**MORSE CODE TRANSLATOR USING IMAGE PROCESSING TECHNIQUES**

A PROJECT REPORT PRESENTED BY

R. D. VIHARAGEDARA

(S/16/494)

**TO THE DEPARTMENT OF STATISTICS AND COMPUTER SCIENCE**

**FACULTY OF SCIENCE**

*in partial fulfillment of the requirement*

*for the award of the degree of*

**B.Sc. (Honours) in Computer Science**

of the

**UNIVERSITY OF PERADENIYA**

**SRI LANKA**

**2020**

**DECLARATION**

I do hereby declare that the work reported in this project report was exclusively carried out by me under the supervision of Dr. Amalka Pinidiyaarachchi. It describes the results of my own independent work except where due reference has been made in the text. No part of this project report has been submitted earlier or concurrently for the same or any other degree.

Date: …………………… ……………………

Signature of the candidate.

Certified by:

1. Supervisor: Dr. Amalka Pinidiyaarachchi

Date: …………………… Signature: ……………………

2. Head of the Department: Dr. Ruwan Nawarathna

Date: …………………… Signature: ……………………

**ACKNOWLEDGMENTS**

**TABLE OF CONTENTS**

**DECLARATION ii**

**ACKNOWLEDGMENTS iii**

**TABLE OF CONTENTS iv**

**TABLE OF FIGURES v**

**LIST OF ABBREVIATIONS vi**

**CHAPTER 1: INTRODUCTION 1**

1.1 Problem Statement **1**

1.2 Aim of The Project **2**

1.3 Objectives of The Project **2  
CHAPTER 2: LITERATURE REVIEW 3**

**CHAPTER 3: METHODOLOGY 4**

3.1 Approach **4**

3.2 Proposed Technologies **5**

3.3 System Design **6**

3.4 Work Plan **7  
CHAPTER 4: EXCLUSIVE SUMMARY 8**

**REFERENCES 10**

**TABLE OF FIGURES**

**Figure 3.1:** Basic Diagram About System Design. **6**

**Figure 3.2:** Work Plan **7**

**LIST OF ABBREVIATIONS**

CSS Cascading Style Sheets  
ER Entity–relationship  
HTML Hypertext Markup Language  
UI User interface  
UX User Experience

**CHAPTER 1**

**INTRODUCTION**

**1.1 Problem Statement**

Morse code is a method encode characters in a text as sequences of two different signal durations, called dots and dashes. Morse code was invented by Samuel F. B. Morse in the 1830s then advanced in the 1840s. Samuel Morse received a U.S. patent, US1647A, for dot-dash telegraphy signals on June 20, 1840.[1][2]

The International Morse Code encodes the 26 English letters A through Z, some non-English letters, the Arabic numerals and a small set of punctuation and procedural signals. There is no distinction between upper- case and lower-case letters. Each Morse code symbol is formed by a sequence of dots and dashes. The dot duration is the basic unit of time measurement in Morse code transmission. The duration of a dash is three times the duration of a dot. Each dot or dash within a character is followed by period of signal absence, called a space, equal to the dot duration. The letters of a word are separated by a space of duration equal to three dots, and the words are separated by a space equal to seven dots.[1]

Morse code is usually transmitted by on-off keying of an information-carrying medium such as electric current, radio waves, visible light, or sound waves. The current or wave is present during the time period of the dot or dash and absent during the time between dots and dashes. Even, it can be used with pen and paper or with the parts of the body like eyes or fingers. [1][2]

**“What hath God wrought”** is the first official message sent by Samuel F.B. Morse on May 24, 1844, to open the Baltimore - Washington telegraph line. [2]

In the past, Morse code had extensive usage, especially in the military. Although Morse code has a limited usage area today, it is still being used in aviation, amateur radio activities, and assistive technology (AT).[2]

Although Morse code had a steep learning curve in the past, thanks to modern applications like Morse code translator [4] or Gboard keyboard [3], it is easier to learn Morse code nowadays.

As my project, I would like to develop a software that translate morse coded message to a text format using image processing techniques.

