



# **TrainMe- A web-based management system for driving school**

**A dissertation submitted for the Degree of Master of  
Information Technology**

**A.H.R.P Mithrasena**

**University of Colombo School of Computing**

**2020**



# **TrainMe- A web-based management system for driving school**

**A.H.R.P Mithrasena**

**2020**

# Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge, it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: A.H.R.P Mithrasena

Registration Number: 2017/MIT/046

Index Number: 17550463

---

Signature:

Date:

This is to certify that this thesis is based on the work of Ms. A.H.R.P Mithrasena under my supervision. The thesis has been prepared according to the format stipulated and is of an acceptable standard.

Certified by:

Supervisor Name: Dr. M.G.N.A.S Fernando

---

Signature:

Date:

# Abstract

In a time when technology is quickly accessible by everyone, the regular businesses are opting to undergo a digital transformation to vary the way they operate to deliver more value to the customer. In recent years, most of the business is predicated on technology. At present, with the fast development of the information technology industry, most of the organizations and industries are going towards the technology advancements. Driving training school is one of the areas that I have noticed to be improved with technology advancement. Automation of the manual process manages the waste of human power and money by utilizing the resources effectively. So, most of the people tend to automate this process to urge the utmost benefit.

Most of the automated systems contain only the registration and viewing some records within the system. The proposed system includes automated scheduling, vehicle tracking, learning portal with an online payment facility with feature-rich web-based system.

# Acknowledgment

I express my heartiest gratitude to my supervisor Dr. M.G.N.A.S Fernando, senior lecturer of the University of Colombo School of Computing of University of Colombo School of Computing, for providing me continuous direction and supervision throughout the project.

I would like to appreciate my loving family, who has always supported me and motivate me throughout this time period. And also, special thanks to Mr. Ashan Maduranga and Ravidu Lashan, who gave technical support as well as helping me in writing the thesis. It's a great wish on behalf of me to accept the help and involvement of all the people that helped me to complete my project.

# Table of Contents

<b>Declaration</b> .....	<b>3</b>
<b>Abstract</b> .....	<b>4</b>
<b>Acknowledgment</b> .....	<b>5</b>
<b>Table of Figures</b> .....	<b>1</b>
<b>Table of Tables</b> .....	<b>2</b>
<b>Chapter 1</b> .....	<b>1</b>
<b>Introduction</b> .....	<b>1</b>
1.1 Background and Motivation.....	1
1.2 Existing Problems .....	2
1.3 Aim and Objectives.....	3
1.4 Methodology .....	3
1.5. Outline of the Dissertation .....	5
1.6. Summary .....	5
<b>Chapter 2</b> .....	<b>6</b>
<b>Similar Systems and Technology</b> .....	<b>6</b>
2.0. Introduction.....	6
2.1. Facts and finding.....	6
2.2. Existing of driving school management systems .....	6
2.3. Discussion .....	10
2.5. Technology .....	11
2.5.1 .Net Platform.....	11
2.5.2 SQL Server 2017.....	11
2.5.3. AngularJs .....	11
2.5. Summary .....	12
<b>Chapter 3</b> .....	<b>13</b>
<b>Problem Analysis</b> .....	<b>13</b>
3.1 Introduction.....	13
3.2 Overview of the project .....	13
3.2.1 Problem Statement analysis .....	14
3.3 Data collection protocol.....	14
3.3.1 Review Similar Systems .....	14

3.4 Requirement Specification for the proposed system.....	15
3.4.1 Functional Requirements .....	15
3.4.2 Non- Functional Requirements .....	16
3.5 Summary .....	16
<b>Chapter 4 .....</b>	<b>17</b>
<b>System Design.....</b>	<b>17</b>
4.1 Introduction.....	17
4.2 Overall System Architecture.....	17
4.2.1 Application Layer .....	19
4.2.2 Data Layer.....	19
4.2.3 Presentation Layer .....	19
4.3 Data Design.....	19
4.3.1 Entity Relationship Diagram of the Proposed System .....	20
4.3.2 Use case Diagram .....	20
4.4 Interface Design .....	20
4.4.1 Login and Register student page .....	21
4.4.2 Scheduling and Learning Portal .....	22
4.4.3 Add, Edit, Delete module page .....	23
4.4.4 Reporting and statistics .....	25
4.5 Summary .....	26
<b>Chapter 5 .....</b>	<b>27</b>
<b>Implementation and Testing .....</b>	<b>27</b>
5.1 Introduction.....	27
5.2 System Implementation .....	27
5.2.1 Technology Consideration .....	27
5.2.2 Selecting the Language .....	27
5.3 Testing.....	28
5.3.2 Testing Strategy .....	30
5.3.2.1 Test Procedure.....	33
5.3.3 Test Plan.....	34
5.4 Summary .....	34
<b>Chapter 6 .....</b>	<b>35</b>
<b>Evaluation.....</b>	<b>35</b>

6.1 Introduction.....	35
6.2 Evaluation of Project Practice.....	35
6.3 Evaluating the application.....	37
6.3.1 Approaches to evaluating the product.....	37
6.3.2 Evaluating Criteria.....	37
6.3.3 Problems Addressed.....	38
6.3.4 Evaluating the Process Solution Provided.....	38
6.3.5 Other Project Accomplishments.....	39
6.4 Summary.....	40
<b>Chapter 7.....</b>	<b>41</b>
<b>Conclusion and Future Work.....</b>	<b>41</b>
7.1 Introduction.....	41
7.2 Project Results Overview.....	41
7.2 Project Result Outline.....	41
7.2.1 Lessons Learnt Via the Achievement of Project Objectives.....	42
7.3 Future Improvements.....	43
7.4 Summary.....	43
<b>References.....</b>	<b>44</b>
<b>Appendix A: Diagrams.....</b>	<b>45</b>
<b>Appendix B: Code snippets for some development.....</b>	<b>47</b>
Figure B.1: Database table structure.....	47
Figure B.4: Service written to add, edit, delete and get student details.....	49
Figure B.5: Service written to add, edit, delete and get vehicle details.....	50
<b>Appendix C: Test Cases.....</b>	<b>51</b>



# Table of Figures

Figure 1.4.1: High-level diagram of the proposed system .....	5
Figure 2.2.1: Manahara Learners online system .....	8
Figure 2.2.2: Bimal Learners online system .....	9
Figure 2.2.3: How to drive online system .....	10
Figure 4.2.1: Overall system architecture .....	19
Figure 4.4.1.1: Login Page .....	22
Figure 4.4.1.2: Registration of a new student.....	22
Figure 4.4.1.3: Schedule page.....	23
Figure 4.4.1.4: Learning portal Page .....	23
Figure 4.4.1.5: Student details .....	24
Figure 4.4.1.6: Add vehicle details window .....	24
Figure 4.4.1.7: View Instructor details window .....	25
Figure 4.4.1.8: Add Instructor details .....	25
Figure 4.4.1.9: Reporting .....	25
Figure 4.4.1.10: Statistics .....	26

# Table of Tables

Table 5.3.2.1: Test Plan- Unit Testing .....	33
Table 5.3.2.2: Test Plan- Integration Testing .....	34
Table 5.3.2.3: Test Plan- System Testing .....	34
Table 6.3.3.1: Identifying evaluators and problems addressed .....	39
Table 6.3.4.1: Evaluating the Process Solution Provided .....	40
Table 7.2.1.1: Accomplishment of Project Objectives .....	44

# Chapter 1

## Introduction

The introduction chapter highlights the details of the overall background for the proposed system of the TrainMe system. Introduction to the project will be given at the beginning of the chapter. Then the identified problem statement relevant to the project is discussed, and the proposed solution is elaborated. The project's aim, objectives, and limitations for the proposed project will be discussed further in this chapter.

At present, there is a multitude of driver training institutes operating across the country, and only a handful of them have opted for the digital transformation. The majority of these institutes are still committed to the manual processes of running the business.

With the proposed system, it can be given a far better solution than the ones already developed. The manual processes that are being utilized by the driver training institutes are time and resources consuming. A feature-rich, digital solution is needed in place of these manual systems under a user-friendly environment with practical resource usage, to deliver maximum value for the end-users.

### 1.1 Background and Motivation

In recent years, most of the business is predicated on technology. For the past recent years, trends of technological advancements came into the picture. All most all the industries are now driving with the technologies. All the manual processes are automated. Automating a manual process helps to save both money and time. One of the essential areas that need to go with the technology is handling manual systems used in the areas like driving schools. In Sri Lanka, most of the organization are behind the technologies [1]. This is another area that is notices by me. So, most of people tend to automate this process to urge the utmost benefit.

Most of the event is merely focused on managing the knowledge and automate the manual process. I even have experienced the difficulties of using a manual system to handle data and problems encountered when allocating time slots to learners.

Collaborating with numbers of students at once, tracking the progress of the students, and handling daily data is a hell of a problem driving schools that use the manual system. This motivates me to implement a sophisticated automated web-based solution for driving schools in Sri Lanka.

## **1.2 Existing Problems**

In a time where technology is readily accessible by everyone, the traditional businesses are opting to undergo a digital transformation to change the way the businesses are operated in order to deliver more value to the customer. Every business, from large scale enterprises to small scale retail shops leverage technology in the most efficient ways to deliver maximum value to the customer at a fraction of the cost [2]. Though each of these success stories and every keynote highlights the importance of the digital transformation, some businesses are still uncertain about embracing the new technology-driven culture. One such business that has held back is the driver training institutes (Learners).

At present, there is a multitude of driver training institutes operating across the country, and only a handful of them have opted for the digital transformation. The majority of these institutes are still committed to the manual processes of running the business. There are different types of users who are coming to get trained, such as workers, students, etc. during different times of the day. Ease of scheduling a driving lesson for a trainee is low as these scheduling typically happens through phone calls, which sometimes require the trainee to make a few phone calls to get a confirmation on their specific time slot.

As there are no follow-ups made to inquire if the trainee showed up at the scheduled time slot, the task of tracking the progress of the trainee has come down to be solely dependent on the instructor's memory. As there is no reliable way to track the real-time location of a training vehicle, the trainees have to get picked up from one designated location and would have to wait their turn with a batch of other trainees. Due to this, the trainees could be waiting up to one and a half hours before arriving at their designated driving makes it inefficient to create, update, or search for a particular entry.

Driver training institutes provide learning resources and past papers to better prepare the candidates for the exams. However, as all these are hard copies, the institute has to bear an expense to them printed [3]. The trainees, on the other hand, have to make multiple trips to the institute to collect these resources and submit any papers which have been completed, to be marked by the instructors. Lack of a digital portal to manage the learning resources makes the overall process time consuming and costly for both the trainees and the instructors. Further, a lack of infrastructure to support online payments makes it mandatory for the trainees to make multiple trips to the institute to pay the dues, which could be a pain point.

### **1.3 Aim and Objectives**

This project aims to provide a feature-rich digital solution to manage limited resources efficiently and effectively in a driver training institute.

The following are the key milestones that need to be achieved successfully during the project to achieve the aim as mentioned earlier.

- Identify the customer requirements and fulfill them to keep a good customer base.
- Designing appropriate interfaces to the system such as Registration, Time scheduling, Payments, learning portal, Vehicle tracking, etc.
- Implement registration method which will help to record all the learner's details properly
- Identify the online payment method to do all the transactions efficiently.
- Provide learning resources for learners to learn online from anywhere with any device
- Identify scheduling mechanism which will help prevents double-booking of time slots.
- Identify vehicle tracking mechanism to identify the live location of the training vehicle.

### **1.4 Methodology**

Develop a web-based system to facilitate customer requirements. The proposed system will use a web server to store information about the learners and other tasks. There will be three user levels, which are system admin, instructors, and students. Students first registered with the system and

made the payment to access the features in the system. The system admin had full permission for the application, but the instructor will not.

The system should support the allocation of predefined training when scheduling training slots to the trainees on a **first-come, first-serve basis**.

