Mini-Project Report On

Smart Bike Helmet

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

Bachelor of Technology

in

Computer Science & Engineering

By

Crispin Mathew (U2003065)
David Johns Denny (U2003066)
Devananth H (U2003068)

Under the guidance of
Dr. Dhanya P.M

Department of Computer Science & Engineering
Rajagiri School of Engineering and Technology (Autonomous)
Rajagiri Valley, Kakkanad, Kochi, 682039

July 2023

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING AND TECHNOLOGY
(AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039

CERTIFICATE

This is to certify that the mini-project report entitled ***"Smart Bike Helmet"*** *is a bonafide work done by* ***Mr. Crispin Mathew (U2003065), Mr. David Johns Denny (U2003066), Mr. Devananth H (U2003068)****, submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2022-2023.*

Dr. Preetha K. G.	**Dr. Sminu Izudheen**	**Dr. Dhanya P M**
Head of Department	Mini-Project Coordinator	Mini-Project Guide
Dept. of CSE	Professor	Asst. Professor
RSET	Dept. of CSE	Dept. of CSE
	RSET	RSET

ACKNOWLEDGEMENTS

We wish to express our sincere gratitude towards **Dr. P. S. Sreejith**, Principal of RSET, and **Dr. Preetha K. G.**, Head of Department of Computer Science and Engineering for providing us with the opportunity to undertake our mini-project, "Smart Bike Helmet".

We are highly indebted to our mini-project coordinators, **Dr. Sminu Izudheen**, Professor, Department of Computer Science and Engineering, **Dr. Renu Mary Daniel**, Assistant Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our mini-project guide **Dr. Dhanya P.M**, for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, we would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Crispin Mathew

David Johns Denny

Devananth H

ABSTRACT

The project Smart Bike Helmet is aimed at preventing deaths and serious casualties occurring as a result of road accidents where help reaches the affected too late. The helmet is designed in such a way that it detects the crash, gets the coordinates of the crash and sends the location as an SMS to emergency contacts listed by the rider. Along with the SMS, a system generated voice call is sent to the contact to make sure that the message does not go unnoticed. Also, a hospital near to the location of the accident is identified and is sent to the emergency contact. The data collected from the sensor, along with the location is stored into a database which can be used for further analysis. This is an IOT based project making use of components like GSM module, GPS module, vibration sensor and an arduino uno board.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

1.1.1 Motorcycle Accidents

Motorcycle accidents present a serious concern as riders are more vulnerable to injury due to the lack of protective enclosures. Such accidents frequently lead to severe injuries and fatalities. Speeding, reckless driving, and not wearing helmets are common contributing factors. Moreover, adverse road conditions, inclement weather, and negligence from other motorists also play significant roles in motorcycle accidents. To mitigate these risks, promoting the use of helmets, educating riders about safe practices, and enhancing driver awareness of motorcyclists are crucial steps. Additionally, improving road infrastructure and implementing traffic regulations that accommodate motorcycle safety can further help in reducing the frequency and severity of motorcycle accidents and ensuring safer roads for everyone.

1.1.2 Importance Of Action In The Golden Hour

The "Golden Hour" is a critical period in the aftermath of a road accident, typically lasting for about 60 minutes immediately after the incident. During this crucial time frame, prompt and effective actions taken by emergency responders and medical personnel can significantly impact the outcome for the injured victims. The importance of actions during the Golden Hour can be summarized as follows:

Maximizing Survival Chances: The Golden Hour is considered the most critical time for saving lives. Swift medical attention and interventions within this hour can increase the chances of survival for severely injured individuals. Immediate assessment and treatment of life-threatening injuries can prevent further complications and improve overall

outcomes.

Preventing Complications: Delayed medical care can lead to complications, worsening injuries, and increased risk of mortality. Early intervention during the Golden Hour can help prevent these complications and reduce the severity of injuries.

Reducing Long-Term Disabilities: Rapid medical response and appropriate treatment during the Golden Hour can minimize the risk of long-term disabilities resulting from road accidents. Timely medical interventions can prevent permanent damage and improve the prospects of a full recovery.

Minimizing Response Time: Emergency medical services and first responders are trained to act quickly during the Golden Hour. Reducing response time is crucial, as it allows medical teams to reach the accident site promptly, assess the situation, and provide immediate care.

Effective Triage: Triage is the process of prioritizing patients based on the severity of their injuries. During the Golden Hour, efficient triage can ensure that the most critically injured victims receive immediate attention and transportation to medical facilities with the appropriate resources to treat their injuries effectively.

Early Stabilization and Transport: The Golden Hour allows for early stabilization of injured individuals at the accident scene. Medical personnel can administer life-saving interventions, such as controlling bleeding, maintaining airway patency, and providing pain relief. Transporting the injured to specialized medical facilities swiftly can further enhance their chances of survival.

Better Utilization of Resources: Acting promptly during the Golden Hour optimizes the use of medical resources, including medical staff, equipment, and facilities. By efficiently managing resources, medical teams can handle multiple patients effectively and provide adequate care to all those in need.

In summary, the importance of action during the Golden Hour in road accidents cannot be overstated. Immediate and effective medical attention can save lives, prevent complications, and improve the overall prognosis for injured individuals. It underscores the critical role of emergency responders, medical personnel, and the entire healthcare system in ensuring the best possible outcomes for accident victims.

1.2 Existing System

Most motorcycle accidents can be prevented with the use of helmets and following the speed limits.

Helmet use is a critical safety measure to prevent severe injuries and fatalities in motorcycle accidents. Helmets provide essential protection for the rider's head and brain, reducing the risk of head trauma and traumatic brain injuries in the event of a crash. By absorbing and dispersing impact forces, helmets help minimize the severity of head impacts during collisions.

Studies consistently demonstrate that wearing a helmet significantly lowers the risk of head injuries and fatalities in motorcycle accidents. In many countries, laws mandate helmet use for all motorcyclists and their passengers. These laws have proven effective in reducing the number of serious injuries and deaths in motorcycle accidents.

Promoting helmet use through public awareness campaigns, education, and enforcement of helmet laws can lead to safer roads for motorcyclists. Encouraging responsible riding practices and consistently wearing helmets can play a vital role in preventing motorcycle accidents from turning into life-altering tragedies.

1.3 Problem Statement

Develop a Smart Helmet System for Motorcycle Riders that Detects and Responds to Crashes

Motorcycle accidents pose a significant risk to riders' safety due to the lack of protective barriers. To address this issue, the project aims to develop a Smart Helmet System equipped with sensors and communication modules. The system's primary objective is to detect when a crash has occurred and promptly alert the rider's designated emergency contact.

1.4 Objectives

The key features of the Smart Helmet System include:

Crash Detection: The helmet must be equipped with appropriate sensors (e.g., accelerometer) to accurately detect the impact of a crash or collision.

Emergency Alert: Once a crash is detected, the system should automatically send an emergency alert to the designated emergency contact. The alert may include details such as the rider's identity and the fact that a crash has occurred.

Location Tracking: The system should use GPS technology to track and obtain the precise location of the accident.

Hospital Nearby Information: After obtaining the location, the system must utilize APIs to identify the five nearest hospitals and their contact details.

SMS Communication: The system should relay the accident location and hospital information as an SMS to the designated emergency contact.

Real-Time Response: The Smart Helmet System should initiate the alert and communication process in real-time to minimize response time.

1.5 Scope

The scope of the Smart Helmet System project is to develop an innovative and robust solution for enhancing motorcycle rider safety. The system will be designed to detect crashes using advanced sensor technology, such as accelerometers, and promptly send emergency alerts to the designated contact. By integrating GPS technology, the system will accurately track the accident's location and provide real-time information on the nearest hospitals. The project will involve developing efficient communication modules to relay accident details via SMS to the emergency contact, ensuring rapid response in case of emergencies. The system's scope includes rigorous testing and validation to ensure accurate crash detection, reliable communication, and seamless integration with the helmet. The project aims to contribute significantly to reducing the severity of injuries and saving lives in motorcycle accidents, making roads safer for riders and their families.

Chapter 2

Literature Review

2.0.1 Improved Crash Detection Algorithm for Vehicle Crash Detection - Byoungman An and YoungSeop Kim

A majority of car crash is affected by careless driving that causes extensive economic and social costs, as well as injuries and fatalities. Thus, the research of precise crash detection systems is very significant issues in automotive safety. A lot of crash detection algorithms have been developed, but the coverage of these algorithms has been limited to few scenarios. Road scenes and situations need to be considered in order to expand the scope of a collision detection system to include a variety of collision modes. The proposed algorithm effectively handles the x, y, and z axes of the sensor, while considering time and suggests a method suitable for various real worlds. To reduce nuisance and false crash detection events, the algorithm discriminated between driving mode and parking mode. The performance of the suggested algorithm was evaluated under various scenarios, and it successfully discriminated between driving and parking modes, and it adjusted crash detection events depending on the real scenario. The proposed algorithm is expected to efficiently manage the space and lifespan of the storage device by allowing the vehicle's black box system to store only necessary crash event's videos.

