A Project Report on

IOT Enabled Traffic Diversion System for Emergency Services

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Pushkar Mohanish Telavane(19104011)
Vatsal Ajay Kumar Singh(19104006)
Shubhangi Arvind Tripathi(19104067)

Under the Guidance of

Prof. Mandar Ganjapurkar
Prof. Apurva Chaudhari

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2022-2023

Approval Sheet

This Project Report entitled ***"IOT Enabled Traffic Diversion System for Emergency Services"*** Submitted by ***"Pushkar Mohanish Telavane"(19104011),"Vatsal Ajay Kumar Singh"(19104006),"Shubhangi Arvind Tripathi"(19104067),***is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

Prof. Apurva Chaudhari
Co-Guide

Prof. Mandar Ganjapurkar
Guide

Dr. Kiran Deshpande
Head Department of Information Technology

Place: A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"IOT Enabled Traffic Diversion System for Emergency Services"*** submitted by ***"Pushkar Mohanish Telavane" (19104011), "Vatsal Ajay Kumar Singh" (19104006), "Shubhangi Arvind Tripathi" (19104067)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2022-2023.

Prof. Apurva Chaudhari
Co-Guide

Prof. Mandar Ganjapurkar
Guide

Dr. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **IOT Enabled Traffic Diversion System for Emergency Services.** We take this opportunity to express our sincere thanks towards our guide **Prof. Mandar Ganjapurkar** & Co-Guide **Prof. Apurva Chaudhari** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Dr. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Sonal Jain** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Pushkar Mohanish Telavane
19104011

Vatsal Ajay Kumar Singh
19104006

Shubhangi Arvind Tripathi
19104067

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

_____

(Pushkar Mohanish Telavane - 19104011)

_____

(Vatsal Ajay Kumar Singh - 19104006)

_____

(Shubhangi Arvind Tripathi - 19104067)

Date:

Abstract

The rapid rise in vehicle numbers, which has grown serious in the wake of the startling increase in global population, is one of the most significant effects of modern technology development. The delay in getting to the destination is therefore the main problem here. The delay sometimes is attributed to the time it takes to get to the hospital or to get the first assistance. The sad scenario is made more worse in India by the traffic. Even if the government is taking a number of initiatives, a comprehensive solution is needed to reduce the time taken at each step, such as assessing the patient's status, calling the medical aid, or making the nearest aid available. As a result, traffic congestion is now a major issue in the majority of the world's nations. With the use of IOT, to create a system that will allow emergency vehicles, such as ambulances, to get at their location as quickly as possible. A smartphone application that will allow drivers and traffic controllers to communicate with each other is currently being developed. We would provide the ambulance's driver the ability to specify the locations of the source and destination. Digital screens placed near traffic lights will show the lane that the traffic controller chose. Other highway users will be able to separate themselves or change lanes before the ambulance arrives if they can observe the notification on the screen. A lack of effective traffic management can result in thousands of deaths since an ambulance won't be able to get to the hospital in time. However, the issue is getting worse since annual vehicle growth is currently at 11% while annual road expansion is just around 4% in developing nations like India. So, to address this issue, the paper offers a fresh, simple solution for traffic control in emergency scenarios.

Contents

List of Figures

List of Tables

List of Abbreviations

IOT:	Internet of Things
LCD:	Liquid Crystal Display
API:	Application Programming Interface
GSM:	Global System for Mobile Communications
WiFi:	Wireless Fidelity
GPS:	Global Positioning System
GIS:	Geographic Information System

Chapter 1

Introduction

The pace of global development is really rapid right now. As a result of industrialization and urbanization, there are more people on the road, which has boosted automobile usage. Roadblocks are the result in major cities. Transportation is severely hampered by traffic jams. The ambulance service is the one most impacted by traffic. Patients in need of immediate medical attention or those in critical condition may be sent in an ambulance to the local hospital to the earliest . As soon as the patient arrives at the hospital, the proper assistance will be provided to them in order to maximize their chances of survival in critical situations. Getting to the hospital on time and without any traffic problems. It is crucial to make room for the ambulance. When an ambulance get caught in traffic, it wastes a lot of time and kills people while waiting for the traffic to clear. With the help of emerging technology like IoT, or the Internet of Things, we can solve these issues. If it takes a long time for the ambulance to get to the hospital, many lives will be lost. According to surveys, heart attacks can be successfully treated in 95% of instances if the ambulance arrives at the hospital promptly and without incident. It is crucial to make room for the ambulance. When an ambulance becomes occasionally caught in traffic, it wastes a lot of time and kills people while waiting for the traffic to clear. With the help of emerging technologies like the Internet of Things, we can solve these issues. In IoT, all of the components are linked and controlled through the internet. As a result, IoT has a huge impact on today's world because it offers numerous solutions to solve problems. The rapid rise in vehicle numbers, which has grown serious in the wake of the startling increase in global population, is one of the most significant effects of modern technology development. As a result, traffic congestion is now a major issue in the majority of the world's nations. Furthermore, in nations like India, the rate of road extension is just a third of the rate of increase of vehicles, which makes the issue even worse. Traffic congestion has increased as a result of increased vehicles, which makes it difficult for ambulances to transport patients to the right locations on time in an emergency. According to statistics, more than 20% of patients in need of urgent care pass away in transit to the hospital as a result of delays. In large, developing cities all around the world, traffic difficulties are an unavoidable situation. Absolute congestion has developed as a result of rising car ownership due to population increase. Because life is sacred, it cannot be returned until it is lost. In the face of calamities, the emergency response required reaction time is crucial. This could be a medical, emergency, or defence department. Congestion from flow is their biggest problem. In order to solve this, a pre-emptive equation-based intelligent traffic system that can quickly adapt to changing en- vironmental conditions and offer solutions are required. There are several instances of ambulances getting snarled in

traffic and having to wait for it to clear. This delay can occasionally last for minutes or even hours, causing the patient to pass away for lack of timely treatment. Proposing the solution for above condition, which is divided into three main parts, taking into consideration the current situation of emergency services and the traffic system.

1.1 Components

1.1.1 Mobile App

The system's user-facing component deals with them di- rectly. Ambulance driver is our final client. Complex graphic user interfaces are typically beyond the capabilities of drivers. The goal is to develop a user-friendly interface that is simple for the user to browse. In the beginning of the journey, the river would be able to select the source and destination locations. The system's user-facing component deals with them di- rectly. Ambulance driver is our final client. Complex graphic user interfaces are typically beyond the capabilities of drivers. The goal is to develop a user-friendly interface that is simple for the user to browse. In the beginning of the journey, the river would be able to select the source and destination locations.

1.1.2 Digital Screens

With nodeMCU, the digital Displays will be wirelessly connected to the mobile app. Digital panels will display the driver's chosen lane along with a notice such as "Ambulance Approaching maintain lane XYZ unoccupied."

1.1.3 Driver's Feedback

The driver will have the ability to provide feedback in the form of a rating scale, emojis, or stars when the entire service is completed. Our future goals include using machine learning algorithms to analyse user feedback and understand trends.

1.2 Algorithm

A* Search algorithm is one of the best and popular technique used in path-finding and graph traversals.It helps to approximate the shortest path in real-life situations, like- in maps, games where there can be many hindrances. We can consider a 2D Grid having several obstacles and we start from a source cell (colored red below) to reach towards a goal cell (colored green below) using A* Search Algorithm for Google API Services to find the shortest route for Driver.

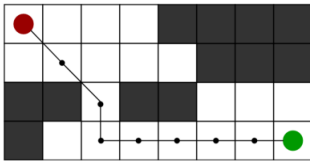


Figure 1.1: A* Search Algorithm

Chapter 2

Literature Review

The purpose of literature review is to gain an understanding of the existing research on ambulance monitoring and directions services relevant to area of study.