The class schedule is an essential part of every driving school. A logged-in trainee should have the option to select an available time slot for his/her training, on a preferred date. Once locked in, other trainees should not be able to lock the same time slot, to ensure there are no conflicts on the schedule. Need to find a proper algorithm to avoid conflicts of booking the same schedule date by two persons. The system needs to use effective mechanism when allocating limited resources to the trainee. E.g., Vehicle, Drivers, etc.

The proposed system architecture will be a **client-server system**.

The client accesses the web application over the internet, and the web application requests the required URL over the webserver. Web server process the data and send back to the web application. Then the client should be able to view his/her request results.

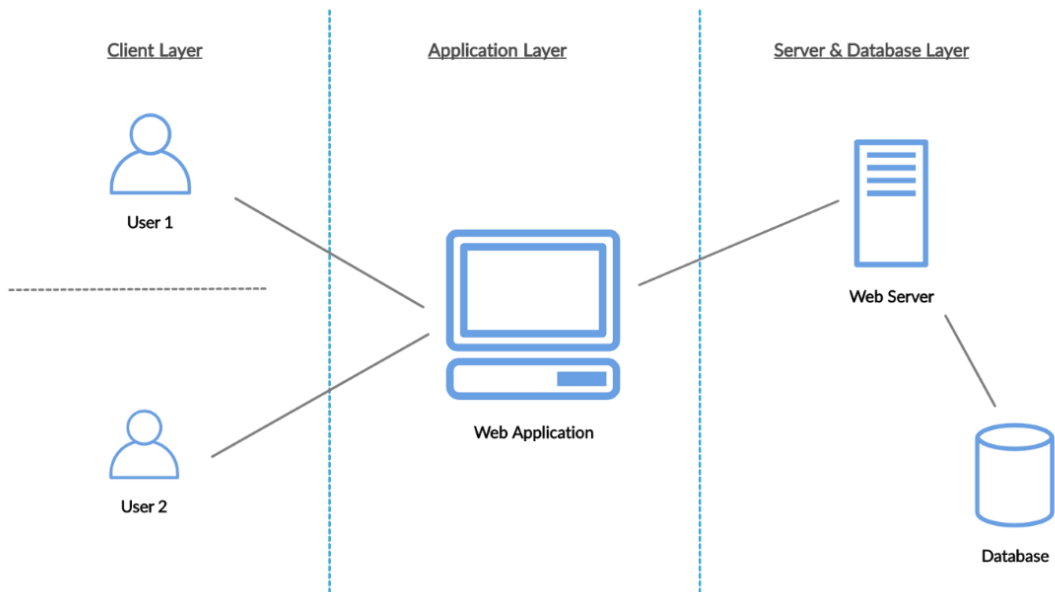


Figure 1.4.1: High-level diagram of the proposed system

## **1.5. Outline of the Dissertation**

The structure of the dissertation as follows. The second chapter describes the background of the project with an analysis. Requirement Analysis is included in this chapter. A detailed description of the similar systems is included as well. You should also consider different competing design strategies for your system with the comparison of alternate design strategies. The proposed research design and methodology will be discussed in chapter three. Critical evaluation of the system will describe under chapter four, and the last chapter will be discussing the conclusion of the work with future improvements.

## **1.6. Summary**

Basics for the dissertation are discussed in this chapter. It presented the current problem, which is manual processes that are being utilized by the driver training institutes are time and resources consuming. Hence, the utmost aim of this project is to supply a feature-rich digital solution to manage limited resources efficiently and effectively in a driver training institute. Considering this aim, the objectives of the implementation were presented, followed by justification, methodology, and the limitations of the design.

# Chapter 2

## Similar Systems and Technology

### 2.0. Introduction

Under this chapter will discuss a similar system to the proposed solution. To build the system properly, critical analysis of similar systems is very important. That will give the knowledge and strength to the developers to build the right system. All of this information helps to select the best way to develop the system. In order to build a good system, researching and doing analysis of the existing system is a good practice. As an enhancement for the existing system, new implementations can be done. Critical analysis of the literature review will give the purpose of collecting information related to the development of driving school system.

### 2.1. Facts and finding

The main goal of this study is to recognize the difficulties and problems of the current system. Other than that, this helps to identify and analyze problems in the current system and come up with a new solution to the existing drawbacks. In order to identify the requirements for future systems, the study of the management process is important. Finally, it is to complete the future system and to organize the important modules for the system.

### 2.2. Existing of driving school management systems

Under this section will discuss the current online system of driving school management systems around Sri Lanka.

#### 1. Manahara Learners

Refer to figure 2.2.1; this is one of the famous online websites that is used by one of the driving school in Sri Lanka. Manahara Learners are using their system to provide information about their driving school to outside users. This displays six main menus in the system. The initial one is the Home page. The home page will display a brief introduction about the learning school, some basic



topic descriptions, some of the feedback which is given by their students, and all of the recent announcements of their driving school.

About us, the tab will describe who they are, and the third past paper page will provide a couple of past papers to the students. The fourth tab shows the services that they are going to provide to the students. The fifth one mentioned the gallery, which will show some photos of their vehicle. The last one is their contact details.

This only provides normal functions like other online systems. This allows students to go through the system and to get an idea about the driving school, how they teach to students, and help to get some general ideas like driver awareness, driver techniques, Road Safety, and new license, etc.

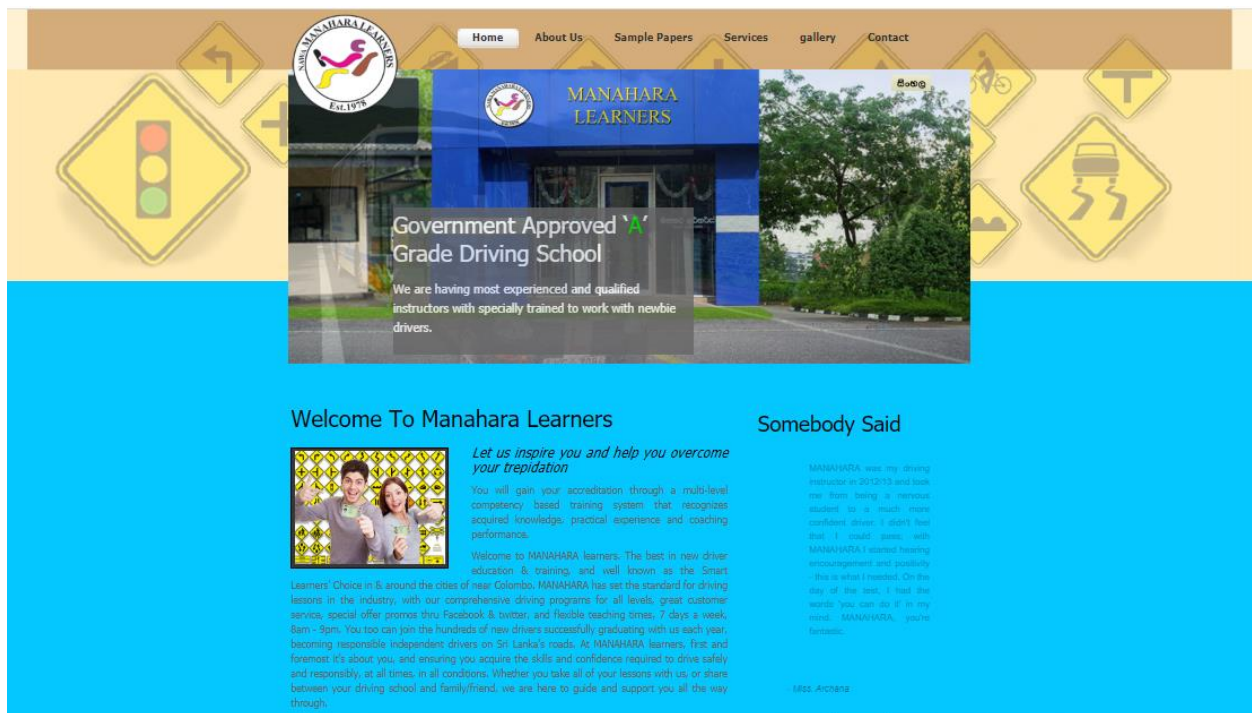


Figure 2.2.1: Manahara Leaners online system

## 2. Bimal Learners

The second online system is from “Bimal Learners” Refer to figure 2.2.2. This is another similar system to usual driving school system. Mainly this also has six menus. The Home page shows the provided service and a couple of basic information such as required documents, prices, and charges for practical lessons, government charges, and the things user should know when lost the license, etc. In the contact menu, they have mentioned all of the branch details with the location. They especially have a sign-up button. But it’s not functioning at the moment.

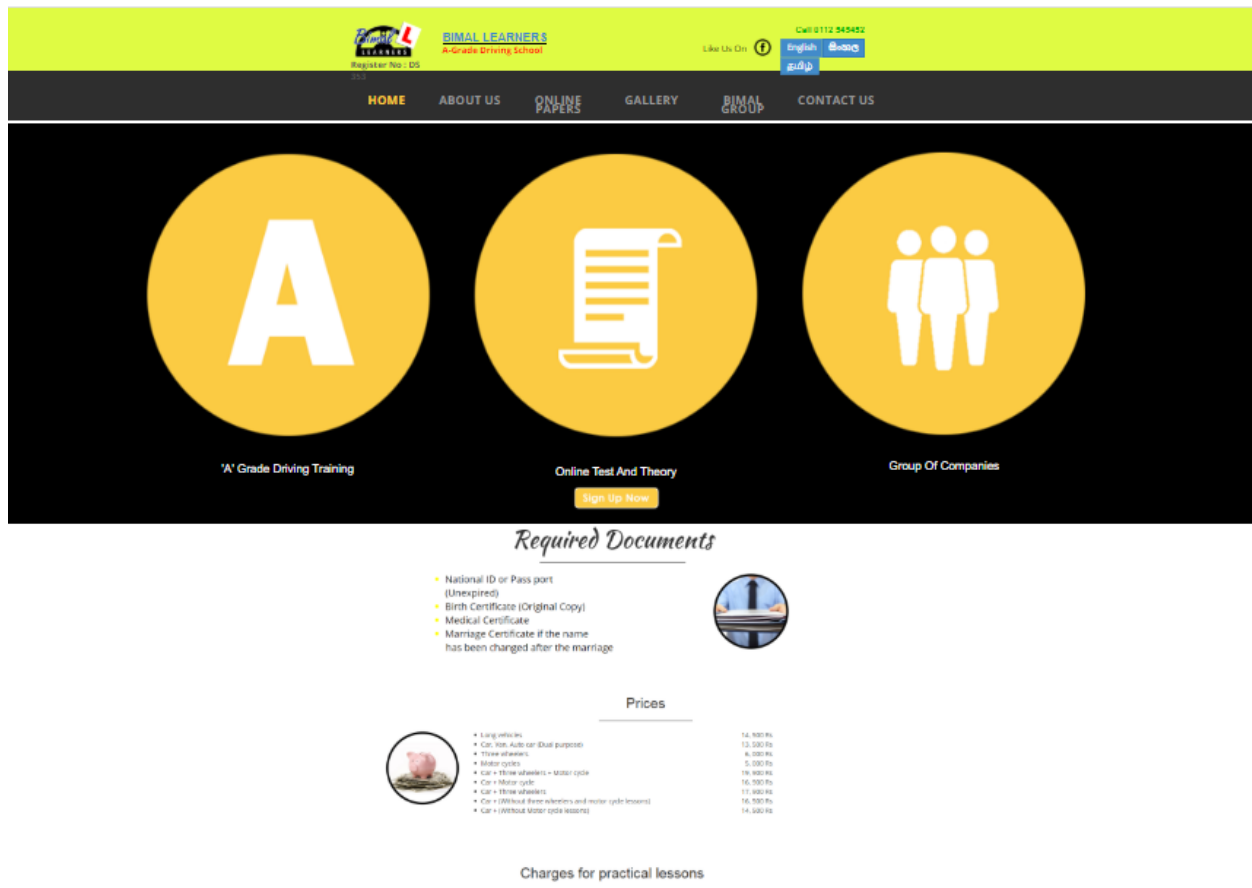


Figure 2.2.2: Bimal Learners online system

On the home page, they were using graphics as well. About us, the menu is for their overall information. Then the next page is for the online test. This contains two options, namely, reading materials and online papers with three main languages. They have a gallery menu that shows some images of their vehicles. The next is data about their franchise. They list all of their chains here, all of it situated around Sri Lanka.