There are various methods for determining the crash event, using images or using autonomous driving techniques, but in this paper, we propose a method using g-sensor, which is a basic method.

G-sensor : The g-sensor is commonly referred to as an accelerometer. They are used in various devices such as smartphones, vehicles and of course black boxes. The black box's g-sensor monitors the appropriate acceleration called G-Force. The 3- axis accelerometer contains 3 accelerometers, one for each axis, which can measure the acceleration in the

±x, ±y, ±z, axes as shown in Fig. 1. The accelerometer output is highly dependent on the selected sensitivity expressed as G-force. For example, an accelerometer with 2G sensitivity can output an acceleration of up to 2G. The value is read from 2 bytes register and the precision when using high sensitivity are sufficient for the consumer to use. Decreasing the accelerometer's sensitivity also decreases precision because the length of the register where the value is stored is the same. There are 3 registers where acceleration is stored, X acceleration register, Y acceleration register and Z acceleration register. The data collected on each axis is averaged and the values obtained are used to offset the output each time the sensor is read. Most accelerometers have offset registers, and writing the values obtained after calibration into these registers will offset the output. For instance, the accelerometer's digital output has a full programmable range of ±2G, ±4G, ±8G and ±16GYou can select the appropriate sensitivity according to the application. In this paper, STMicroelectronics g-sensor was used and 2G was set up for measuring the accelerations of a vehicle

Raw Data : The raw values of the accelerometer are read by the microcontroller and are obtained by using complementary filters to perform data fusion. In this case, the accelerometer is used to correct the drift of the gyroscope. Complementary filter is an equation that creates a weighted arithmetic mean between the values of the gyroscope and accelerometer. angle = 0.90 x (angle + gyroData x t) + 0.10 x accData (1) The weight chosen for the data coming from both sensors depends on the target application. Increasing the weight of the accelerometer data improves the responsiveness of the system, but increases its sensitivity to vibration, which makes the system unstable. If the gyro weighs more than 0.90, it makes a slow but very stable system.

Car Crash Detection System : The arrangement of g-sensors in a vehicle may vary depending on the vehicle's space and design. In general, the black box is placed in the center of the vehicle because it is considered to be placed in a safe location. In general, g-sensor uses the first value read after system booting as the default value and measures the change of the corresponding value. In this paper, the initial value is specified as 1.000 for convenience and understanding of calculation. There are countless crash detection algorithms.

Chapter 3

System Analysis

3.1 Expected System Requirements

The user is expected to wear the helmet for every ride.

The emergency contact is expected to have a smartphone with sms connectivity which is regularly checked.

3.2 Feasibility Analysis

3.2.1 Technical Feasibility

The project is technically feasible since majority of motorcycle users already use helmets and majority of people are in possession of a smartphone with sms capabilities. .

3.2.2 Operational Feasibility

The operations are built in a simple and easy to use manner. The rider must only sign up to our website to order a helmet. The emergency contact must be provided at the time of sign up. The rest of the functions are carried out by the development team.

3.2.3 Economic Feasibility

The helmet is economically feasible as it is built from readily available hardware. The cost of the entire product is just the cost of the hardware as the software element is completely free to develop.

3.3 Hardware Requirements

The following are the system requirements to develop the Smart helmet.

- Arduino Uno
- GPS Neo6m
- Adxl335 Accelerometer
- NodeMCU board

3.4 Software Requirements

The following are the softwares used in the development of the project.

3.4.1 Arduino IDE for programming Arduino UNO and NodeMCU

The Arduino Integrated Development Environment (IDE) is a user-friendly software platform designed for programming Arduino microcontrollers. It provides a code editor with features like syntax highlighting and auto-completion to simplify writing Arduino sketches in C/C++. The IDE includes a Library Manager for easy installation and management of pre-written code libraries, expanding the capabilities of Arduino boards. It also offers a Serial Monitor tool for debugging and communication between Arduino boards and computers. With the Board Manager, users can select and configure the appropriate board for programming. The IDE allows users to verify and upload their code to Arduino boards via a USB connection. Arduino IDE's collection of examples and tutorials further aids beginners in learning and developing various electronic projects and IoT applications. As an open-source software, Arduino IDE fosters community collaboration and customization.

3.4.2 MySQL Database

is an open-source relational database management system (RDBMS) widely used for storing, managing, and retrieving data in various applications. It provides a structured query language (SQL) to interact with the database, allowing users to perform tasks such as creating, modifying, and querying data. MySQL offers a robust and scalable solution for organizing and storing data, making it suitable for small to large-scale applications. Its high performance and efficient data retrieval capabilities make it popular in web development, e-commerce, content management systems, and other data-driven

applications. MySQL also prioritizes data security with features like user authentication, access controls, and data encryption. As an open-source software, it encourages community collaboration, leading to continuous improvement and customization. Its widespread adoption, user-friendly interface, and active community support have solidified MySQL's position as one of the most prominent and reliable database management systems in the industry.

3.4.3 MongoDB

MongoDB is an open-source NoSQL database management system that employs a document-based data model for storing and managing unstructured or semi-structured data. It stores data in collections, each containing BSON documents with varying fields, providing flexibility and scalability. MongoDB's dynamic schemas allow easy modification of data without disrupting existing records. It offers fast data retrieval by eliminating the need for complex joins commonly found in relational databases. Its distributed architecture enables horizontal scaling, ensuring high availability and performance by distributing data across multiple servers. MongoDB's adaptability, speed, and ability to handle diverse data types make it a popular choice for modern applications, particularly in scenarios where data structures evolve and grow rapidly.

3.4.4 Twillio API

Twilio is a cloud communications platform that provides Application Programming Interfaces (APIs) for developers to integrate voice, messaging, and video functionalities into their applications. The Twilio API allows developers to send and receive SMS messages, make and receive phone calls, and implement other real-time communication features programmatically. It offers easy-to-use RESTful APIs and SDKs in various programming languages, simplifying the process of integrating communication capabilities into web and mobile applications. Twilio's scalability, reliability, and global reach make it a popular choice for businesses to enhance customer engagement and streamline communication processes. Developers can leverage Twilio API to build interactive and personalized communication experiences, enabling seamless and efficient communication with customers and users.

3.4.5 Here API

HERE API is a powerful location-based platform that provides Application Programming Interfaces (APIs) for developers to access a wide range of location-related services and data. It offers functionalities such as geocoding, reverse geocoding, routing, traffic information, and mapping services. With HERE API, developers can integrate location-based features into their applications, enabling accurate and real-time location tracking, navigation, and geospatial analysis. HERE API offers SDKs and RESTful APIs in various programming languages, ensuring ease of integration into web and mobile applications. Its extensive coverage and reliable geospatial data make it a popular choice for businesses in industries like transportation, logistics, and mapping, empowering them to deliver location-aware and location-based services to their users efficiently.

Chapter 4

Methodology

4.1 Proposed Method

- The project aims to develop a smart bike helmet with an integrated accident alert system. The helmet is equipped with sensors capable of detecting potential accidents or impacts during a motorcyclists ride.
- The project involves creating a user-friendly web page where riders can sign in and enter their emergency contact details.
- To process the sensor data and enable communication between components, the system utilizes the Arduino Uno micro controller board. ADXL 335 is used for impact detection, GSM module for getting the coordinates and NodeMCU for network connection .
- Once an accident is detected by the sensors, the system triggers an immediate alert. The alert is sent to the registered emergency contact through an SMS, which contains the accident location and details of the most nearby hospitals. A call is also send to the number to attract the responder's attention.

4.1.1 Accident detection

In the Smart Bike Helmet Accident Alert System, the processing of sensor data and communication between components is crucial for its effective operation. The system utilizes specific hardware components, including the Arduino Uno microcontroller board, ADXL335 accelerometer, GPS neo 6m module, and NodeMCU, to achieve its objectives.

