VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi - 590 018

MINIPROJECT REPORT ON

IoT Based Automatic Vehicle Accident Detection and Rescue System

Mini Project Report submitted in partial Fulfillment for the Award of Degree of

Bachelor of Engineering

in

Electronics and Communication Engineering

Submitted by

Bhargava V Desai	1RN18EC032
Biswajit Bhunia	1RN18EC033
Dheeraj D	1RN18EC043
Dokku Balaji	1RN18EC049

Under the Guidance of

Ms. Leena Chandrashekar

Asst. Professor

ESTD 2001

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

(Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-21)

RNS INSTITUTE OF TECHNOLOGY

(AICTE Approved, VTU Affiliated and NAAC 'A' Accredited)

(UG Programs - CSE, ECE, ISE, EIE and EEE have been Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-2021)

Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560098

2020 - 21

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi - 590 018

MINIPROJECT REPORT ON

IoT Based Automatic Vehicle Accident Detection and Rescue System

Mini Project Report submitted in partial Fulfillment for the Award of Degree of

Bachelor of Engineering

in

Electronics and Communication Engineering

Submitted by

Bhargava V Desai	1RN18EC032
Biswajit Bhunia	1RN18EC033
Dheeraj D	1RN18EC043
Dokku Balaji	1RN18EC049

Under the Guidance of

Ms. Leena Chandrashekar

Asst. Professor

ESTD 2001

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

(Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-21)

RNS INSTITUTE OF TECHNOLOGY

(AICTE Approved, VTU Affiliated and NAAC 'A' Accredited)

(UG Programs - CSE, ECE, ISE, EIE and EEE have been Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-2021)

Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560098

2020 - 21

RNS INSTITUTE OF TECHNOLOGY

(AICTE Approved, VTU Affiliated and NAAC 'A' Accredited)

(UG Programs - CSE, ECE, ISE, EIE and EEE have been Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-2021)

Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560098

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

(Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-21)

ESTD 2001

CERTIFICATE

Certified that the Mini Project report entitled **"Iot Based Automatic Vehicle Accident Detection and Rescue System"** is carried out by **Bhargava V Desai (1RN18EC032), Biswajit Bhunia (IRN18EC033),Dheeraj D (1RN18EC043) and Dokku Balaji(1RN18EC049)** in partial fulfillment for the award of degree of Bachelor of Engineering in Electronics and Communication Engineeringof Visvesvaraya Technological University, Belagavi, during the year 2020-2021. It is certified that all corrections / suggestions indicated during internal assessment have been incorporated in the report. The mini project report has been approved as it satisfies the academic requirements in aspect of the project work prescribed for the award of degree of Bachelor of Engineering.

....	....	....
Ms. Leena Chandrashekar	Dr. Vipula Singh	Dr. M K Venkatesha
Asst. Professor	Head Of Department	Principal

External Viva

Name of the examiners	Signature with date
1	....
2	....

RNS INSTITUTE OF TECHNOLOGY

(AICTE Approved, VTU Affiliated and NAAC 'A' Accredited)

(UG Programs - CSE, ECE, ISE, EIE and EEE have been Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-2021)

Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560098

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

(Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-21)

ESTD 2001

DECLARATION

We here by declare that the entire work embodied in this mini project report titled, "**IoT based Automatic Vehicle Accident Detection and Rescue System**" submitted to **Visvesvaraya Technological University**, Belagavi, is carried out at the department of **Electronics and Communication Engineering, RNS Institue of Technology, Bengaluru** under the guidance of **Ms. Leena Chandrashekar**, Asst. Professor. This report has not been submitted for the award of any Diploma or Degree of this or any other University

Name	USN	Signature
1. Bhargava V Desai	1RN18EC032	....
2. Biswajit Bhunia	1RN18EC033	....
3. Dheeraj D	1RN18EC043	....
4. Dokku Balaji	1RN18EC049	....

Acknowledgement

The joy and satisfaction that accompany the successful completion of any task would be incomplete without thanking those who made it possible. We consider ourselves proud to be a part of RNS Institute of Technology, the institution which moulded us in all our endeavors.

We express our gratitude to our beloved Chairman late **Dr. R N Shetty** and **MD Satish R Shetty** for providing state of art facilities.

We would like to express our sincere thanks to **Dr. M K Venkatesha**, Principal and **Dr. Vipula Singh**, Professor and Head, Department of ECE, for their valuable guidance and encouragement throughout our program.