- The system in paper[1] displays how to use the android app to find the closest hospital. The hospital and the controller receive the data that the emergency car driver inputs into the android application. If there is any traffic, the controller will clear it, and the quickest route to get to the local hospital will be selected.
- In paper[2] When an accident occurs, the injured person must be taken to an ambulance, which must travel to the scene and use an Infrared sensor to identify the patient on a stretcher. Once identified, the patient's heart rate is then monitored by a heart rate sensor. The nurse must enter the type of injury using the serial monitor so that requests are sent to nearby hospitals regularly.
- Paper[3] says that a traffic signal contains a zigbee receiver that is controlled by a microcontroller when it gets a signal, clearing the way for the vehicle to move. The ambulance is equipped with a zigbee transmitter to transmit signals.
- In paper[4] the ambulance driver uses the web application for choosing the route and navigating the ambulance. Communication between the ambulance and traffic signal is established using the cloud server. Arduino UNO is interfaced with Wi-Fi module and the Wi-Fi module is used as a traffic signal in this system.
- In paper[5] The online application is used by the ambulance driver to choose and navigate routes. Using the cloud server, communication is created between the traffic signal and ambulance. The Wi-Fi module is connected to Arduino UNO and utilised as a traffic signal in this system.
- In paper[6], they created a digital notice board that displays messages sent by users via mobile phone (SMS) and designed a straightforward, user-friendly system that can receive and display messages and information in a specific way with respect to date and time. This will help the user easily keep track of the notice board each day and every time he uses the system. Sender and receiver are the two sections that make up a system. The message can be entered directly by the application sender.Sender is in charge of transmitting important data using the GSM system. The sender must input

the relevant mobile number in order to access the digital notice board. Correct mobile number entry allows the user to type a message and receive space for information transmission. They created an Android application to improve the suggested system's usability. With this programme, the message can be entered immediately by the sender.

- The system that was suggested in the study paper[7] has not yet been put into place. The system contains a wrist-worn sensor that continuously records data and heartbeat. Using algorithms, the data is further examined on the cloud. The system alerts the patient's emergency contacts, a nearby hospital, an ambulance, and others in case of a heart attack. The device also assists the ambulance, much like the Ola app, by directing it along the quickest route.

- In paper[8] The paper's feedback management system will address institutional weaknesses, which will help you do well in NIRF evaluations. By the use of a structured feedback management system, higher education can enhance its teaching strategies, learning process, curriculum, core and soft subjects, skill development, practical applications, and experiential learning in accordance with market expectations.

- In paper[9] The location of the ambulance will be presented in Google Maps in the exhibits output of the system, which will allow hospital management to reroute the ambulance to the location of the closest patient. To inform the ambulance of the patient's position, the hospital can employ GSM technology.

- The infrared sensors will be utilised to gather data from the lane and fetch the acquired data to the micro controller, according to article[10]. Four infrared sensors, one on each side of each road, will be positioned at a predetermined distance from the intersection on each road, dividing the length of the road from the intersection into two zones: a high density zone and a low density zone. Two proximity infrared sensors installed at either side of the road in the opposite direction detect the presence of vehicles in each area. The sensors are positioned by maintaining a specific spacing between them so that they do not overlap. The use of two sensors removes the possibility that "vehicles are present along one side alone," giving us a true picture of how the vehicles coincide while situating them because doing so will lead to inaccurate data readings.

Chapter 3

Objectives

We intend to do this project implementation to meet following objectives:

- To make the ambulances services faster and efficient by using Google API for finding the shortest route to destination.
- To guide the traffic controller to manage traffic during peak hours as well as normal hours and make way for ambulance.
- To help the traffic controller to move traffic in a systematic order by displaying the lane chosen for the ambulance driver.
- To provide an efficient service to driver by opting for our user-friendly application using Flutter.
- To get the feedback from the driver of the ambulance about assistance provided through system using grading scale, emoticons or stars.

Chapter 4

Project Design

The project's key features, structure, criteria for success, and major deliverables are all planned out in this steps. The aim is to develop design in a way so that it can differ from existing system that can be used to achieve the desired project goals.

4.0.1 Existing System

- The evolving problem of traffic is an inevitable situation in wide and rising cities worldwide. Population growth has led to a growth in the number of cars leading to absolute congestion development.
- The reaction time required by the emergency response plays a vital role in the face of disasters. To address this, an intelligent traffic system is needed using a pre-emptive equation that adapts rapidly to environmental conditions.
- People travelling on roads often do not notice ambulances coming and when they do it becomes difficult to give side to the vehicle due to congestion.

4.0.2 Proposed System

- In Fig 4.1 the Ambulance Driver Interface and operations are explained. The drivers first have to first register themselves if they are new user or else just simply login to the application. The driver has to select Pickup and Destination Location. Then the driver will be provided with the shortest route to the destination.

- In Fig 4.1 the Traffic Controller side of the mobile interface is explained.The Traffic Controller first have to first register themselves if they are new user or else just simply login to the application.The Controller can now Input the message to digital screens which are installed near the highways.

- Mobile App - Created an user friendly application to which both user and traffic controller both will have access to.

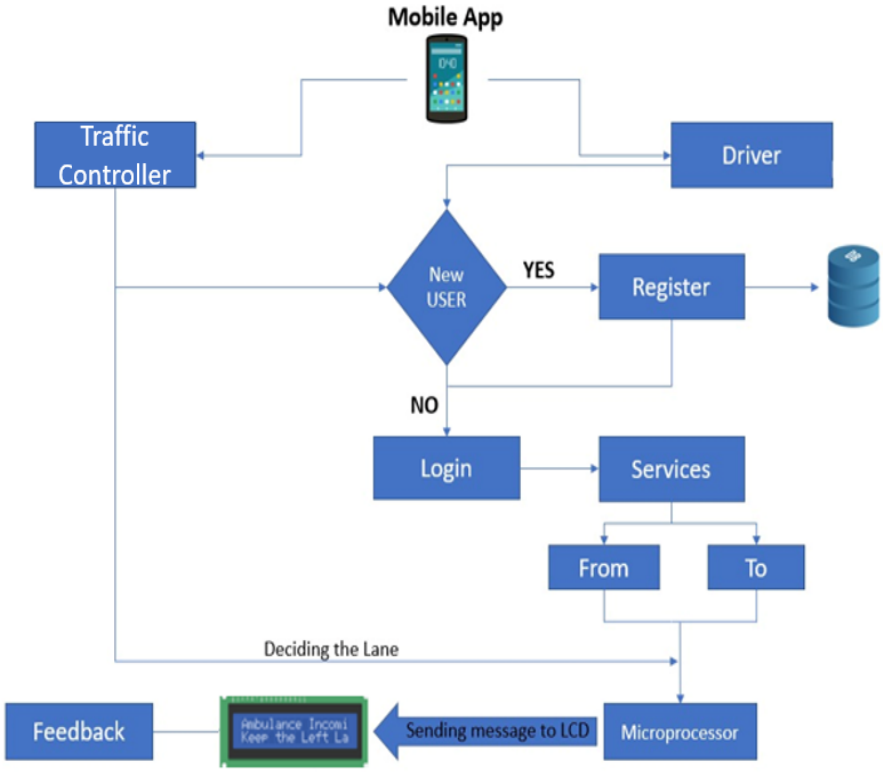


Figure 4.1: Flow Diagram

- Traffic Controller - A Traffic Controller, also known as a Traffic Signal Operator or Traffic Management Coordinator, is responsible for managing the flow of traffic at intersections and roadways. Their main goal is to ensure the safety of drivers, pedestrians, and cyclists by controlling the timing and sequencing of traffic lights, signs, and signals. Here the controller has to decide a dedicated lane of incoming ambulance on that route.

- Driver - The driver of the ambulance will have access to Mobile APP where he has to enter basic information and then will be directed to other page where they are supposed to enter the locations.

- Registrations and Login - New users to the app have to register themselves and already existing users can login themselves in order to avail the services.

- Services - The user will be given the fields to enter source and destination location. They will be provided with the shortest route to the hospital using Google maps.

- Microprocessors - NodeMCU is an open-source firmware and development board that is based on the ESP8266 Wi-Fi chip. It is designed to provide an easy-to-use platform for developing and prototyping Internet of Things (IoT) applications. The NodeMCU board has a built-in Wi-Fi module, making it easy to connect to the internet and interact with other devices and services. It also has a microcontroller unit (MCU) that can be programmed using the Lua scripting language or the Arduino IDE.

- Digital Screen - Digital screens are commonly used in IoT (Internet of Things) applications to display real-time data, alerts, and other information. These screens can be used to provide visual feedback and alerts to users, or to display data from sensors and other IoT devices.
- Feedback- The user will be given option to give feedback of their experience of using the app using star ratings. These feedbacks will be analyzed for getting better results in future.

4.0.3 System Diagram

A system diagram is a graphical depiction of the various components that make up a system and the relationships between them. It helps to visualize the overall structure of the system, its processes, inputs, outputs, and the flow of data or materials. System diagrams can be used to design, analyze, or communicate complex systems, including engineering systems, business processes, software applications, and more. They provide a high-level view of the system, helping readers to understand its architecture and identify potential problems or opportunities for improvement.

- In ambulance section, the app is used by ambulance driver to navigate through the nearest hospital in case of emergency.The information about the route of the ambulance will also be shared with traffic controller. The traffic controller has the ability to input message on the display screens and also have the facility to select a particular lane to that will be access by driver of the incoming ambulance.