The Arduino Uno serves as the central control unit of the system. It acts as the brain, receiving data from various sensors and executing the accident detection algorithm. The system is equipped with the ADXL335 accelerometer sensor, which continuously measures

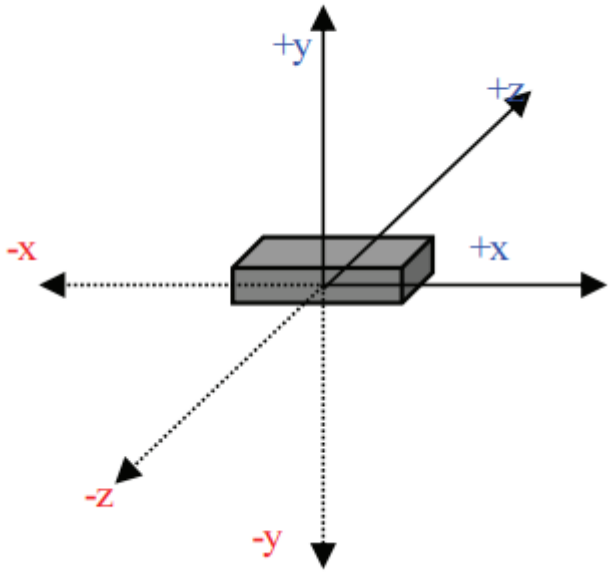


Figure 4.1: G-sensor (accelerometer) axes

acceleration along three axes (X, Y, and Z). The accelerometer detects any sudden changes in acceleration that could indicate an impact or collision.

The algorithm used in the Smart Bike Helmet Accident Alert System effectively converts the X, Y, and Z axis readings from the accelerometer into a magnitude value. This magnitude value is essential for detecting significant impacts or collisions during the motorcyclist's ride. The algorithm follows a systematic approach of sampling data every 2 seconds to calculate the magnitude. The detailed explanation of the algorithm is as follows: At the beginning of the process, the algorithm initializes variables to store the old and new accelerometer values for each axis (X, Y, and Z). It also sets variables to store the differences between the old and new values for each axis.During each sampling period of 2 seconds (or any desired time interval), the algorithm collects new readings from the accelerometer. These new readings represent the current acceleration along the X, Y, and Z axes .After collecting the new values, the algorithm computes the differences between the old and new values for each axis. These differences represent the change in acceleration along each axis during the 2-second interval. To calculate the magnitude value, the algorithm computes the sum of the squares of the differences for each axis. This is done by squaring each individual difference and summing them up. The result is the square of the differences. Finally, the algorithm takes the square root of the sum of the

squares of differences to obtain the magnitude value. This magnitude value represents the overall acceleration during the 2-second interval and indicates the intensity of any impact or collision. After calculating the magnitude, the algorithm updates the old accelerometer values for each axis with the latest readings. This prepares the algorithm for the next 2-second sampling period.

To determine if an accident has occurred, the recorded acceleration is compared to a predefined threshold value. If the acceleration surpasses this threshold, it is considered a potential accident. Upon detecting an impact exceeding the threshold, the Arduino Uno, which acts as the system's central controller, receives this information from the ADXL335 accelerometer. The Arduino Uno then activates the GPS module, which is integrated into the system, to retrieve the current location of the helmet at the time of the impact. The GPS module utilizes Global Positioning System technology to determine the accurate latitude and longitude coordinates of the accident location. Once the GPS module acquires the accident location, the Arduino Uno processes this data. Subsequently, the Arduino Uno sends the location data to the NodeMCU. The NodeMCU, functioning as a Wi-Fi module, establishes a network connection to enable communication with external devices and databases. With the network connection established, the NodeMCU uploads the accident data in JSON format to a Python backend. The Python backend is responsible for further processing and inserting the data into a MySQL database. The MySQL database is specifically set up to store and manage accident-related information effectively.

4.1.2 Alert Mechanism

The Smart Bike Helmet incorporates an alert mechanism to promptly notify emergency contacts and provide information about nearby hospitals in case of an accident. The Arduino Uno sends the accident data, including the GPS coordinates, to the NodeMCU. The NodeMCU, functioning as a Wi-Fi module, establishes a network connection and sends the received data in JSON format to the Python backend for additional analysis.

The Python backend plays a crucial role in the system. It processes the received JSON data and extracts the accident location coordinates. To facilitate locating nearby hospitals, the backend integrates the HERE API. By providing the HERE API with the accident coordinates as input, the backend retrieves a JSON file consisting of a list of nearby hospitals based on the accident location.

The system implements the Alert Mechanism using the Twilio API integrated into the Python backend. First, the backend filters and selects the required hospitals from the HERE API response. Then, it utilizes the Twilio API to initiate a call to the designated emergency responders' numbers, effectively attracting their attention. Subsequently, the backend sends SMS messages to the registered emergency contacts, utilizing the accident location coordinates and links to the nearest hospitals obtained from the HERE API response.

Chapter 5

System Design

5.1 Architecture Diagram

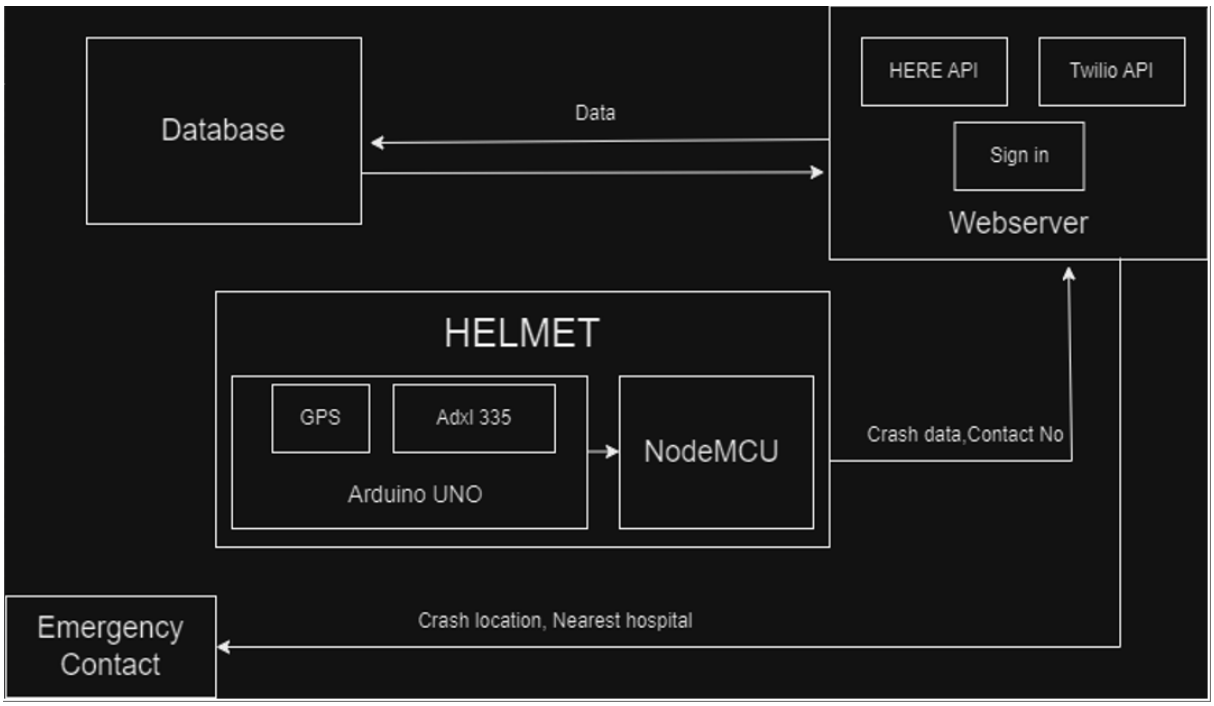
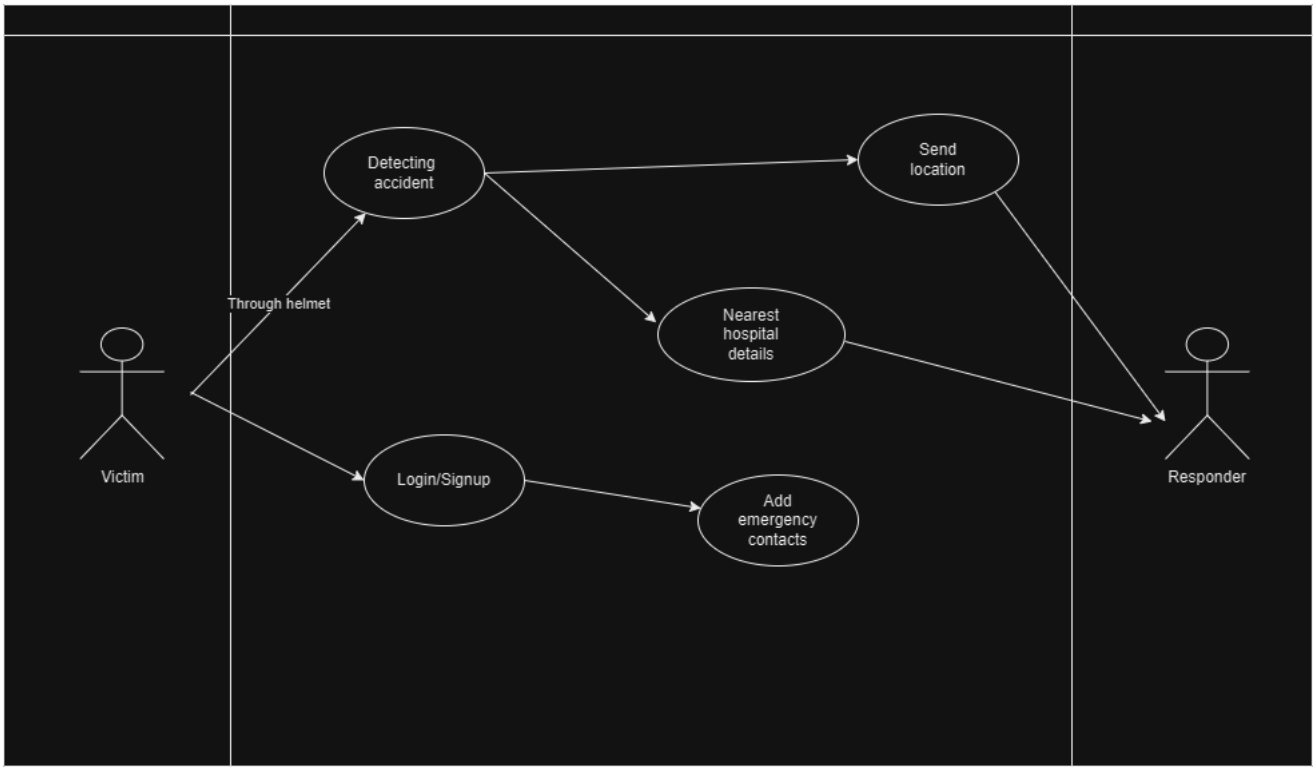