This online system that is using by Bimal Learners also not too different than Manahara Learners online system. Only difference in their website is they have a sign-up feature, and in the online papers, they provide both reading materials as well as past papers in three main languages.

### 3. How to drive driving school

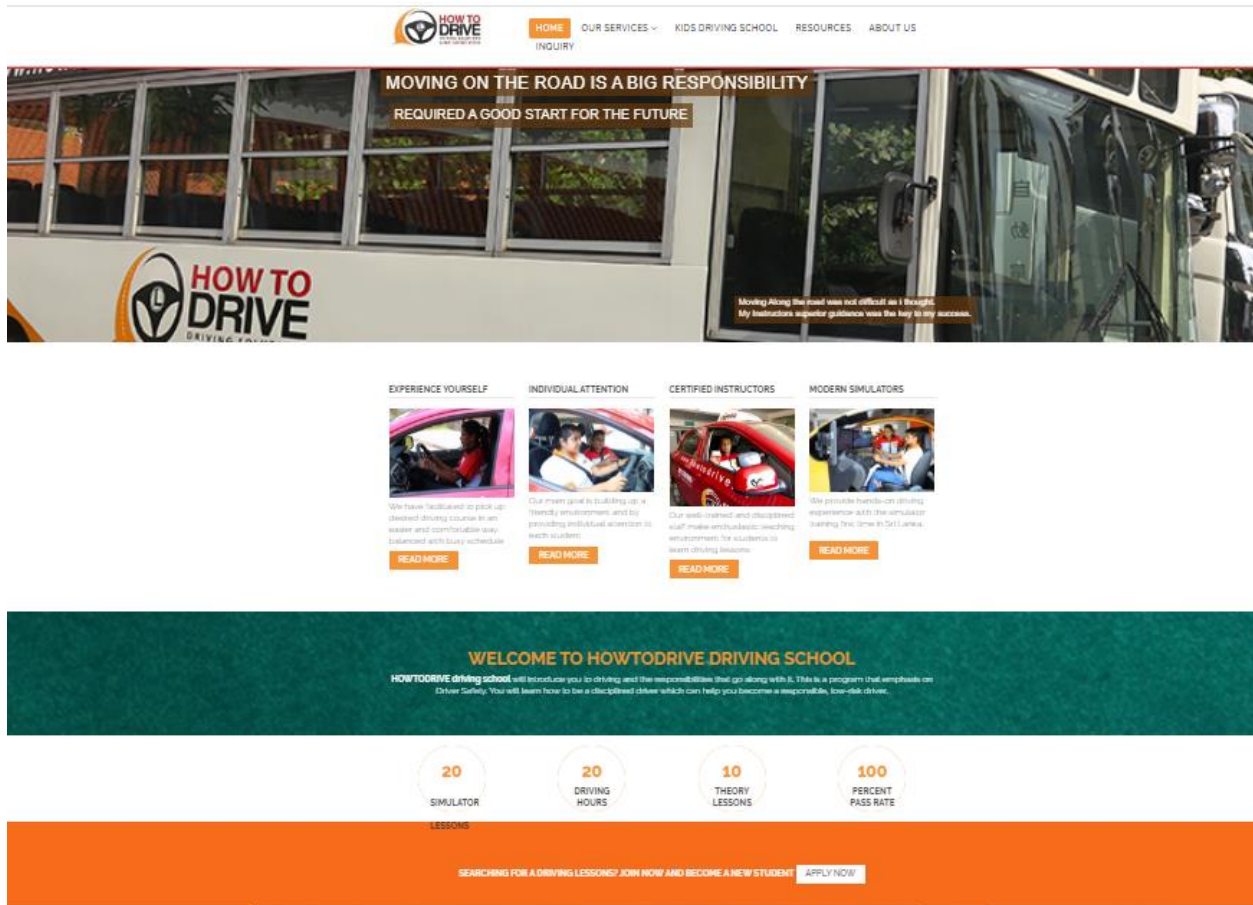


Figure 2.2.3: How to drive online system

"How to drive" driving school is the other system that is going to be reviewed. Refer to figure 2.2.3 for a detailed view picture. The home page is the first tab as default. On this page, they have included a brief introduction about their driving school, display some pictures, and the latest news. There is no sign-up or sign-in page. The next menu is displayed about the services that they are provided from the institute.

The next menu is the kids driving school option, which other systems doesn't have. They have given different age groups and a description of how and what they are going to teach for kids.

The next menu is resources, and the next one is about us tab, which shows vision, mission, and

values. Finally, the inquiry tab, which has all contact details and, users can include inquiry about anything over the site.

## **2.3. Discussion**

As discussed above, there are three websites of driving school online system. All of the above mentioned three systems are used to give information about their services rather than interacting with the users. It knows as an informative online system for the users. This depicts all the information related to the driving practical, how to apply for a driving license, and some common rules. These websites are very useful, but there is no contact or interaction between the users.

Today, automating a manual system is an easy task, but matching the exact client requirement is not an easy task. Always should target customer requirements. User-friendliness and efficiency are very important when it comes to online systems. Users should be able to access the system at anytime from anywhere. The proposed system will fill the gap of these online sites.

## **2.4 Comparison of alternative design strategies**

As discussed above, at present, there are different solutions provided for the driving schools. But any of these solutions will not address the problems discussed in the earlier chapters. Developing a mobile app would be the best option for this problem. But with time constraints, it was not going to be an easy task. And also, the mobile app will only be efficient for students, not for the instructors or system admin. For implementing the scheduling, the feature should come up with a good solution to avoid conflicts between two users. Giving one date for a couple of students will make the logic too complex, so due to that proposed of having one day to book by only one student at a time. The payment option is used any may systems, but integrating a payment gateway is going to be a costly task. Because of that, the best option is to integrate a pay pal account with the payment feature and just display the user can make a payment online. Tracking is another tricky feature that is very difficult to implement. There are so many popular vehicle tracking systems like uber, pick me app available in Sri Lanka. So, I had to integrate that kind of tracking mechanism into the system also.

## **2.5. Technology**

This section summarizes the list of tools and technologies that were used to develop the system. Additional details of the web application, web server, database server, and the programming languages used are detailed below.

### **2.5.1 .Net Platform**

The APIs that are developed as a part of this system is written in the C# programming language. The .NET developer platform was used to develop the APIs as it has inbuilt support for C# language and it's compilers, supporting libraries, and editors and tools necessary for all major operating systems. The .NET platform supports an array of tools, programming languages, and libraries that could be used to build a variety of different applications [4].

### **2.5.2 SQL Server 2017**

To fulfill the database server requirement of the project, Microsoft SQL Server has been used. It is a relational database management system owned and developed by Microsoft. With the facilities to host the server on a local machine, over a network or the internet [5] provides much-needed flexibility to the project.

### **2.5.3. AngularJs**

AngularJS is one of the most popular frameworks that is currently being used to develop dynamic web applications. In the development of this application, the same framework has been used as it allows you to extend HTML syntax to define application components clearly. With the support for data binding and dependency injection, AngularJS effectively eliminates the need to write a large amount of code that is otherwise required.

### **2.3.4 Server Support**

TrainMe is a web-based system where students and instructors can do the functionalities independently. In order to support the system, servers of the likings of XAMPP or IIS were considered. However, having the requirement of being compatible with the underlying technologies such as .NET in mind, IIS was chosen for the scope of this project.

## **2.5. Summary**

This chapter focused on providing an general review of the driving school system. It detailed the similar applications that are in the market at the moment and the technologies and tools that are associated with the project.

# Chapter 3

## Problem Analysis

### 3.1 Introduction

Analysis of the entire system is deliberated in this section. Since this system didn't have a particular client at the moment, requirement gathering protocol is not needed. But studying a similar system will help to get an idea and identify the requirement for the development. The recognized existing process of the system is described in detail. The components of the proposed system are known through the functional and non-functional requirements identified in analyzing. UML is employed in demonstrating the functionality and, therefore, the reform of the modularity in both current systems utilized in Sri Lanka and the proposed system.

### 3.2 Overview of the project

In recent years, most of the business is based on technology. All most all the industries are now driving with the technologies. All the manual process is automated. Automating a manual process helps to save money and time both. One of the essential areas that need to go with the technology is driving schools. In Sri Lanka, most of the organizations are behind the technologies. So, most of the people tend to automate this process to get the maximum benefit. The main features supported by the system as follows;

- Register new user
- Add, edit, and delete vehicle student and instructor.
- Scheduling
- Vehicle tracking
- Learning portal
- Payments
- Reporting
- Statistics

Collaborating with numbers of students at once, tracking the progress of the students, and managing daily data is a hell of a problem driving schools that use the manual system. This motivates me to implement a sophisticated automated web-based solution for driving schools in Sri Lanka.

### **3.2.1 Problem Statement analysis**

Many driver training institutes are operating across the country, and only a handful of them have opted for the digital transformation. The majority of these institutes are still committed to the manual processes of running the business. Different types of users come to get training, such as workers, students, etc. during different times of the day. Ease of scheduling a driving lesson for a trainee is low as these scheduling typically happens through phone calls, which sometimes require the trainee to make a few phone calls to get a confirmation on their specific time slot. As there is no reliable way to track the real-time location of a training vehicle[6].

Lack of a digital portal to manage the learning resources makes the overall process time consuming and costly for both the trainees and the instructors. Further, the lack of infrastructure to support online payments makes it mandatory for the trainees to make multiple trips to the institute to pay the dues, which could be a pain point.

## **3.3 Data collection protocol**

Since there is no particular client for the system, there is only one data collection protocol used and Justification of the data gathering methods used as follows.

### **3.3.1 Review Similar Systems**

Reviewing similar systems is a way of collecting data by reviewing existing systems. This is a good way to find out the requirements because once reviewing the system can find the drawbacks of the system and identify the missing features. Identify new improvements will help to identify the requirements. Those will be helpful to get as requirements for the current system. There is some similar system which shows the user feedbacks as well. This kind of feedback will more appreciate because these things can be considered as the new requirements.



## 3.4 Requirement Specification for the proposed system

The following section comprehends with a complete explanation about the proposed system, which should be essential to accomplish. Moreover, the UML diagrams will facilitate the enhanced idea about the functionalities, components afforded to the user by the future system.

### 3.4.1 Functional Requirements

Below describe the functional requirements of the system.

- The system should be able to register new users.

End-users of the system should have the option to sign up with the system. The admin user has to approve the signed-up user before the user can login to the system.

- Add, edit, and delete student, vehicle, and instructor details should be able to do via the system.

- The learning portal is another feature supported by the system.

The learning portal supported by the system will allow the trainees to download the learning resources and to assess their level by doing online quizzes.

- Scheduling is another feature supported by the system.

The system should support the allocation of predefined training slots to the trainees on a first-come first-serve basis. A logged-in trainee should have the option to select a feasible date & time slot for his/her training, on a preferred date.

- The system should be able to track the vehicle.

All logged-in users must be able to see the live location of the active training vehicles through the system.

- The system allow user to handle the online payments.

The system allow user to do online transactions securely. The system may use a dummy payment gateway for implementation purposes since some real-time payment gateways need to buy and use.