We express our profound gratitude to the coordinators who have given valuable suggestions and guidance throughout the project. We would like to express our sincere gratitude to our guide **Ms. Leena Chandrashekar**, Asst. Professor, for her guidance, continuous support and motivation in completing the project successfully.

Finally, we take this opportunity to extend our earnest gratitude and respect to our parents, teaching and non-teaching staff of the department, the library staff and all our friends who have directly or indirectly supported us

Bhargava V Desai

Biswajit Bhunia

Dheeraj D

Dokku Balaji

Abstract

The rapid rise of technology and infrastructure has made our lives easier and convenient. The increase in demand for automotive has also increased traffic hazards and road accidents. Delay in reaching the ambulance to the accident location due to traffic congestion or late information about the accident can cause severe damage to victims or even lead to death in the worst scenario. To reduce this risk to the bare minimum, this Automatic Vehicle Accident detection and Rescue Systems will play a huge role.

In the proposed IoT-based accident detection and rescue system, detection of accidents can be done by using various sensors. These sensors continuously check for untimely deviation from the predefined optimal cases for safe driving. The accident location can be traced by using a GPS module and Wi-Fi module for sending messages to the concerned people and the rescue team through the Cloud IoT network. All of these operations are monitored and controlled by Atmega Microcontroller used in the approached method. The system that has been laid before reduces the loss of damage due to accidents by early detection, also reducing the time lapse for the rescue operation and taking the victim to the hospital.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In 2015, there were about five lakh road accidents in India, which killed about 1.5 lakh people and injured about five lakh people. India, as a signatory to the Brasilia declaration, intends to reduce road accidents and traffic fatalities by 50% by 2022. The Motor Vehicles (Amendment) Bill, 2016 has been listed for consideration and passage in the current Budget Session of Parliament. It seeks to address issues related to road accidents, third party insurance and road safety measures. In this context, we present some data on road accidents, causes of accidents, and motor vehicle third party insurance. Road length in India has increased from about 4 lakh kilometres in the 1950s to about 55 lakh kilometres in 2015. Majority of this growth has been in rural roads and roads constructed by the Public Works Department (PWD). Rural roads account for 61%, and PWD roads for 20% of the total road length. In comparison, urban roads have a 9% share in the road network. The growth in rural roads may be attributed to schemes such as the Pradhan Mantri Gram Sadak Yojana, which was launched in 2000 and aimed to improve road connectivity in rural areas.

National Highways constitute 2%, and State Highways 3% of the total road length. Project Roads, account for 7% of the total road length, and include roads built by various state departments such as forest, irrigation, electricity, public sector undertakings such as Steel Authority of India, and the Border Roads Organisation. Since 2000, while road network in the country has grown by 39%, the number of registered vehicles has grown by about 158%. While growth in road network will be limited (due to physical constraints), a constant increase in the number of vehicles on roads may lead to congestion and road fatalities. [1]

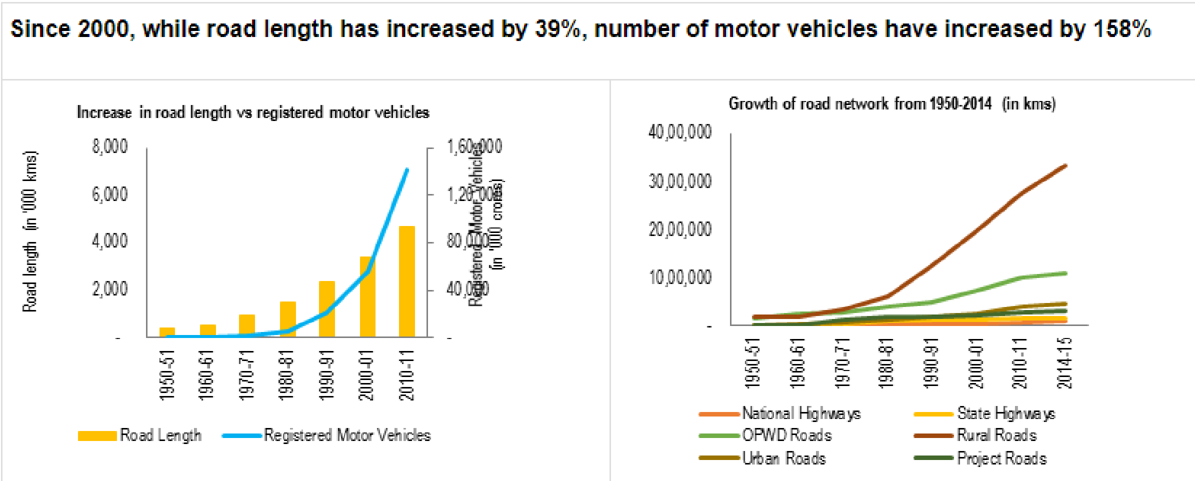


Figure 1.1: Statistics of roads

To protect the vehicle and to track it, so many advanced technologies are available nowadays. In olden days the information of accidents can be transferred, but the place of the accident spot cannot be identified.