Figure 5.1: Architecture diagram

5.2 UML Diagram



Chapter 6

System Implementation

6.1 Sign up/log in

A web page has been designed with a clean and intuitive user interface, allowing users to easily sign up or log in. Once logged in, users can enter their preferred emergency contact number, which will be used to alert them in the event of any crash or accident. This contact number is then manually uploaded onto the Arduino UNO before delivering the helmet to the customer.

6.2 Crash detection

The code uploaded onto the Arduino UNO is equipped to detect impact arising from any crash or hit. The output from the X,Y,Z pins of the ADXL 335 sensor is collected every two seconds, compared with the previous values, the corresponding magnitude is computed and verified to determine whether it has crossed a predefined threshold value. Whenever the specified threshold is crossed, the Arduino UNO extracts the geographical coordinates and timestamp of impact from GPS Neo 6m and forwards this data, along with the crash magnitude and emergency contact number, which was initially uploaded to the Arduino UNO upon purchase, to the NodeMCU. The task assigned to NodeMCU is straightforward: Send the received data in JSON format to the python backend for further processing and transmission via APIs.

6.3 Python server

6.3.1 Insertion into mySQL Database

The JSON data transmitted by the NodeMCU is tokenized into magnitude, latitude, longitude, timestamp, and emergency contact number after which it is entered into the

MySQL database.

6.3.2 Tracing the nearby hospitals

Thanks to the HERE API, what might seem difficult at first couldn't be any simpler. Upon inserting the latitude and longitude of the crash into a URL provided to the registered users of HERE API, the nearby hospitals and its details are obtained in the format of a JSON document from which only the hospital names are extracted and inserted into a python list.

6.3.3 Alerting the emergency contact

The Twilio API provides the functionalities of sending calls and SMS to verified numbers. The SMS consists of the following details

- A google maps URL of the crash location obtained by pasting the latitude and longitude into it.
- The google search page of each nearby hospital obtained using HERE API, which consists of all the essential details such as address, ratings, contact number etc.

Finally a call with default voice contents, as designed by Twilio API is made to the emergency contact to make sure he/she doesn't miss the crash notification.

Chapter 7

Testing

7.1 Unit testing

All the hardware components and the sign up web page were tested individually to evaluate if they worked as intended.

7.1.1 ADXL 335

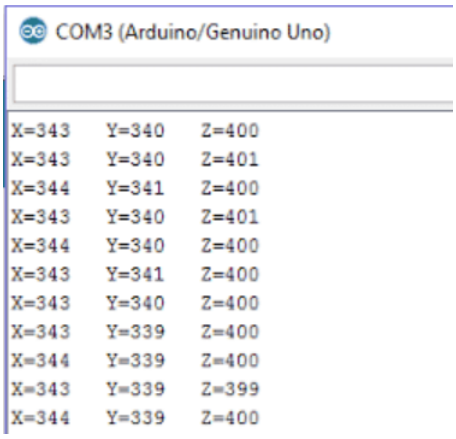


Figure 7.1: ADXL 335 logs

7.1.2 GPS Neo 6m

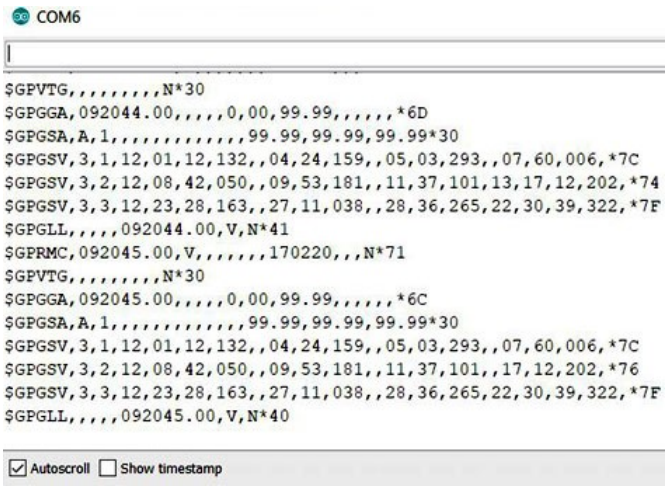


Figure 7.2: GPS Neo 6m logs

7.1.3 Sign up

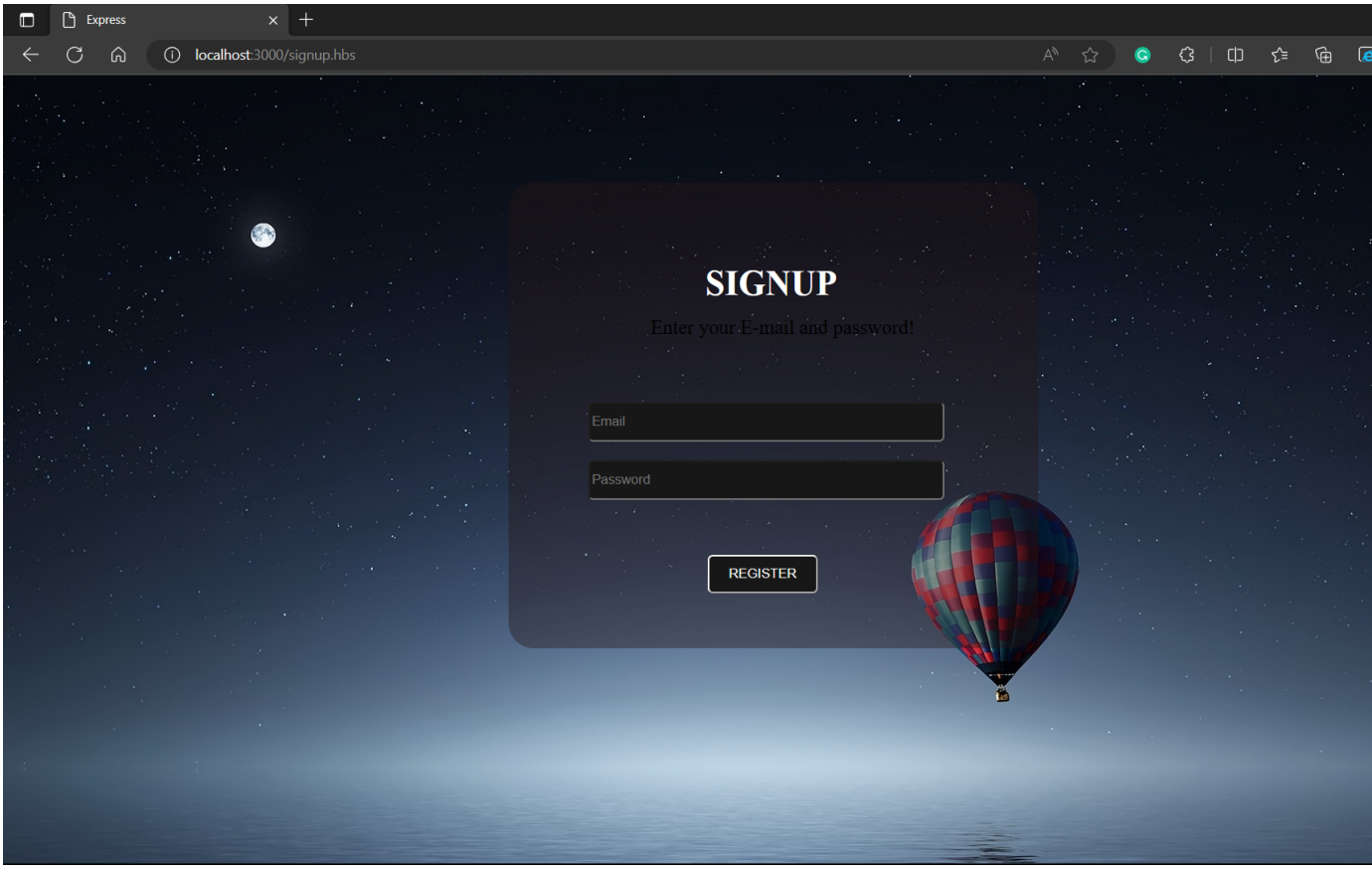


Figure 7.3: Sign up page

7.2 Integration testing

Data transmitted from the NodeMCU was successfully inserted into the mySQL database.