**1.2 Aim of The Project**

The aim of this project will be to develop a

**1.3 Objectives** **of The Project**

The objectives of the project as follow,

* To develop a platform for On-Demand Vehicle Detailing and Repair Service
* Create a platform that to improve the interconnectivity between customers and vehicle repair service provider.
* Create a platform that allow customers to request their vehicle repair or maintenance needs online.
* Create a platform that allow customers to choose the most suitable service for them.
* Create a platform that customers can keep an easily manage their vehicle maintenance records.
* Create new job opportunities (mobile mechanic)
* Create a brand-new startup in Sri Lanka (On-Demand Vehicle Detailing and Repair)

**CHAPTER 2**

**LITERATURE REVIEW**

Most people are accustomed to taking their vehicle to a repair shop for service. As they say, old habits die hard. Usually going to a repair shop can be cumbersome, especially if it’s just for a minor service, such as an oil change or battery replacement. Some people don't have the time, or perhaps no way of getting it there because it was completely broken down. Regardless of the reason, having to bring a vehicle into the shop always feels like a dreaded chore [5].

But using this proposed solution, customers can get just about any vehicle maintenance or repair needs done from the comfort of their own home. Not only it is incredibly convenient, it also offers several other benefits over traditional, brick and mortar auto repair shops.

Vehicle owners keep their vehicle maintenance records manually. This is very inconvenient and it takes another effort to keep them in order. They could be easily destroying or can be missed placed because most of the records are written on papers. And if someone has more than one vehicle it’s very hard to keep track on records of all the vehicles they have. But in my proposed solution the customer has the ability to add more than one vehicle under their account. And also, they can see the past and upcoming repair and maintenance records related to each vehicle they have. So, they can easily manage their vehicle’s repair and maintenance.

It’s hard to know the prices of the services by going to a repair shop and you can’t guarantee the service you had. In my solution the customers are able to see the prices and other customers feedbacks about the services they had. So, they can choose the most suitable service for them.

