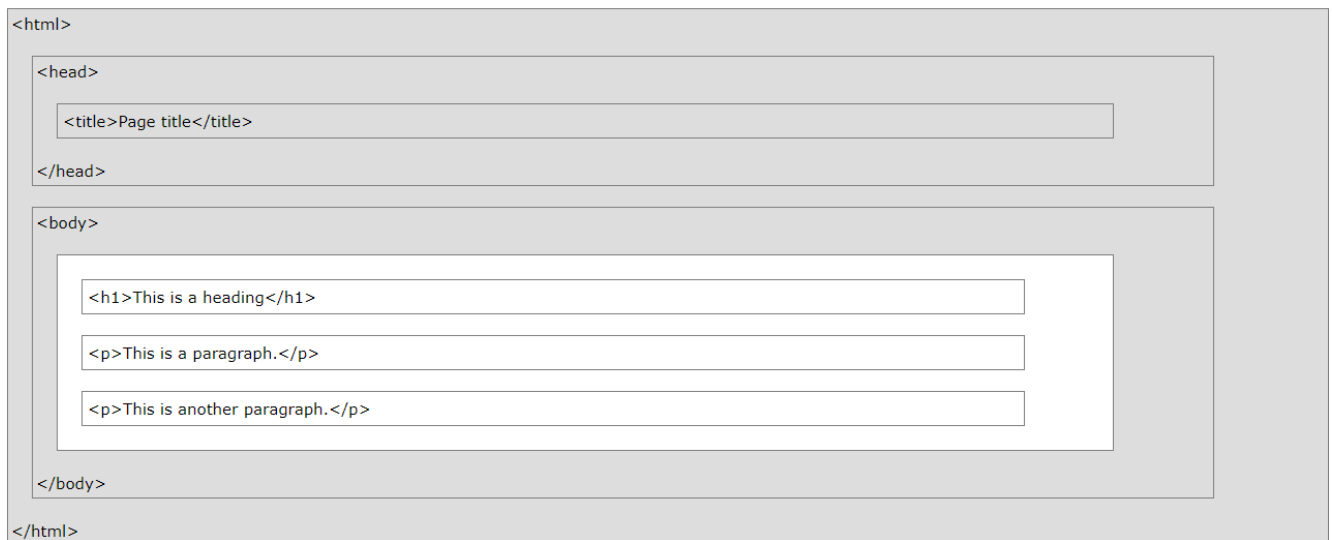


FIGURE 2. HTML page structure (w3schools, 2020).

2.1.2 Cascading Style Sheet

Cascading Style Sheet is known as CSS in its short form. The second magical elements follows by HTML. CSS is used for designing the front-end part of the website. Hakon Wium Lie introduced this form of language to the world in 1994. A CSS syntax consists of a selector and a declaration. CSS is a language that compiles the layout of a web page. CSS3 is the latest CSS standard. The CSS3 rule consists of a selector and a declarer. The picker is the element which can change to HTML style. It can be divided by punctuation. The declaration must contain the name and value of the property. It is separated by a colon. CSS files can be saved not only in HTML but also as separate files. The style tag is used to link the CSS file in the title tag of the HTML page so that it works correctly in the browser. There are two ways to write code. One way to write code is to wrap the HTML header in the tag. Inside the title tag, one can encode the CSS code after the style tag (<style>) opens and closes the style tag (</ style>) after the code is executed. Once one have done that, everyone should see the CSS code output in the browser. Another way to code CSS is to write the CSS code to another file with * .CSS extension, which can be embedded inside the title tag as follows in the body. CSS provides full access to HTML elements for various screen designs such as mobile screens, tablet screens, or desktop screens. This can be called web design, maintenance, color selection, or layout. CSS has been implemented in this thesis to preserve layout, colors, finishes, backgrounds, and layouts. A sample CSS code is shown in Figure 3 below (w3schools, 2020.)

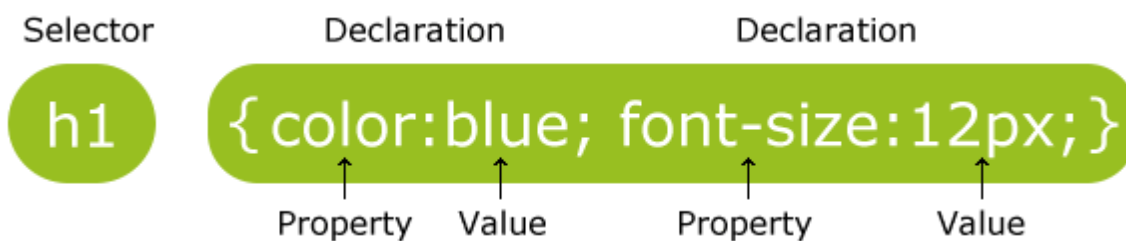


FIGURE 3. Sample code of CSS (w3schools, 2020).

2.1.3 Bootstrap

Bootstrap is a free tool for basic web development. It is more faster and convenient. This is a simpler way to build responsive web design from using design templates like HTML and CSS, as well as for all modern browsers. Bootstrap can be downloaded from CDN (Content Delivery Network). Bootstrap has one of the most well-known terminal structures and open source extends on the planet. Bootstrap was made on Twitter by Mdo and Fat in mid-2010. Before being open source, Bootstrap was called Twitter Blueprint. (w3schools, 2020.) The source code of the bootstrap is given in Figure 4.

```

1 <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
  integrity="sha384-
  J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwjlyYfoRSJoZ+n"
  crossorigin="anonymous"></script>
2 <script
  src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
  integrity="sha384-
  Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
  crossorigin="anonymous"></script>

```

FIGURE 4. Bootstrap link (GetBootstrap 2020)

The links used in Figure 4 are Boot Series 4, the new component of Bootstrap, with a faster design and faster response. They support the latest and most stable versions of all browsers and platforms. Since the first release on Friday, August 19, 2011. It has released twenty versions until now, including two major versions of v2 and v3. With Bootstrap 2, a variety of functions work as optional tables were added. It rewrote the library to create a mobile-friendly code by Bootstrap 3. In Bootstrap 4, there are two major changes to the project architecture like Sass migration and flexbox CSS. The goal is to help web developers move faster by inviting newer CSS features, fewer dependencies, and newer technologies. (Get-Bootstrap, 2020.)

2.1.4 JavaScript

JavaScript is the third part of the magical elements in web development, in short for "JS". For ten years now, it becomes one of the most popular scripting languages. It is the only programming language that can run without having installed in a browser or an extension. Brendan Heart is the creator of JavaScript. It has a big community support. It is not only useful for web applications but also mobile applications. The investment in JavaScript is getting higher in the technological society. It is the most popular and powerful scripting language for programmers due to its simplicity and understanding. Furthermore, everything on the website can be done with the help of JavaScript, such as real-time networking chatting applications, video streaming services, command-line tools, and even games. Also, JS is a versatile script that defines an object as a dynamic first-class function and helps control the operation of a web page. In a nutshell, it can be run inside of a browser or in node as well. This language is useful for building properties that will differentiate the user from a beginner. JS can work in other environments, such as the NodeJS compiler on the remote server or in operating system scripts. The popularity of modern browsers has created a new wave of JavaScript frameworks. Like CSS, JS is encoded in the HTML in the header and can also be used just like CSS. However, the file extension is different, ".js". (w3schools, 2020.)

JavaScript is the most popular of all programming languages and was developed by Brendan Eich on Netscape. Besides, JS is a versatile script that describes the object as a dynamic, first-class function and helps control the behavior of web pages. JS has been extended to use HTML5. It helps the developer to open a new window, to control its appearance, such as print window, resize, scroll, fade and highlight pages new window. The display layout and history objects have been modified by JS and the screen objects as well. Moreover, different types of browser's pop-up boxes, current layout, and cookies are all integrated into JavaScript. (Jsx, 2020.)

2.2 Backend Web Development

Web development is ultimately responsible for aggregating server-side web software and front-side web development work. Final encryption is never seen by end-users or client users. The most useful code is in the background and backend code available on the web. However, web developers code cannot be viewed directly in the backend. A developer working on embedded web development must be familiar with C ++, C #, Java, and PHP known as Hypertext processor or other programming languages. Starting with development, the encoder needs to program the data and services sought by the program or front-end system. Below display in Figure 5 is the tree structure of the whole backend development programming language with tools and technologies. (Dose & Lilja 2020.)

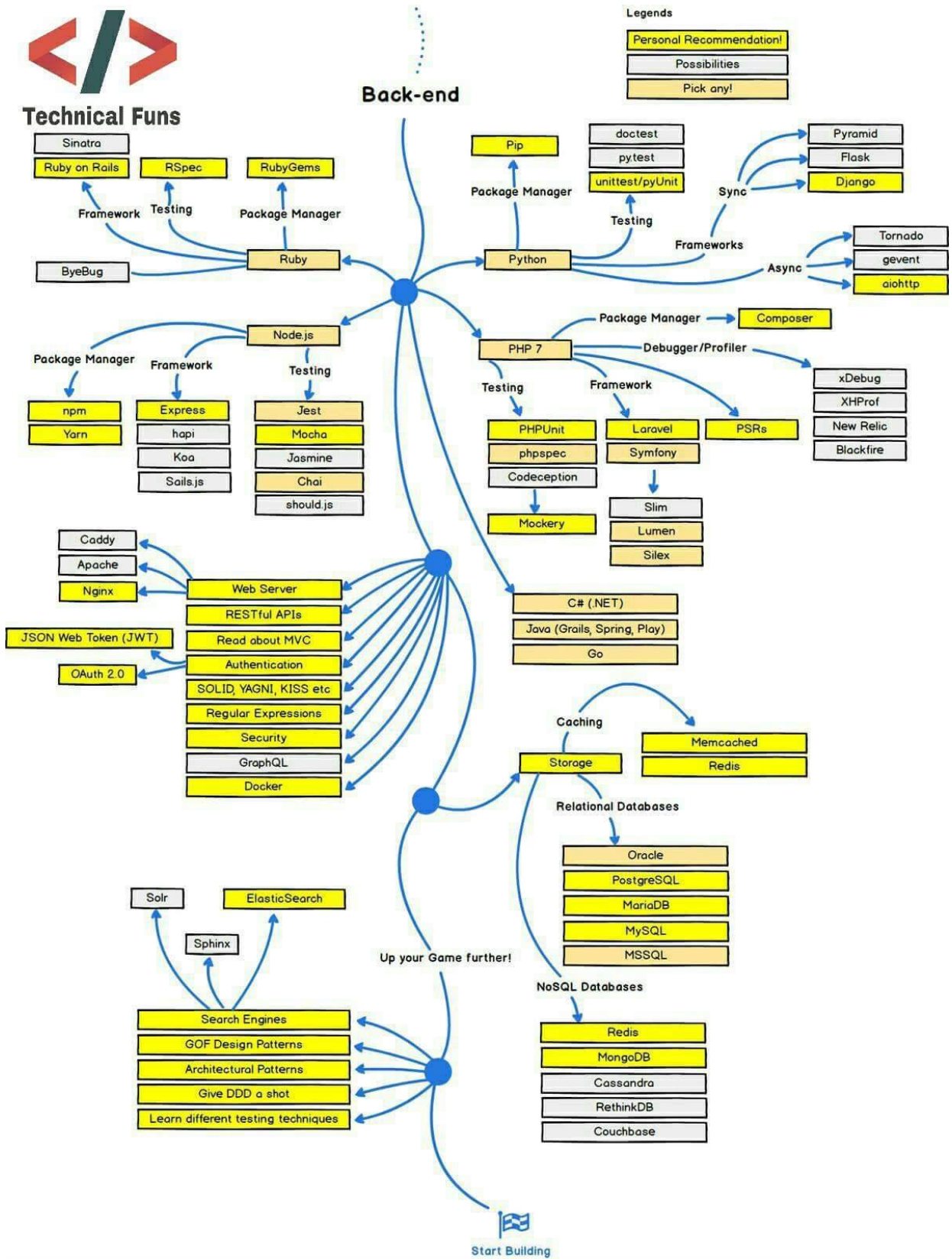


FIGURE 5. Tree-structure of backend development (codeburst.io, 2020).

2.2.1 PHP

PHP is the most powerful scripting language for creating dynamic websites. It helps to provide dynamic functionality, operating requests, and the system to respond automatically to the developer's website. PHP stands for hypertext preprocessor. PHP alone is not an independent language. It is an embedded language for HTML. Thus it is called HTML-embedded scripting language. PHP is almost similar to JAVA, C, and Pearl but it has its special and unique features. The most common use of PHP is to do form processing and math calculation at the backend server. However, it provides the contents result in the browser supporting format such as HTML pages. Moreover, it connects the database dynamically in the webpages (In this thesis mostly the PHP functionality and the integration between PHP were using in the author's website. w3schools, 2020.)

2.2.2 ASP.NET

ASP is a great local and web building structure. ASP means an active server page. With the help of ASP.NET HTML, CSS, and JavaScript, it is easier to build and manage web applications. Moreover, developers are using the mentioned invention of web development to build applications. Inside the extension, ASP.NET, provides engineers with two systems to get started creating web applications: ASP.Net and ASP.NET MVC web forms. ASP.NET Web Forms ASP.NET is the most experienced web design system. The application was deleted on January 16, 2002. After that ASP.NET MVC is a renewed system that was first released in December 2009 as ASP.NET MVC 1.0. When ASP.NET has become the most used plan Web advertising template. The ASP.NET adaptation time is approaching, there are many rumors about ASP.NET. The following web forms are exempt from the following strikes. Nowadays, it is one of the most popular server-side scripting pages. The requirements for moving from ASP.NET Web Forms to ASP.NET MVC is very high at some point. (Liberty and Herwitz 2003.)

2.2.3 Node.js

Node.js is open source and is used to run JavaScript on a server. It is designed to assemble and build versatile system applications and is ideal for handling a large number of synchronous connections with more capacity. Node.js is used to produce applications that require a reliable connection to the server. Node.js is often the most important decision of programming designers chose it to avoid highlighting events such as outdated, randomly managed, fast code execution, single-character design, and buffering. In Node.js, JavaScript is used to build the program in the same way as RESTAPI and all the code base

is integrated into a single programming language for both front and back end. Preferably, the use of Node.js to create CPU-centric applications is very common at present. (Node.js, 2020.)

2.3 Databases

Database means the collection data. Organized data is collected for the record via different kinds of systems. The system for collecting data can be an app or website depending on the management agreement. However, the relevant data is a more restrictive method. SQL and NoSQL are two types of data language systems. SQL is a data language and used in an organized way and users can have access to the data very easily. SQL is a collection of schemas, tables, references, views, and other elements. There are also different types of databases. The database designer supports an IT-based process like search for empty modeling in ways that helps to find model rooms. (Shi, 2007.)