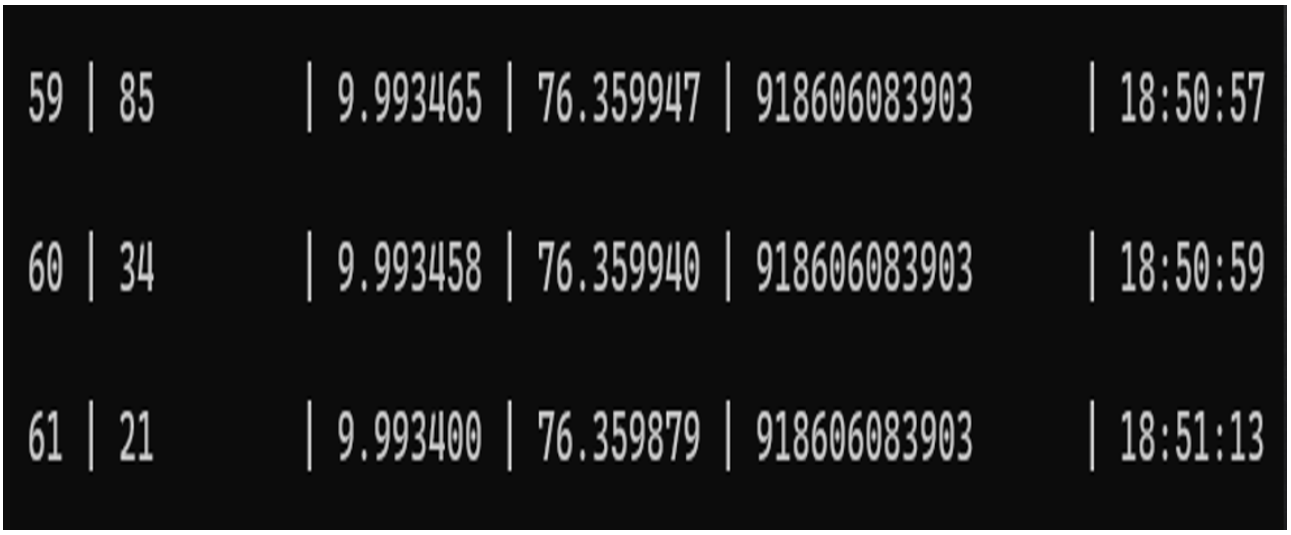


Figure 7.4: Database entries after insert operation

7.3 System testing

All components were successfully integrated to deliver a successful SMS to a dummy emergency contact.