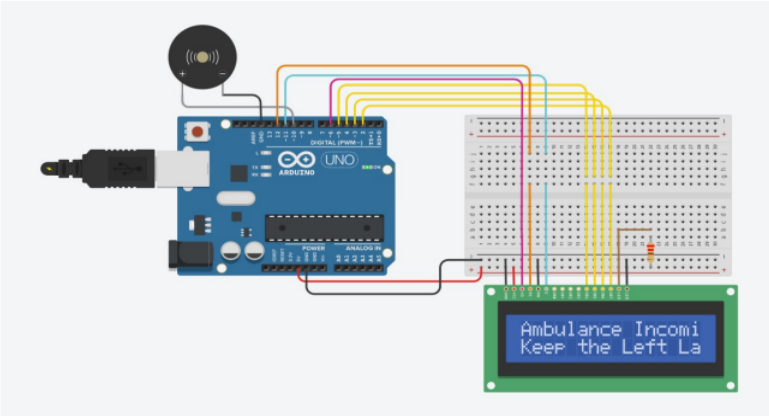


Figure 4.2: Tinkercad Simulation

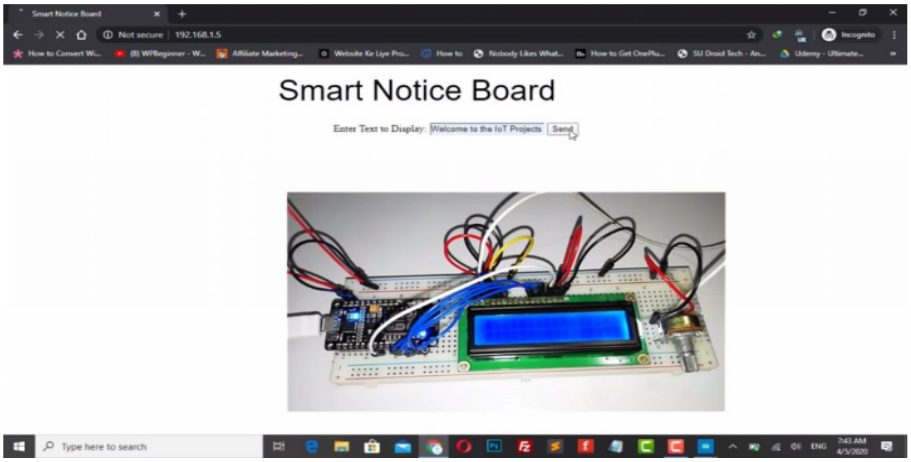


Figure 4.3: Message Input Interface

- Use Case Diagram
  A use case diagram is a visual representation of actors, use cases, and their interactions in a system, used to describe system functionalities and requirements. It is a type of UML diagram that represents the interactions between actors (users or external systems) and a system or software application. Use case diagrams are often used in software development to identify the different ways that users will interact with a system and to define the functional requirements of the system.

  The Use Case Diagram below shows different actors as Driver and Traffic Controller. The relation between actors and what they can do with the system. The Traffic Controller will manage the traffic using smart screens with the help of IOT. Ambulance driver will be provided with the shortest path according to the source and destination entered earlier. The location co-ordinates will be provided as inputs for the traffic controller to manage the traffic. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design related requirements.

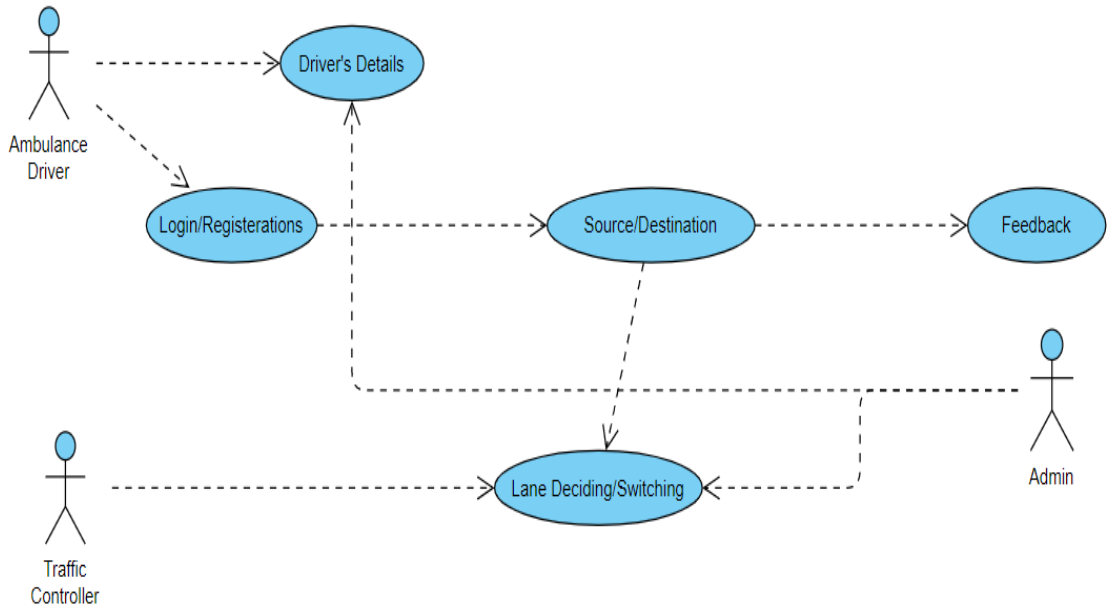


Figure 4.4: Usecase Diagram

- Sequence Diagram
  The sequence diagram shows the entire sequence of flow of the data in the system.In the initial step the the user has to login or register themselves if they are new to the app. After logging in the user has to enter source and destination location. The app then finds the shortest route to the destination.When the ambulance has reached destination the user will be asked for the feedback which they can give in form of star ratings.