- The system allows to generate various reports  
The system allow admin to generate different reports like annual user registrations, annual user vehicle category registration and how many sessions conducted by the instructor.
- The system allows to get system statistics  
The system admin able to get an idea about student's attempted quizzes with the session participated and also from the instructor side how many sessions conducted by the instructor.

### **3.4.2 Non- Functional Requirements**

Non-functional requirements of the proposed system as follows,

- There are access levels in the system. Each student access is restricted to his/her registration records. Only authorized system administrator(s) has access to all or any student registration records.
- The database design of the future system will enhance the efficiency of the system. Maximum speed is going to be provided to the info retrieval functionality.
- Maintaining the common standard need for data transmission with the other systems will help for the system's interoperability. Transfer information from one to another place without any problem will improve the interoperability of the system.
- The usability of the system is highly gained by the user manual guide presented at the interface and with the user-friendly interfaces designed.
- Testing the software will be important to verify the implementation is a bug-free software and debugging the system components ranging from the phase of initial development.

### **3.5 Summary**

Data which is gathered through the data gathering are analyzed. Detailed description regarding the procedures and functionality of the current system is mentioned. The final phase of the chapter describes the functional, non-functional, and technical necessities of the developing system.

# Chapter 4

## System Design

### 4.1 Introduction

This chapter highlights the database construction using the Conceptual Data Modeling diagram and Database Relationship diagram. The conceptual design of the database gives an abstract view of the entities of the database. Entity Relationship Diagram reveals all the tables that the implementation should proceed with.

System Architectural design gives an idea about the overall system architecture. The application layer will decide the logic of the system. The data link layer considers the data storage while the presentation layer interacts with the system's interfaces. Software Architectural plan is implemented to show the interconnection of the system modules of the proposed system. The integration of these modules makes the entire final system of TrainMe. Detailed descriptions of these consistent components are also discussed in this chapter. In the design report, the designed interfaces are demonstrated to show how the system flows.

### 4.2 Overall System Architecture

Following System Architecture illustrates the generic flow of handling the main components of the proposed system of TrainMe. Overall System Architecture segmented the whole system into three layers consecutively, and all three layers are interconnected. Application layer, Data layer, Presentation layer are the three segments [7]. The application layer is focusing on the logic of how the users interact with the application while the Data layer is responsible for evaluating the method of how the data is kept in the database server. The presentation layer evaluates the interaction of the user with the system via the interfaces.

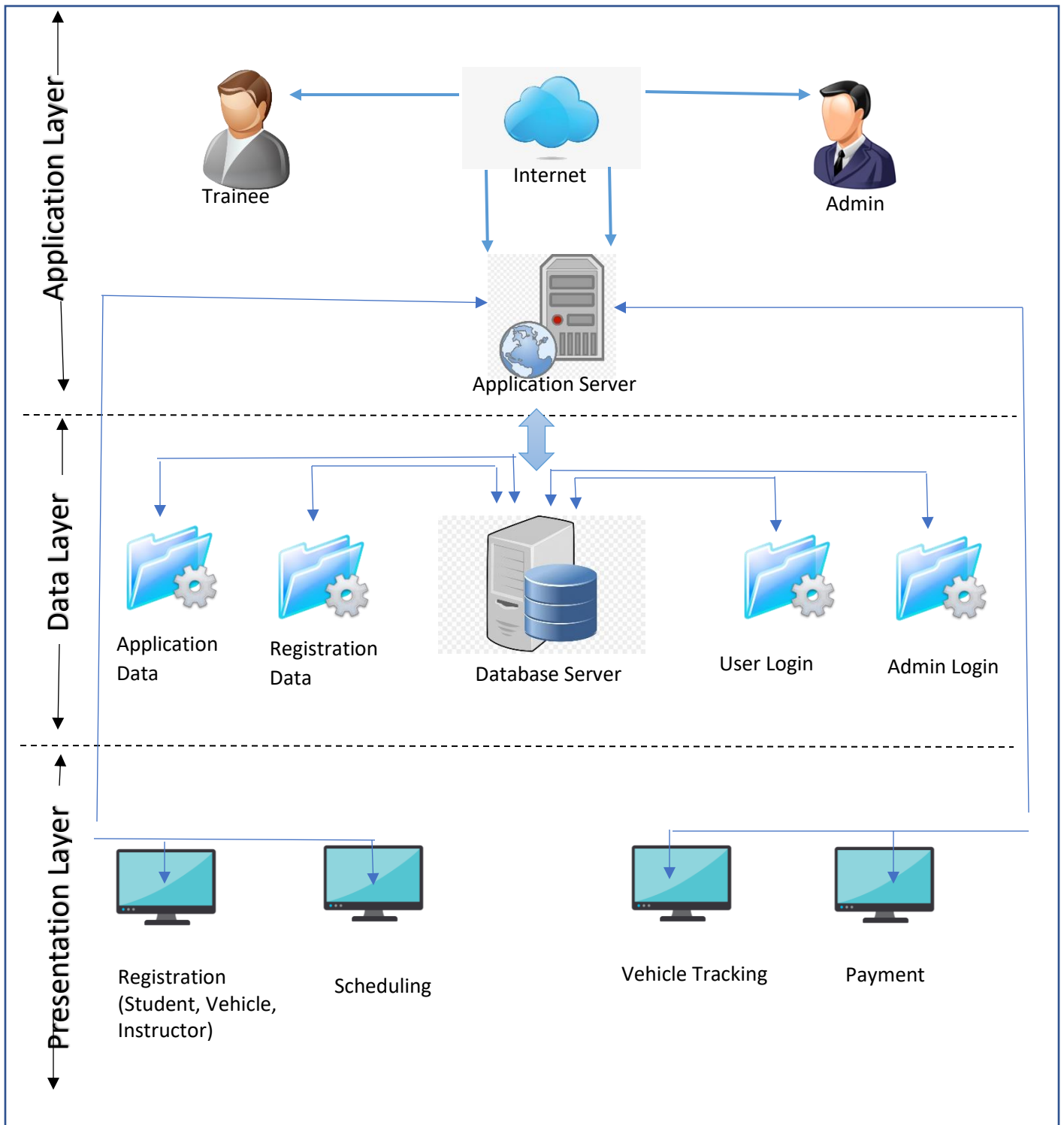


Figure 4.2.1: Overall system architecture

### **4.2.1 Application Layer**

The logical architecture of the proposed system is realized in the application layer. While being consistent with the method flow and architectural design of the system, functionality, efficiency, accuracy, and productivity of the system is supported at the application layer. The authentication to the system is supported by binding the application layer and the data layer of the system together. Once a user enters the username and the password details to the system, it goes through a verification process. Once completed, the data layer is accessed in order to correctly authenticate the user, providing him/her access to the proposed TrainMe system.

### **4.2.2 Data Layer**

The definition of how the different entities of the database interact with each other is designed at the data layer. The entirety of the information required to process the system functionality is stored in the database server to be implemented. In order to maintain the efficiency of the system, the storage of information is done effectively, using proper techniques. The data layer facilitates creating an info base, manipulate the data to create a knowledge base, and supports the basic crud operations. The application layer, data layer, and presentation layer work together to provide a holistic view of the general system architecture.

### **4.2.3 Presentation Layer**

Presentation layer dictates the usability of the system via the well-constructed user-friendly interfaces. The web-based interfaces act as the single communication channel of all the users with the proposed system.

The candidate and admin users' relevant interfaces were designed after taking all the identified functionalities of the system into consideration.

## **4.3 Data Design**

Databases are a critical component in any application development process, and therefore, it is imperative to pay attention to the multitude of database technologies that exist at the time of writing. The selection of a proper database depends on the requirements of each project. The

capabilities of the database in question should also be taken into consideration. For example, when considering an object database, the entities and relationships map to object classes and named relationships [9].

The term database design does not merely apply to the base structure of a database any longer. Database designing encompasses the overall design process, including the design of forms and queries required to create the overall database application within the database management system.

### **4.3.1 Entity Relationship Diagram of the Proposed System**

Entity Relationship diagram of the proposed system of TrainMe gives a detailed view of the database structure. The design of ER diagrams is an essential phase in the design process. The diagrams help apply the required normalizations and other techniques to the database design, ensuring the system can meet both functional and non-functional requirements. ER diagram is attached with the appendix A.

### **4.3.2 Use case Diagram**

Use case diagrams depict the entire procedure of the proposed system concerning the users of the system. They are attached with the appendix A.

## **4.4 Interface Design**

A login interface acts as an entry point for both admin and regular users to interact with the system. The hosted interfaces are accessible via the internet. Once entered into the system, the changes carried out by both types of users are stored within the web-based database server [10].

Interfaces are designed with having user-friendliness in mind. With the increased user-friendliness of the interfaces, the efficiency of the overall system increases.

## 4.4.1 Login and Register student page

Once the user types the URL, this will be the main page which will navigate to the home. With the valid user name and a password already registered user can log in to the system successfully.



Figure 4.4.1.1: Login Page

By clicking the create new account button student can register to the system.

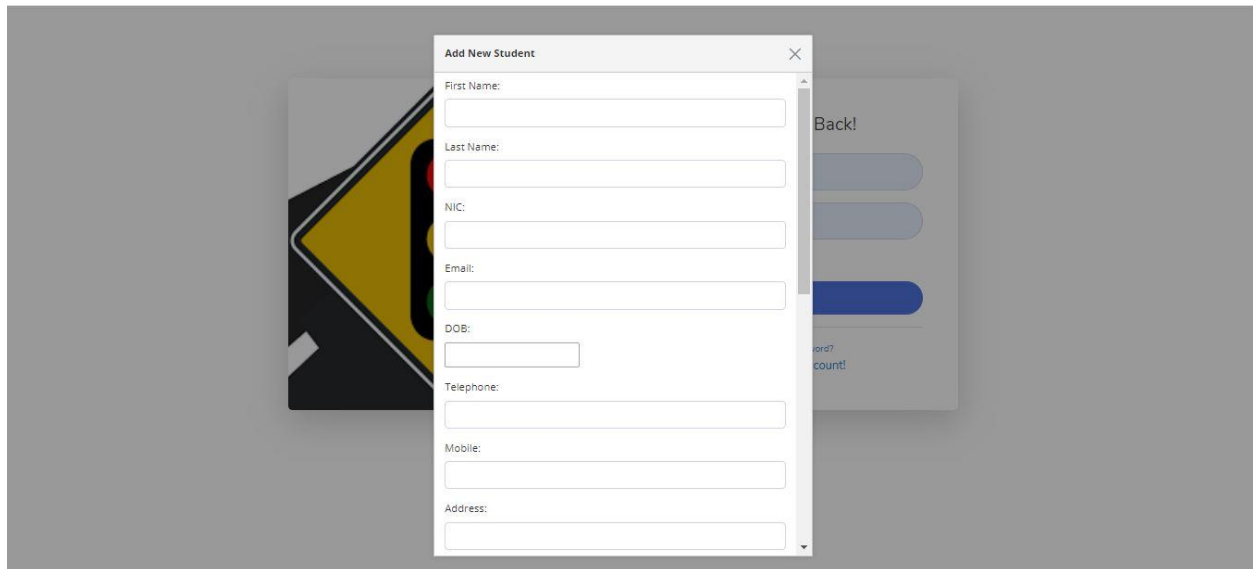


Figure 4.4.1.2: Registration of a student

## 4.4.2 Scheduling and Learning Portal

Once system admin clicks on the scheduling tab, he will navigate to the scheduling page. This allow admin to schedule a new session.