In any vehicle airbags are designed, air bags are used for security and safety travels. TPMS (Tire Pressure Monitoring System) is a system designed to control the pressure inside the pneumatic tires on vehicles that provides different operating conditions such as a lower tire pressure is desired in order to maximize traction, maneuvering through challenging terrain, pulling a heavy load out of an incline at slow speeds, crawling out of 2 soft dirt. Many other systems have been proposed to deduce the accident.

The proposed system deals with two sensors where an Accelerometer sensor is used to detect the angle of roll and pitch of the vehicle. Vibration sensor is used for detection of the frequency change in the vehicle's body. To constantly remind the person who is driving about the status of the vehicle, to check its functionality by detecting anomalies using various sensors. After the accident has been detected, a direct message is sent to the nearby hospital and the chosen guardians.

1.2 Motivation

According to the World Health Organization (WHO) report, nearly 1.35 million people died in road accidents, making road traffic injuries the eight leading cause of death globally. The number of fatalities associated with road accidents is extremely high, thus, measures must be taken to improve road safety. Most injuries incurred by accidents are not serious, and the victim's life can be saved if rescued timely. However, it takes additional delay to manually notify the emergency teams due to poor communication mechanisms, thus, victims are left unattended for a long time, resulting in an increased death rate. The consequences of road accidents are not just constrained to the loss of human lives yet, but also incorporate the destruction of property, traffic blockages, and immense economic loss. Thus automatic accident detection systems are the need of time, which can speed up the rescue operations and limit the casualties after the mishap and numerous lives can be saved. This paper features existing mechanisms to detect accidents, its working, and limitations. Furthermore, accident prevention methodologies, accident contributing factors are highlighted as well. This study critically reviews existing literature on accident detection and prevention techniques, with the objective that smart systems can be developed with improved accuracy and better strategies to control 3 accident causing factors while watching out for the existing challenges in the current systems.

1.3 Scope

Our project works on the principle of detection and tracking of accidents. If the vehicle is stable, no information is sent to the rescue team. Whenever an accident occurs, the vehicle changes its direction randomly and vibrates with high frequency. The Vibration Sensor and Accelerometer detect these happenings with the vehicle. The controller receives the input from the sensors and sends the accident alert information, to the rescue team and family members, along with location of the accident place using GPS and WIFI Module and finally through the messaging algorithm. It also facilitates alerting the nearest hospital and provides the necessary details to prepare for the medical assistance. limitations As our proposed methodology requires internet connectivity ,it might be difficult to detct accidents at areas with poor signal. The sensor used can be damaged easily hence care must be taken in this regard. Other than that the sensors play a vital role in detecting anamolies and the sensitivity of the sensors can also be varied, and requires low power. All data collection happens in real time.

1.4 Applicability

Applicability direct applications through the proposed methodology we can easily detect accidents due to collision,engine failure and vehicle rollover. indirect applications

Through the bolt cloud the status of the vehicle is checked constantly, if there are any problems with the engine parts they are warned by the alarming signal and the vehicle gets recovered after a quick servicing of vehicle. It is mainly applicable to all kinds of vehicles with certain modifications in the analysis.

1.5 Objectives

- Collect the data like temperature of the engine, angle of tilt of the vehicle, vibration sensor readings, Proximity sensors readings.
- Compare the data from these sensors with the predetermined threshold values and to display an alert message to the driver, if any anomaly occurs in order to prevent accidents.
- Send the data collected from all these sensors to the IOT platform using a wifi module.
- Detect an accident using the data from sensors readings and to collect the GPS location of the vehicle at the place of accident.
- Send the GPS location of the vehicle to the guardians and nearest ambulance for immediate rescue operation.