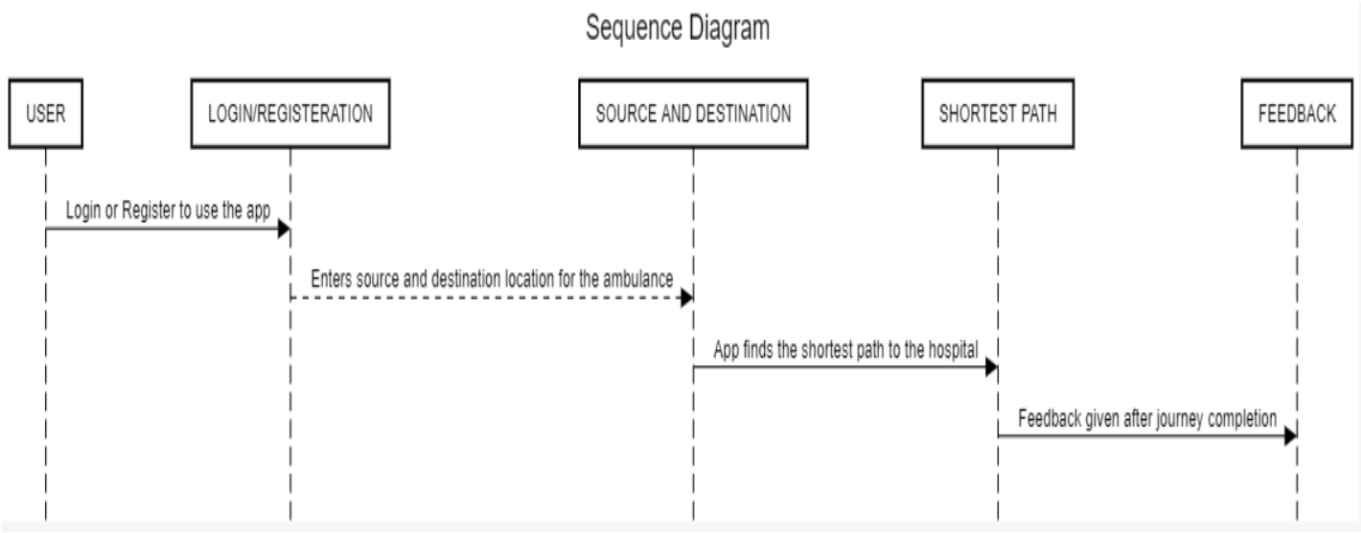


Figure 4.5: Sequence Diagram

Chapter 5

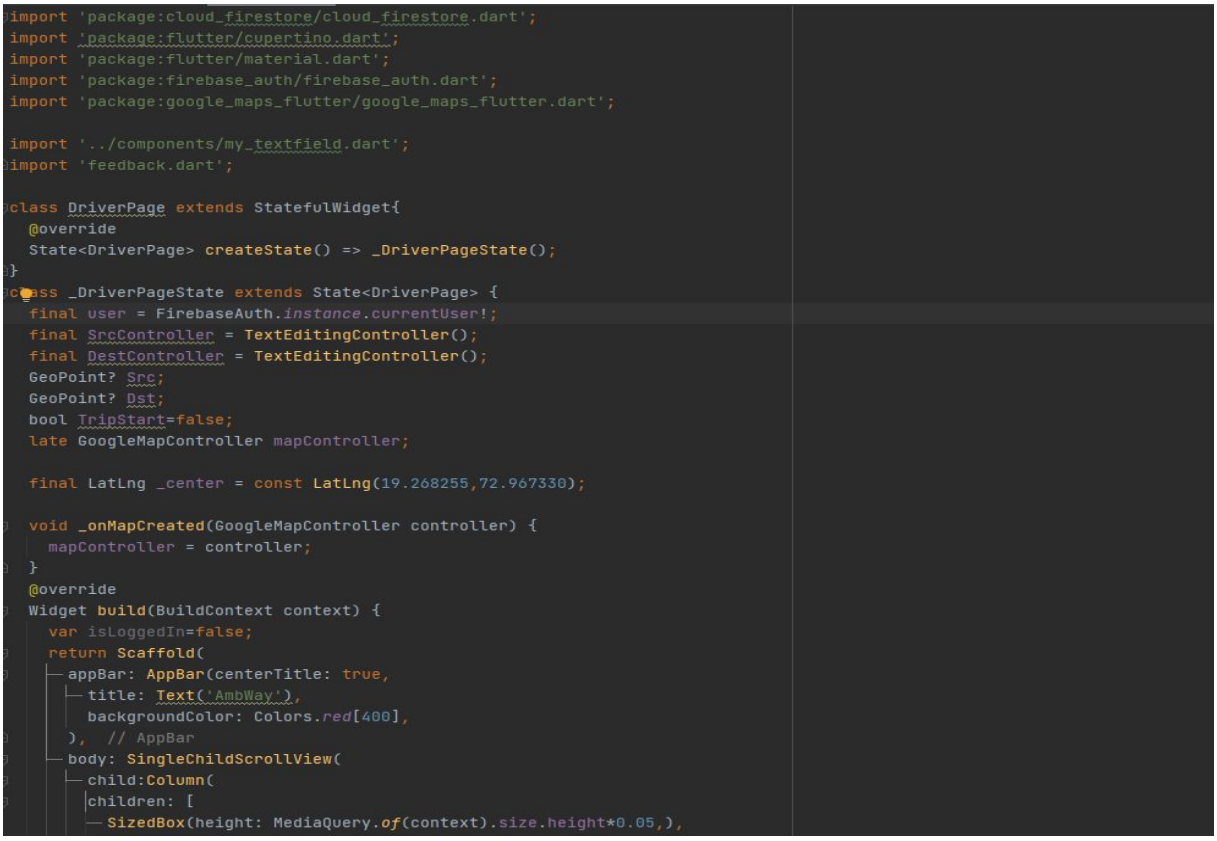
Project Implementation

Project implementation consists of visions and plans with which we are supposed to build the end product. This includes the logical conclusion, after evaluating, deciding, visioning, planning and finding the other resources for the project. Technical implementation is one of the major aspects of executing a project.

```dart
import 'package:major/services/auth_service.dart';
import 'package:humanitarian_icons/humanitarian_icons.dart';


class LoginPage extends StatefulWidget {
  // ignore: prefer_const_constructors_in_immutables
  const LoginPage({super.key});

  @override
  State<LoginPage> createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  // text editing controllers
  final emailController = TextEditingController();

  final passwordController = TextEditingController();

  // sign user in method
  void signUserIn() async {
    showDialog(
      context: context,
      builder: (context) {
        return const Center(
          child: CircularProgressIndicator(),
        );  // Center
      },
    );

    try {
      await FirebaseAuth.instance.signInWithEmailAndPassword(
        email: emailController.text,
        password: passwordController.text,
      );
      // ignore: use_build_context_synchronously
      Navigator.pop(context);
    } on FirebaseAuthException catch (e) {
      Navigator.pop(context);
      showErrorMessage(e.code);
    }
  }
```

Figure 5.1: Login Page

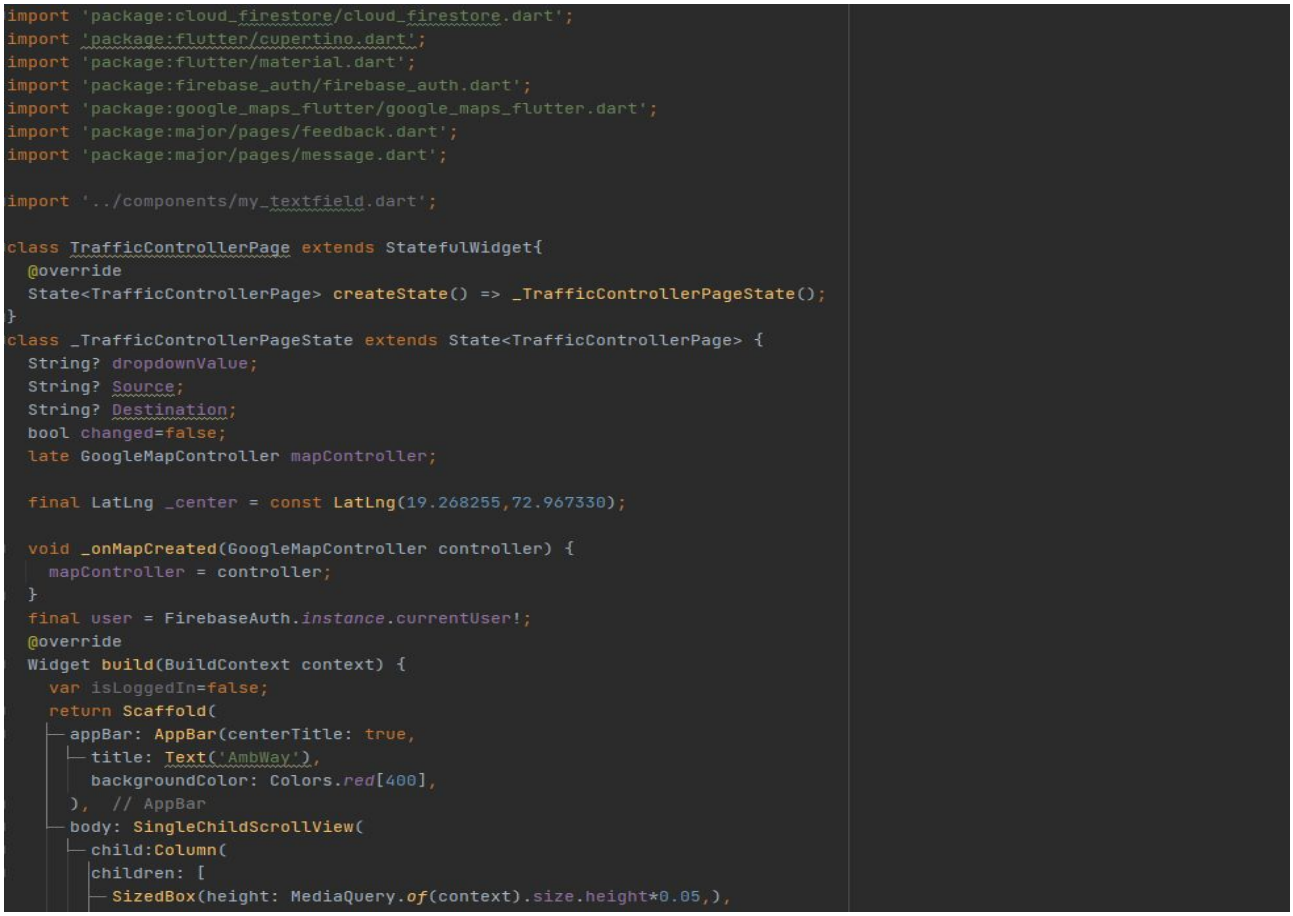
For Logging in user has to enter username and password which will be authenticated by firebase and same will be reflected to the user.



```dart
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:google_maps_flutter/google_maps_flutter.dart';

import '../components/my_textfield.dart';
import 'feedback.dart';

class DriverPage extends StatefulWidget{
  @override
  State<DriverPage> createState() => _DriverPageState();
}
class _DriverPageState extends State<DriverPage> {
  final user = FirebaseAuth.instance.currentUser!;
  final SrcController = TextEditingController();
  final DestController = TextEditingController();
  GeoPoint? Src;
  GeoPoint? Dst;
  bool TripStart=false;
  late GoogleMapController mapController;

  final LatLng _center = const LatLng(19.268255,72.967330);

  void _onMapCreated(GoogleMapController controller) {
    mapController = controller;
  }
  @override
  Widget build(BuildContext context) {
    var isLoggedIn=false;
    return Scaffold(
      appBar: AppBar(centerTitle: true,
        title: Text('AmbWay'),
        backgroundColor: Colors.red[400],
      ),  // AppBar
      body: SingleChildScrollView(
        child:Column(
          children: [
            SizedBox(height: MediaQuery.of(context).size.height*0.05,),
```