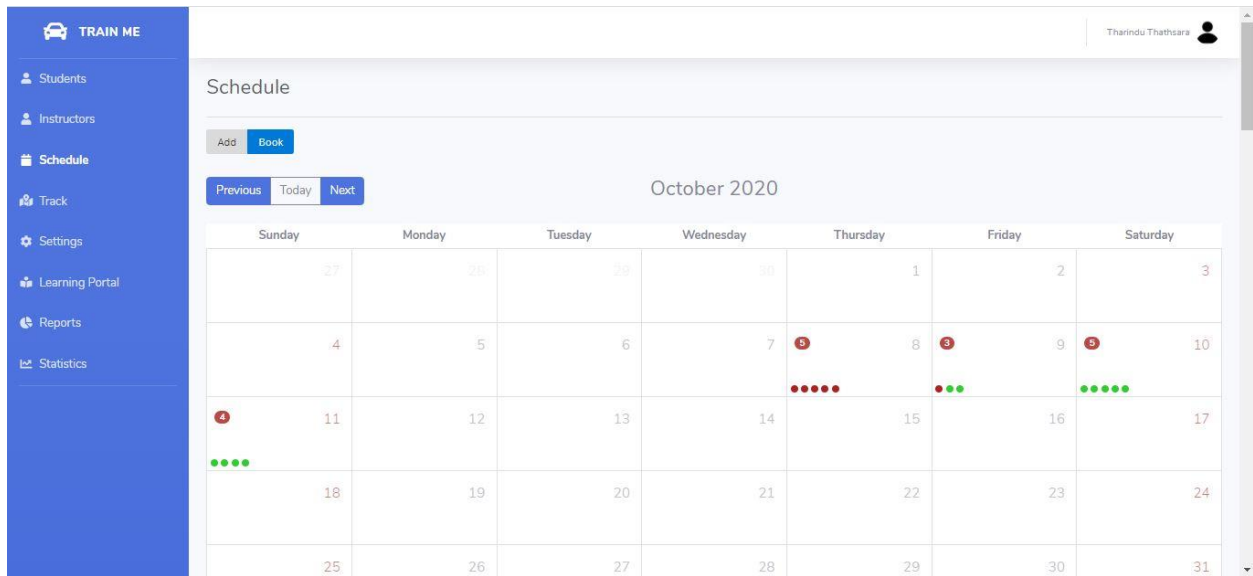


Figure 4.4.1.3: Schedule Page

When click on the learning portal allow system admin to upload learning material and also able to add multiple quizzes.



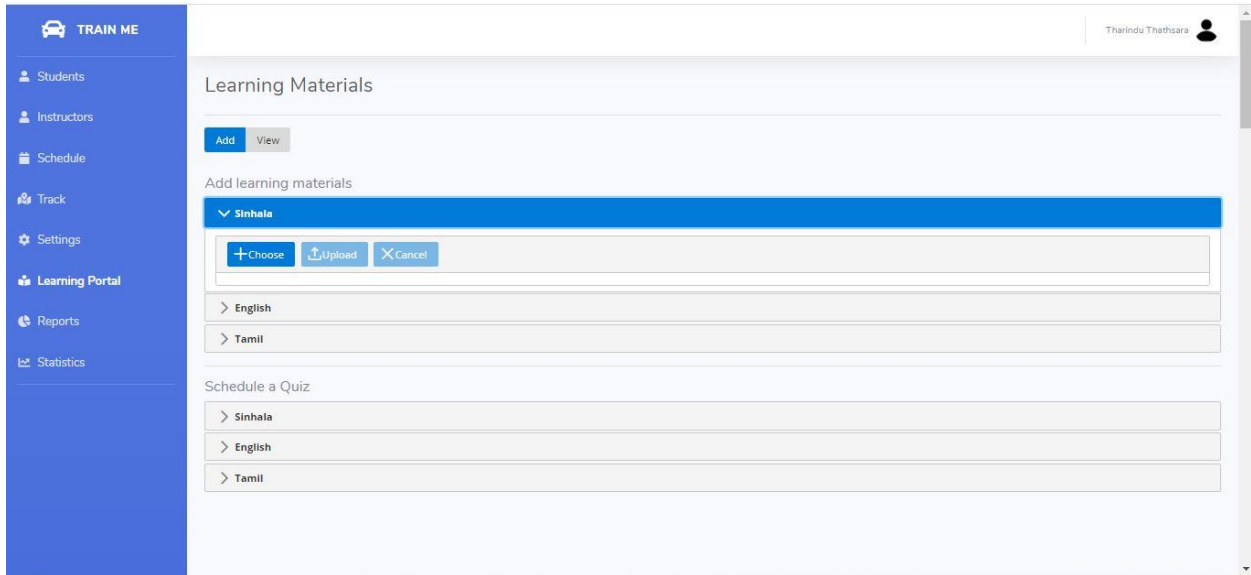


Figure 4.4.1.4: Learning portal

### 4.4.3 Add, Edit, Delete module page

The following windows are three separate modules that show add, edit and delete students, vehicle, and instructor details.

First Name	NIC	Email	Mobile	Vehicle Classes	Is Active	Has Paid	Assign Quiz
Akalaneka	926788018V	akalaneka@yopmail.com	0772489825	A1	Yes	Yes	Choose ▾
Akila	925811810V	akila@yopmail.com	0772489825	A1	Yes	No	Choose ▾
Amila	967867110V	amila@mailinator.com	0772489825	A1	Yes	No	Choose ▾
Amith	345345453	amith@yopmail.com	3453453	A1	Yes	No	Choose ▾
Anuradha	926788018V	anuradha@yopmail.com	0772489825	A1,C,G,B1	Yes	Yes	Choose ▾
Anusha	915718810V	anusha@yopmail.com	0774567156	G	Yes	Yes	Choose ▾
Aruna	926788018V	aruna@mailinator	0772489825	A1	Yes	No	Choose ▾
Aruni	923456719V	aruni@mailinator.com	0774567465	B - Manual	Yes	Yes	A1 - Quiz two ▾
Asanka	926788018V	asanka@yopmail.com	0772489825	A1	Yes	Yes	Choose ▾
Bandara	926788018V	bandara@yopmail.com	0774567156	A1,G,C,B1	Yes	No	Choose ▾

Figure 4.4.1.5: Student details

TRAIN ME

Tharindu Thathsara

Students

Instructors

Schedule

Track

Settings

Learning Portal

Reports

Statistics

> Add Vehicle Category

▼ Add/Edit Vehicle

Add Edit

Vehicle Registration Number:

Type: Bike

Capacity:

Vehicle Class: A1

Save Reset

> Add/Edit Instructor Phone Id

Figure 4.4.1.6: Add vehicle details window

TRAIN ME

Tharindu Thathsara

Students

Instructors

Schedule

Track

Settings

Learning Portal

Reports

Statistics

Instructors

Add

FirstName	LastName	NIC	Email	Mobile	IsActive
Gamini	Perera	926788018V	gamini@yopmail.com	0772489825	Yes
Isuru	Gamiath	873413410V	isuru@yopmail.com	0772489815	Yes
Radheesha	Perera	9256477819V	radheesha@yopmail.com	0714567278	Yes
Randinu	Perera	926788018V	randinu@mailinator.com	0772489825	Yes
Ranjith	Premarathne	926788018V	ranjith@mailinator.com	0772489825	Yes
Sarath	Asoka	926788018V	sarath@yopmail.com	0772489825	Yes

1

Figure 4.4.1.7: View Instructor details window

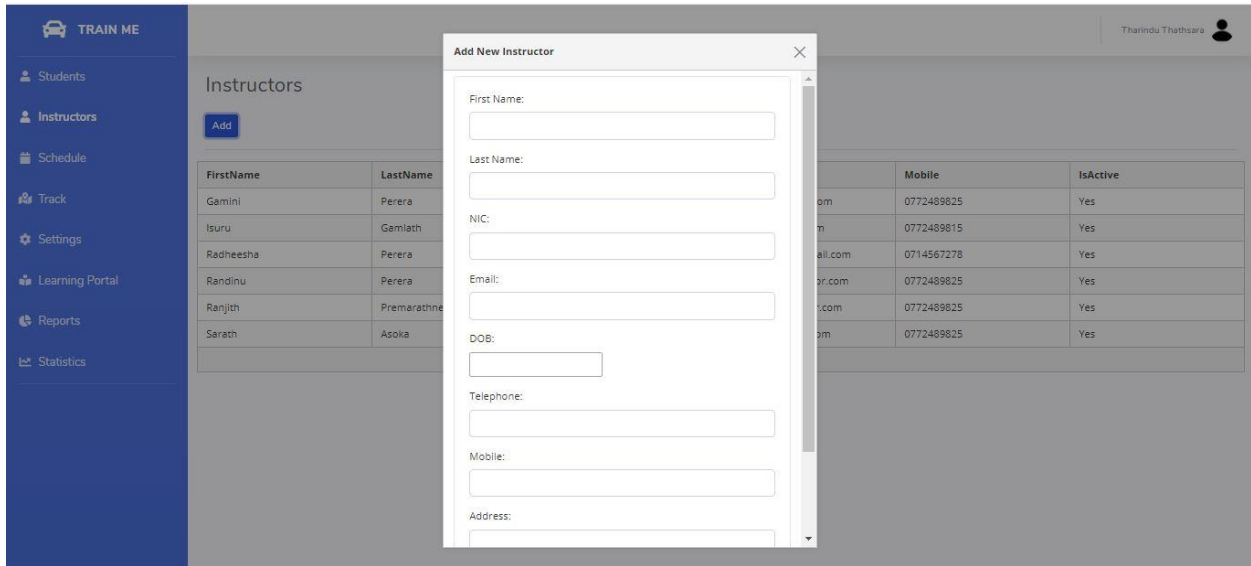


Figure 4.4.1.8: Add Instructor details

#### 4.4.4 Reporting and statistics

These UI allow system admin to generate various reports and also to get some statistics regarding the number of sessions that the student participated and also how many quizzes that user attempted. And also, statistics regarding the how many sessions conducted by an instructor.



Figure 4.4.1.9: Reporting

First Name	Last Name	NIC	Quizes Attempted	Sessions Participated
Amila	Sankalpa	967867110V		0
Sanka	Dineth	926788018V		0
Aruna	Perera	926788018V		0
Danika	Sampath	926788018V		3
Niluka	Damayanthi	926788018V	A1 - Quiz two	1
Sanaka	Danuka	926788018V	Shedule one	1
Dedunu	Perera	926788018V		0
Aruni	Dunukara	923456719V	A1 - Quiz two	2
Udari	Wimalarathna	914567119V	B- Auto Quiz 1	1
Nayomi	Wijesinghe	902311410V	B- Auto Quiz 1	2
Bimal	Jayakodi	874517781V	B- Auto Quiz 1	1

Figure 4.4.1.10: Statistics

## 4.5 Summary

This document presents details about designing the application registration, adding, editing, and delete student, vehicle, and instructor with the dashboard module.

The document explains the application's complete system architecture and how it will function through the three main layers. And also, this report illustrates the software architecture of the system.

And also, this report describes the particulars about the modules of the system. The Application submission and recruitment system contains several modules and how they work. Furthermore, this document describes the database structure and the relationships of the System. ER diagram is used to show how the database designing was working.

This summary concludes the designing details of the documents for the proposed System.

# Chapter 5

## Implementation and Testing

### 5.1 Introduction

The implementation and testing details of the overall system are discussed in this section. Selecting proper tools for the development of the application is of paramount importance. Therefore, additional details about these selections are further discussed in this section. In addition to that, the testing methodologies and objectives that are employed in developing the application are discussed intimately in this section. Testing helps identify the proper functionality of the system.

### 5.2 System Implementation

#### 5.2.1 Technology Consideration

When selecting an appropriate toolset to develop a system, the domain, and the specific requirements of the system plays a significant role. A proper set of tools should be used in order to develop a quality product.

As the complexity of a project grows, an array of tools may be needed in order to develop a proper system. Tools should be chosen wisely after carefully considering required efficiency, performance, usability, flexibility, and other system functionalities. The used set of tools should ensure that the system achieves all its functional and non-functional requirements.

#### 5.2.2 Selecting the Language

One of the critical components that have to be decided prior to the start of development is the programming language/languages. When having a layered architecture, an array of programming languages may be needed for the development tasks at each of the different layers. When selecting

these languages, compatibility naturally becomes an essential factor. Ease of development and existing compatible tools that support the development process are two other factors that should be considered when selecting a programming language.