1.6 Methodology

The happening of an accident can be detected by collecting and computing all the data coming from the sensor dynamically. For the computation part, the threshold values are predefined for each data obtained from the sensor and during runtime, sensor data are compared with those threshold values and if all the values are above threshold values, then an accident is detected.
Once the accident is detected, the location of the accident site is sent to the selected gaurdians and the medical team as a message in text format with the help of IoT.

1.7 Achievements

With the project we can detect the happening of an accident and we can even prevent it, if the causes are like small system failures. For future scope, we can aid ambulance to reach the location, where accident has taken place, quickly by controlling the flow of traffic in that route. We can also use the data collected by the sensors to get better analysis behind the accident

Chapter 2

Literature survey

2.1 Related Studies

Khan et al.[4] described a system for accident detection using a three-axis accelerometer sensor to the cloud server. The framework automatically initiates the process of dispatching the nearest ambulance by processing the Global Positioning System (GPS) coordinates of the incident and providing a specific route to the accident spot in question. By using the related android application, ambulance driver can reach the spot in a rather efficient manner.

Saga et al. [4] designed a framework where soon as the accident occurs, the airbag will open and auto-lock the breaks. Then a buzzer willbe switched on. The system incorporates a GPS module, through which a microcontroller determines the coordinates of the location. Then the GSM (Global System for Mobile Communication) module sends this accident information to the victim's family for urgent attention.

Khan et al. [5] proposed a smart rescue system based on an Android application. Under this system, the user will be automatically monitored at all times. However, the user can turn the system off if the need be. This system can detect any kinds of a jerk and generate an alert, a false alert can be cancelled by the driver. If the driver does not cancel this alert within 15 seconds, then it is assumed that a substantial accident has taken place and victim's location is sent via an API to the nearest emergency responder so that emergency rescue operations can be initiated.

Ramya et al. [6] described an intelligent traffic light system (ITLS) where an ambulance can move easily without traffic light stopping it on the way. Through the GPRS (General Packet Radio Service, a packet-oriented mobile data standard on the 2G and 3G cellular communication networks) 3G modem, the traffic controller will get the location of the ambulance. If the ambulance is near to the traffic junction, then the corresponding signal will be green. Automated systems in traffic detection are also on the rise.

Fernandez et al. [3] put forward a system of accident detection that uses GPS, GPRMC and MCU through a speed monitoring algorithm. GPRMC is the most common sentence transmitted by almost all GPS devices, this sentence or line of data contains nearly everything a GPS application requires. If an accident occurs then the place of accident, speed before accident, time and date will be readily available through SMS using GPRS.

Ali and Eid [2] have considered the number of factors such as sudden alteration in acceleration, rotation and an impact force at the rear into their automatic accident detection system, the system uses fuzzy logic to finalize a decision and attained an accuracy of 98.67%. However, it only works if the accident occurs from behind.

Yee et al. [7] proposed a method which uses GPS and Accelerometer-based accident detection system. The system not only detects accidents but also provide the degree of severity of the crash. A limitation of the system is that it can only detect a service provider within 10 KM of the radius. Also, it is not clear how the system calculates many injured onboard passengers.

2.2 IoT based vehicle accident detection and protection

This proposed IOT based accident detection system helps to reduce the loss of life due to accidents and also reduces the time taken by the ambulance to reach the hospital. To detect the accident there is an accelerometer sensor present in this rescue system and the GSM module sends messages about the location to the respective guardian and rescue team. With the help of the accelerometer sensor signal, a severe accident due to an obstacle can be recognized. Microcontroller used, sends the alert message through the GSM module including the location to the guardian or a rescue team. So, the emergency help team can immediately trace the location through the GPS module, after receiving the accident location information, action can be taken immediately.

Chapter 3

Requirements and Analysis

In this Chapter, it is defined what are the hardware and software requirements and the analysis of the project has been presented in this chapter in a descriptive manner starting from problem statement to designing the model of the project.

3.1 Problem Definition

Prevention: Real time sensing and monitoring of vehicle body conditions using electronic sensors.
Detection: Identification of accident occurrence and location of the site.
Rescue: Providing Emergency assistance by calling for ambulance and chosen guardians for help.

3.2 Requirements Specification

Vibration Sensor: To measure the vibrations produced on the body of the vehicle.
Accelerometer : To measure the stability of the vehicle.
Temperature Sensor : To measure the temperature of the engine and its surroundings.
IR Sensor : To measure the distance between the bumper and the objects in front.