Figure 5.2: Google API for shortest path

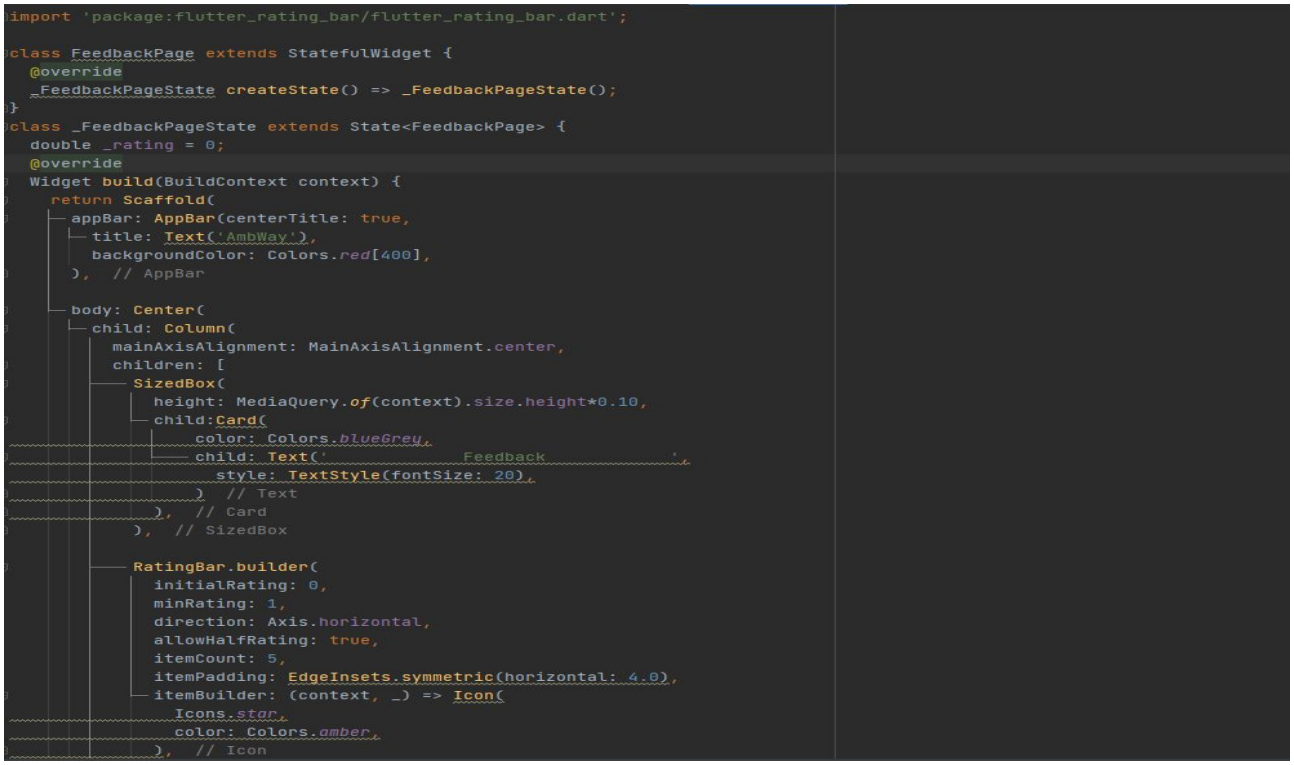
The source and destination fields will be provided to the users, after which they will be displayed the shortest path to the destination.



```dart
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:google_maps_flutter/google_maps_flutter.dart';
import 'package:major/pages/feedback.dart';
import 'package:major/pages/message.dart';

import '../components/my_textfield.dart';

class TrafficControllerPage extends StatefulWidget{
  @override
  State<TrafficControllerPage> createState() => _TrafficControllerPageState();
}
class _TrafficControllerPageState extends State<TrafficControllerPage> {
  String? dropdownValue;
  String? Source;
  String? Destination;
  bool changed=false;
  late GoogleMapController mapController;

  final LatLng _center = const LatLng(19.268255,72.967330);

  void _onMapCreated(GoogleMapController controller) {
    mapController = controller;
  }
  final user = FirebaseAuth.instance.currentUser!;
  @override
  Widget build(BuildContext context) {
    var isLoggedIn=false;
    return Scaffold(
      appBar: AppBar(centerTitle: true,
        title: Text('AmbWay'),
        backgroundColor: Colors.red[400],
      ),  // AppBar
      body: SingleChildScrollView(
        child:Column(
          children: [
            SizedBox(height: MediaQuery.of(context).size.height*0.05,),
```

Figure 5.3: Controller Side

The source and destination will be sent to controller as inputs and he/she will be able to switch between drivers running status.



```dart
import 'package:flutter_rating_bar/flutter_rating_bar.dart';

class FeedbackPage extends StatefulWidget {
  @override
  _FeedbackPageState createState() => _FeedbackPageState();
}
class _FeedbackPageState extends State<FeedbackPage> {
  double _rating = 0;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(centerTitle: true,
        title: Text('AmbWay'),
        backgroundColor: Colors.red[400],
      ),  // AppBar

      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            SizedBox(
              height: MediaQuery.of(context).size.height*0.10,
              child:Card(
                  color: Colors.blueGrey,
                  child: Text('              Feedback              ',
                    style: TextStyle(fontSize: 20),
                  )  // Text
              ),  // Card
            ),  // SizedBox

            RatingBar.builder(
              initialRating: 0,
              minRating: 1,
              direction: Axis.horizontal,
              allowHalfRating: true,
              itemCount: 5,
              itemPadding: EdgeInsets.symmetric(horizontal: 4.0),
              itemBuilder: (context, _) => Icon(
                Icons.star,
                color: Colors.amber,
              ),  // Icon
```

Figure 5.4: Feedback Page

Imported the module of rating bar present in Flutter which will help user to give feedback for the completed journey in form of star ratings.

```
#include <ESP8266WiFi.h>

#include <ESPAsyncTCP.h>

#include <ESPAsyncWebServer.h>

#include <LiquidCrystal_I2C.h>


LiquidCrystal_I2C lcd(0x27,16,2);

AsyncWebServer server(80);


 //replace ssid and password with your wifi network credentials

const char* ssid = "redmi"; // your SSID

const char* password = "12345678";  //your WIFI Password


const char* PARAM_INPUT_1 = "input1";


const char index_html[] PROGMEM = R"=====(

<!DOCTYPE HTML><html><head>

  <title>Smart Notice Board</title>

  <meta name="viewport" content="width=device-width, initial-scale=5">

<p> <font size="9" face="sans-serif"> <marquee> IoT Wireless Smart Notice Board </marquee> </font> </p>
```

Figure 5.5: IOT code for the prototype

Imported various modules such as ESP8266WiFi, ESPAsyncWebServer and LiquidCrystal for the purpose of compelling and deploying the code through Ardunio IDE in the IOT prototype.

```
void setup() {
  Serial.begin(115200);
  lcd.init();                      // initialize the lcd
  // Print a message to the LCD.
  lcd.backlight();
   lcd.begin(16, 2);
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("Ambulance Incoming");
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  if (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("WiFi Failed!");
    return;
  }
  Serial.println();
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
```

Figure 5.6: IOT code for the prototype

Setuping the function which will initalize the LiquidCrystal also defining the Wifi credentials to connect the prototype with the front-end application.

```
  server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){

    request->send_P(200, "text/html", index_html);

  });


  server.on("/get", HTTP_GET, [] (AsyncWebServerRequest *request) {
    String message;
    String inputParam;
    if (request->hasParam(PARAM_INPUT_1)) {
      message = request->getParam(PARAM_INPUT_1)->value();
      inputParam = PARAM_INPUT_1;

   //    lcd.clear();
       lcd.setCursor(0,1);
      lcd.print(message);
    }
    else {
      message = "No message sent";
      inputParam = "none";
    }
    Serial.println(message);
  request->send(200, "text/html", index_html);
  });
  server.onNotFound(notFound);
  server.begin();
}

void loop() {
    for (int positionCounter = 0; positionCounter < 29; positionCounter++) {
    lcd.scrollDisplayLeft();
    delay(500);
  }
}
```

Figure 5.7: IOT code for the prototype

Setuping the LiquidCrystal in the specific format message will be displayed. Also connecting the prototype with application through TCP protocol with use of AsyncWebServer-Request method.