After carefully considering an array of programming languages that are applicable at different layers of the application, the following conclusions were made. Front-end is to be developed in AngularJS. All the web-interfaces will follow the same standard. The back-end application layer is to be developed in ASP.NET and C#. (Refer to Appendix B to see the code snippets for some features.)

## 5.3 Testing

To determine if the project was able to achieve the functionality goals expected of the system, testing is required. Testing could be carried out during all the phases of the development process. A proper presentation of a test case is capable of depicting an errorless, well-performing system with the required functionalities and modules. Further, the objectives made on system testing could be used as a gauge to determine the testing levels needed to be used to accomplish the task.

### 5.3.1 Objectives of Testing a System

System development should be carried out in such a way to produce a system that is usable, intuitive, simple, and functional. Following are the set of objectives of system testing, and these should be achieved for the successful development of the application.

- **To discover the functionality of the system.**

To verify that the system is functioning correctly with the implemented functionalities, systems tests are being conducted. Going through the test cases will help to identify if the functions match the client requirement. Most of the time, prototypes are being used in test cases to explore the system's functional successes.

- **To identify any missed requirements.**

At the initial stage, the main requirements are gathered and analyzed. Testing will identify the exact requirements. Therefore, testing helps to accurately get the feedback from the user about the requirement contentment.

- **To identify any issues with connectivity with the database.**

The developed system must definitely be checked against for the connectivity issues. Critical issues will be generated if the system is not correctly connected to the database since connectivity to the database is imperative to test the system's functionality.

- **To identify errors in development.**

It is imperative to have test runs in order to detect errors in system development. Error checking of the system can be started from the initial stage of the system development. These identified issues may occur due to coding errors, errors in connecting to the database (as mentioned above), and system compatibility issues with different platforms and with the server.

- **To recognize the fulfillment of the non-functional requirements of the system.**

Through the testing process, non-functional requirements like usability, performance, efficiency, and security can be checked.

- **To correct the flow of the system.**

Conducting integration testing by connecting system modules will ensure the correct flow of the system. By achieving this goal, it can be ensured that the incompatibilities between modules are rectified.

### 5.3.2 Testing Strategy

Test strategies encompass the entirety of the levels of tests performed and different descriptions of these testing activities. The three primary levels of testing are unit testing, integration testing, and system testing.

- **Unit Testing**

Unit testing is carried out at an individual component level to identify bugs and issues at the initial development phase. Typically, these tests are carried out by programmers, and the primary objective of this is to ensure the observed outcome matches the desired outcome at a component level. Identifying usability issues early on helps with the faster delivery of a project.

Table 08 shows the module and the objective of testing the particular component.

<b>Module</b>	<b>Details</b>
User Authentication/Registration Module	This module allows registered users to successfully log in to the system via the correct username and password. Users who don't have an account are able to create a new account by clicking the registered button. Verify the valid and invalid usernames and passwords. Students and Instructors registration is done by filling the registration form.
View Module	The system should support different views and functionality to the end-users, based on the role associated with the user (admin, trainee, instructor). Test whether users can view their records based on the role.
Learning Portal	The system should support a learning portal from which the trainees could download the learning resources. Verify the uploaded learning materials can be successfully downloaded.



	Moreover, check the format of the uploaded document. If the documents are not in the correct format, they cannot be uploaded
Scheduling a training Module	The system supports the allocation of predefined training slots to the trainees on a first-come first serve basis. A logged-in trainee should have the option to select a feasible date & time slot for his/her training, on a preferred date. Test whether no conflicts occur for the same date. The same date cannot be booked by two users
Vehicle Tracking	All logged-in users must be able to see the live location of the active training vehicles through the system. A user with a training slot can use this to get on the training vehicle, traveling on a predefined route, from a predefined location. Test whether the feature functions properly.
Payments	The system allows us to handle online transactions securely. Verify by testing; the payments can be made successfully.
Reporting	The system allow admin to generate various reports. This only page should only visible to admins only. Verify by generating reports like user registrations, annual user vehicle category registration and instructor session count.
Statistics	Verify only admin can visible this page and get the statistics about the total number of session that a student participated and quizzes attempted. And also, total number of sessions conducted by the instructor.

Table 5.3.2.1: Test Plan- Unit Testing

- **Integration Testing**

In integration testing connecting all the individual modules that are identified over the unit testing phase. During the integration testing, make sure that when the modules are integrated, the existing features or new features are not broken.

<b>Module</b>	<b>Detail</b>
User Authentication	Only logged in users can be able to edit and view their profile. The delete can be done only from an admin user
View Module	
User Authentication	Every logged-in user should be able to see and download the materials in the learning portal. Only valid user can download the materials
Learning Portal	
User Authentication	Every valid user can schedule a time slot for themselves. Other users can view the calendar but can't edit the dates of some other user has booked
Scheduling a time slot	
User Authentication	Every user can track the vehicle with their logins. So, the particular assigned vehicle only can be seen via the login.
Vehicle Tracking	
User Authentication	Every user can log in from their account and safely can make an online payment. Only admin users can see all the payment records
Payment	
User Authentication	Every admin can log in from their accounts and generate reports successfully.
Reports	
User Authentication	Every admin can log in from their accounts and can view the statistics about total number of students participated for sessions and attempted quizzes. And also total number of sessions conducted by instructor.
Statistics	

Table 5.3.2.2: Test Plan- Integration Testing

- **System Testing**

The final phase of testing is system testing. At this phase, the entire system is tested to identify if it aligns with the Software Requirement Specification document. In order to carry out this particular testing, the complete implementation of the system has to be deployed with correct environment details. At this level of testing, the entirety of functional and non-functional requirements is to be tested. The primary focus of carrying out this level of testing is to identify how the developed system interacts in the real-world environment.

<b>Module</b>	<b>Details</b>
View Module	The entire flow of the whole system is checked by testing all the modules in the system. Related page redirection by clicking on the particular link or the button. Typing the valid admin username and password, identifying the admin username and password and proceed with admin functionalities. And checking the candidate username and password and allow them to proceed with their functionalities.
Learning Portal module	
Scheduling	
Vehicle Tracking	
Payment	
Reports	
Statistics	

Table 5.3.2.3: Test Plan- System Testing

### 5.3.2.1 Test Procedure

The testing of the system is carried out sequentially to verify the proper functionality of the proposed system. Developers are responsible for creating unit tests at the initial development time to verify the correct functionality at the component level. Subsequently, integration tests should be carried out in order to identify errors at a collection level. Finally, acceptance testing should be carried out, involving the users of the system to ensure the developed system meets client requirements.

### **5.3.3 Test Plan**

The testing of the system is carried out sequentially to verify the proper functionality of the proposed system. Developers are responsible for creating unit tests at the initial development time to verify the correct functionality at the component level. Subsequently, integration tests should be carried out in order to identify errors at a collection level. Finally, acceptance testing should be carried out, involving the users of the system to ensure the developed system meets client requirements.

- Recognize the system functions and modules to be tested
- Recognize the system functions and modules not to be tested
- Preparing the test cases for executing the testing on the application
- With respect to the requirement, specification evaluates test results, and output.
- Preparing actions to debug the errors

(Refer to Appendix C for test cases)

## **5.4 Summary**

When developing the software application system testing and implementation is a very important task. In this chapter, it has been described how to do the testing in module wise as well as system-wise. The next chapter of the thesis will describe the system evaluation process.

# Chapter 6

## Evaluation

### 6.1 Introduction

The previous chapter detailed the implementation phase of the software development process. Further, the testing processes of the complete system, which should be completed before handing the system over to the client to ensure an errorless real-world deployment was discussed in depth. This chapter is going to detail the factors associated with the evaluation of the application. Evaluation is going to be helpful to spot whether we achieved the expected level of functionality and performance from the developed system.

### 6.2 Evaluation of Project Practice

During the phase of project evaluation, the focus mainly lies on discovering information and evaluating the several phases used in the development cycle. Each of the above phases should be thoroughly evaluated to understand the procedures and practices that were conducted during each of these phases. Below is a list of the main phases.

- Analysis phase
- Design Phase
- Development phase
- Testing phase
- Implementation phase

#### **Analysis Phase**

The analysis phase lays the foundation for the entire project. Based on the level of success obtained at this phase, the success of the subsequent phases can be measured. During this phase, the investigation of the related domain's business process has to be done. In order to acquire the client's requirements adequately, techniques such as interviews, observations, document reviews, questionnaires, and field visits can be used. By being thorough in following this process, the exact

client requirements can be acquired. As the project does not have a client, similar systems have been analyzed to understand the requirements associated with the project.

### **Design Phase**

Having a good design helps accomplish the development of the software rapidly. The inverse of the above is true as well. With a bad design at the project's core, it becomes challenging to execute the rest of the functions in the development phase. In other words, it is at this phase that the client's requirements are converted into a logical design.

In order to develop the logical design, an array of tools, such as UML diagrams (use-case diagrams, activity diagrams, etc.), can be used. With the logical design, it is easy to conceptualize the requirements that are collected about the business process.

Two of the most important tasks that fall under the design phase are the Business process design and the user-interface design. Throughout the process, continuous comparisons should be made with the logical design to ensure that the process and the outputs are adhering to the logical design.

### **Development Phase**

At the end of the design stage, the logical design has to be converted into a real-world application. This process is carried out with the design of the interfaces and programming of the business process. By implementing a process to receive continuous feedback from the users will, in turn, help to analyze the success in the evaluation of the design phase.

### **Testing Phase**

During the testing phase, it has to be verified that the development work and end application are on the right track. This particular phase of testing includes unit testing, integration testing, system testing, and acceptance testing. Having a proper test plan from the beginning is of paramount importance to complete this phase. Proper test cases should be executed to cover all captured scenarios during the test planning.

### **Implementation phase**

The final phase discussed in this section is the implementation phase. During the implementation phase, the designs that were built in previous phases have to be materialized into a real-world application. Some of the key components that come to play in this phase are software design,

hardware design, and infrastructure design. Once the entire system is completed, we can move on to the next section, which covers the evaluation phase.

## **6.3 Evaluating the application**

This section covers how the final system is evaluated. In order to evaluate the system, the functional requirements that were gathered at the beginning should be revisited.

Different methodologies can be used to gather data about the system. Interviews and observations can be used parallelly to gather some vital information on how the system is performing in a real-world setting.

The following section will cover the techniques that are used to evaluate the end application.

### **6.3.1 Approaches to evaluating the product**

- **Interviews**

Interviews are a proper way to gain valuable information about how an end-user perceives a specific product. Though resource-intensive, this method guarantees verifiable, detailed feedback, which could be invaluable. In the future, when the proposed system establishes a proper client base, the interviews can be utilized to gain insights into the system's usage.

- **Observation**

Observation is another very effective technique at gaining valuable user feedback. The information gathered through monitoring the system in a live environment may help in determining the overall user-friendliness of the system and about user training requirements.

### **6.3.2 Evaluating Criteria**

The following section highlights the functional improvements of the new system and addresses the learnings from the requirement analysis phase.

- The functionality of the system
- User-friendliness of the system

- Accuracy of the system
- The efficiency of the system
- The productivity of the system
- Compatibility of the system

### 6.3.3 Problems Addressed

Trainees, instructors, and admin of the learning center can be taken as evaluators. And the following are the problems identified with the system studied.

Evaluators	Problems addressed in the current system.
Trainees, Instructors and Admin staff in the learning center	<ul style="list-style-type: none"> <li>• Difficult to schedule a training session</li> <li>• There is no way to keep a track on the driving vehicle.</li> <li>• There is no way to make an online payment.</li> <li>• Trainees do not have a way to download or find out the learning materials.</li> </ul>

Table 6.3.3.1: Identifying evaluators and problems addressed

### 6.3.4 Evaluating the Process Solution Provided

Here, the effectiveness of the solutions for the problems in the current system is addressed by the users.