3.3 Project Schedule:

Table 3.1: Project Schedule

# WEEK	DATES	PROGRESS
1	19/04 to 25/04	Formation of Team and Research on the Project Topics
2	26/04 to 02/05	Submission of Project Title and Literature Survey
3	03/05 to 09/05	Research for Project Related References and Submission of Synopsis
4	10/05 to 16/05	Guide Allotment, Hardware Purchase, Software Installation and Lab Setup
5	17/05 to 23/05	Temperature Sensor and Vibration Sensor Interfacing and Data Collection
6	24/05 to 30/05	Phase One Presentation
7	31/05 to 06/06	LCD Display Interfacing and Circuit Verification
8	07/06 to 13/06	Accelerometer Interfacing and Data Collection
9	14/06 to 20/06	GPS Module Interfacing
10	21/06 to 27/06	Message Alert Generation System Setup
11	28/06 to 04/07	Ambulance and Hospital Assist System Setup
12	05/07 to 11/07	Phase Two Presentation
13	12/07 to 18/07	Report Preparation
	19/07/2021	Final Submission of Model and Report Submission

3.4 Hardware Requirement

Wifi Module ESP8266, Neo-6 GPS Module, SW-420 Vibration Sensor module, ADXL345 Accelerometer Sensor, LM35 Temperature Sensor, IR Sensors, Power Supply, Connecting Wires, Breadboard, LEDs, Buzzer, Arduino UNO.

ESP 8266 Wifi Module : The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network at an operating frequency of 2.4 GHz. The ESP8266 is capable of either hosting an application or offloading all WiFi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much WiFi-ability as a WiFi Shield offers (and that's just out of the box)! The ESP8266 module is an extremely cost effective board with a huge, and ever growing, community. This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, which is designed to occupy minimal PCB area.

GPS Module (NEO-6M): The GPS module is found in the most mobile phones and cars and becomes smaller and cheaper over time. These modules receive coordinates from satellites.A GPS navigation device, GPS receiver, or simply GPS is a device that is capable of receiving information from GPS satellites and then to calculate the device's geographical position. Using suitable software, the device may display the position on a map and also provide directions. The Global Positioning System (GPS) uses a global navigation satellite system (GNSS) made up of a network of a minimum of 24, but currently 30, satellites placed into orbit The GPS module has a Receiver with antenna which provides the location of the vehicle. The GPS system is commonly used to get information about coordinates, speed, time and distance. In this module, a GPS system is adopted to implement the in-vehicle device. The GPS module is illustrated in the *figure* 3.1

Figure 3.1: NEO-6M GPSModule

Accelerometer(ADXL345 Breakout Board) : It requires an electromechanical sensor to detect sudden and abnormal vehicle behaviors. The accelerometer measures the acceleration of the static and dynamic forces. The Accelerometer is illustrated in the*figure*3.2

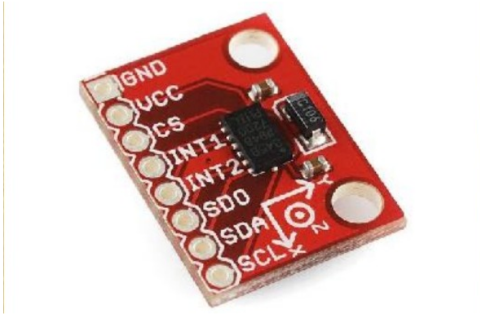


Figure 3.2: ADXL345 Accelerometer

IR sensor(LM358P driver): sensor is an electronic device that measures and detects infrared radiation in its surrounding environment.Mostly IR Sensor is used for Interfacing with Arduino,8051, AVR, PIC, ARM, Raspberry Pi and All Micro-controllers so that IR Infrared Obstacle Avoidance Sensor can also be used in Home, office,Industrial Automation IoT Research and Development etc. The IR Sensor is illustrated in the *figure*3.3

Figure 3.3: LMP358P IR Sensor

Temperature Sensor(LM35 IC) : Temperature Sensor(LM35 IC) : The LM35 is a high precision temperature sensor. It has an analog output, which is linearly proportional to the surrounding temperature. Just hook it up to a ADC pin on your microcontroller to get temperature readings. Output calibrated in ° Celsius 0.5° C accuracy at 25° C. Wide range of operation between -55° to +150° C Wide range of operating voltage ranging between 4 to 30 volts Current consumption of less than 60 uA. The Temperature sensor is illustrated in the *figure*3.4