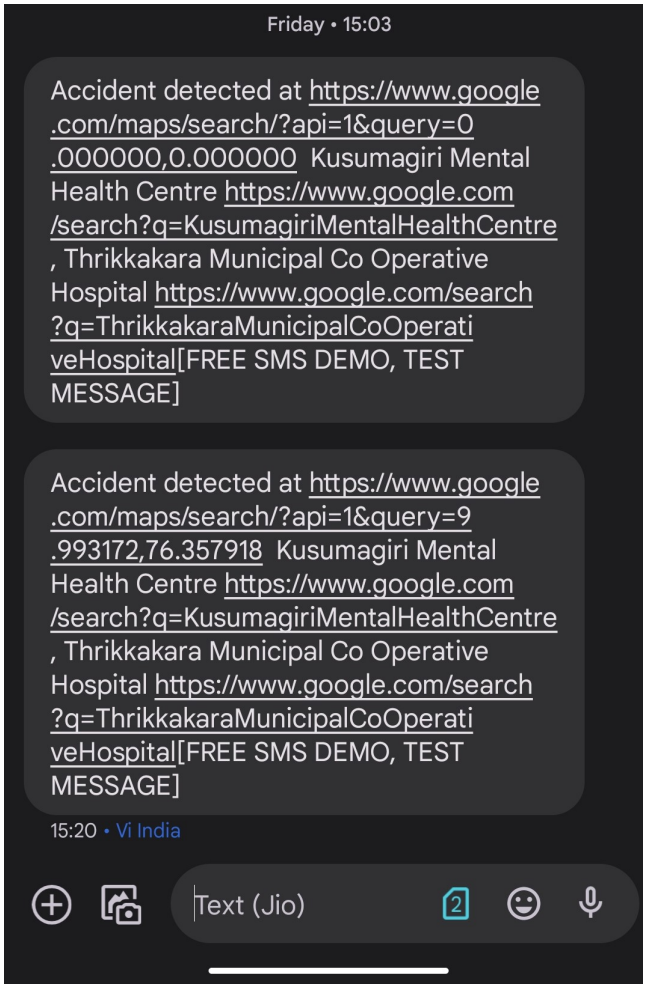


Figure 7.5: SMS incorporting the gmaps URL and hospital search pages

Chapter 8

Results

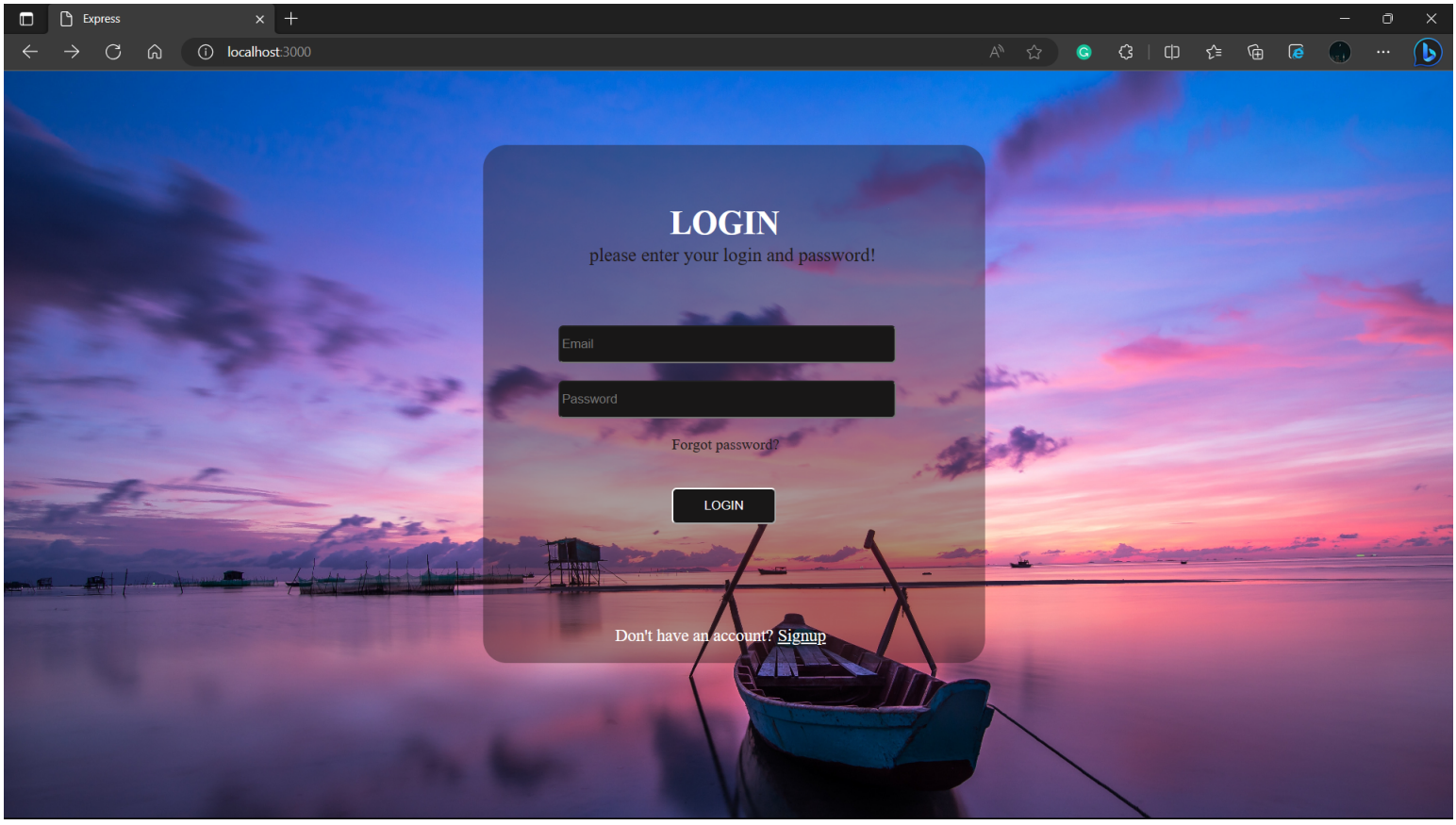


Figure 8.1: User Login interface

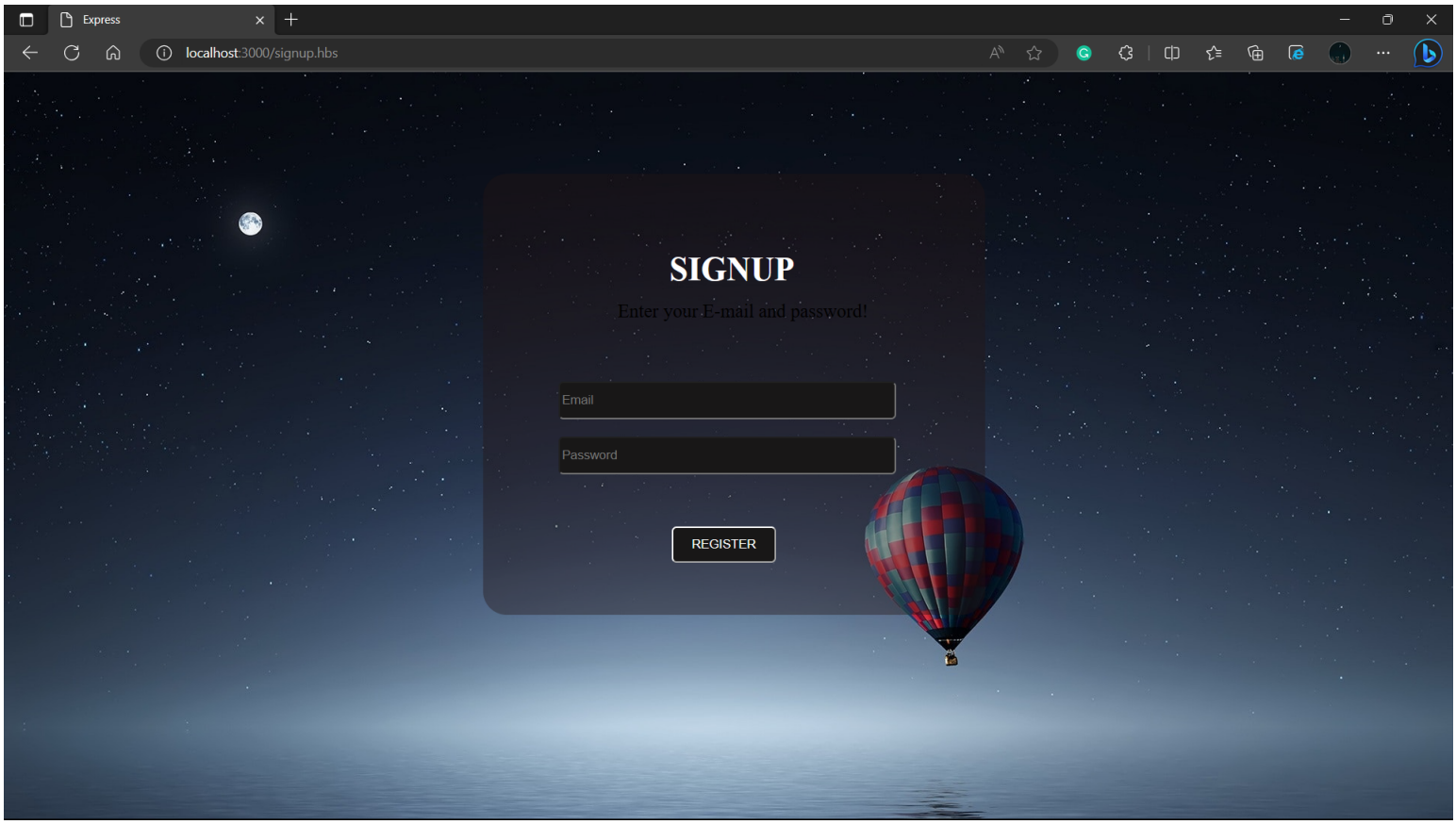


Figure 8.2: User Signup interface