Problems addressed in the current system	Solution Provided	User Satisfactory Level (%)
<ul style="list-style-type: none"> <li>• Difficult to schedule a training session</li> </ul>	The system should support the allocation of predefined training slots to the trainees	



	on a first-come first-serve basis. A logged-in trainee should have the option to select a feasible time slot for his/her training, on a preferred date.	
<ul style="list-style-type: none"> <li>• Vehicle tracking is very difficult.</li> </ul>	All logged-in users must be able to see the live location of the active training vehicles through the system.	
<ul style="list-style-type: none"> <li>• There is no way to do an online payment method.</li> </ul>	The system allows to handle online transactions securely.	
<ul style="list-style-type: none"> <li>• Trainees do not have a way to download or find out the learning materials.</li> </ul>	The system should support a learning portal from which the trainees could download the learning resources.	

Table 6.3.4.1: Evaluating the Process Solution Provided

### 6.3.5 Other Project Accomplishments

The following lists down the achievements acquired other than the achievements mentioned above. The knowledge gained during the development of the project is necessary to face future project development challenges. It’s a pleasure to apply project management and programming skills, which I have learned beforehand, in a real-world environment has been a rewarding experience. During this time period following shows the things learned,

- Sharpen the project management skills learned
- Gained knowledge on comparing and contrasting different programming languages to fit a suitable objective
- Learned about time management.

- Writing different reports help to improve writing skills.

## **6.4 Summary**

This chapter was assigned to discuss the evaluation process of the entire development process. The evaluation process is a good measuring rod to determine the development success according to the user request and right business process. Moreover, during this chapter, discuss the techniques used for the evaluation and also the results as well as the user feedback. The final chapter will give the future work and the brief on the project.

# Chapter 7

## Conclusion and Future Work

### 7.1 Introduction

The above chapter separated to discuss the evaluation process of the project. The goal of this chapter is to conclude the development of the TrainMe web-based system.

This chapter defines the conclusion of the problem. Mainly focused on the finding and how those findings apply for different situations. Moreover, this includes the lessons learned, project practices, project results, and future enhancement.

### 7.2 Project Results Overview

The goal of this project is to provide a feature-rich digital solution to manage limited resources efficiently and effectively in a driver training institute. The proposed system is feasible enough to provide user registration, vehicle tracking, scheduling a time slot for training, learning portal to access learning materials, and payment to do online transactions. Key usable people of this system are trainees who come to get driving lessons, instructors, and learning center owners (Super Admin) of the product.

Since there is no specific client for the project, the results have been collected using fact-gathering techniques such as reviewing existing systems. This information helps to study and find an analytical perspective like operational, economic, and social easily.

### 7.2 Project Result Outline

The success of reaching the intention relies on the value of accomplishing the objectives of the system. As the postgraduate student retrieving the aim in the development is helpful to get the real-time experience in the system development.

## 7.2.1 Lessons Learnt Via the Achievement of Project Objectives

This session emphasizes the learning outcomes of the project against the project objectives.

Purposes	Lessons Learnt
<ul style="list-style-type: none"> <li>System domain planning</li> </ul>	<ul style="list-style-type: none"> <li>In system development, identifying and designing the scope of the system is important.</li> <li>From the business study on the development gathered new knowledge.</li> </ul>
<ul style="list-style-type: none"> <li>Identify the proper requirement gathering techniques for the requirement gathering.</li> </ul>	<ul style="list-style-type: none"> <li>Based on the domain identified, identified the most suitable fact-gathering protocol</li> </ul>
<ul style="list-style-type: none"> <li>Gathered data analysis</li> </ul>	<p>Able to get the following results,</p> <ul style="list-style-type: none"> <li>The ability to solve problems.</li> <li>Developed analytical skills.</li> <li>Develop decision-making ability.</li> </ul>
<ul style="list-style-type: none"> <li>Recognizing and plan the modules of the proposed system.</li> </ul>	<ul style="list-style-type: none"> <li>Getting the retrieving of important data and classifying them into units will be getting by the design phase.</li> <li>Able to get hands-on experience with some structured diagrams, flow charts, and pseudo-codes.</li> </ul>

	<ul style="list-style-type: none"> <li>• Learned how far the design theories come into practice on the real situation</li> </ul>
--	--

Table 7.2.1.1: Accomplishment of Project Objectives

### 7.3 Future Improvements

With the limited time period, implementing all the features for a state becomes a difficult situation. But introducing some other components to the development will help to improve the efficiency of the associated method. As a future improvement expecting to introduce a mobile app to track the vehicle so it will be easy for users to use the app to track the vehicles in real-time. Giving them access to both mobile app as well as web applications, will add more usability and richness to the product.

Moreover, planning to add an SMS feature that can give real-time updates regarding the cancellation of the time schedule and if some change was done to the time slot, the user would be able to get to know about as soon as possible.

### 7.4 Summary

This thesis documents all procedures and methods related to the implementation process of the TrainMe web-based management system. The thesis's flow has come across the discussion of similar systems, technologies used, design, testing, evaluation, and the conclusion of the project, which includes a walkthrough on possible future improvements to this project. This chapter marks the end of the thesis.

# References

- [1] *Umpir.ump.edu.my*, 2019. [Online]
- [2] TENGKU. IBRAHIM, "DRIVING SCHOOL STUDENT MANAGEMENT SYSTEM," vol 24, 2005.
- [3] *Matchinglabs.com*, 2019. [Online]
- [4] MengXF, Lu HJ, et al. Data extraction from the Web-based on predefined schema. *Journal of Computers Sciences and Technology*, 2001,17(4):377~388.
- [5] Silberschatz A, Stonebraker M, Ullman JD. Database systems: Achievements and opportunities. *CACM*, 1991,34(10):110~120.
- [6] <https://www.scribd.com/doc/65535197/Main-Project-Proposal-Driving-School-Management-Information-System>
- [7] <https://prezi.com/y57a5dguv-k7/driving-school-managment-system/>
- [8] <http://www.matchinglabs.com/wp-content/uploads/2015/10/DrivingLABS-Driving-School-Management-System.pdf>
- [9] <https://pdfs.semanticscholar.org/add3/3d6ad4958cb9346824c062b370fe98022478.pdf>
- [10] [https://www.researchgate.net/publication/200629237\\_The\\_graduated\\_driver\\_licensing\\_system\\_in\\_New\\_South\\_Wales](https://www.researchgate.net/publication/200629237_The_graduated_driver_licensing_system_in_New_South_Wales)
- [11] <https://www.scribd.com/document/385103982/147-Motor-Driving-School-Management-System-Synopsis-pdf>

# Appendix A: Diagrams

Figure A.1: ER Diagram

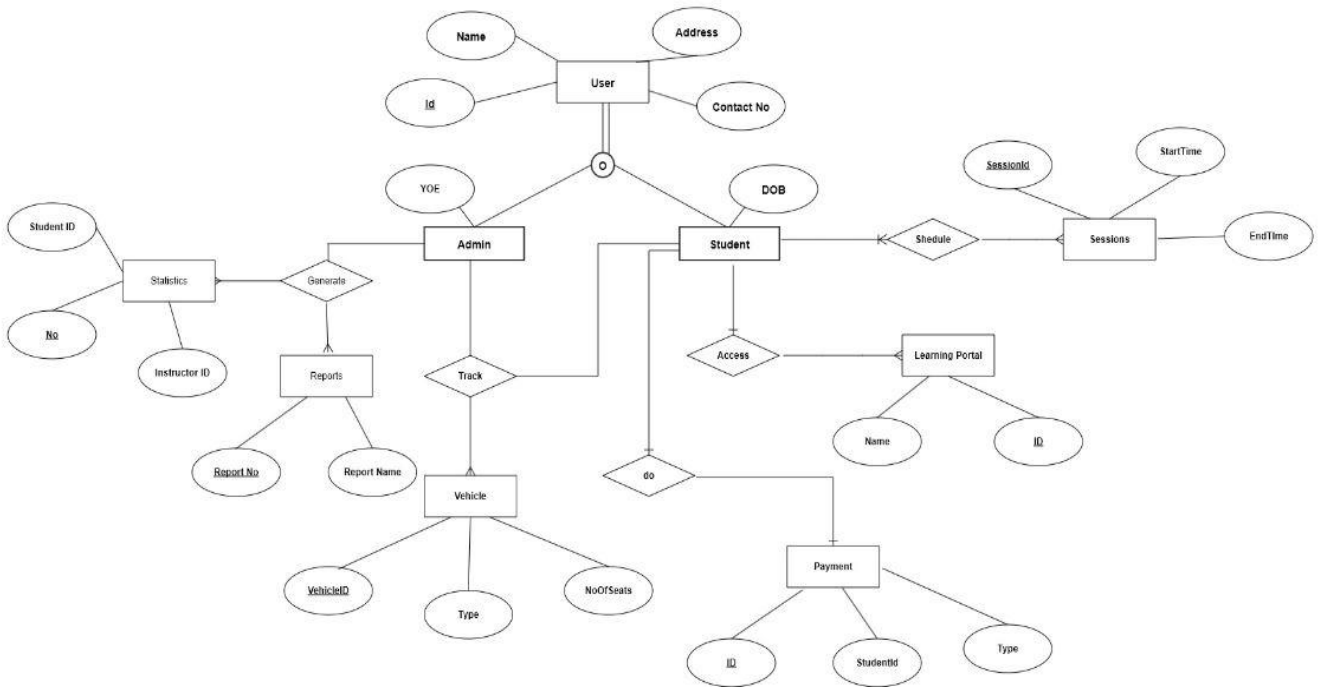
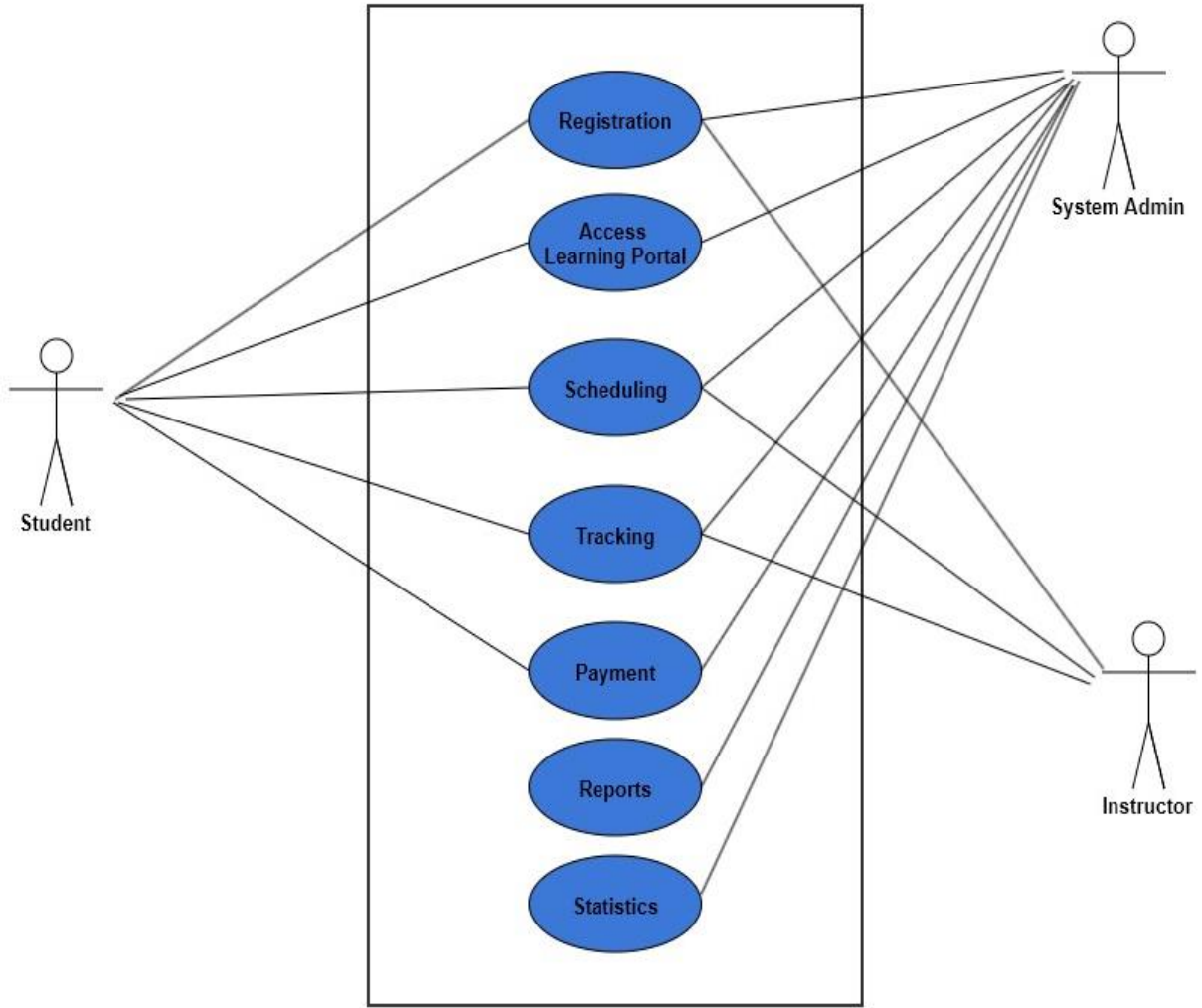