Figure 3.4: LM35 Temperature Sensor

Vibration sensor(SW-420): Vibration sensor module alarm Motion sensor module vibration switch SW-420 module based on the vibration sensor SW-420 and Comparator LM393 to detect if there is any vibration that exceeds the threshold. The threshold can adjust using an onboard potentiometer. When this no vibration, this module outputs logic LOW the signal indicates LED light, and vice versa. If the module does not vibrate, the vibration switch is in the close state, the output of low output, the green indicator light. The product vibrates, vibration switches momentary disconnect, the output is driven high, the green light does not shine. The output can connect to the microcontroller, which to detect high and low levels; so as to detect whether the environment has vibration, plays a role in the alarm. The Vibration sensor is illustrated in the *figure*3.5

Figure 3.5: SW-420 Vibration Sensor

3.5 Software Details

BOLT IOT Cloud Platform, Bolt IoT Javascript library, Bolt Cloud API, Arduino IDE.

Preliminary Product Description:

- Prevent and Accident causing breakdown of the vehicle by raising an alert
- Provide a Quickest possible response by alerting the Rescue Team and the Nearby Hospital about the occurrence of the accident by sharing the location of occurrence
- Alert the Chosen Guardian after the occurrence of an accident and inform them about the location of occurrence of the accident

3.6 Flowcharts

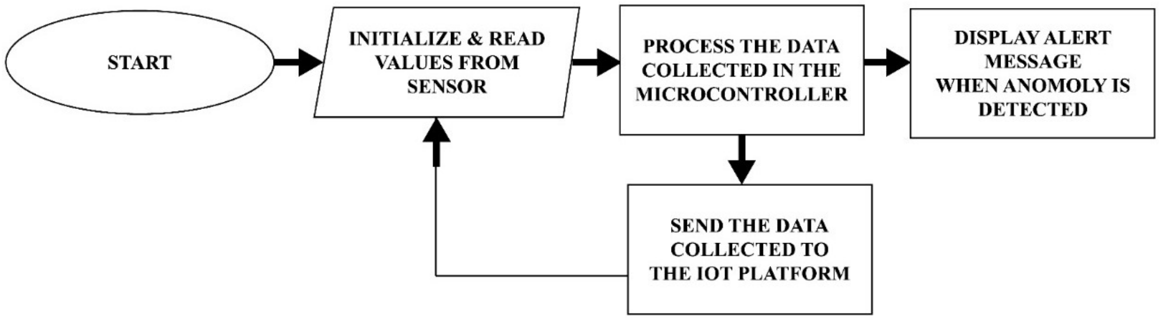


Figure 3.6: Prevention Mechanism

In the Prevention Mechanism, first the values from the sensors are initialised and read. Then this collected data is processed in the microcontroller and sent to the IoT platform and are checked for any unusuality. If everything is ok, this process is repeated. If any abnormality is found, then an alert message is displayed.

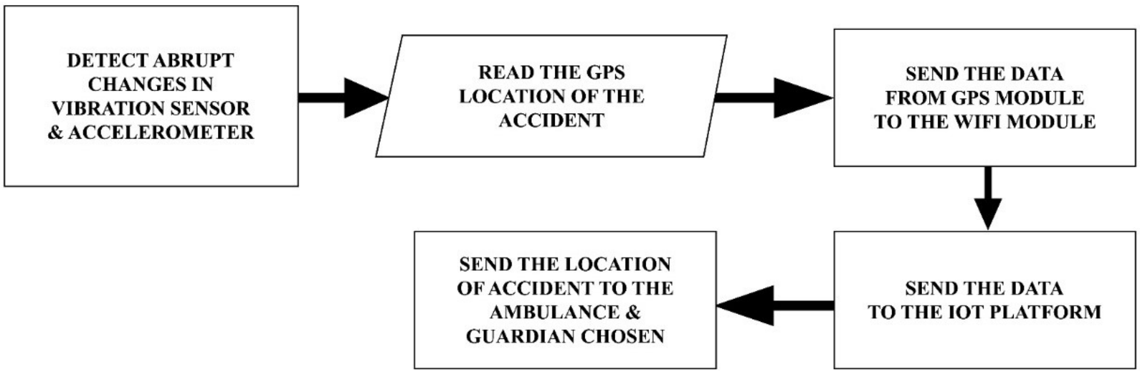


Figure 3.7: Detection Mechanism

In the Detection Mechanism, abrupt changes in the accelerometer and vibration sensor are detected when an accident happens. The GPS location of the site is read via GPS module and this information is sent to the IoT platform via WiFi module, which further sends the location to the guardian and the ambulance.