2.3.1 MySQL

SQL is a standard language for accessing and manipulating databases. SQL stands for Structured Query Language. SQL lets the user access and manipulate databases. SQL was a standard of the American National Standards Institute (ANSI) in 1986, and the International Organization for Standardization (ISO) in 1987. SQL can execute queries against a database, retrieve data from a database, insert records in a database, update records in a database, delete records from a database, create new databases, create new tables in a database, create stored procedures in a database, create views in a database and most importantly set permissions on tables, procedures, and views. This is the most well-known database framework utilized in PHP. It is a most famous open-source social information base administration framework. Organized Query Language is the extended version of SQL. Inside the MySQL, information is protected/put away in an assortment of lines and sections. Each site needs to store some valuable information for which they need MySQL. Login page, enlistment page, search page and to store the information, MySQL makes it more convenient. (Welling & Thomson 2003.)

2.3.2 NoSQL

NoSQL database is a database that stores information in positions other than correlation tables. A typical confusion is that a NoSQL database or a non-correspondence database does not store correspondence information well. The NoSQL database can store correspondence information, which is uniquely in con-

trast to a relative database. They do not use the SQL language. Indeed, for some individuals, correspondence information in a NoSQL database is simpler to think about than a SQL database because there is no compelling reason to part the pertinent information between tables. NoSQL information models permit the user to incorporate significant information into a solitary information structure. In the late 2000s, the NoSQL database was used, and right after then, capacity costs started to fall dramatically. It is important to make an unpredictable and hard to oversee data model to diminish information duplication. Engineers enhances the NoSQL database for designer profitability since it was the main expense of the product. (Vaish, Gaurav 2020.)

2.4 Content Management System

The acronym for the content management system is CMS. It is the use of software to create and manage website content. It is a program used to supply a handy way for editing and adding content to web pages. This allows users to view the site code and edit the code as user requirements. The best examples of this are Drupal, Joomla, and WordPress. (Mening 2020.)

2.4.1 WordPress

WordPress is the most mainstream web content administration framework system site with a variety of tools to help the developer create and manage websites. Anyone can create a website using a WordPress website without the coding and design skills to create a website. User does not need to install anything on the computer to run a WordPress website. Just an internet connection is enough to update the website. It is open-source software and the user can choose any design. The selected code is included with the template so it can be customized. WordPress.org is the official WordPress website. Individuals can utilize a free website from Word-Press.com, or they can utilize their website easily. WordPress is an easy way to maintain the entire system of a website. people are using WordPress because of its simplicity and the number of users is increasing rapidly. (Biglione 2016.)

2.4.2 Joomla

Joomla is an open-source platform for creating websites or applications. Joomla uses object-oriented programming (OOP) and PHP techniques. It was introduced in 2005. It is called Mambo and it symbolizes its nature as an open-source project owned by a group of people. This is the second most popular CMS in the technology world. Joomla is also based on PHP and SQL databases. It allows the developer

to perform a variety of tasks, such as editing metadata directly on the front of the page and including automatic mode for entering information. Joomla provides its security extensions and the encoder creates a list of known vulnerabilities. In terms of content management and affordability, Joomla is less popular than WordPress. However, Joomla is well known for its complexity. (Mening 2020.)

2.4.3 Drupal

The most difficult and powerful content management system is called Drupal. The first version of Drupal was introduced in 2001. Recently, it is the third most popular content management system in the technology world. Drupal is the choice of a business website. Drupal encoding is written in PHP. It is designed to perform faster because it uses less hardware than other software. (Mening 2020.)

2.5 WEB SERVER

Web server signifies hardware and software. From the view of hardware, a web server is a PC that stores web server programming and site part records for example HTML archives, pictures, CSS styles, JavaScript documents. It is associated with the Internet and permits the trading of physical information with different gadgets associated with the web. From the view of software, a web server contains in any event, an HTTP server that controls the web as clients. An HTTP server is a server or client-side programming which has URLs (web address) and HTTP.(Mozila.org, 2020.)

2.5.1 Apache

A web server is a server programming intended to run programming that can address the issues of clients on the internet. A web server can be a large container for maintaining a website. The web server forms approaching system demands as indicated by HTTP and other related conventions. Apache is the most generally utilized web server application. Apache has a strong programming library. Apache is a free, open-source application. It cuts away at 67% of all sites on the planet. It is quick, solid, and trustworthy. Apache is a profoundly equipped for addressing the requirements of an assortment of situations utilizing augmentations and modules. Most WordPress facilitating suppliers use Apache as their web server programming. However, WordPress can be a good platform for apache based web server programming. (Sabharwal, Navin 2020.)

2.5.2 Nginx

Nginx was initially made by Igor Sysoev, with its first introduced in October 2004. Igor Sysoev at first considered the product as a response to the C10k issue, which is an issue concerning the presentation issue of taking care of 10,000 simultaneous associations. Nginx knots as "motor ex", is an open-source web server. Moreover, it is an underlying accomplishment as a web server. Nginx is presently additionally utilized as an HTTP reserve, a converse intermediary, and a burden balancer. Some prominent organizations utilizing Nginx in Autodesk, Atlassian, Intuit, T-Mobile, GitLab, DuckDuckGo, Microsoft, IBM, Google, Adobe, Salesforce, VMWare, Xerox, LinkedIn, Cisco, Facebook, Target, Citrix Systems, Twitter, Apple, Intel, and some more. Since its underlying foundations are in execution enhancement under scale, Nginx regularly is better than other well-known web servers in benchmark tests, particularly in circumstances with static substance as well as high simultaneous solicitations, which is the reason Kinsta utilizes Nginx to control its facilitating. Nginx is worked to offer low memory use and high simultaneousness. As opposed to making new procedures for each web demand, Nginx utilizes an off-beat, occasion driven methodology where solicitations are taken care of in a solitary string. With Nginx, one ace procedure can control different laborer forms. The ace keeps up the laborer forms, while the laborers do the genuine handling. Since Nginx is offbeat, each solicitation can be executed by the laborer simultaneously without blocking different solicitations. (Nedelcu 2013.)

2.5.3 XAMPP

XAMP was made by Apache companions. It represents Cross-Platform (X), Apache (A), MySQL (M), PHP (P), and Perl (P). It assists with making a neighborhood web server for engineers and they use it for their testing purposes. Apache, MySQL, and PHP language is used to make a web server. However, every one of those is remembered for a basic XAMPP record. XAMPP similarly deals with Windows, Mac, and Linus because it is a cross-stage item. Clients make a neighborhood test server through XAMPP for testing their site. There are four significant segments in XAMPP. Furthermore, Apache is the most utilized web server application, and web substances are conveyed and handled by this Apache to a PC. Practically 54% of sites are utilizing Apache since it is one of the well-known web servers. Besides, MySQL is a database and a database association required for gathering information to make a web application. It is very useful when it is about making a database utilizing MySQL. It assists with including, expelling and changing information in a web server. Thirdly, PHP is a web programming code. Apache should begin from the XAMPP control board to run PHP code on the webserver. (Mikoluk 2020.)

3 WEBSITE DEVELOPMENT PROCESS

At the beginning of the site development, the required question must be answered therefore the writer can comprehend the significance of the site. The improvement of the site is done gradually and it is about the thought which was required in the readiness of the site. The advancement periods of the site were required previously and during the creation of the site. To make a decent site, introduction and arranging is required before developing. Moreover, data assembling and arranging is a useful and an important stage in the beginning. Without arranging and research, it is difficult to finish the project in time which causes regular mistakes. Finally, the author clarifies each progression in this proposal are maintained properly in the project. (Mugugesan, Deshpande, Hansen & Ginige 2016.)

3.1 Phases

Contribution in the various process must require in the correct continuation of the website. Before beginning with the website, what, why, and how questions must be replied so the task will get the best possible startup before beginning. The author needs to know what addresses will be raised and what is the task. Why addresses will cause the author to comprehend the reason for making the task and how addresses will make the section to assemble everything about the website, so it is simpler to experience the undertaking with fewer mistakes. New thoughts will rise through the what questions followed by why addresses and which will cause the author to comprehend the significance of the undertaking while how addresses will help to achieves view of the arranging and data assembling. (Shelly, Woods & Dorin 2009.)

Foundation needs data and without preparation, one cannot begin with the task. That is the reason gathering data is required before the improvement of the website. Data assembling is the most important part of the commercial website during the project. It is one of the off chance that the designer is confounded in some programming task, at that point the data that is assembled by him will cause the developer to conquer the snags that emerge in the undertaking. Arranging is done after the separation of the different kinds of data. Proper arrangement of each progression in legitimate configuration and arranging of that data ought to require which will lead the project in good format. (Shelly, Woods & Dorin 2009.)

3.2 User Requirements

Data arranging is insufficient for a successful project. Prerequisites are expected to complete the website within the objects and timeframe. Web programming knowledge is required for making the sites are the prerequisites for the development of the site. Different apparatuses and advancements can assist with making the site. Picking of the right software and hardware must be legitimate because more elevated level innovations can provide a better site. Moreover, the requirement of the client will give more concentration and help with the site. (Welling and Thomson, 2003.) The desired website should meet all the prerequisites that clients need while handed to the client and arranged each progression exactly how the client has asked.

UML is a general modeling language. It is used to document the development process. UML stands for Unified Model Language. Simply put, UML is a modern approach to modeling and documentation software. It is the most popular way to model a business process. It is based on a schematic representation of software components. An old proverb says: a picture has a thousand words. Using visuals will help the developer for the better understand the errors and omissions that may occur in an application or business process. UML is the result of environmental documentation and software. In the 1990s, there were many ways to introduce and document software systems. In 1994-1996, UML was developed by three software engineers working on Rational Software, who needed to create a more consistent way to visualize these systems. It was later adopted as a standard in 1997 and has remained standard since several reforms. (Uml Diagram, 2020.)

3.3 Design

Planning of the site includes various stages like server structure, database plan, and website pages structure. These plans make the configuration for the site in the tech world. A site is the assortment of well-structured website pages. Planning the pages includes the structuring of the web pages in an organized way. Developers uses different kind of designs according to the requirements. Below in Figure 6 is the structure of the pages in a site. (Shelly, Woods & Dorin 2009)

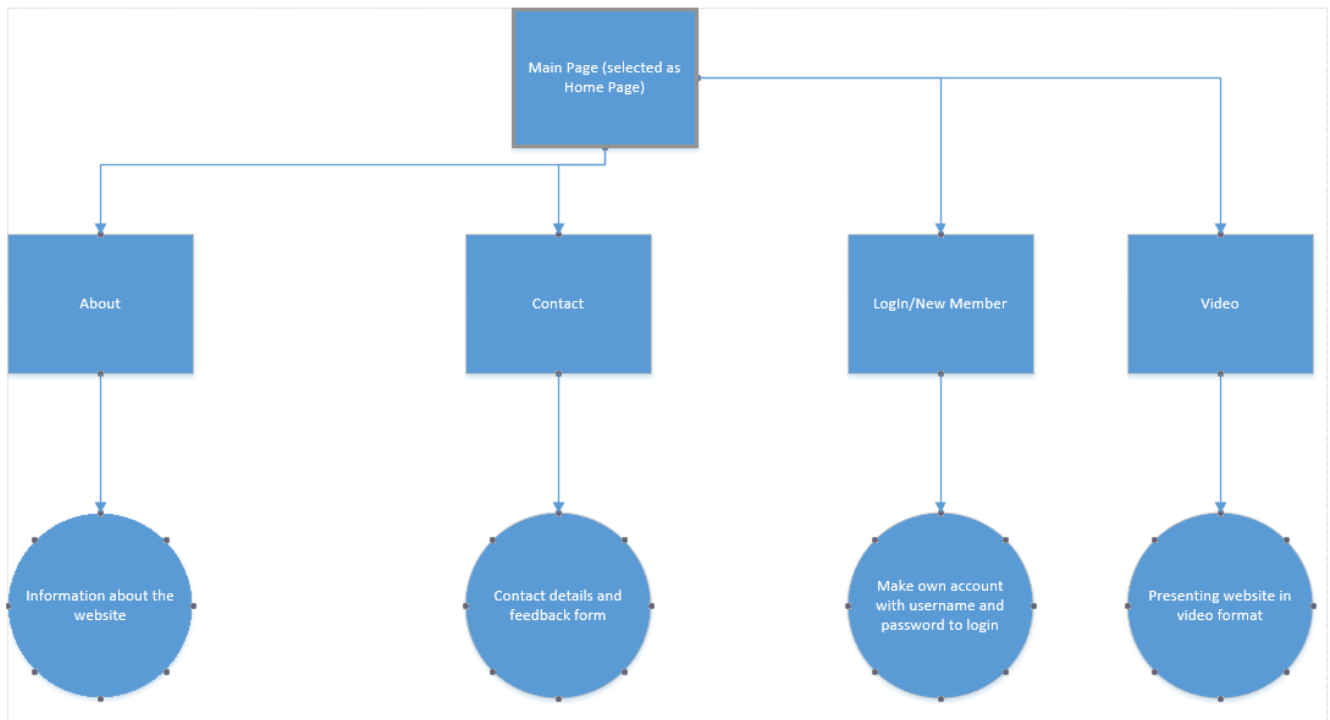


FIGURE 6. designing of a website (Shelly, Woods & Dorin 2009.)

The server configuration is required for publishing a website. To run the site properly, there must be the server. After planning the server, the web architecture and database configuration can be structured. A database is utilized to store data. Structuring of the database is expected to store the data are assembling to the site. Database configuration assists the websites with the client requirement while marking and it likewise having the information for the client for which they pay to the website developer. Figure 7 below is the database plan with its properties. (Shelly, Woods & Dorin 2009.)