Chapter 6

Testing

6.1 Functional Testing

6.1.1 Unit Testing

Unit testing is the first level of testing, which is typically performed by the developers themselves. At the code level, it is the process of ensuring that individual components of software are functional and work as intended. Unit testing can be done manually, however automating the process will reduce delivery times and boost test coverage. Because flaws will be detected earlier in the testing process and will take less time to fix than if they were discovered later, debugging will be easier as a result of unit testing. It helped us in finding the shortest route on the basis of algorithm that Google API has implemented. Google API will help us to generate the possible shortest path after entering the source and destination locations for the driver side application. The main objective of unit testing is to isolate written code to test and determine if it works as intended. Also make the changes to improve the efficiency.

6.1.2 Prototype Testing

Prototype testing is the process of testing your hardware prototype with proper connections and integration to validate design and feasibility decisions before development stage is initiated. The goal is to identify problems and areas of improvement early so you can make the necessary changes prior to development and build a product that meets users' needs and expectations and also works properly with underlying hardware and software.

6.1.3 Test Cases

Test Case No.	Test Case Condition	Test Steps/Procedures	Expected Results	Actual Results	Pass/Fail
1	Route Test	User enters the source and destination locations	The system should present the shortest path from source and destination location	The system successfully provided the shortest route	Pass
2	Duration Test	Record the time taken for completing one journey to the desired destination	The ambulance should take comparatively less time to reach the destination	The ambulance took somewhat same time to reach the destination	Fail(ambulance took same time as it took with earlier systems)
3	Message Display Test	The controller would enter the display message in the field available on the application	The exact same message should be displayed on all digital screens which was entered by controller	The same message was displayed on all screens	Pass

Table 6.1: Software Testing for Traffic Divergence System

Chapter 7

Result

This shall form the penultimate chapter of the report and shall include a thorough evaluation of the investigation carried out and bring out the contributions from the study. The discussion shall logically lead to inferences and conclusions as well as scope for possible further future work.

7.0.1 Flutter Module

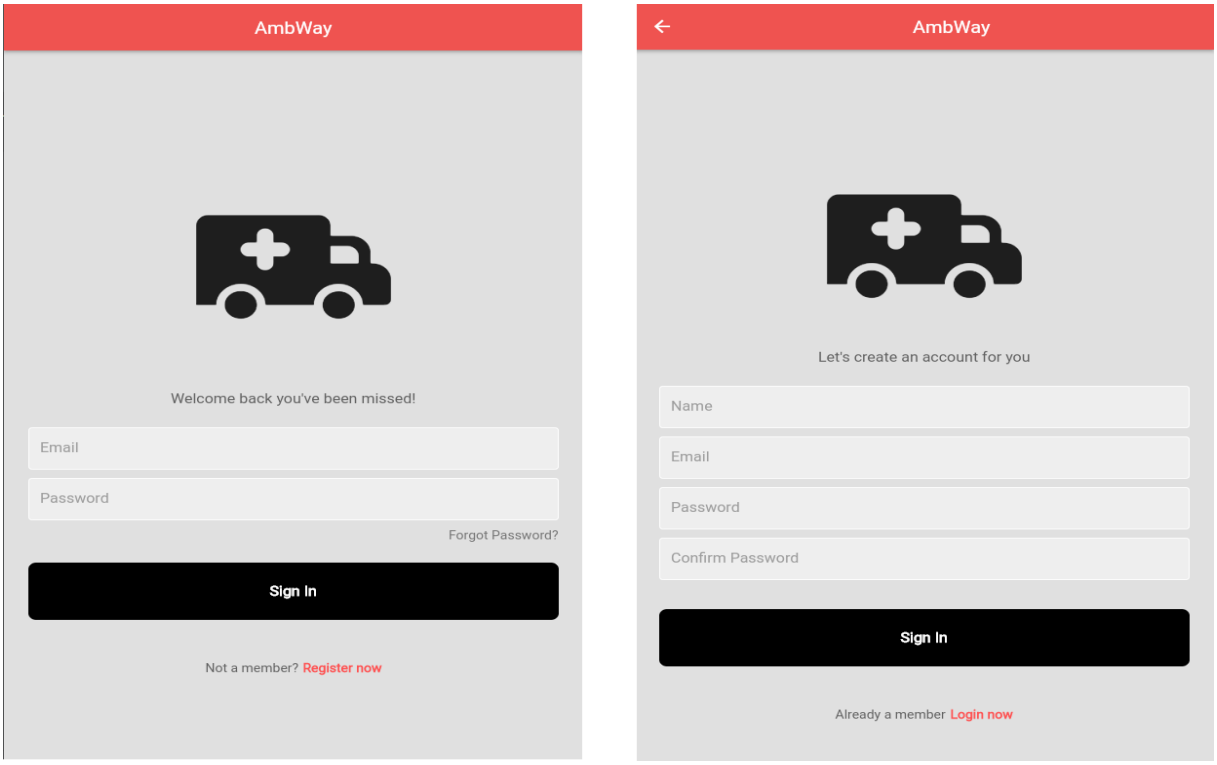


Figure 7.1: Login and Registration

Login page has two fields Email and Password for the user who have a account in the app. For the registration the user has to enter name,Email,Password creation and confirming the password.

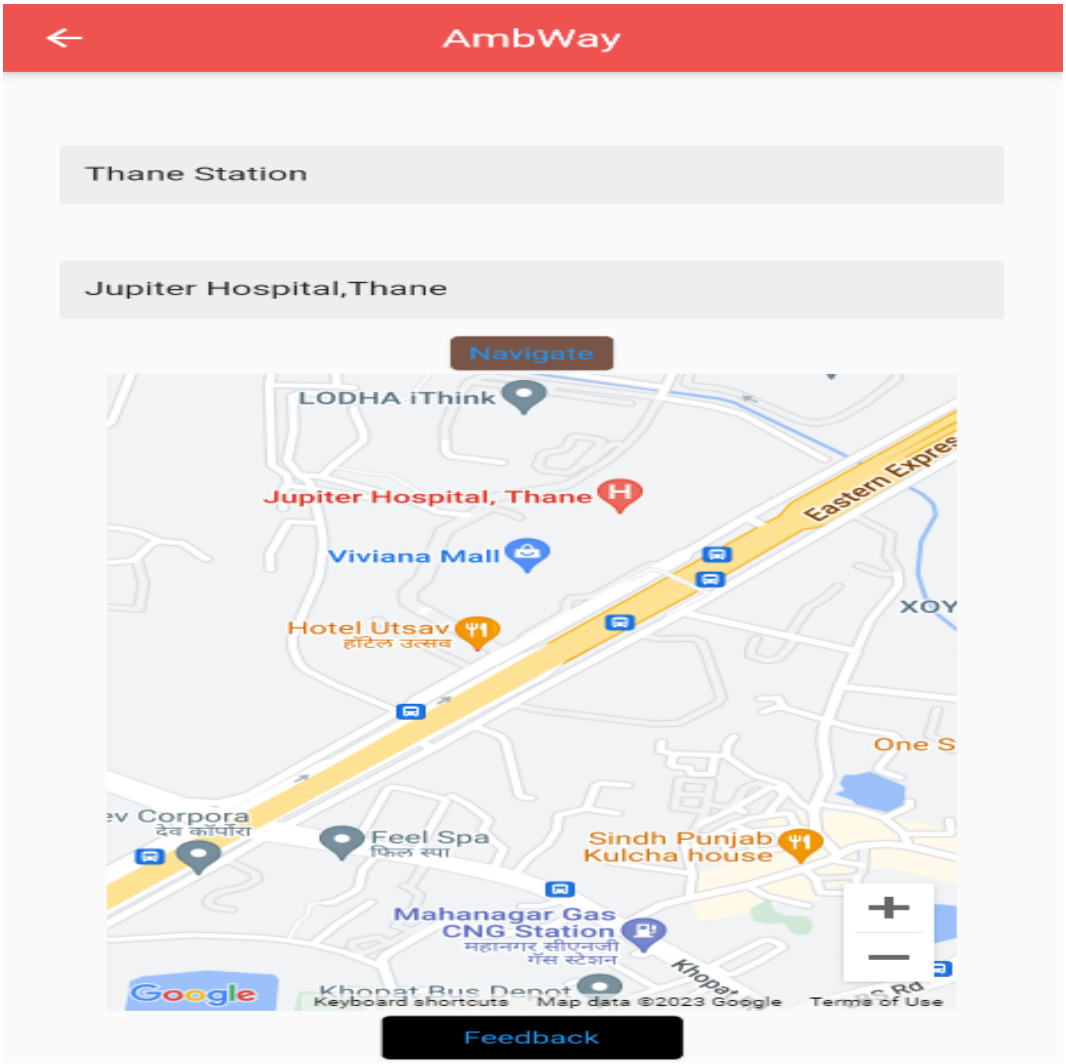


Figure 7.2: AmbWay driver's page

The driver will get to see two fields where he has to enter source and destination location and he will be displayed with shortest path to the destination

Figure 7.3: AmbWay traffic controller's page

The traffic controller interface will have the option to input the message that has to be displayed on all connected digital screens. Traffic controller can also change between different ambulance drivers available on the route.