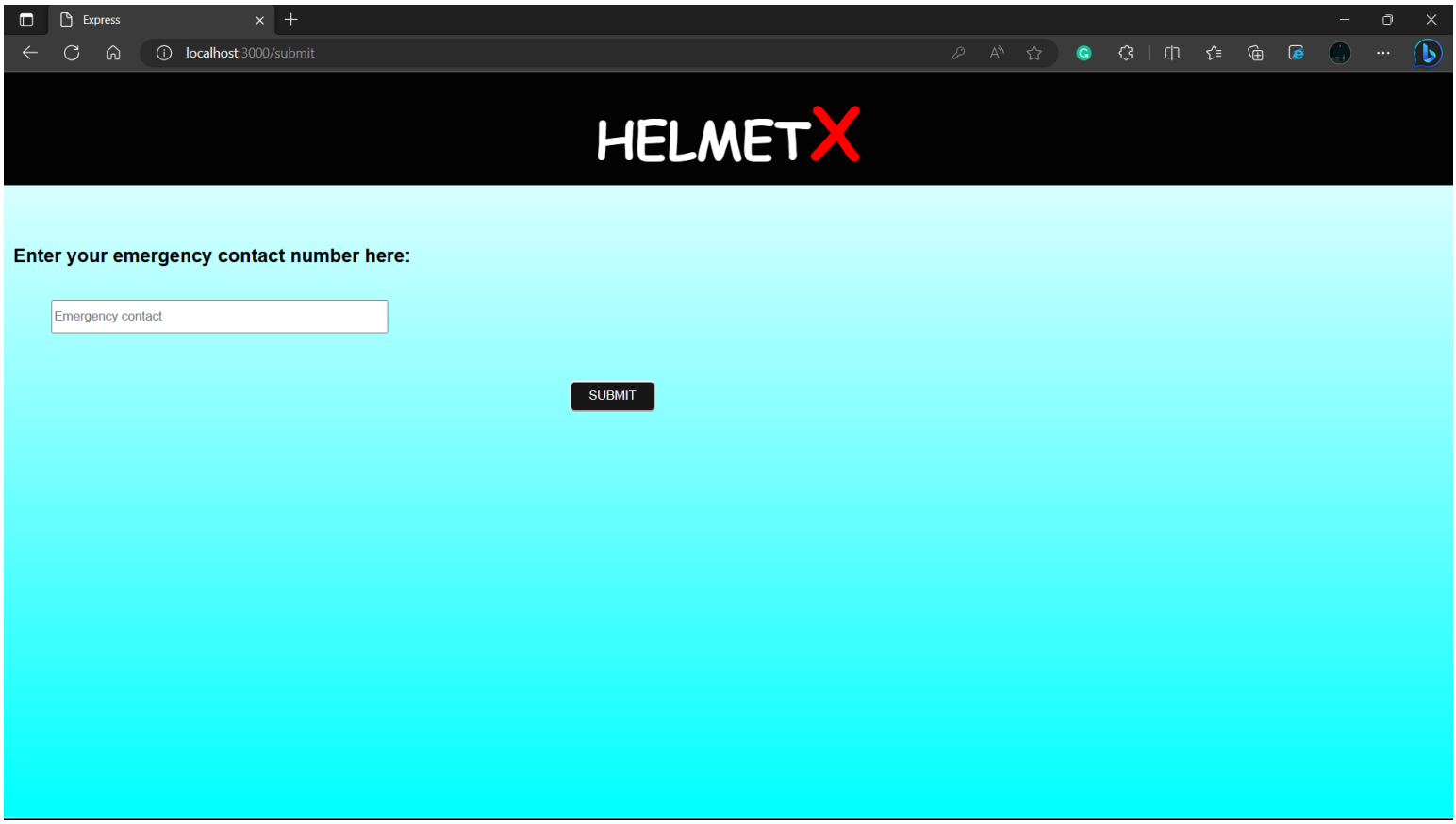


Figure 8.3: Emergency number submission interface

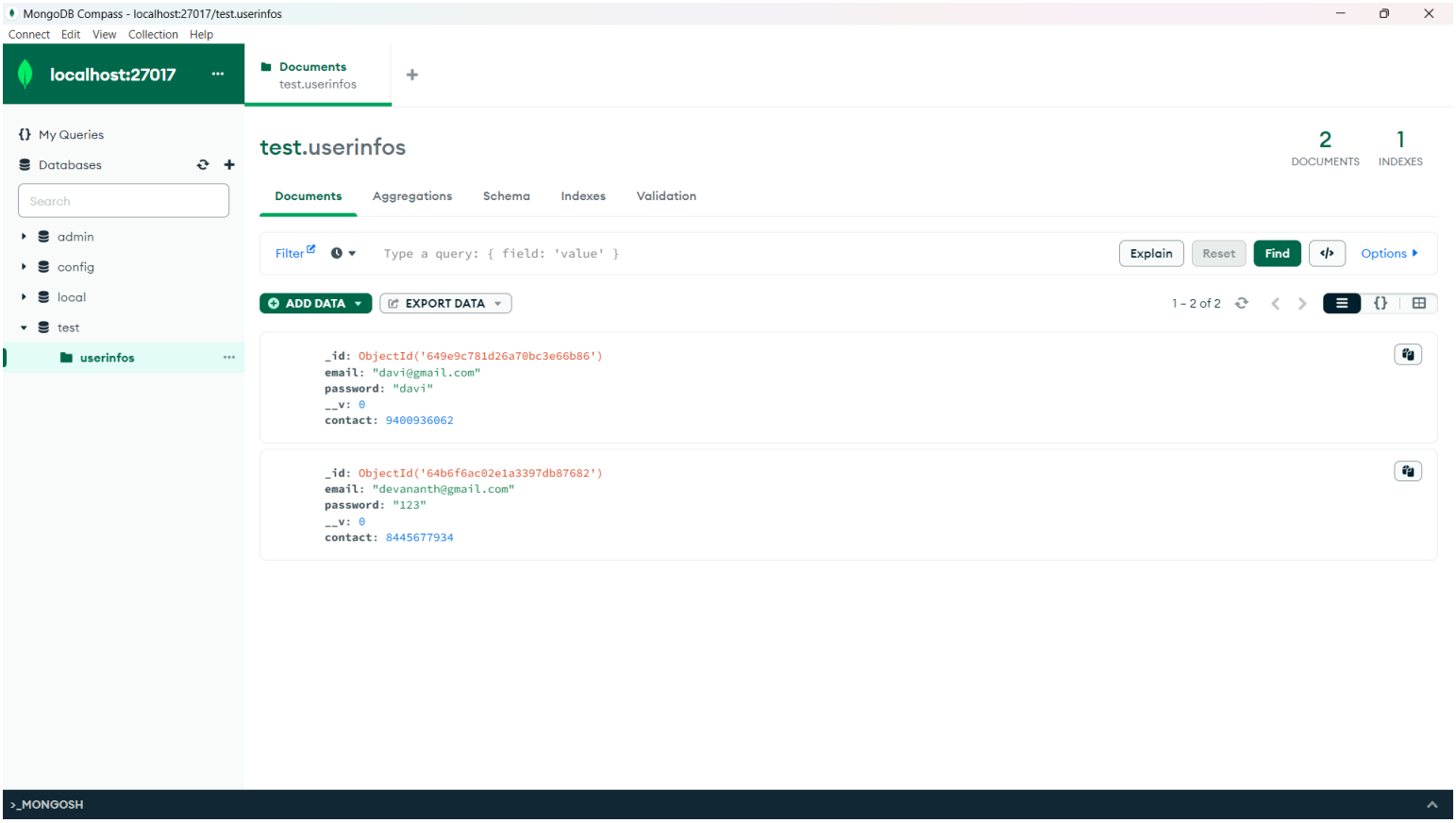


Figure 8.4: User authentication MongoDB database

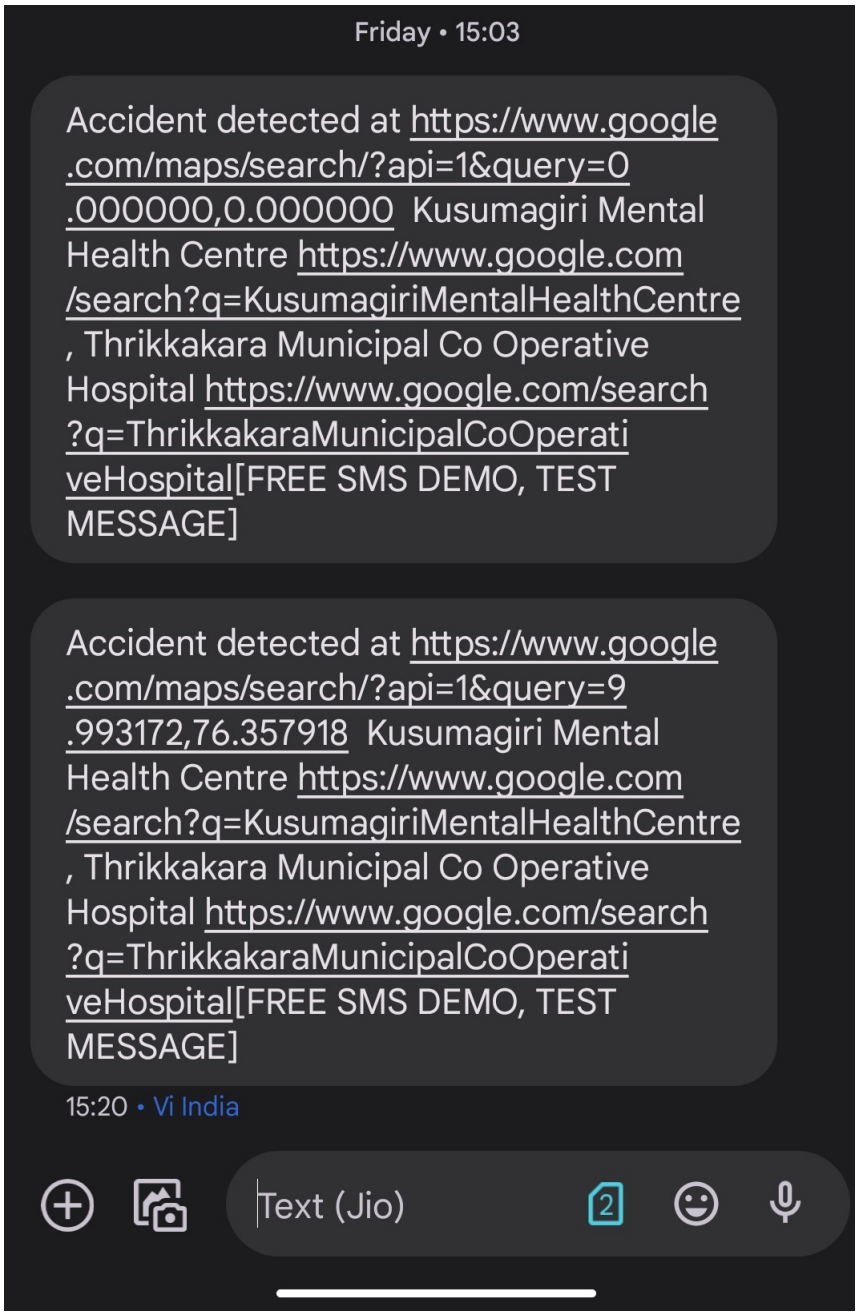
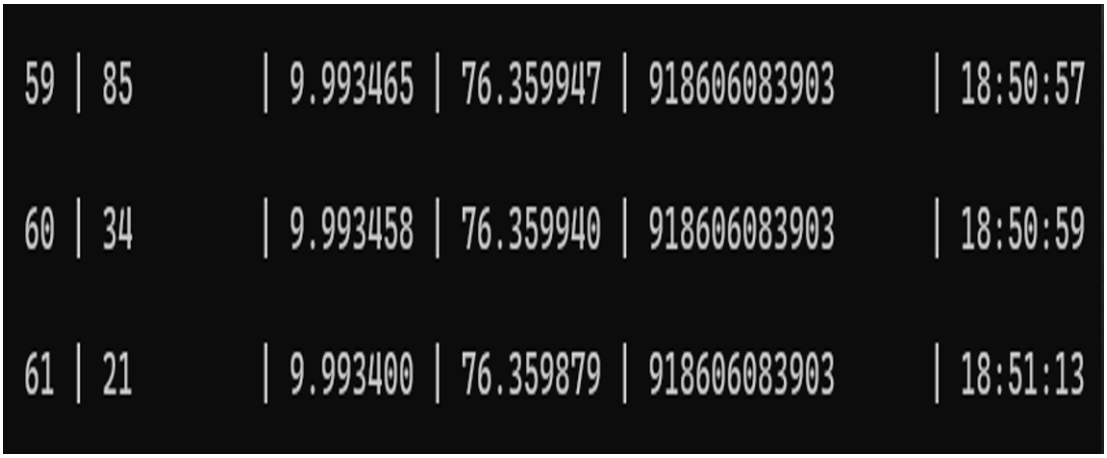