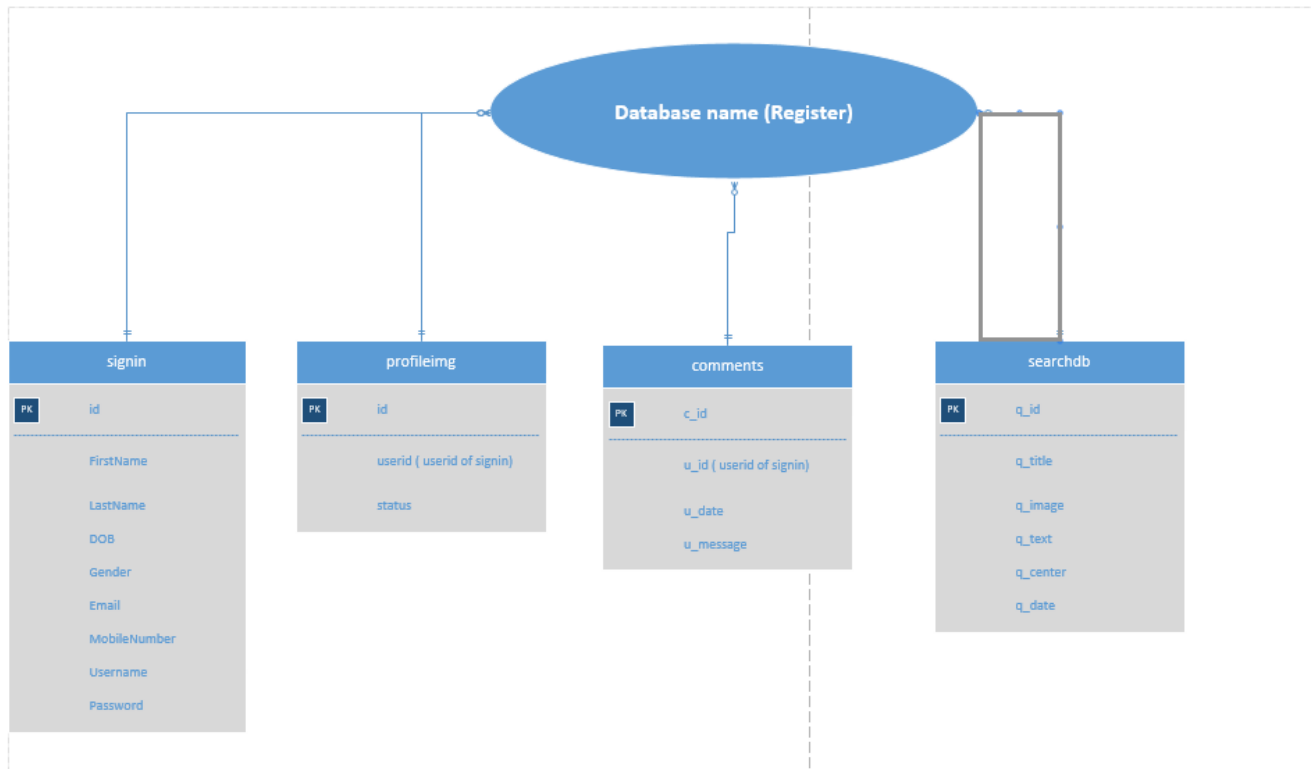


FIGURE 7. Tables and its components(Shelly, Woods & Dorin 2009.)

3.4 Implementation

After completing all the requirements are described above in the above sections is time to implement the website for the improvement of the site and are prepared for the distribution. Finally, it needs to display the exhibitions which are connected with the coding and testing of the site. The word testing itself depicts after the implementation. Site testing is utilized to perceive how the program is functioning. Every possible means of testing is required as it is about the presentation of the site. Developers need to test the site again while coding to see the aftereffect of the code. The execution of the site advancement is a significant assignment for the improvement of the site. At lasts, representing the whole database, server structure, and all subpages plan in a well-structured way is called a proper site implementation. (Sklar 2011)

3.5 Publishing

Even though distributing the site is an advance on the site improvement, it includes various variables for transferring on the web. Publishing of the site needs an internet address or IP address with the goal that anybody can gain admittance to the site from their place which is more convenient for the webserver. To have one's area name, one should get it on the web. One ought to have their area name to distribute the site. Thus, with the user name, one can log in to the site. Another factor for distributing the site online relies on what kind of site is it. On the off chance that it is a static site, Microsoft IIS and Apache are the best apparatuses and on the off chance that it is dynamic, databases are embedded into it. The web server will be able to run the site when it arranges the area names and the IP address. (Sklar 2011)

4 CAR RENTAL PROJECT

This car rental website was created to manage the business process of a car company. This website is aimed to help the customer and the owner. The owner of the company can manage his car and the customer as well as all kinds of arrangements with the customer. This website is on the local server. Moreover, it can work also in the public server when it will be published. This is a sample website with limited cars now. However, owner as admin can add his product or service anytime.

4.1 User Requirements

It is a good framework structured website for managing car rental business. The vehicle rental website gives an easy system of posting and booking cars. HTML, CSS, BOOTSTRAP, JQUERY, JAVASCRIPT is used as the frontend followed by PHP and MYSQL as backend for making this the website.

However, there are many websites out there on the internet for managing the car and tourism business. This website is also working with the same concept in the car management system. Moreover, there is still manual paperwork is an existing system in so many companies. The client must go to the company where the client can get the vehicle on lease and book their vehicle. In the current framework, the client can give testimony after using the service to the administrator on the web. Moreover, the new framework is completely automated. This framework gives highlights like time chart to show vehicle subtleties, client profiles, and the feedback from the customer to the administrator. A new service request can additionally be added by the client on the website. Figure 8 shows the use case diagram for the project.

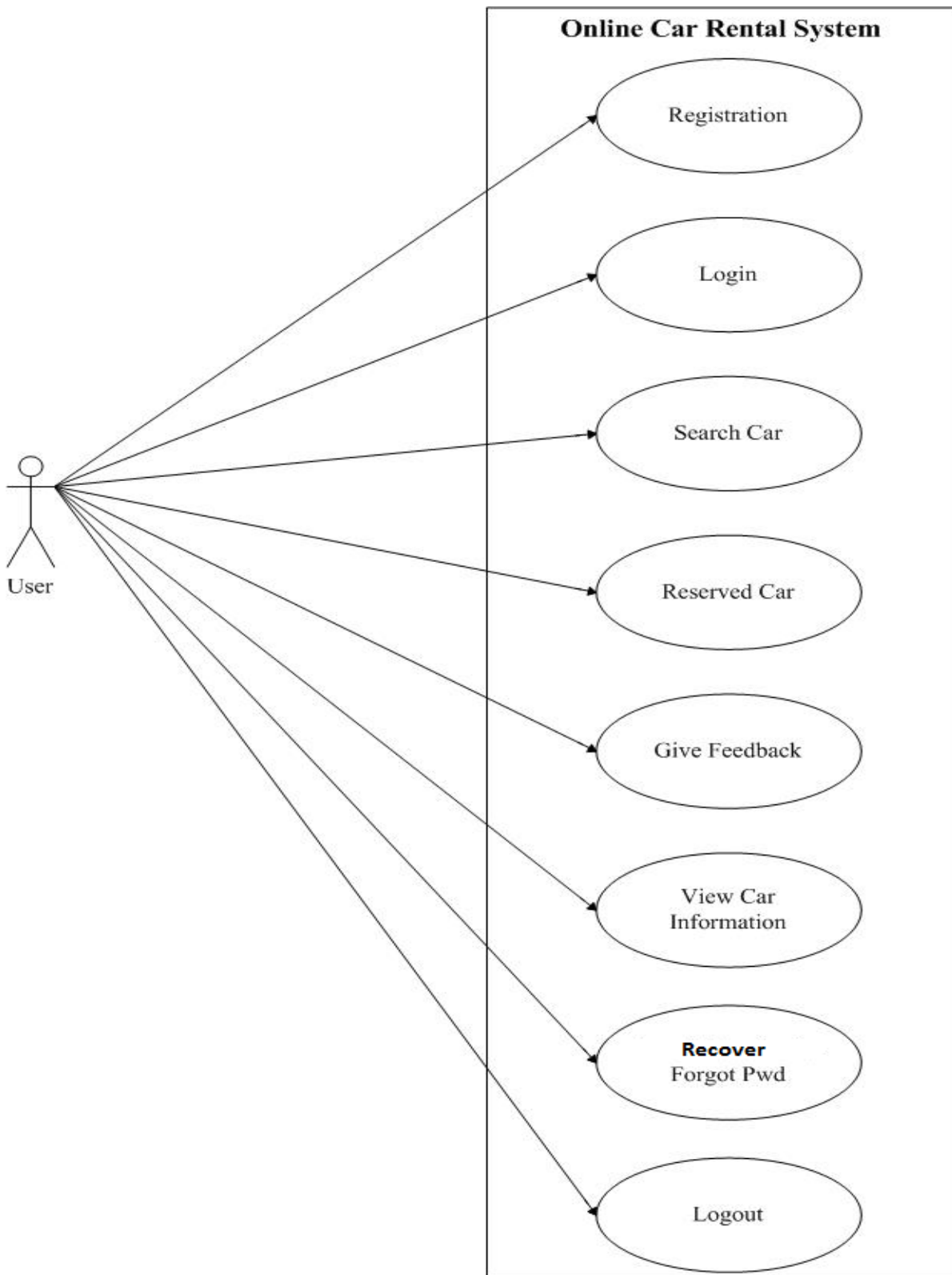


FIGURE 8. Usecase diagram for user

4.1.1 Hardware Requirement

Client-side requires a minimum of 512 Megabytes of RAM(Random Access Memory). Hard disk requirement is 10GB and a minimum level of processor is 1.0 GHz. For instance, to manage server-side needs a minimum of 1 Gigabyte of RAM(Random Access Memory). The hard disk requirement is 20GB and the processor level is minimum 1.0 GHz.

4.1.2 Software Requirement

User needs a web browser like Google Chrome or any compatible browser and operating system such as Windows or Os. The website will run properly after having all the requirements. However, the server-side requirement is high because it is a PHP based site. Web server Apache is mandatory, PHP as a server-side language, and for the database, SQL is needed. The operating system as a running machine and any web browser like Chrome is necessary.

4.2 Webpage Design

In this site administrator is the admin and client functions as user. Moreover, Admin can post a new vehicle, oversee booking vehicle and lease and view input and inquiry. The client can see data of accessible vehicles, booking vehicles, can get the vehicle on lease from the admin, and give feedback and post an inquiry request to the admin. The client can see available vehicles and the client can book the desired vehicle. The client can get the vehicle at whatever requirement they have to on the lease with the help of the website. The client can post testimony to the administrator. The Admin can add new vehicles as the client can get the vehicle according to their choice and requirements, see the accessible vehicles and confirm the booking of the vehicle when the client order. The Admin can deal with the lease so the client can see the bookings of the vehicle. The administrator sees the input and fulfil the inquiry requests from the client. Therefore, there are two designed websites address, one is for the admin and another is for the client user. Figure 9 illustrates the flow of the website below from the start to the end.

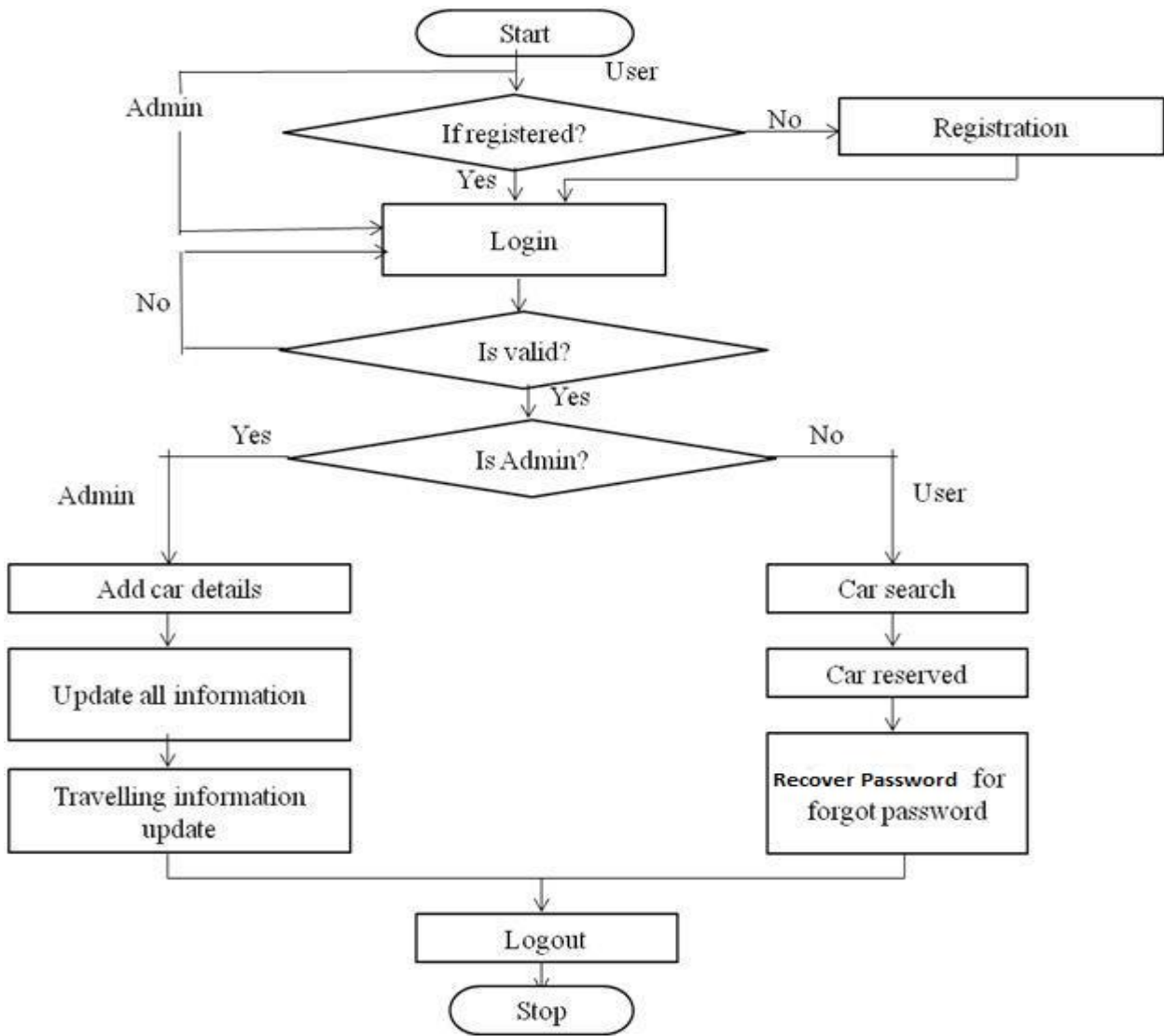


FIGURE 9. Flowchart of the website

4.2.1 Entity Relationship Diagrams

The ER diagram means the entity-relationship diagram in short ERD. It shows all kinds of possible relational data with the user and the admin. Admin has his or her own user name and password. Admin can view the customer request and feedback. Moreover, the admin can send back the answer of the inquiry to the client user. However, the relation between the client and admin is also Figured out by the ERD diagrams. Figure the 10 diagram shows the related branches of the website.