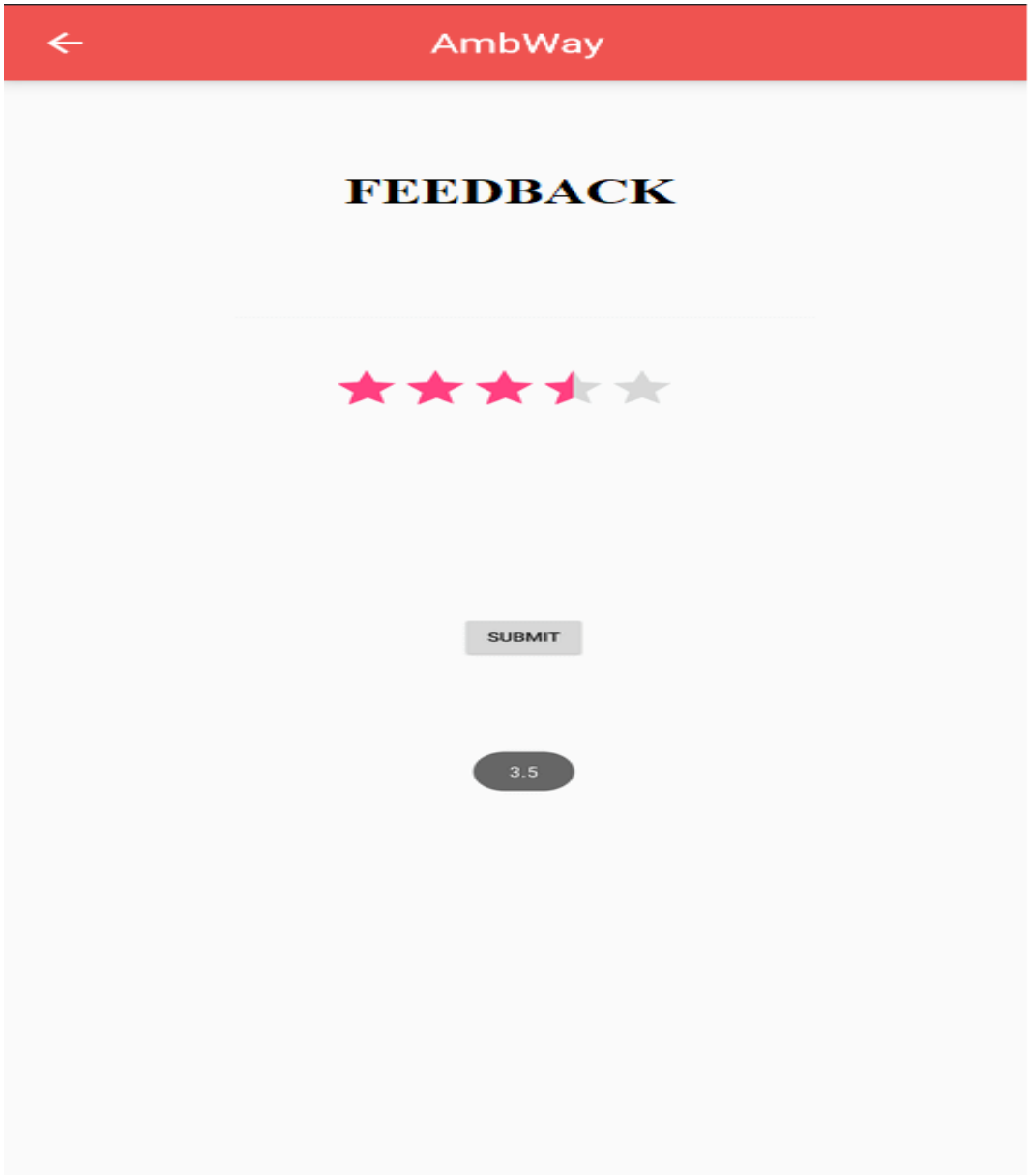


Figure 7.4: AmbWay Feedback page

The user will have option to rate the overall service offered by AMBWAY with the help of star ratings. These ratings will be utilized for future analysis.

Figure 7.5: Smart Screen

The smart screen has a field for traffic controller where they will be input the message that has to be displayed on all digital screens on the route to destination.

7.0.2 IOT Module

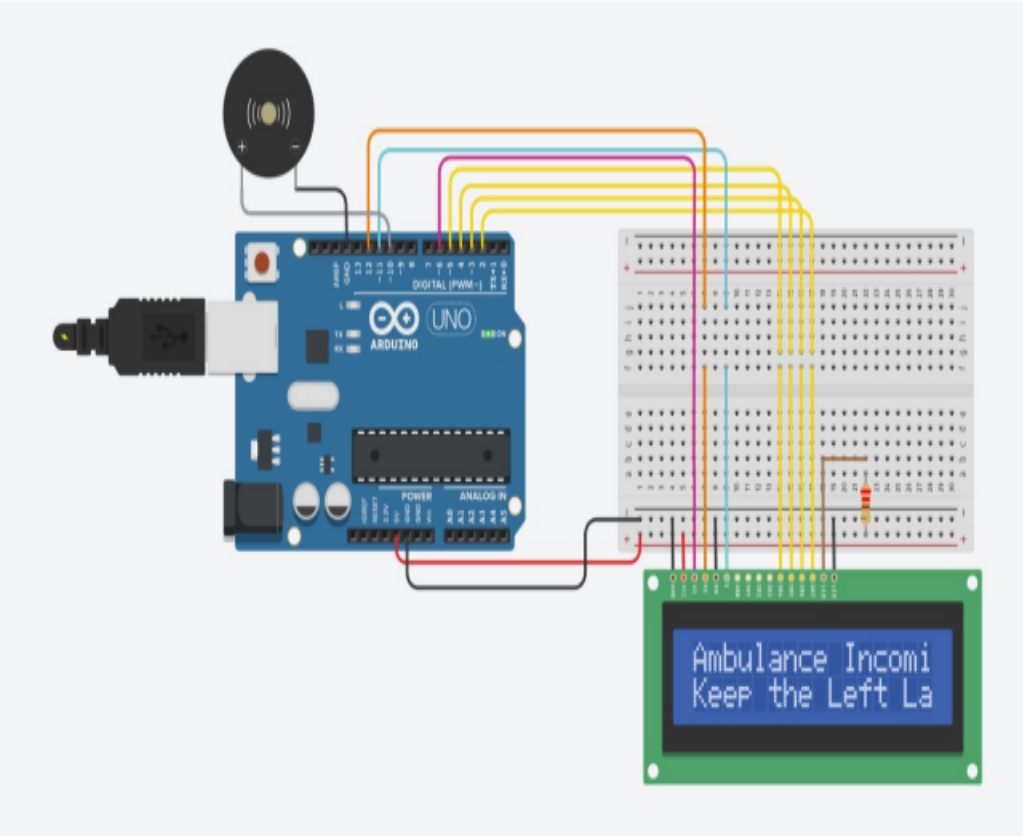


Figure 7.6: Tinkercad Simulation

We implemented the simulation on Tinkercad to check the proper working of circuits before physically making the protoype.