Figure A.2: High-Level Use case diagram





## Appendix B: Code snippets for some development

Figure B.1: Database table structure

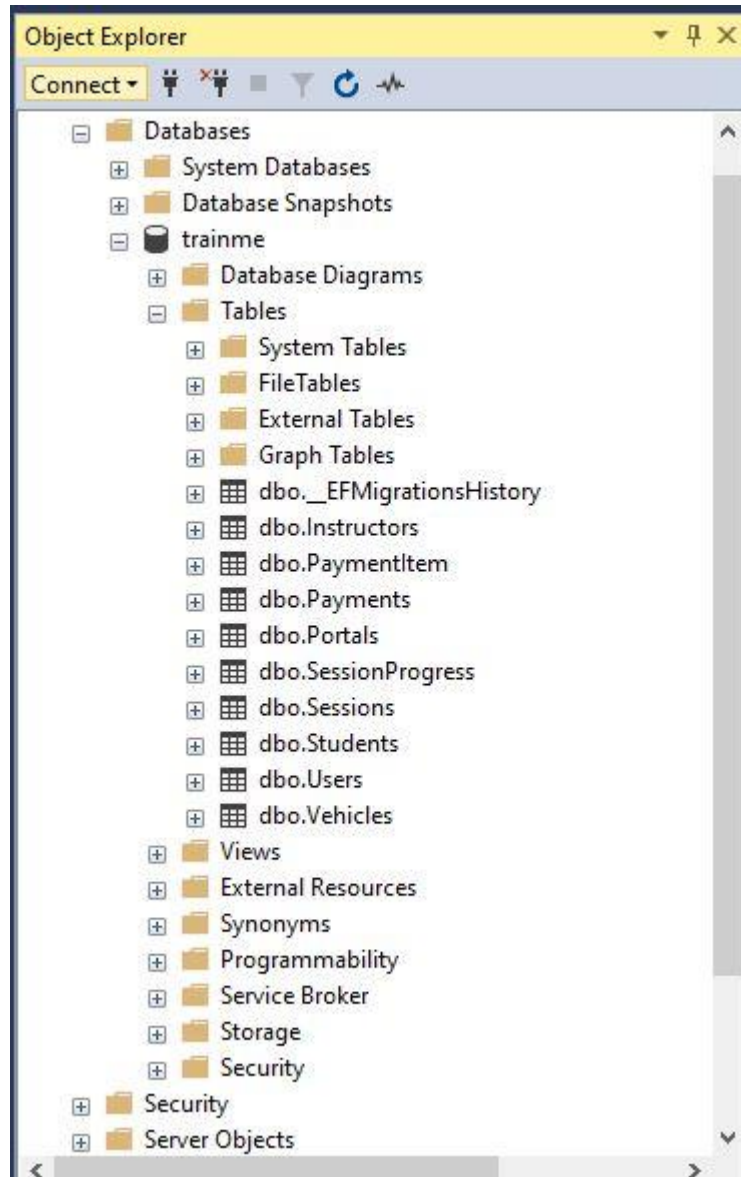


Figure B.2: Login Page

```
src > app > components > login > login-form > login-form.component.html > form.border.form-login
1 <form [formGroup]="loginForm" (ngSubmit)="onSubmit()" class="border form-login">
2   <div class="form-group">
3     <label for="email">Email</label>
4     <input type="email" class="form-control" id="email" aria-describedby="emailHelp" formControlName="email" placeholder="Enter email">
5   </div>
6   <div class="form-group">
7     <label for="password">Password</label>
8     <input type="password" class="form-control" id="password" formControlName="password">
9   </div>
10  <button type="submit" class="btn btn-primary">Login</button>
11 </form>
```

Figure B.3: Services written for Login with different user levels

```
src > app > components > login > login-form > TS login-form.component.ts > ...
12
13 loginForm: FormGroup;
14
15 constructor(
16   private readonly loginService: LoginService,
17   private readonly router: Router
18 ) { }
19
20 ngOnInit() {
21   this.initForm();
22 }
23
24 initForm() {
25   this.loginForm = new FormGroup({
26     email: new FormControl(''),
27     password: new FormControl('')
28   })
29 }
30
31 onSubmit() {
32   console.log(this.loginForm.value);
33   this.loginService.login(this.loginForm.value).subscribe(
34     (data: any) => {
35       console.log(data);
36       this.loginService.storeCredentials(data.name, data.token, data.role, data.email);
37       if(data.role === 1) {
38         this.router.navigate(['/student']);
39       } else if(data.role === 2) {
40         this.router.navigate(['/instructor']);
41       } else if (data.role === 3) {
42         this.router.navigate(['/admin']);
43       } else {
44       }
45     }, (err: any) => {
46       console.error(err);
47     }
48   )
49 }
50 }
```

Figure B.4: Service written to add, edit, delete and get student details

```
src > app > services > student > TS student.service.ts > StudentService > getStudent
14 constructor(private api: ApiService) { }
15
16 getStudents(): Observable<any> {
17   return this.api.httpGet(this.baseUrl);
18 }
19
20 addStudent(data: Student) {
21   console.log(data);
22   return this.api.httpPost(this.baseUrl, data);
23 }
24
25 getStudent(id: string) {
26   console.log(id);
27   return this.api.httpGet(`${this.baseUrl}/${id}`);
28 }
29
30 getStudentByEmail(email: string) {
31   console.log(email);
32   return this.api.httpGet(`${this.baseUrl}/email/${email}`);
33 }
34
35 editStudent(id: string, data: string) {
36   console.log(data);
37   return this.api.httpPut(this.baseUrl, id, data);
38 }
39
40 removeStudent(id: string) {
41   console.log(id);
42   this.api.httpDelete(this.baseUrl, id);
43 }
```

Figure B.5: Service written to add, edit, delete and get vehicle details

```
src > app > services > vehicle > TS vehicle.service.ts > VehicleService
14
15   getVehicles(): Observable<any> {
16     return this.api.httpGet(this.baseUrl);
17   }
18
19   addVehicle(data: Vehicle) {
20     console.log(data);
21     return this.api.httpPost(this.baseUrl, data);
22   }
23
24   getVehicle(id: string) {
25     console.log(id);
26     return this.api.httpGet(`${this.baseUrl}/${id}`);
27   }
28
29   editVehicle(id: string, data: string) {
30     console.log(data);
31     return this.api.httpPut(this.baseUrl, id, data);
32   }
33
34   removeVehicle(id: string) {
35     console.log(id);
36     return this.api.httpDelete(this.baseUrl, id);
37   }
38
39   getVehicleForDropdown() {
40     return this.api.httpGet(`${this.baseUrl}/short`);
41   }
42 }
43
```

# Appendix C: Test Cases

Figure C.1: Test cases written for Student Details

Select Suite:

- Root
- Student
- Instructor
- Vehicle

Search suites (type keyword and press enter)

ID	Title	Suite	Priority	Last Run Status	Last Run	Runs	Avg. Execution Time	Created By
1	Verify the functionality of saving student details	Student	high	✓	4 days ago	1	a few seconds	Ru
2	Verify the page UI	Student	normal	...	N/A	0	N/A	Ru
3	Verify the functionality of the back button	Student	normal	...	N/A	0	N/A	Ru
4	Verify the data in the table shown properly	Student	high	...	N/A	0	N/A	Ru
5	Verify the pagination and the maximum no of records in the table	Student	normal	...	N/A	0	N/A	Ru
6	Verify functionality of editing student details	Student	high	...	N/A	0	N/A	Ru
7	Verify functionality of deleting student details	Student	high	...	N/A	0	N/A	Ru
8	Verify the responsiveness in major browsers	Student	normal	...	N/A	0	N/A	Ru
23	Verify the functionality of saving student details	Student	high	...	N/A	0	N/A	Ru
24	Verify the page UI	Student	normal	...	N/A	0	N/A	Ru

Figure C.2: Test cases written for Instructor Details

TrainMe
Overview
Test Cases
Milestones
Test Executions
Reports
Settings
Hello Runolgirly
Jump to: TrainMe

Manage
Add suite
Add test case
Import
Export
Reusable Steps

Select Suite:

- Root
- Student
- Instructor
- Vehicle

Search suites (type keyword and press enter)

**Test Cases in Instructor**

that match all filters + Add Filter

7 test cases of 30 Add test case Execute

ID	Title	Suite	Priority	Last Run Status	Last Run	Runs	Avg. Execution Time	Created By
9	Verify the functionality of saving Instructor details	Instructor	high	...	N/A	0	N/A	Ra
10	Verify the page UI	Instructor	normal	...	N/A	0	N/A	Ru
11	Verify the functionality of the back button	Instructor	normal	...	N/A	0	N/A	Ru
12	Verify the data in the table shown properly	Instructor	normal	...	N/A	0	N/A	Ra
13	Verify the pagination and the maximum no of records in the table	Instructor	normal	...	N/A	0	N/A	Ru
14	Verify functionality of editing Instructor details	Instructor	high	...	N/A	0	N/A	Ru
15	Verify functionality of deleting Instructor details	Instructor	high	...	N/A	0	N/A	Ra

Figure C.3: Test cases written for Vehicle module

The screenshot displays the TrainMe application interface. The top navigation bar includes tabs for Overview, Test Cases, Milestones, Test Executions, Reports, and Settings. The Test Cases tab is active, showing a sub-header 'Test Cases in Vehicle'. Below this, there are filters and an 'Execute' button. The main content area contains a table with 7 test cases, each with a unique ID, title, suite name, priority, last run status, last run date, number of runs, average execution time, and a 'Created By' field.

ID	Title	Suite	Priority	Last Run Status	Last Run	Runs	Avg. Execution Time	Created By
16	Verify the functionality of saving vehicle details	Vehicle	high	...	N/A	0	N/A	Ru
17	Verify the page UI	Vehicle	normal	...	N/A	0	N/A	Ru
18	Verify the functionality of the back button	Vehicle	normal	...	N/A	0	N/A	Ru
19	Verify the data in the table shown properly	Vehicle	high	...	N/A	0	N/A	Ru
20	Verify the pagination and the maximum no of records in the table	Vehicle	normal	...	N/A	0	N/A	Ru
21	Verify functionality of editing student details	Vehicle	high	...	N/A	0	N/A	Ru
22	Verify functionality of deleting student details	Vehicle	high	...	N/A	0	N/A	Ru