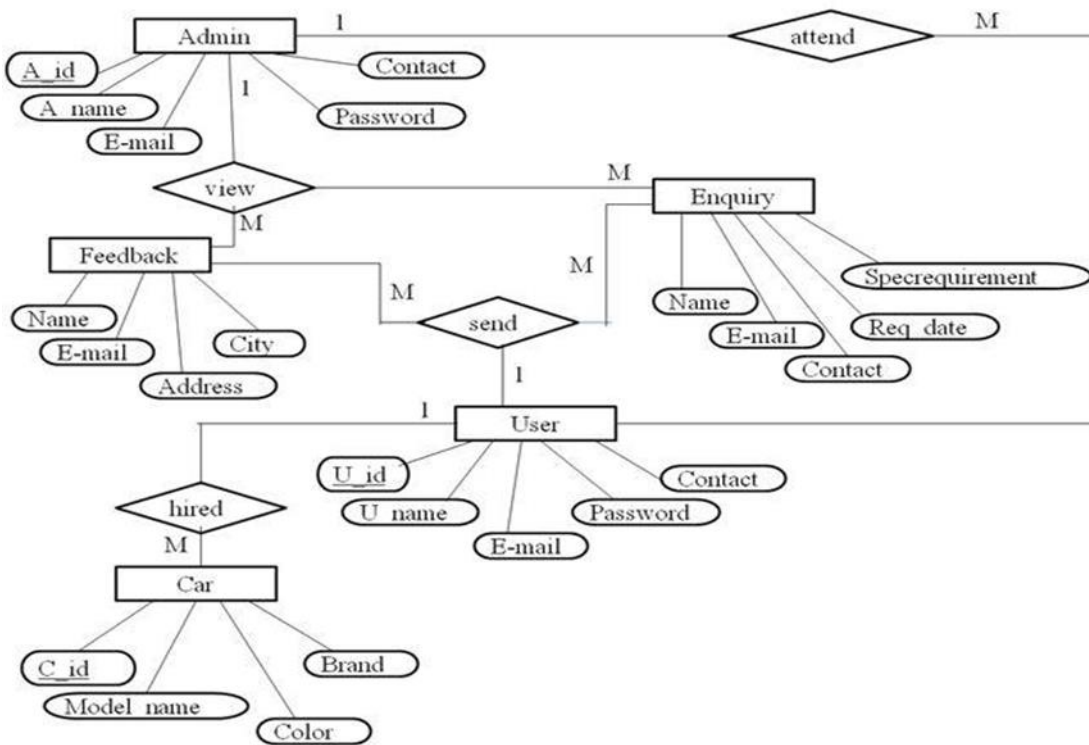


FIGURE 10. ER Diagram

4.2.2 Usecase Diagrams

Admin is the superuser of the website. Figure 11 shows the graphical description of the usecase diagram for the admin. Admin can log in to the system by his or her user name and password. Moreover, the admin can manage the booking and car listing details. Furthermore, the admin can view and post the testimony of the customer in the system as it will show on the website. However, the admin can cancel the booking any time as well.

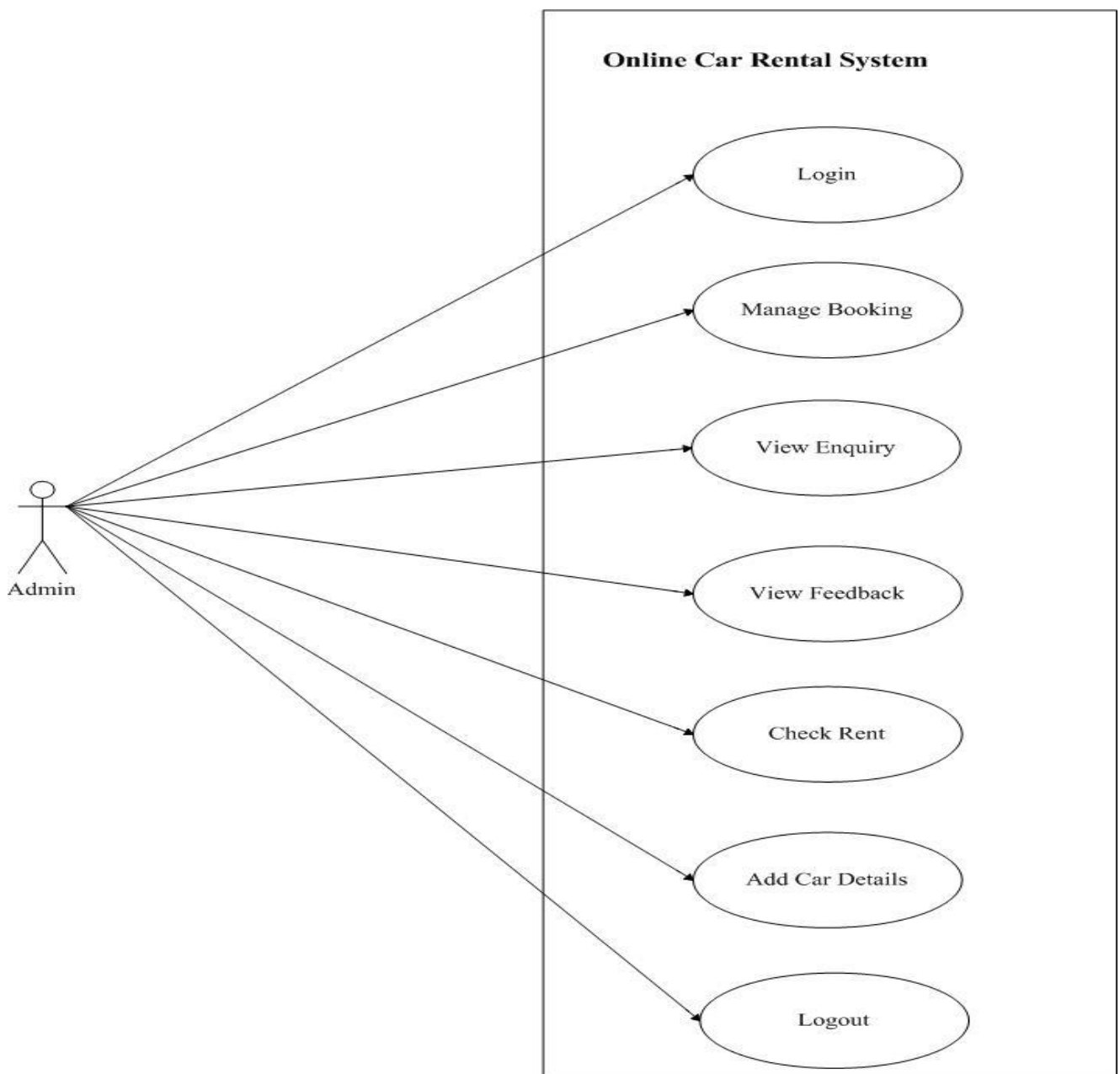


FIGURE 12. Usecase diagram for admin.

4.2.3 Activity Diagrams

Figure 13 shows the graphical description of the activity diagram for the admin panel and Figure 14 shows the graphical description of the activity diagram for the user. Admin and user can log in to the system after having successful input in the forms. The activities of the admin are management of car booking, viewing and posting feedback, pricing the new vehicle according to their value, and adding extra features. However, the most important work is to manage the business for the client by the website and it depends on the activity of the user. The client can log in to the website as a registered user with valid details. Moreover, the user can search for a new car according to the choice and requirements, sees the car information, and reserves a car when needed. Furthermore, the user can give feedback after using the service on-premises and recover his or her password.

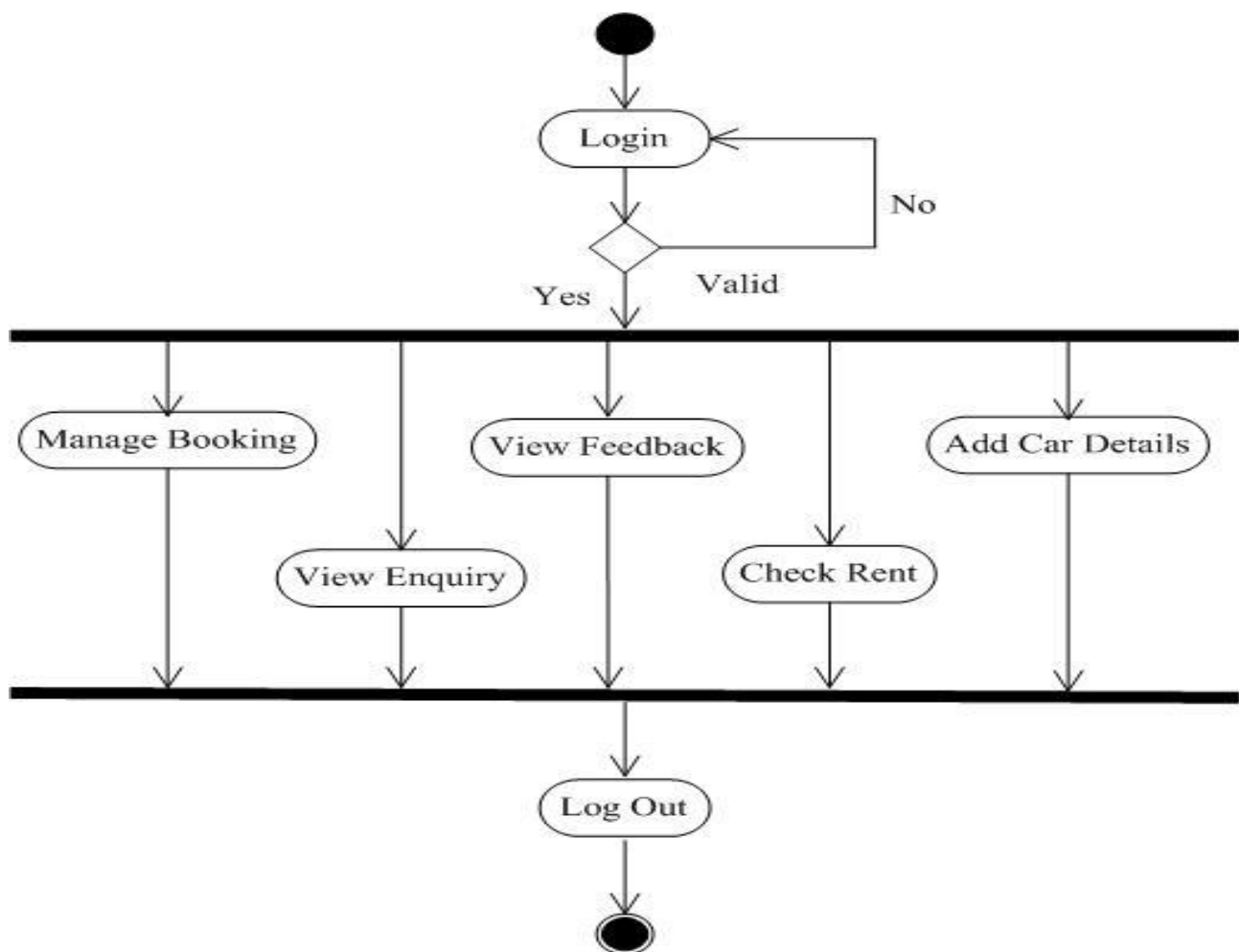


FIGURE 13. Activity Diagram for the admin

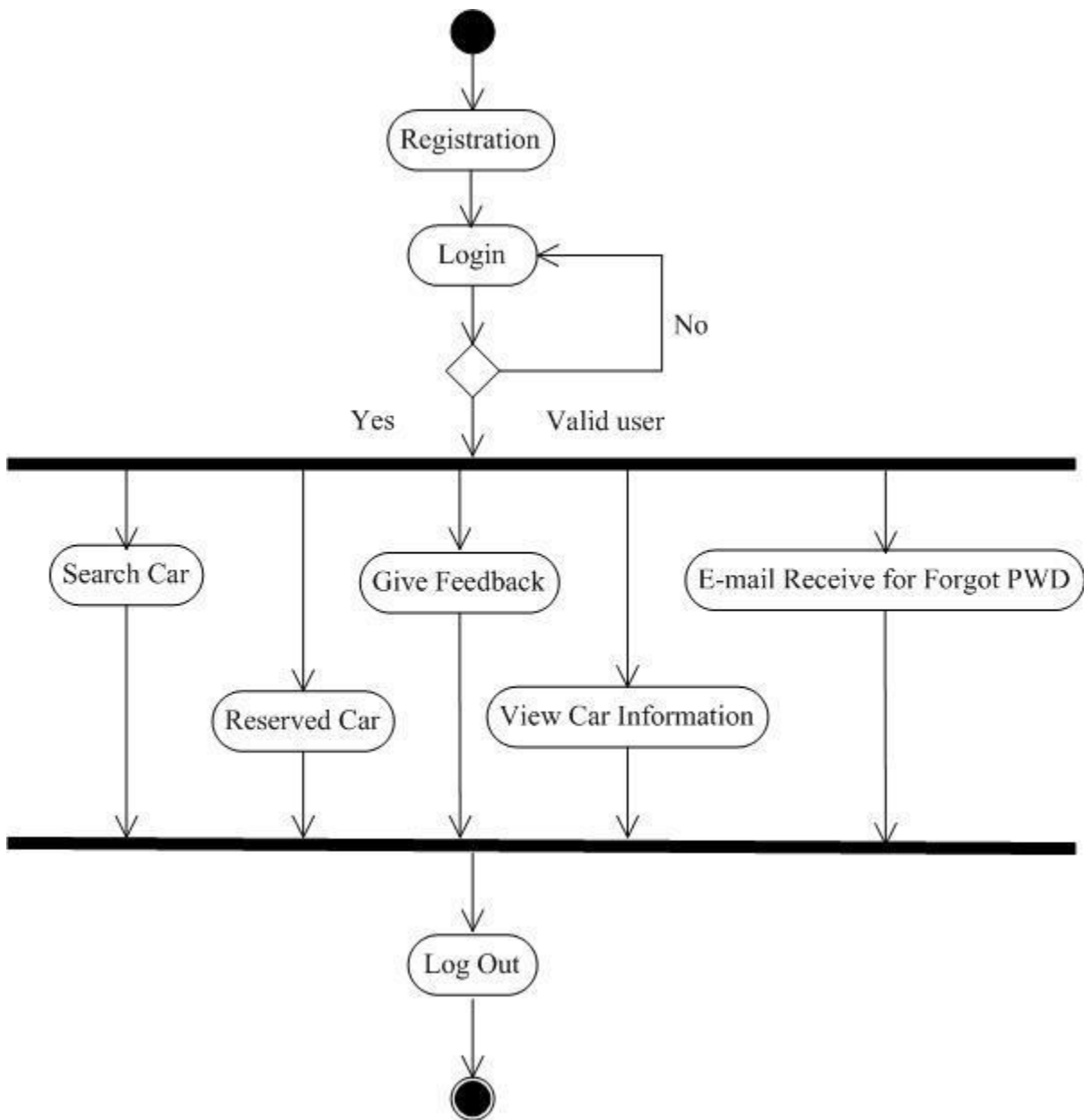


FIGURE 14. Activity Diagram for the user

4.3 Database Design

SQL database tables are sequenced with the order of the data from Figure 15 to Figure 34 in the Appendix. The created tables explain the data hierarchy of the whole project from the admin and the user side. Database tables show the relationship between all the data and it is extremely important to have database design on the website. All the data were arranged in a specific way that both the user and admin can use them when necessary.