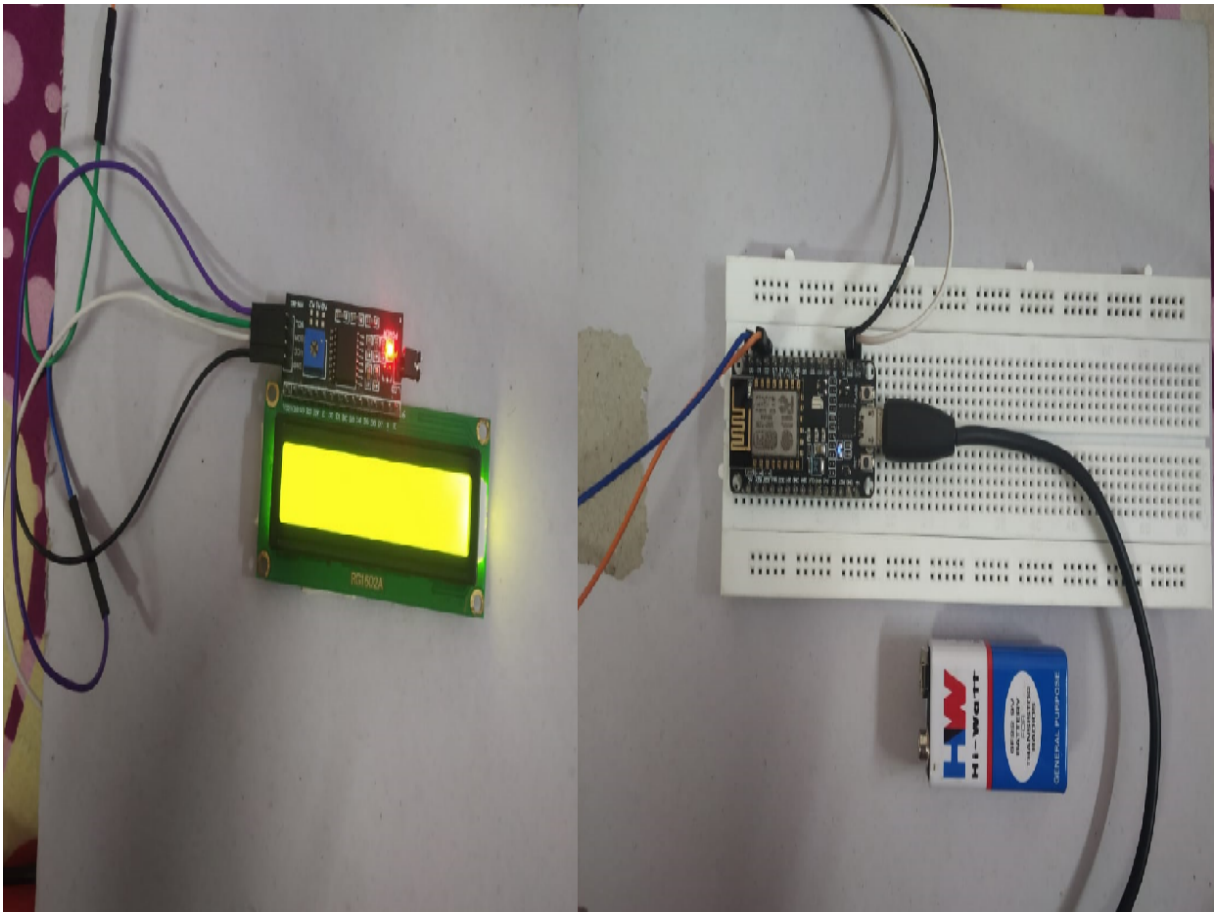


Figure 7.7: IOT Prototype

Connecting the nodeMCU with the LCD and establishing the required connections. Further integrating Physical prototype with the Flutter Application.

Chapter 8

Conclusions and Future Scope

The work presents review of the existing research done in field and tries to develop a system suitable for developing countries. The project has two objectives, which are, first, finding the shortest route to hospital for ambulance and second for displaying the message of ambulance arrival on all digital screens present on the route of the ambulance. The micro-controller can be programmed easily which gives scope for deployment better algorithms in future. This system for emergency services will prove life changing for all citizens residing as it eases the work of traffic controllers and also saves precious lives.

Right now the system is just focused towards helping the Ambulance reach its location faster and quicker. Hence it can be used in different emergency situations such as for fire brigade, police, bomb squad etc. The proposed system holds a vast area to grow in future. The smart board can be used in future for the purpose of certain different public awareness programs too. One of our future scope holds the usage of sensors, which would be enabled to collect real-time data of traffic around particular areas. This would help in making wise decision before starting the journey to destination.

Bibliography

[1] Retraction: Smart Traffic Assistance System for Ambulance Vehicles using Internet of Things (J. Phys.: Conf. Ser. 1916 012127) Published 23 February 2022

[2] IoT Based Intelligent Ambulance Monitoring and Traffic Control System Chapter February 2021 Santhosh Krishna B V New Horizon College of Engineering

[3] Implementation of intelligent Traffic clearance system for Ambulance using Voice Assistance UG Scholars, Department of ECE, AURORA'S TECHNOLOGICAL AND RESEARCH INSTITUTE,Uppal, Hyderabad, India Issue III, 2021 August

[4] IoT Based Traffic Management System for Ambulances Preprint · April 2020 Central University of Kashmir

[5] Development of an IoT based real-time traffic monitoring system for city governance Communication and Information Research Center, Sultan Qaboos University, Muscat, Oman Accepted on 23 September 2020

[6] IOT based Smart Notice Board Gaurav Bhardwaj UG Student, Department of ECE MIT School of Engineering Pune, Maharashtra, India Vol. 9 Issue 06, June-2020

[7] IoT based heart monitoring and alerting system with cloud computing and managing the traffic for an ambulance in India by Khushboo Bhagchandani, D. Peter Augustine Department of Computer Science, CHRIST (Deemed to be University), India Accepted Jul 3, 2019

[8] Feedback Management System with reference to Institutions of Higher Education: Opportunities Challenges - An Exploratory Study, Vol 4, No 1, March 2019 by Pradeep M D Kalicharan M L

[9] Ambulance Assistance for Emergency Services Using GPS Navigation Issue: 09 Sep-2016 by Shantanu Sarkar School of Computer Science, VIT University, Vellore

[10] Intelligent Traffic Monitoring System Conference Paper · July 2015 by Paromita Roy West Bengal University of Technology

[11] Integrating GIS, GPS and GSM technologies for the effective management of ambulances.Accepted 27 July 2000 by G. Derekenaris J. Garofalakis , C. Makris J. Prentzas S. Sioutas A. Tsakalidis Department of Computer Engineering and Informatics, Graphics, Multimedia & GIS Lab, School of Engineering, University of Patras, 26500 Rion, Patras, Greece

[12] A Comparative Study of IoT Based Smart Traffic Management System.Date of Conference: 04-05 December 2021.Published in: 2021 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE) by Sunny Hossain; Farzana Shabnam

[13] A Clean Approach to Flutter Development through the Flutter Clean Architecture Package Published in: 2019 International Conference on Computational Science and Computational Intelligence (CSCI) by Shady Boukhary; Eduardo Colmenares

[14] Traffic Management System Using IoT Technology - A Comparative Review.Published in: 2018 IEEE International Conference on Electro/Information Technology (EIT) Date of Conference: 03-05 May 2018 by Omid Avatefipour and Froogh Sadry

[15] An IoT based Intelligent Traffic Congestion Control System for Road Crossings.Published in: 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT).Date of Conference: 15-17 February 2018 by Pampa Sadhukhan; Firoj Gazi

Appendices

Appendix-A: Flutter Modules Download and Installation

1. flutter pub add shared_preferences: 2.0.15

2.flutter pub add url_launcher: 6.1.5

3. flutter pub add humanitarian icons:

4. flutter pub add icons:

5. flutter pub add http: 0.13.5

6. flutter pub add web3dart: 2.5.1

7. flutter pub add websocket: 0.0.5

8. flutter pub add web_socket_channel: 2.3.0

9. flutter pub add firebase_core: 2.9.0

10. flutter pub add firebase_auth: 4.4.1

11. flutter pub add firebase_firestore: 4.5.1

Appendix-B: IOT Modules Download and Installation

1.Download and Install Ardunio IDE

2.Import below listed modules and packages

ESP8266WiFi.h
ESPAsyncTCP.h
ESPAsyncWebServer.h
LiquidCrystalI2C.h

Publication

Paper entitled **"IOT Enabled Traffic Diversion System for Medical Emergency Services."** is presented at **" 2023 IEEE International Conference on Contemporary Challenges in Science and its Engineering Applications (IC3SEA 2023)"** by **"Pushkar Telavane, Vatsal Singh, Shubhangi Tripathi, Mandar Ganjapurkar, Apurva Chaudhari".**