Figure 8.5: Emergency contact alerting SMS



59	85	9.993465	76.359947	918606083903	18:50:57
60	34	9.993458	76.359940	918606083903	18:50:59
61	21	9.993400	76.359879	918606083903	18:51:13

Figure 8.6: MySQL database for data collection

Chapter 9

Risks and Challenges

1. Helmet must always be used by the rider.
2. False alarms must be dealt with.
3. The sensor may suffer damages in major accidents.
4. GPS module needs a clear view of the sky to get a signal.
5. The battery must be fully charged whenever necessary.
6. The cost of the smart bike helmet will be more than that of a normal helmet.

Chapter 10

Conclusion

In conclusion, the Smart Bike Helmet project has been successfully developed and implemented, providing a comprehensive and effective solution to enhance the safety of motorcyclists on the roads. The system aims to promptly alert emergency contacts and locate nearby hospitals in the event of an accident, ensuring quick response and medical attention when needed.

A user-friendly web page was developed, allowing riders to sign in and enter their emergency contact details, enabling quick and easy setup of the system. The core functionality of the project, the accident alert system, has been successfully implemented. The helmet is equipped with sensors capable of detecting potential accidents or impacts. Once an accident is detected, the system promptly sends alerts to the registered emergency contacts with the details of the nearby hospitals and accident location. The magnitude of impact, GPS location, and time of impact are automatically uploaded into a MySQL database.

With the help of the data collected, we can identify accident hotspots and peak hours of accidents. Police can intensify patrolling and checks in these locations during specified times, which will aid in reducing road accidents.

References

[1] Mohammad Ehsanul Alim, Sarosh Ahmad, Marzieh Naghdi Dorabati, Ihab Hassoun. Design and Implementation of IoT Based Smart Helmet for Road Accident Detection. IEEE. 2022

[2] Byoungman An and YoungSeop Kim. Improved Crash Detection Algorithm for Vehicle Crash Detection

[3] Weimeng Niu, Liqing Fang, Lei Xu, Xu Li. Summary of Research Status and Application of MEMS Accelerometers. 2018

Appendix A: Sample Code

Arduino code:

_____

```
#include <ArduinoJson.h>

#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>



SoftwareSerial arduinoSerial(D2, D1); // RX, TX pins

const char* ssid = "Moto g52";
const char* password = "dragonblood";
String data;

ESP8266WebServer server(80);

void setup() {
  Serial.begin(9600);
  arduinoSerial.begin(9600);

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("WiFi connected");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
```

```
 server.on("/data", handleDataRequest);  // Data endpoint route
 server.begin();
}

 void loop() {
 if (arduinoSerial.available()) {
    data = arduinoSerial.readString();
   Serial.println(data);
   server.handleClient();
   server.handleClient();
 }

 }

 void handleDataRequest() {
 //String data1 = "{\"crash_data\":data}";  // Replace with your actual data

 DynamicJsonDocument doc(256);
 doc["crash_data"] = data;
 String jsonData;
 serializeJson(doc, jsonData);
 server.send(200, "application/json", jsonData);
 Serial.println(jsonData);
}
```

NodeMCU code:

_____

```
#include <ArduinoJson.h>
```

```
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>


SoftwareSerial arduinoSerial(D2, D1); // RX, TX pins

const char* ssid = "Moto g52";
const char* password = "dragonblood";
String data;

ESP8266WebServer server(80);

void setup() {
  Serial.begin(9600);
  arduinoSerial.begin(9600);

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("WiFi connected");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());

  server.on("/data", handleDataRequest);  // Data endpoint route
  server.begin();
}

  void loop() {
```

```
 if (arduinoSerial.available()) {
    data = arduinoSerial.readString();
   Serial.println(data);
   server.handleClient();
   server.handleClient();
 }

 }

 void handleDataRequest() {
 //String data1 = "{\"crash_data\":data}";  // Replace with your actual data

 DynamicJsonDocument doc(256);
 doc["crash_data"] = data;
 String jsonData;
 serializeJson(doc, jsonData);
 server.send(200, "application/json", jsonData);
 Serial.println(jsonData);
}
```

Python backend

_____

```
import requests
import time
import mysql.connector
import twilio
from twilio.rest import Client

account_sid = "AC30ab565293217237d0c85e126f9aebfd"
```

```
auth_token = "53f0e13246e5efe376e564f7fdc966a1"

url = 'https://discover.search.hereapi.com/v1/discover?at='+peices[1]+','+peices[2]+'&q=hospital&apikey=84q8N9DRx6YAqyJaXUpr9fhiTnCUIBzvHLpk3QU4Oaw&limit=4'

strl=''

strh=''

item2=''

peices=[]


# MySQL database configuration

mysql_config = {

    "host": "localhost",

    "user": "root",

    "password": "1234",

    "database": "project"

}


# NodeMCU IP address and port

nodemcu_ip = "192.168.208.149"

nodemcu_port = 80


# Get data from NodeMCU

def get_data_from_nodemcu():

    url = f"http://{nodemcu_ip}:{nodemcu_port}/data"  # Replace with your NodeMCU data endpoint

    response = requests.get(url)

    if response.status_code == 200:

        data1 = response.json()

        print(data1)

        return data1


    else:

        print("Error retrieving data from NodeMCU:", response.status_code)
```

```
        return None


# Post data to MySQL
def post_data_to_mysql(data1):
    peices = data1["crash_data"].split("\t")
    print(peices)


    try:
        connection = mysql.connector.connect(**mysql_config)
        cursor = connection.cursor()


        # Modify the query below according to your MySQL table structure
        query = "INSERT INTO crash_data (Magnitude, Latitude, Longitude, Emergency_Contact,
Timestamp) VALUES (%s, %s, %s, %s, %s)"
        values = (peices[0], peices[1], peices[2], peices[3], peices[4])
        cursor.execute(query, values)
        connection.commit()


        print("Data posted to MySQL successfully.")


    except mysql.connector.Error as error:
        print("Error posting data to MySQL:", error)


    finally:
        if connection.is_connected():
            cursor.close()
            connection.close()
        return(peices)


def api(arr):
    response = requests.get(url)
```

```
    client = Client(account_sid, auth_token)
    strh = 'Accident detected at
'+'https://www.google.com/maps/search/?api=1&query='+arr[1]+','+arr[2]+'  '
    if response.status_code == 200: #successful http req
            data = response.json()
            filtered_titles = [item['title'] for item in data['items']]
            print(filtered_titles)
            for item in filtered_titles:
               item2=item.replace(" ","")
               strh += item+' '+'https://www.google.com/search?q='+item2+', '

    strh_final=strh[:-2]
    print(strh)
    call = client.calls.create(
            url='http://demo.twilio.com/docs/voice.xml',
            to='+'+arr[3],
            from_='+15738594232'
            )

    print(call.sid)
    message = client.messages.create(
               body=strh_final,  # SMS content
               to='+'+arr[3],
               from_='+15738594232'
               )
    if message.sid:
            print('SMS sent successfully!')
    else:
            print('Failed to send SMS.')
```

```
# Main program
if __name__ == '__main__':

    while True:



        data = get_data_from_nodemcu()

        if data:

            a = post_data_to_mysql(data)
            #print(a)
            api(a)
        else:
            print("Failed to retrieve data from NodeMCU.")

        #time.sleep(2)  # Pause execution for 2 seconds
```

Appendix B: CO-PO And CO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
100003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
100003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
100003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
100003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
100003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
100003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
100003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
100003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
100003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
100003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
100003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
100003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
100003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
100003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

100003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
100003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
100003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
100003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
100003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
100003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
100003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
100003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
100003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
100003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
100003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
100003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
100003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
100003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
100003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

100003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
100003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
100003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
100003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
100003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
100003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

100003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
100003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
100003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
100003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
100003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
100003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
100003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
100003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
100003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
100003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
100003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
100003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.