Table Name	Admin
Description	This table stores information about Admin
Primary Key	Id
Foreign Key	-

FIGURE 15. Database table

Sr. No	Field Name	Data type (Size)	Constraints	Description
1	id (<i>Primary</i>)	int (11)	Primary Key	It stores Admin id
2	UserName	varchar(100)	Not Null	It stores admin user name
3	Password	varchar(100)	Not Null	It stores the password of Admin
4	updataionDate	Timestamp	NotNull	It stores the profile updating date

FIGURE 16. Database table

4.4 Implementation

The website is fully dynamic. There is no need of knowing coding for the user and the admin. Basic uses of the internet are required for the client. XAMP is used as a software for running the server locally. Bracket is used for viewing and editing the code. Hence, the displayed Figure illustrates the whole project such as the homepage of the web page, car listing, booking, manage testimony as well as inquiry and canceling for the user. Moreover, the dashboard can be used for changing passwords, creating new users, managing dashboards for the admin panel. The graphical interface of XAMP, Bracket, and connected database are snapped in Figures 35 to 37. Figure 35 is the screenshot of XAMP. Figure 36 is the view of the code editor bracket. Bracket is used to write and edit code when necessary. Figure 37 shows the connected database in XAMPP. The whole database was created in XAMPP IDE as well.

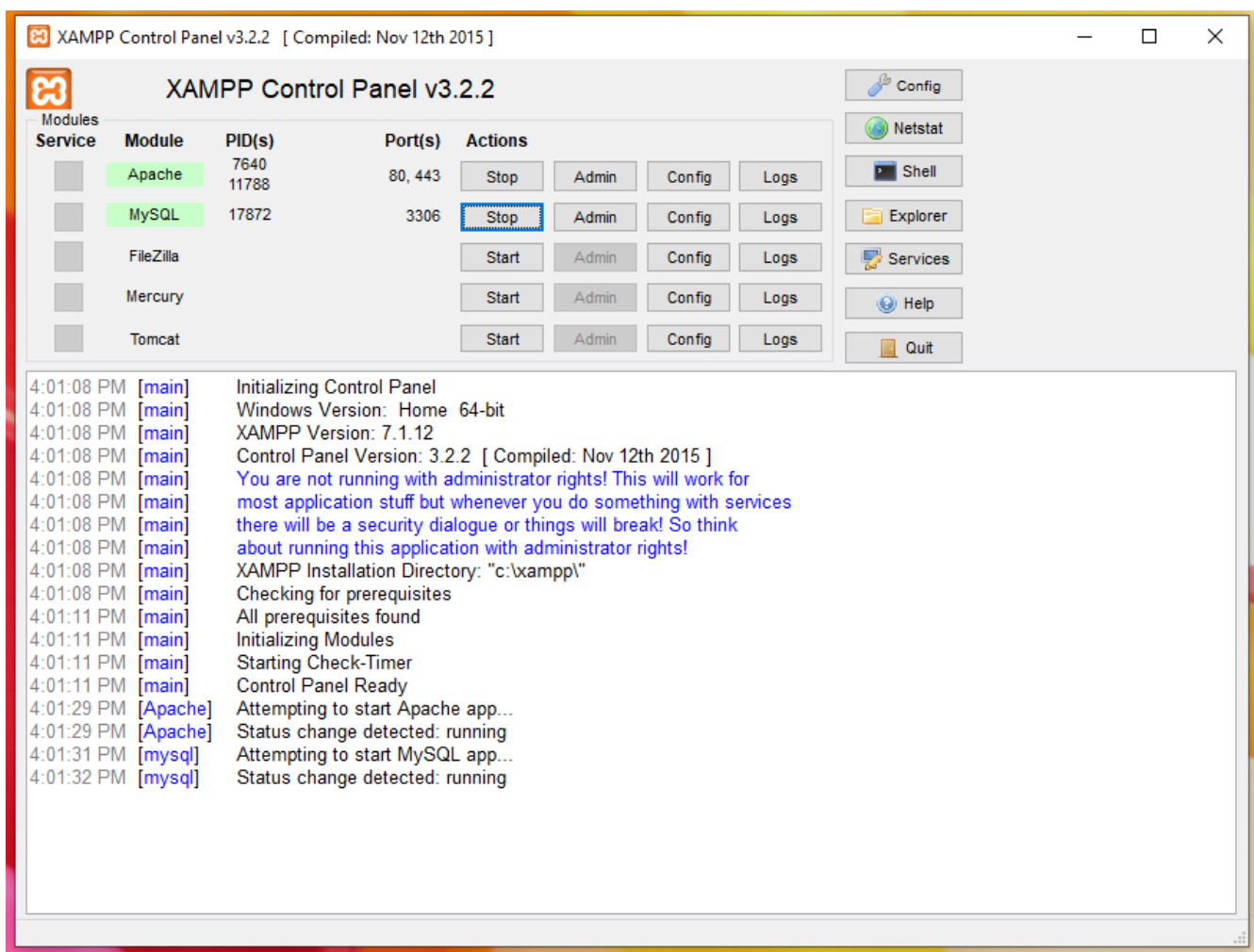


FIGURE 35. Screenshot of XAMP

```

C:/xampp/htdocs/carrental/profile.php (Getting Started) - Brackets
File Edit Find View Navigate Debug Help
Left
logout.php — carrental
my-booking.php
my-testimonials.php
page.php
post-testimonial.php
profile.php
search-carresult.php
update-password.php
vehical-details.php
Right
Getting Started
screenshots
quick-edit.png
index.html
main.css

profile.php
1 |<?php
2 | session_start();
3 | error_reporting(0);
4 | include('includes/config.php');
5 | if(strlen($_SESSION['login'])==0)
6 | {
7 | header('Location:index.php');
8 | }
9 | else{
10 | if(isset($_POST['updateprofile']))
11 | {
12 | $name=$_POST['fullname'];
13 | $mobilen=$_POST['mobilenumber'];
14 | $dob=$_POST['dob'];
15 | $address=$_POST['address'];
16 | $city=$_POST['city'];
17 | $country=$_POST['country'];
18 | $email=$_SESSION['login'];
19 | $sql="update tblusers set
    FullName=:name,ContactNo=:mobilen,dob=:dob,Address=:address,City=:city,Coun
    ry=:country where EmailId=:email";
20 | $query = $dbh->prepare($sql);
21 | $query->bindParam(':name',$name,PDO::PARAM_STR);
22 | $query->bindParam(':mobilen',$mobilen,PDO::PARAM_STR);
23 | $query->bindParam(':dob',$dob,PDO::PARAM_STR);
24 | $query->bindParam(':address',$address,PDO::PARAM_STR);
25 | $query->bindParam(':city',$city,PDO::PARAM_STR);
26 | $query->bindParam(':country',$country,PDO::PARAM_STR);
27 | $query->bindParam(':email',$email,PDO::PARAM_STR);
28 | $query->execute();
29 | $msg="Profile Updated Successfully";
30 | }
31 |
32 | ?>
33 | <!DOCTYPE HTML>
34 | <html lang="en">
35 | <head>
36 | <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
37 | <meta http-equiv="X-UA-Compatible" content="IE=edge">
38 | <meta name="viewport" content="width=device-width,initial-scale=1">
39 | <meta name="keywords" content="">
40 | <meta name="description" content="">
41 | <title>Car Rental Portal | My Profile</title>
42 | <!--Bootstrap -->
43 | <link rel="stylesheet" href="assets/css/bootstrap.min.css" type="text/css">
44 |
Line 1, Column 1 — 237 Lines
INS UTF-8 PHP Spaces: 4

```

FIGURE 36. Screenshot of Bracket

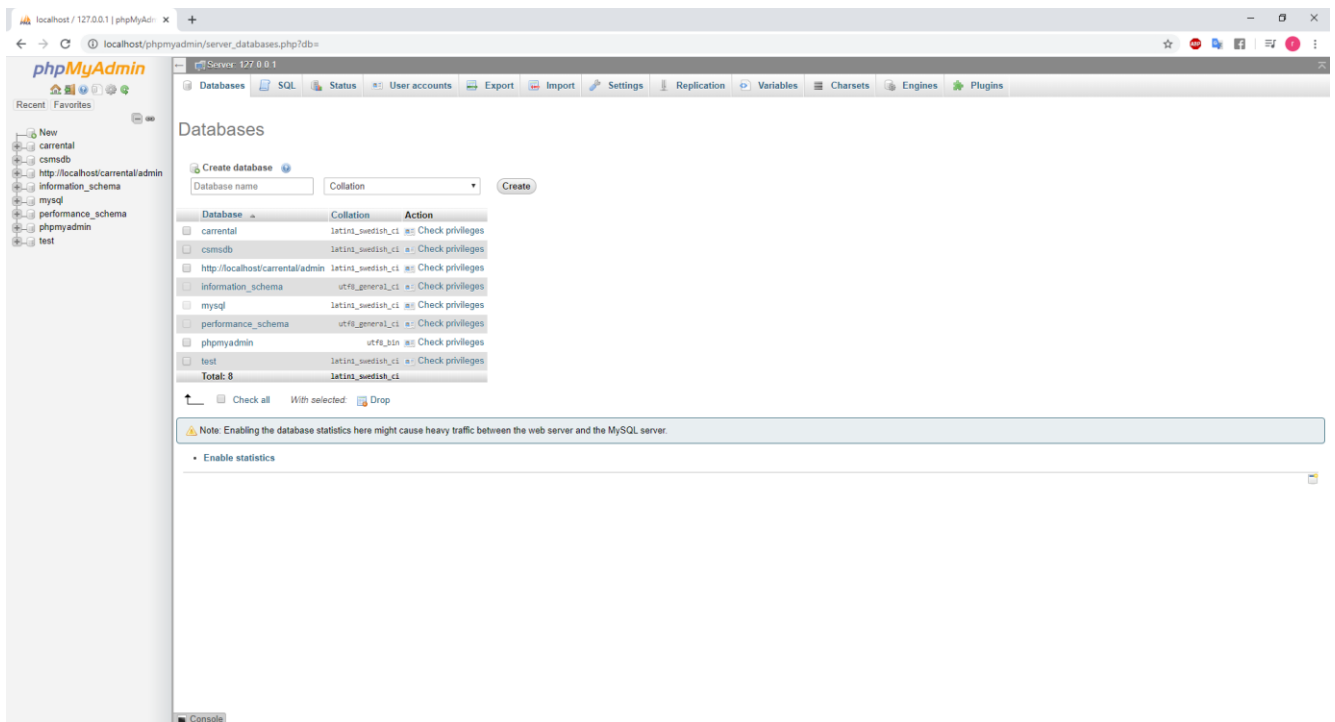


FIGURE 37. Screenshot of SQL database in XAMP

4.4.1 Homepage and sub-pages

The coding of these pages shows in Figures 38, 39 and 40 individually. So when the client login into this site, he or she can see the homepage in the first place. The client can register as a new user and log in to the system from the homepage.

In the home page code, there are referenced JavaScript work which were coded to get click occasion listener which implies when a client clicks in the sidebar, the JavaScript work was being completed and show the sub-pages with the goal that the client can see those pages. Inside the picture tag, the logo of this website was referenced as a picture. There were two catches, one is for login and another is for video when clicked and they open the individual page by coordinating the snap audience utilizing the 'href' to their area. The description of this section image is the homepage code in Figure 38.

```

43 <!-- Navigation -->
44 <nav id="navigation_bar" class="navbar navbar-default">
45 <div class="container">
46 <div class="navbar-header">
47 <button id="menu_slide" data-target="#navigation" aria-expanded="false" data-toggle="collapse"
48 class="navbar-toggle collapsed" type="button"> <span class="sr-only">Toggle navigation
49 </span> <span class="icon-bar"></span> <span class="icon-bar"></span> <span class="icon-bar"></span> </button>
50 </div>
51 <div class="header_wrap">
52 <div class="user_login">
53 <ul>
54 <li class="dropdown"> <a href="#" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
55 <i class="fa fa-user-circle" aria-hidden="true"></i>
56 <?php
57 $email=$_SESSION['login'];
58 $sql ="SELECT FullName FROM tblusers WHERE EmailId=:email ";
59 $query= $dbh -> prepare($sql);
60 $query-> bindParam(':email', $email, PDO::PARAM_STR);
61 $query-> execute();
62 $results=$query->fetchAll(PDO::FETCH_OBJ);
63 if($query->rowCount() > 0)
64 {
65 foreach($results as $result)
66 {
67
68 echo htmlentities($result->FullName); }?><i class="fa fa-angle-down" aria-hidden="true"></i></a>
69 <ul class="dropdown-menu">
70 <?php if($_SESSION['login']){?>
71 <li><a href="profile.php">Profile Settings</a></li>
72 <li><a href="update-password.php">Update Password</a></li>
73 <li><a href="my-booking.php">My Booking</a></li>
74 <li><a href="post-testimonial.php">Post a Testimonial</a></li>
75 <li><a href="my-testimonials.php">My Testimonial</a></li>
76 <li><a href="logout.php">Sign Out</a></li>
77 <?php } else { ?>
78 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">Profile Settings</a></li>
79 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">Update Password</a></li>
80 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">My Booking</a></li>
81 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">Post a Testimonial</a></li>
82 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">My Testimonial</a></li>
83 <li><a href="#loginform" data-toggle="modal" data-dismiss="modal">Sign Out</a></li>
84 <?php ?>
85 </ul>
86 </li>
87 </ul>
88 </div>
89 <div class="header_search">
90 <div id="search_toggle"><i class="fa fa-search" aria-hidden="true"></i></div>
91 <form action="#" method="get" id="header-search-form">
92 <input type="text" placeholder="Search..." class="form-control">
93 <button type="submit"><i class="fa fa-search" aria-hidden="true"></i></button>
94 </form>
95 </div>
96 </div>
97 <div class="collapse navbar-collapse" id="navigation">
98 <ul class="nav navbar-nav">
99 <li><a href="index.php">Home</a> </li>
100
101 <li><a href="page.php?type=aboutus">About Us</a></li>
102 <li><a href="car-listing.php">Car Listing</a>