**CHAPTER 3**

**METHODOLOGY**

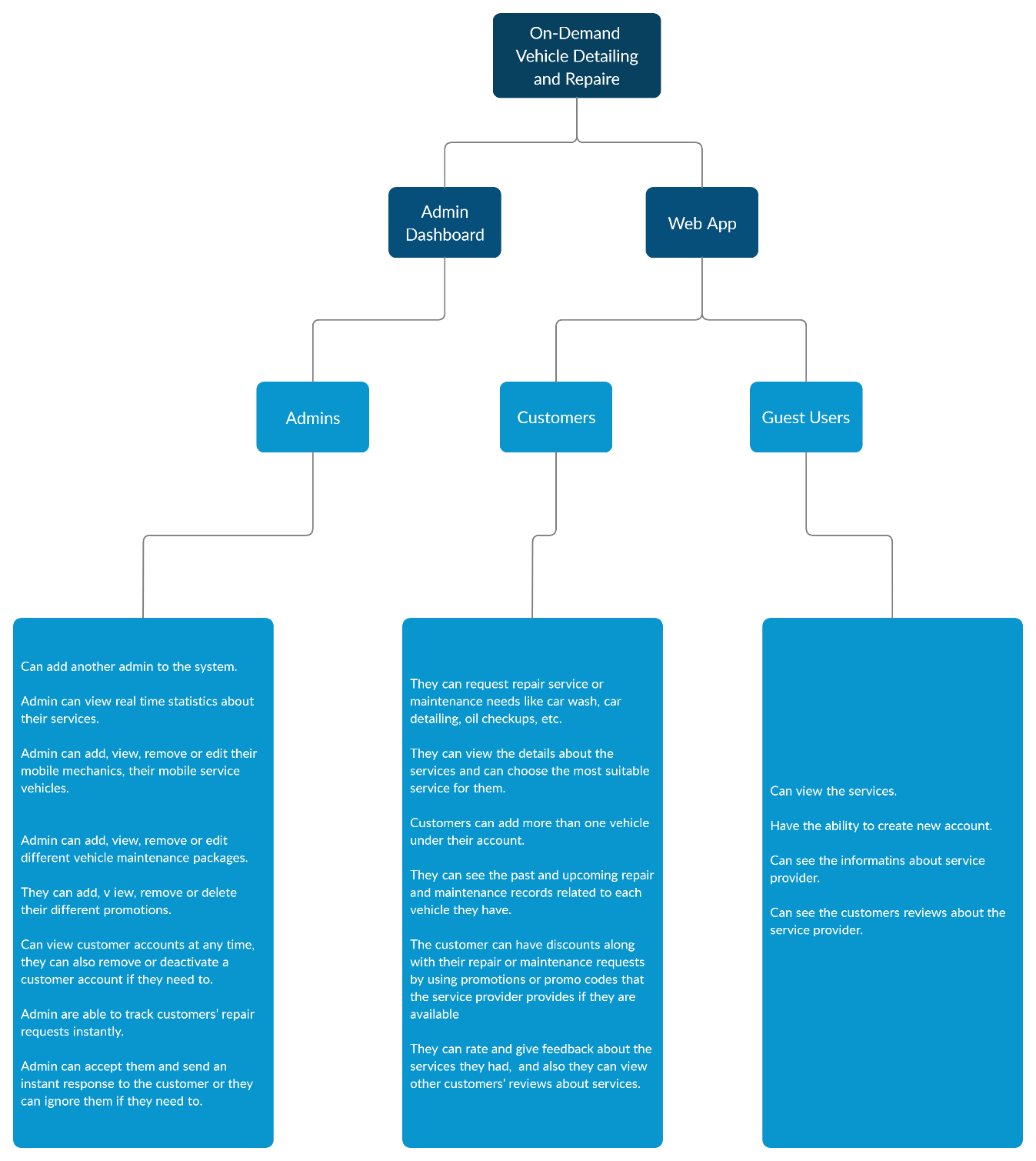
**3.1 Approach**

* Requirements Analysis
  + Gathering the client requirements.
  + Understanding the deficiencies of the current systems.
* Design
  + Design the database of the system using ER diagrams.
  + Design the UI/UX.
* Development
  + Develop the system using the proposed technologies.
* Testing
  + Testing the developed system functionalities with test data stored in the database.
* Deploy
  + Deploy the tested system for customer use.
* Maintenance
  + Maintaining the deployed system according to the future changes requirements from the client.

**3.2 Proposed Technologies**

* Frontend
  + HTML and CSS
  + React
* Backend
  + Spring boot
* Database
  + MySQL
* Testing
  + Junit
  + Spring Test

**3.3 System Design**



**Figure 3.1**: Basic Diagram About System Design.

**3.4 Work Plan**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Action | Duration | | | | | | | | | | | | | | | | | |
| Jun | | Jul | | Aug | | Sep | | Oct | | Nov | | Dec | | Jan | | Feb | |
| Project Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requirements Analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design  ER Diagram |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design  UI/UX |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Deploy |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Maintainence |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Figure 3.2**: Work Plan

**CHAPTER 4**

**EXCLUSIVE SUMMARY**

In this project there will be two modules with complete user interfaces such as an admin dashboard for service provider and a customer web app. The customers have to register to the system by using an email or phone number as they prefer. Admin accounts will be provided by the service provider.

After successfully registering to the system, customers can login to their accounts. Beside the basic profile functionalities like login, logout, search, edit and remove, the system will provide various functionalities and services such as requesting repair service or maintenance needs like car wash, car detailing, oil checkups, etc.

The customer can add more than one vehicle under their account. They can view, edit or remove them. Also, they can see the past and upcoming repair and maintenance records related to each vehicle they have, so they can easily manage their vehicle’s repair and maintenance.

The customer can have discounts along with their repair or maintenance requests by using promotions or promo codes that the service provider provides if they are available. They can rate and give feedback about the services they had, and also, they can view other customers' reviews about services.

For admins to access the system, they need to use the username and password issued by the service provider. Only an admin can add another admin to the system. After login to their profile they have the opportunity to edit their personal information as they wish. The system will provide a number of functionalities to the admins in order to keep the service management under control.

They can view real time statistics about their services like the number of mechanics and mobile service vehicles available, the number of service requests completed, ongoing service requests, latest service requests, notifications, most and least popular service options, heat maps (Areas that most service requests came from), etc.

Admins can easily manage the mobile mechanics and mobile service vehicles by using the system. They can add, view, remove or edit their mobile mechanics, their mobile service vehicles and different vehicle maintenance packages they provide like car wash, car detailing, oil changes, etc. Also, they can add, view, remove or delete their different promotions.

Admins can view customer accounts at any time, they can also remove or deactivate a customer account if they need to. They are able to track customers’ repair requests instantly. They can accept them and send an instant response to the customer or they can ignore them if they need to.

Besides them there will be some special functionalities to the admins, like send push notifications to all, disable the system temporarily, etc.

All the information and the data in the whole system would be permanently stored in the database along with a backup for recovery if any system crashes occur.

**REFERENCES**

1 - <https://en.wikipedia.org/wiki/Morse_code>

2 - <https://morsedecoder.com/>