```

FIGURE 38. code of Homepage

Contact page itself portrays its significance. Moreover, the contact page of this website incorporates the contact subtitles of the page. In the contact page underneath in Figure 39, the email address for the input mail and contact number was given for the assistance of the client.

```

116 <!--Contact-us-->
117 <section class="contact_us section-padding">
118 <div class="container">
119 <div class="row">
120 <div class="col-md-6">
121 <h3>Get in touch using the form below</h3>
122 <?php if($error){?><div class="errorWrap"><strong>ERROR</strong><?php echo htmlentities($error); ?> </div><?php }
123 else if($msg){?><div class="succWrap"><strong>SUCCESS</strong><?php echo htmlentities($msg); ?> </div><?php }?>
124 <div class="contact_form gray-bg">
125 <form method="post">
126 <div class="form-group">
127 <label class="control-label">Full Name <span>*</span></label>
128 <input type="text" name="fullname" class="form-control white_bg" id="fullname" required>
129 </div>
130 <div class="form-group">
131 <label class="control-label">Email Address <span>*</span></label>
132 <input type="email" name="email" class="form-control white_bg" id="emailaddress" required>
133 </div>
134 <div class="form-group">
135 <label class="control-label">Phone Number <span>*</span></label>
136 <input type="text" name="contactno" class="form-control white_bg" id="phonenumber" required>
137 </div>
138 <div class="form-group">
139 <label class="control-label">Message <span>*</span></label>
140 <textarea class="form-control white_bg" name="message" rows="4" required></textarea>
141 </div>
142 <div class="form-group">
143 <button class="btn" type="submit" name="send" type="submit">Send Message <span class="angle_arrow">
144 <i class="fa fa-angle-right" aria-hidden="true"></i></span></button>
145 </div>
146 </form>
147 </div>
148 </div>
149 <div class="col-md-6">
150 <h3>Contact Info</h3>
151 <div class="contact_detail">
152 <?php
153 $pagetype=$_GET['type'];
154 $sql = "SELECT Address,EmailId,ContactNo from tblcontactusinfo";
155 $query = $dbh -> prepare($sql);
156 $query->bindParam(':pagetype', $pagetype, PDO::PARAM_STR);
157 $query->execute();
158 $results=$query->fetchAll(PDO::FETCH_OBJ);
159 $cnt=1;

```

FIGURE 39. code of contact page

In this task, the about page portrays this page just as how this site can be utilized to include in the business. Everything about this page is coded inside the section tag including the rundown label which is pictured beneath in Figure 40.

```

Working Files
├── ...
├── contact-us.php
├── index.php -- carrental
├── logoot.php -- carrental
├── my-booking.php
├── my-testimonials.php
├── page.php
├── post-testimonial.php
├── profile.php
├── search-carresult.php
├── Getting Started
├── screenshots
├── quick-edit.png
├── index.html
└── main.css

```

```

1 <?php
2 session_start();
3 error_reporting(0);
4 include('includes/config.php');
5 ?>
6
7 <!DOCTYPE HTML>
8 <html lang="en">
9 <head>
10 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
11 <meta http-equiv="X-UA-Compatible" content="IE=edge">
12 <meta name="viewport" content="width=device-width,initial-scale=1">
13 <meta name="keywords" content="">
14 <meta name="description" content="">
15 <title>Car Rental Portal | Page details</title>
16 <!--Bootstrap -->
17 <link rel="stylesheet" href="assets/css/bootstrap.min.css" type="text/css">
18 <!--Custom Style -->
19 <link rel="stylesheet" href="assets/css/style.css" type="text/css">
20 <!--OWL Carousel slider-->
21 <link rel="stylesheet" href="assets/css/owl.carousel.css" type="text/css">
22 <link rel="stylesheet" href="assets/css/owl.transitions.css" type="text/css">
23 <!--slick slider -->
24 <link href="assets/css/slick.css" rel="stylesheet">
25 <!--bootstrap slider -->
26 <link href="assets/css/bootstrap-slider.min.css" rel="stylesheet">
27 <!--FontAwesome Font Style -->
28 <link href="assets/css/font-awesome.min.css" rel="stylesheet">
29
30 <!-- SWITCHER -->
31 <link rel="stylesheet" id="switcher-css" type="text/css" href="assets/switcher/css/switcher.css" media="all" />
32 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/red.css" title="red" media="all" data-default-color="true" />
33 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/orange.css" title="orange" media="all" />
34 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/blue.css" title="blue" media="all" />
35 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/pink.css" title="pink" media="all" />
36 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/green.css" title="green" media="all" />
37 <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/purple.css" title="purple" media="all" />
38
39 <!-- Fav and touch icons -->
40 <link rel="apple-touch-icon-precomposed" sizes="144x144" href="assets/images/favicon-icon/apple-touch-icon-144-precomposed.png">
41 <link rel="apple-touch-icon-precomposed" sizes="114x114" href="assets/images/favicon-icon/apple-touch-icon-114-precomposed.html">
42 <link rel="apple-touch-icon-precomposed" sizes="72x72" href="assets/images/favicon-icon/apple-touch-icon-72-precomposed.png">
43 <link rel="apple-touch-icon-precomposed" href="assets/images/favicon-icon/apple-touch-icon-57-precomposed.png">
44 <link rel="shortcut icon" href="assets/images/favicon-icon/favicon.png">
45 <link href="https://fonts.googleapis.com/css?family=Lato:300,400,700,900" rel="stylesheet">
46 </head>
47 <body>
48 <!-- Start Switcher -->
49 <?php include('includes/colourswitcher.php');?>
50 <!-- /Switcher -->
51
52 <!--Header-->
53 <?php include('includes/header.php');?>
54 <?php
55 $pagetype=$_GET['type'];
56 $sql = "SELECT type,detail,PageName from tblpages where type=:pagetype";
57 $query = $dbh -> prepare($sql);
58 $query->bindParam(':pagetype',$pagetype,PDO::PARAM_STR);
59 $query->execute();
60 $results=$query->fetchAll(PDO::FETCH_OBJ);

```

Line 1, Column 1 -- 134 Lines

FIGURE 40. code of pages

4.4.2 Login and Registration page

Clients can register themselves as a new user by providing some information on the login page and have a user name and password for login for the next time. The graphical interface and some coding part is given below from Figure 41 to Figure 45. Figure 41 is the code of the login page. Both the client and the admin can log in to the website via the login page. Users can change the password, add or remove user details as well. Figure 42 is the registration page code for the client. Users can create a new account on the registration page. Figure 43 shows the final view of the homepage. The homepage is the first outlook of the website from where the client can view the whole website. Figure 44 shows the graphical interface for posting a new vehicle. Admin can post or manage vehicles dynamically whenever a new vehicle arrives in the garage. Figure 45 is the graphical view of the testimony page. When a client post feedback on the website, it will show in the website when it is approved by the admin.


```

Working Files
switcher.css
switcher.js
colorswitcher.php
config.php — carrental/includes
footer.php
forgotpassword.php
header.php — carrental/includes
login.php
registration.php
Getting Started
screenshots
quick-edit.png
index.html
main.css

2
3 {
4 $email=$_POST['email'];
5 $password=md5($_POST['password']);
6 $sql ="SELECT EmailId,Password,FullName FROM tblusers WHERE EmailId=:email and Password=:password";
7 $query= $dbh -> prepare($sql);
8 $query-> bindParam(':email', $email, PDO::PARAM_STR);
9 $query-> bindParam(':password', $password, PDO::PARAM_STR);
10 $query-> execute();
11 $results=$query->fetchAll(PDO::FETCH_OBJ);
12 if($query->rowCount() > 0)
13 {
14 $_SESSION['login']=$_POST['email'];
15 $_SESSION['fname']=$results->FullName;
16 $currentpage=$_SERVER['REQUEST_URI'];
17 echo "<script type='text/javascript'> document.location = '$currentpage'; </script>";
18 } else{
19
20     echo "<script>alert('Invalid Details!');</script>";
21 }
22 }
23 ?>
24 <div class="modal fade" id="loginform">
25 <div class="modal-dialog" role="document">
26 <div class="modal-content">
27 <div class="modal-header">
28 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
29 <span aria-hidden="true">&times;</span></button>
30 <h3 class="modal-title">Login</h3>
31 </div>
32 <div class="modal-body">
33 <div class="row">
34 <div class="login_wrap">
35 <div class="col-md-12 col-sm-6">
36 <form method="post">
37 <div class="form-group">
38 <input type="email" class="form-control" name="email" placeholder="Email address*">
39 </div>
40 <div class="form-group">
41 <input type="password" class="form-control" name="password" placeholder="Password*">
42 </div>
43 <div class="form-group checkbox">
44 <input type="checkbox" id="remember">
45 </div>
46 <div class="form-group">
47 <input type="submit" name="login" value="Login" class="btn btn-block">
48 </div>
49 </form>
50 </div>
51 </div>
52 </div>
53 </div>
54 </div>
55 <div class="modal-footer text-center">
56 <p>Don't have an account? <a href="#signupform" data-toggle="modal" data-dismiss="modal">Signup Here</a></p>
57 <p><a href="#forgotpassword" data-toggle="modal" data-dismiss="modal">Forgot Password ?</a></p>
58 </div>
59 </div>
60 </div>
61 </div>

```

FIGURE 41. code of login pages

```

Working Files
switcher.css
switcher.js
colorswitcher.php
config.php -- carrental/includes
footer.php
forgotpassword.php
header.php -- carrental/includes
login.php
registration.php
Getting Started
screenshots
quick-edit.png
index.html
main.css

30 <script>
31 function checkAvailability() {
32   $("#loaderIcon").show();
33   jQuery.ajax({
34     url: "check_availability.php",
35     data: 'emailid='+$("#emailid").val(),
36     type: "POST",
37     success:function(data){
38       $("#user-availability-status").html(data);
39       $("#loaderIcon").hide();
40     },
41     error:function (){}
42   });
43 }
44 </script>
45 <script type="text/javascript">
46 function valid()
47 {
48   if(document.signup.password.value!= document.signup.confirmpassword.value)
49   {
50     alert("Password and Confirm Password Field do not match !!");
51     document.signup.confirmpassword.focus();
52     return false;
53   }
54   return true;
55 }
56 </script>
57 <div class="modal fade" id="signupform">
58   <div class="modal-dialog" role="document">
59     <div class="modal-content">
60       <div class="modal-header">
61         <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>
62         <h3 class="modal-title">Sign Up</h3>
63       </div>
64       <div class="modal-body">
65         <div class="row">
66           <div class="signup_wrap">
67             <div class="col-md-12 col-sm-6">
68               <form method="post" name="signup" onSubmit="return valid();">
69                 <div class="form-group">
70                   <input type="text" class="form-control" name="fullname" placeholder="Full Name" required="required">
71                 </div>
72                 <div class="form-group">
73                   <input type="text" class="form-control" name="mobilenumber" placeholder="Mobile Number" maxlength="10" required="required">
74                 </div>
75                 <div class="form-group">
76                   <input type="email" class="form-control" name="emailid" id="emailid" onBlur="checkAvailability()" placeholder="Email Address" required="required">
77                   <span id="user-availability-status" style="font-size:12px;"></span>
78                 </div>
79                 <div class="form-group">
80                   <input type="password" class="form-control" name="password" placeholder="Password" required="required">
81                 </div>
82                 <div class="form-group">
83                   <input type="password" class="form-control" name="confirmpassword" placeholder="Confirm Password" required="required">
84                 </div>
85                 <div class="form-group checkbox">
86                   <input type="checkbox" id="terms_agree" required="required" checked="">
87                   <label for="terms_agree">I Agree with <a href="#">Terms and Conditions</a></label>
88                 </div>
89                 <div class="form-group">

```

FIGURE 42. code of registration page

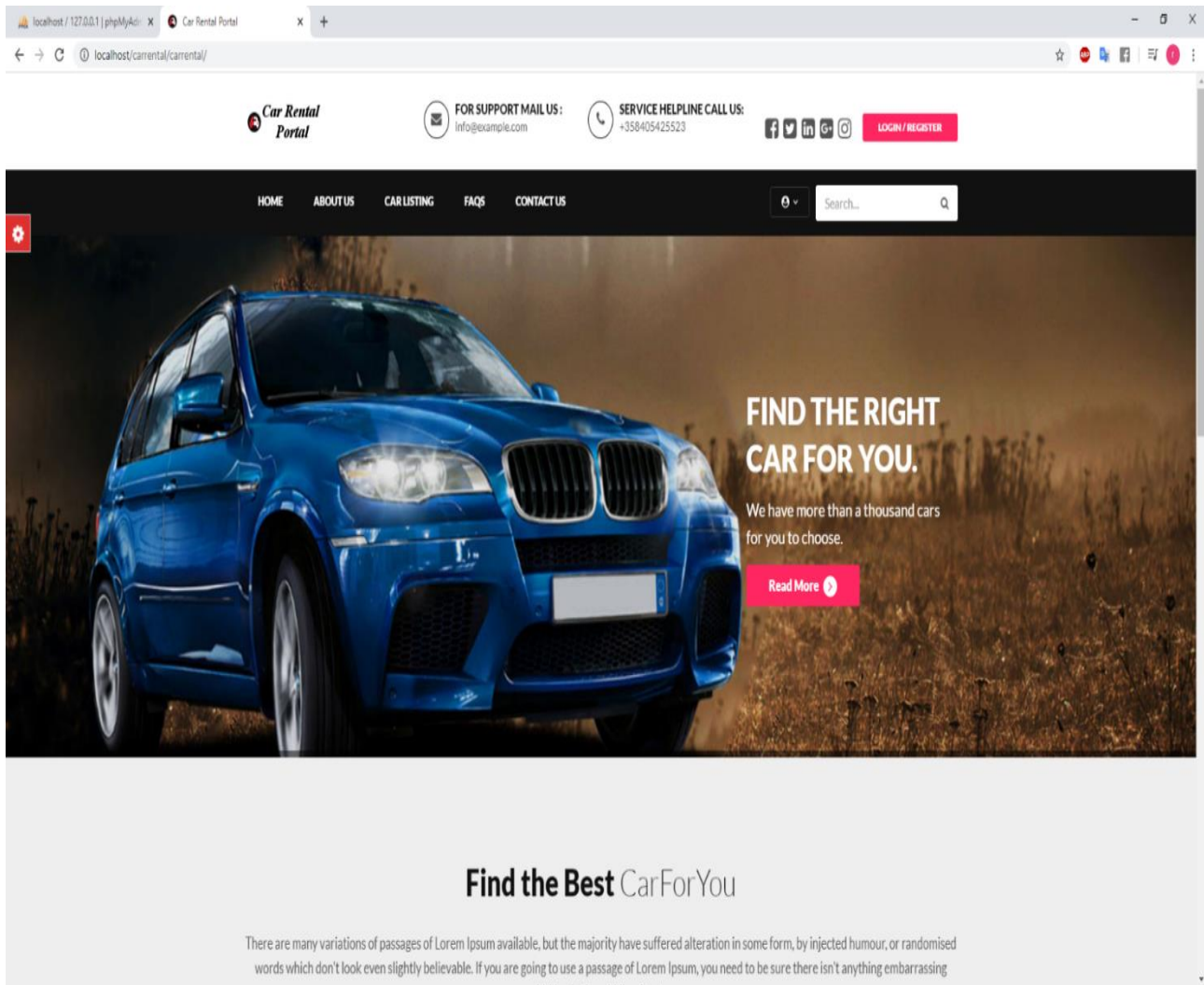


FIGURE 43. Screenshot of homepage

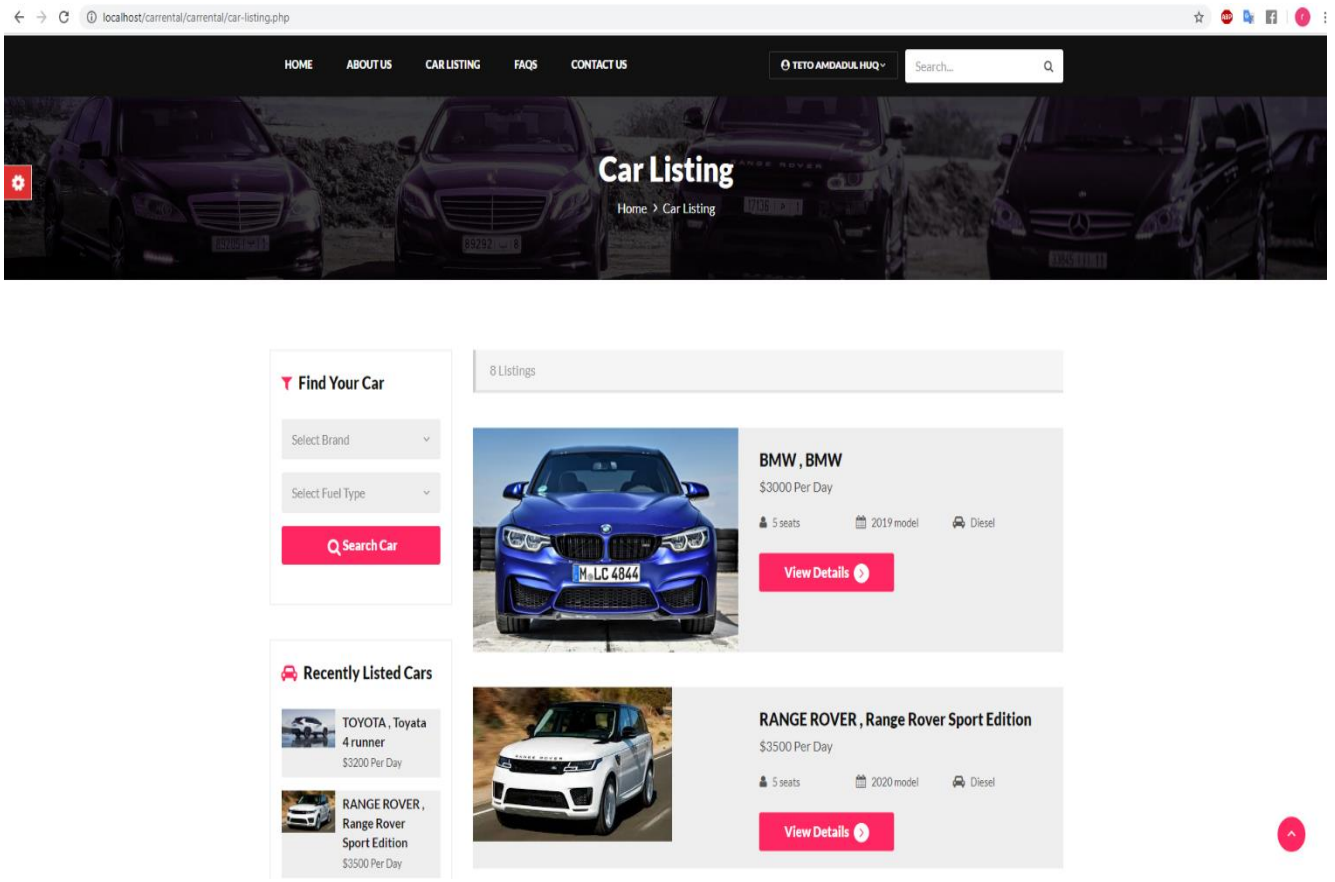


FIGURE 44. Screenshot of car listing

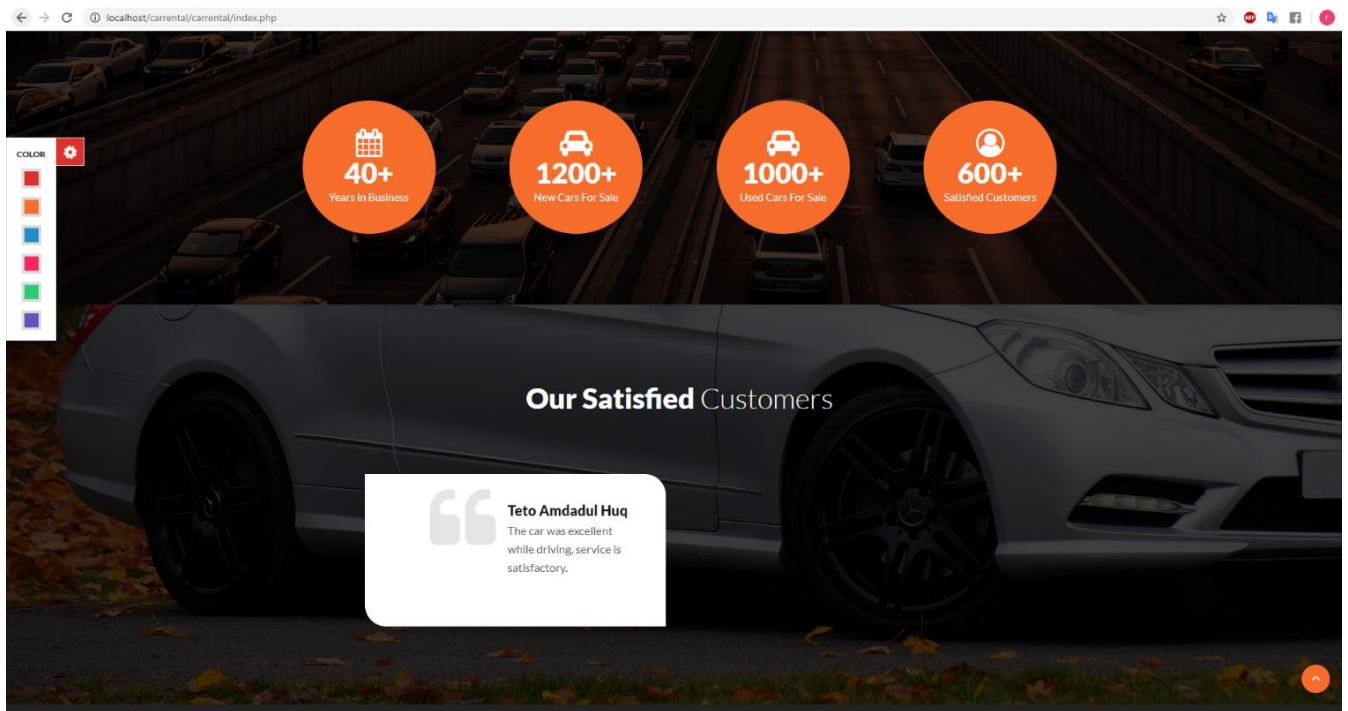


FIGURE 45. Screenshot of Testimony

4.4.3 Admin Panel

Admin is the supreme user of the website. Admin has both the viewing and controlling rights of the website. Admin can see and manage everything from the dashboard. Below the admin panel and code example of a post, a new car is given in Figures 46 and 47. Figure 47 shows the code of posting a new vehicle. Admin can post or manage vehicles dynamically whenever a new car is in the garage.

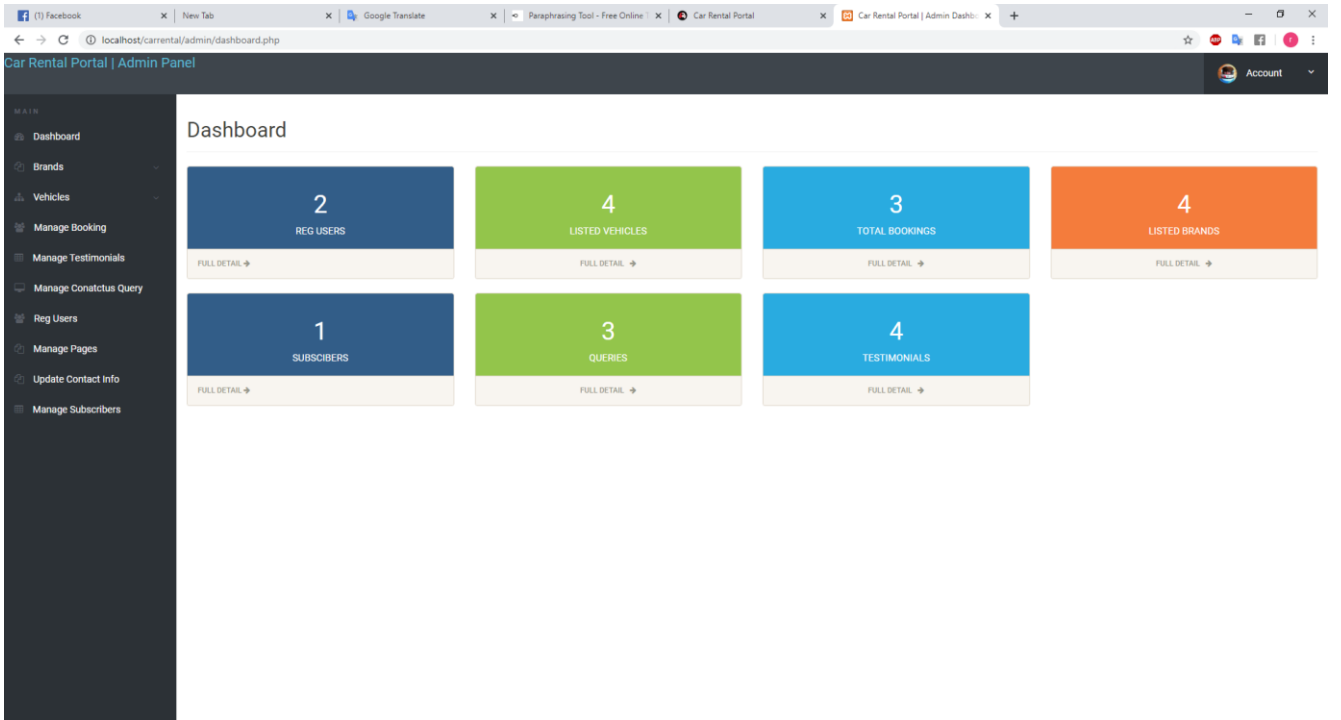


FIGURE 46. Screenshot of admin dashboard

```

Working Files
leftbar.php
logout.php – admin
manage-bookings.php
manage-brands.php
manage-conactusquery.php
manage-pages.php
manage-subscribers.php
manage-vehicles.php
post-avehical.php
Getting Started
screenshots
quick-edit.png
index.html
main.css

1 <?php
2 session_start();
3 error_reporting(0);
4 include('includes/config.php');
5 if(strlen($_SESSION['alogin'])==0)
6 {
7 header('location:index.php');
8 }
9 else{
10
11 if(isset($_POST['submit']))
12 {
13 $vehicletitle=$_POST['vehicletitle'];
14 $brand=$_POST['brandname'];
15 $vehicleoverview=$_POST['vehicalorcview'];
16 $priceperday=$_POST['priceperday'];
17 $fueltype=$_POST['fueltype'];
18 $modelyear=$_POST['modelyear'];
19 $seatingcapacity=$_POST['seatingcapacity'];
20 $vimage1=$_FILES["img1"]["name"];
21 $vimage2=$_FILES["img2"]["name"];
22 $vimage3=$_FILES["img3"]["name"];
23 $vimage4=$_FILES["img4"]["name"];
24 $vimage5=$_FILES["img5"]["name"];
25 $airconditioner=$_POST['airconditioner'];
26 $powerdoorlocks=$_POST['powerdoorlocks'];
27 $antilockbrakingsys=$_POST['antilockbrakingsys'];
28 $brakeassist=$_POST['brakeassist'];
29 $powersteering=$_POST['powersteering'];
30 $driverairbag=$_POST['driverairbag'];
31 $passengerairbag=$_POST['passengerairbag'];
32 $powerwindow=$_POST['powerwindow'];
33 $cdplayer=$_POST['cdplayer'];
34 $centrallocking=$_POST['centrallocking'];
35 $crashcensor=$_POST['crashcensor'];
36 $leatherseats=$_POST['leatherseats'];
37 move_uploaded_file($_FILES["img1"]["tmp_name"],"img/vehicleimages/".$_FILES["img1"]["name"]);
38 move_uploaded_file($_FILES["img2"]["tmp_name"],"img/vehicleimages/".$_FILES["img2"]["name"]);
39 move_uploaded_file($_FILES["img3"]["tmp_name"],"img/vehicleimages/".$_FILES["img3"]["name"]);
40 move_uploaded_file($_FILES["img4"]["tmp_name"],"img/vehicleimages/".$_FILES["img4"]["name"]);
41 move_uploaded_file($_FILES["img5"]["tmp_name"],"img/vehicleimages/".$_FILES["img5"]["name"]);
42
43 $sql="INSERT INTO tblvehicles(VehiclesTitle,VehiclesBrand,VehiclesOverview,PricePerDay,FuelType,
44 ModelYear,SeatingCapacity,Vimage1,Vimage2,Vimage3,Vimage4,Vimage5,AirConditioner,PowerDoorLocks,
45 AntiLockBrakingSystem,BrakeAssist,PowerSteering,DriverAirbag,PassengerAirbag,PowerWindows,CDPlayer,
46 CentralLocking,CrashSensor,LeatherSeats) VALUES(:vehicletitle,:brand,:vehicleoverview,:priceperday,:
47 fueltype,:modelyear,:seatingcapacity,:vimage1,:vimage2,:vimage3,:vimage4,:vimage5,:airconditioner,
48 :powerdoorlocks,:antilockbrakingsys,:brakeassist,:powersteering,:driverairbag,:passengerairbag,
49 |:powerwindow,:cdplayer,:centrallocking,:crashcensor,:leatherseats)";
50 $query = $dbh->prepare($sql);
51 $query->bindParam(':vehicletitle',$vehicletitle,PDO::PARAM_STR);
52 $query->bindParam(':brand',$brand,PDO::PARAM_STR);
53 $query->bindParam(':vehicleoverview',$vehicleoverview,PDO::PARAM_STR);
54 $query->bindParam(':priceperday',$priceperday,PDO::PARAM_STR);
55 $query->bindParam(':fueltype',$fueltype,PDO::PARAM_STR);
56 $query->bindParam(':modelyear',$modelyear,PDO::PARAM_STR);
57 $query->bindParam(':seatingcapacity',$seatingcapacity,PDO::PARAM_STR);
58 $query->bindParam(':vimage1',$vimage1,PDO::PARAM_STR);
59 $query->bindParam(':vimage2',$vimage2,PDO::PARAM_STR);
60 $query->bindParam(':vimage3',$vimage3,PDO::PARAM_STR);
61 $query->bindParam(':vimage4',$vimage4,PDO::PARAM_STR);
62 $query->bindParam(':vimage5',$vimage5,PDO::PARAM_STR);
63 $query->execute();
64 $msg="Your Vehicle has been posted successfully";
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }

```

FIGURE 47. Code of posting a vehicle

4.5 Publishing

Publishing the website is the most important part of the website. After all the testing is done then it is time to publish the website. The author publishes the website on the local server only. XAMP is used for running the Apache and MySQL data. In Figure 48, the localhost address is shown in the address bar.

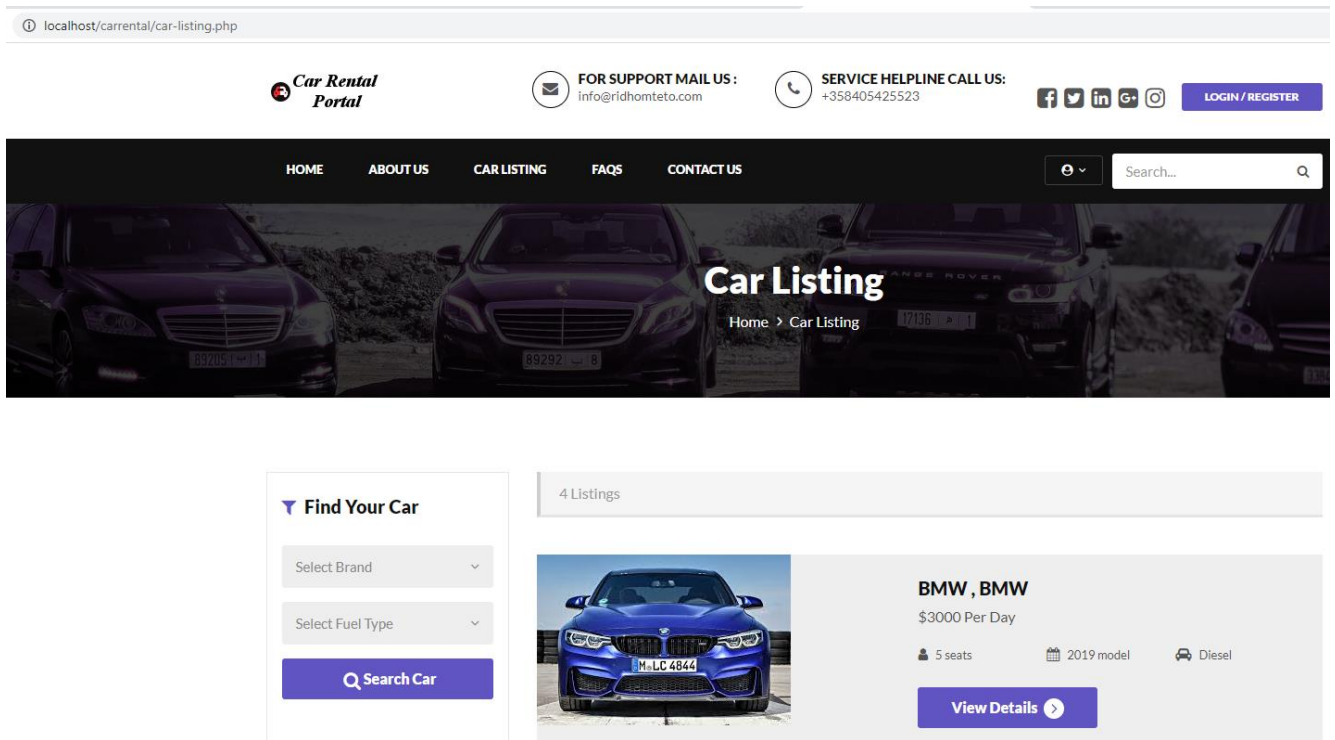


FIGURE 48. Publishing address of the website.

5 CONCLUSION

The primary motivation behind this website was to show how this theory functions for clients and how this site was made. Because of time limitations, the emphasis is on the most proficient method to make this site and how clients can convey it. It tends to be hard to begin a site venture without a legitimate vision and arranging. Legitimate arranging is required before beginning a website, which will make it simpler to finish in a brief timeframe. The way toward building this website has very interesting effects on the author in various significant stages. By following all the steps, assembling a site is simple, successful, and quick. Moreover, the idea of the site must be clear and data regarding the site ought to be gathered. After the arrangement was created, programmers give more consideration to the site plan. Developers make the ideal web architecture with an assortment of websites for web application. At this stage, they must invest more energy at different stages and the website construction is additionally hard without proper planning. When their plan is finished, they will begin working in the execution stage and will have the option to get exact outcomes while making the site. This implies the site is fit to be looked at. Finally, the development work is finished through a site test. After all, everything works out in a good way and the site is well prepared for use.

In summary, the general thought of a web-based business site is introduced in this project. Moreover, this makes it simpler for the developer to build a site like this. Picking the correct method to build a site is a harder choice indeed without having good knowledge and skills in a particular field. By picking up information on this site creation the author will have the option to utilize various positions and encounters to make an increasingly unique site later.

REFERENCES

- Biglione, K. 2016. What is WordPress? Available: <http://wpapprentice.com/blog/what-is-wordpress/>. Accessed 02 January 2020.
- Dose M & Lilja H. 2015 Page: 4-6. A schematic for comparing web backend application frameworks. Available: <http://publications.lib.chalmers.se/records/fulltext/219826/219826.pdf>. Accessed 02 January 2020.
- Duckett J. 2014. JavaScript and jQuery: Interactive Front-End Web Development 1st. Accessed 10 January 2020.
- Get bootstrap. Available: <https://getbootstrap.com/docs/4.4/about/overview/> Accessed 15 January 2020.
- Horton W & Horton K, 2003. E-learning Tools and Technologies: A consumer's guide for trainers, teachers, educators, and instructional designers. Available: <https://books.google.fi/books?isbn=0471456780>. Accessed 02 January 2020.
- Jiaquan Shi, 2007. Database Management Theory, 1st Edition. 1, 1-4. Accessed 21 January 2020.
- JSX. 2015. About JavaScript. Available: https://developer.mozilla.org/enUS/docs/Web/JavaScript/About_JavaScript. Accessed 22 January 2020.
- Liberty, J. & Hurwitz, D., 2003. Programming ASP.NET. Accessed 22 January 2020
- Mozilla.org, 2020. Available: https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_web_server. Accessed 13 May 2020.
- Vaish, Gaurav. NoSQL Starter, Packt Publishing, Limited, 2013. Available: <https://ebookcentral.proquest.com/lib/cop-ebooks/detail.action?docID=1142875>. Accessed 13 May 2020.
- Mikoluk, K. 2014. How to use XAMPP to Run Your Web Server. Available: <https://blog.udemy.com/xampp-tutorial/>. Accessed 13 May 2020.
- Mening R. 2018. WordPress vs Joomla vs Drupal (comparison). Available: <https://websitesetup.org/cms-comparison-wordpress-vs-joomla-drupal/>. Accessed 13 January 2020.
- Murugesan S, Deshpande Y, Hansen S & Ginige A, 2016. Web Engineering: A New Discipline for Development of Web-Based System. Accessed 13 January 2020
- Nedelcu, Clément. Nginx HTTP Server, Packt Publishing, Limited, 2013. Available: <https://ebookcentral.proquest.com/lib/cop-ebooks/detail.action?docID=3027386>.
- Nodejs.org., 2020. Available: <https://nodejs.org/en/about/> Accessed 13 May 2020.
- Tree-structure of backend development 2020. Available: codeburst.io Accessed 09 January 2020
- Sabharwal, Navin, et al. Apache CloudStack Cloud Computing, Packt Publishing, Limited, 2013. Available: <https://ebookcentral.proquest.com/lib/cop-ebooks/detail.action?docID=1142883>. Accessed 01 January 2020.

Sklar, J. 2011. Principles of web Design: The Web Technologies series (5th Edition). Boston, MA: Cengage Learning, Inc.

Shelly, G & Woods, D & Dorin, W. 2009. HTML: Comprehensive Concepts and Techniques (5th Edition). Boston, MA: Cengage Learning, Inc.

W3schools. 2020. Ajax Intro. Available: https://www.w3schools.com/js/js_ajax_intro.asp. Accessed 01 January 2020.

W3schools. 2020. CSS Syntax. Available: https://www.w3schools.com/CSS/CSS_syntax.asp. Accessed 01 January 2020.

W3schools. 2020. HTML Introduction. Available: https://www.w3schools.com/HTML/HTML_intro.asp. Accessed 01 January 2020.

W3schools. 2020. MaxContent Delivery Network, Bootstrap. Available: https://www.w3schools.com/bootstrap/bootstrap_get_started.asp. Accessed 12 January 2020.

Welling L & Thomson L. 2003. PHP and MySQL Web Development. Available: <https://books.google.fi/books?isbn=8131729877>. Accessed 14 May 2020

UML Diagram 2020. Available: <https://www.uml-diagrams.org/> Accessed 20 June 2020.

User Registration:

Table Name	Tblusers
Description	This table provides the information about User registration
Primary Key	Id
Foreign Key	-

FIGURE 17. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores User id
2	FullName	varchar(120)	Null	It stores User name
3	EmailId	varchar(100)	Null	It stores email address of User
4	Password	varchar(100)	Null	It stores Password
5	ContactNo	char(11)	Null	It stores Contact no
6	Dob	varchar(100)	Null	It stores Birthdate

FIGURE 18. Database table

Brands Table:

Table Name	Tblbrands
Description	This table provides the information about Car brands
Primary Key	Id
Foreign Key	-

FIGURE 19. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores brand id
2	BrandName	varchar(120)	Not Null	It stores Brand name
3	CreationDate	Timestamp	CURRENT_ TIMESTAMP	It stores brand creation date
4	UpdationDate	Timestamp	NotNull	It stores brand updation date

FIGURE 20. Database table

Contact us details Table:

Table Name	tblcontactusinfo
Description	This table provides the contact information at the website
Primary Key	Id
Foreign Key	-

FIGURE 21. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	Id	Int	Primary Key	It is the id of the record
2	Address	Tinytext	Null	It stores the name of the company
3	EmailId	varchar(255)	Null	It stores of the company email
4	ContactNo	char(11)	Null	It stores of the company contact no

FIGURE 22. Database table

Inquiry Table:

Table Name	tblcontactusquery
Description	This table will store the information of car inquiry of user
Primary Key	Id
Foreign Key	-

FIGURE 23. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores inquiry id
2	Name	varchar(100)	Null	It stores user
3	EmailId	varchar(120)	Null	It stores email id
4	ContactNum- ber	char(11)	Null	It stores contact no for user
5	Message	Longtext	Null	It stores user message for inquiry
6	PostingDate	Timestamp	CUR- RENT_TIMEST AMP	It stores inquiry date
7	Status	int(11)	NotNull	It stores status 0 for read and 1 for read

FIGURE 24. Database table

Feedback Table:

Table Name	Tbltestimonial
Description	This table store information about feedback
Primary Key	F_Id
Foreign Key	-

FIGURE 25. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores feedback id
2	UserEmail	varchar(100)	Not Null	It stores user email
3	Testimonial	medium text	Not Null	It stores feedback
4	PostingDate	Timestamp	NotNull	It stores posting date of feedback
5	Status	int(11)	NotNull	It stores staus(0 for inactive and 1 active)

FIGURE 26. Database table

Pages Content Table:

Table Name	Tbtpages
Description	This table store information about website pages
Primary Key	Id
Foreign Key	-

FIGURE 27. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores page id
2	PageName	varchar(255)	Null	It stores page name
3	Type	varchar(255)	Not Null	It stores page type
4	Detail	Longtext	Not Null	It stores pages info

FIGURE 28. Database table

Subscriber Table:

Table Name	Tblsubscriber
Description	This table store email address of subscriber
Primary Key	Id
Foreign Key	-

FIGURE 29. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores subscriber id
2	Sub- scriberEmail	varchar(120)	Null	It stores subscriber email is
3	PostingDate	Timestamp	Null	It stores subscription date

FIGURE 30. Database table

Vehicles Info Table:

Table Name	Tblvehicles
Description	This table provides the information about cars
Primary Key	Id
Foreign Key	-

FIGURE 31. Database table

APPENDIX 1/10

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores User id
2	VehiclesTitle	varchar(150)	Null	It stores vehicle title
3	VehiclesBrand	int(11)	Null	It stores vehicle brand id
4	VehiclesOver-view	Longtext	Null	It stores vehicle overview
5	PricePerDay	int(11)	Null	It stores vehicle rent per day
6	FuelType	varchar(100)	Null	It stores fuel type of vehicle
7	ModelYear	int(6)	Null	It stores model year of vehicle
8	SeatingCapacity	int(11)	Null	It stores seating capacity of vehicles
9	Vimage1	varchar(120)	Null	It stores vehicle image 1
10	Vimage2	varchar(120)	Null	It stores vehicle image 2

11	Vimage3	vvarchar(120)	Null	It stores vehicle image 3
12	Vimage4	vvarchar(120)	Null	It stores vehicle image 4
13	Vimage5	vvarchar(120)	Null	It stores vehicle image 5
14	AirConditioner	int(11)	Null	It stores the availability of air conditioner in the vehicle
15	PowerDoor-Locks	int(11)	Null	It stores availability of power door locks in vehicle
16	AntiLockBrakingSystem	int(11)	Null	It stores availability of Anti-lock Braking System in vehicle
17	BrakeAssist	int(11)	Null	It stores availability of Brake Assist in vehicle
18	PowerSteering	int(11)	Null	It stores availability of Power steering in vehicle
19	DriverAirbag	int(11)	Null	It stores availability of Driver Airbag in vehicle
20	PassengerAirbag	int(11)	Null	It stores availability of Passenger airbag in vehicle
21	PowerWindows	int(11)	Null	It stores availability of Power windows in vehicle
22	CDPlayer	int(11)	Null	It stores availability of CD Player in vehicle
23	CentralLocking	int(11)	Null	It stores availability of Central locking in vehicle
24	CrashSensor	int(11)	Null	It stores availability of crash sensor in the vehicle

25	LeatherSeats	int(11)	Null	It stores availability of leather seats in vehicle
26	RegDate	Timestamp	Null	It stores vehicle creation date
27	UpdationDate	Timestamp	Null	It stores vehicle updation date

FIGURE 32. Database table

Booking table:

Table Name	Tblbooking
Description	This table provides the information about booking
Primary Key	Id
Foreign Key	-

FIGURE 33. Database table

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	id (<i>Primary</i>)	int(11)	Primary Key	It stores booking id
2	userEmail	varchar(100)	Null	It stores User email
3	VehicleId	int(11)	Null	It stores vehicle id
4	FromDate	varchar(20)	Null	It stores booking from date
5	ToDate	varchar(20)	Null	It stores booking date
6	message	varchar(255)	Null	It stores message
7	Status	int(11)	Null	It stores confirmation and cancellation status
8	PostingDate	Timestamp	Null	It stores Booking date

FIGURE 34. Database table