Chapter 4

System Design

This chapter describes the detailed pattern of accident detection and various design approaches for rescue operations. The scheme of design displays the reaction of sensors and actuators upon detection of accident. Procedural and Algorithmic design section trace the proceedings of the action required to be taken. The connection to the bolt iot network used for indication and alert msg has been listed in the user interface design section. The continuous data collection in real time pass through different test checking parameters mentioned in this section. Any deviation from the optimal condition will be alerted immediately to the people concerned through the bolt iot network. This chapter also includes the security problem that need to be taken under great consideration.

4.1 Basic Modules

Collection
In this phase we will collect the data from the sensors on board and upload it onto the cloud in real time. The variations or anomalies in the reading are analysed and a warning message is displayed to the Driver.
1.Vibration Sensor : To measure the vibrations produced on the body of the vehicle.
2.Accelerometer : To measure the stability of the vehicle
3.Temperature Sensor : To measure the temperature of the engine and its surroundings.
4.IR Sensor : To measure the distance between the bumper and the objects in front.

Triggering
This phase begins when any internal or external factors cause the accident to occur. This accident will trigger the sensor used for accident detection . Normally if the

vehicle is stable, no information is sent to the rescue team. Whenever an accident occurs, the vehicle changes its direction randomly and vibrates with high frequency. After this trigger, the happening of an accident detected as all sensor parameters will show a spike above the threshold value.

Tracing
After triggering, the GPS location of the accident spot is sent to the rescue team with the help of the wifi module and the messaging algorithm. The accident alert will also be sent to the important contacts automatically

4.2 Data Design

- We will collect the data from the sensors on board one sensor in real time.
- Sensor data is checked and computed in the microcontroller used and based on the results a partiular task is done.
- In the Bolt IoT Cloud Platform, the variations or anomalies in the reading are analysed and a warning message is displayed to the Driver.

4.3 Scheme of Design

Temperature check: Here initially,the temperature readings are checked whether it falls under the optimum range.In case if it does not fall under the optimum range then warning signal is sent to the driver via the bolt cloud. Normally the car engine works in temperature range of 80 degrees to 100 degrees celsius.
IR Sensor: Here ,the obstacle causing accident is detected, IR light is invisible to us as its wavelength (700nm – 1mm) is much higher than the visible light range. IR LEDs have light emitting angle of approx. 20-60 degree and range of approx. few centimeters to several feets,Led starts blinking after the obstacle causing accident is detected.
Vibration check: This module features an adjustable potentiometer, a vibration sensor, and a LM393 comparator chip to give an adjustable digital output based on the amount of vibration. The potentiometer can be adjusted to both increase and decrease the sensitivity to the desired amount. The module outputs a logic level high (VCC) when it is triggered and a low (GND) when it isn't. Additionally there is an onboard LED that turns on when the module is triggered. The vehicle after the accident emits vibration of high frequency which detects the vibration and the information is sent to

rescue team.
GPS The accident location is tracked and is sent to the bolt iot website,and the rescue team after getting the location ,arrives at the accident location and rescues the driver.
Accelerometer: Here the x,y,z component of the vehicle is checked,if all the components are unity then there is no accident happened,if there are variations in the x, y,z components then the position of the vehicle is abnormal and we can know the accident has taken place.The data is read in a digital format which gets transferred to the controller and can be viewed in the Bolt cloud website.

Bolt Iot Module: Interface the esp8266 bolt wifi module and read the values from Arduino UNO and store the data in cloud.

4.4 Procedural design

- Real time collection of data through various sensors.
- Conditional checking of data with the standard threshold values.
- Anomaly Detection.
- Executing the main task (Retrieving GPS location of the accident site).

First step includes the collection of the data through various sensors where the sensors ae exposed to the environment or surroundings. It takes the raw data and sends it to the microcontroller. The second step includes the conditional checking of the raw data with the predefined threshold values in accepted system units. The third step includes detection of anomaly, where if the sensor data goes above the threshold values. The condition becomes true and the anomaly is detected. Fourth step includes excetion of main task which includes sending location of the accident to the guardians and the medical team.

4.5 Logic Diagrams

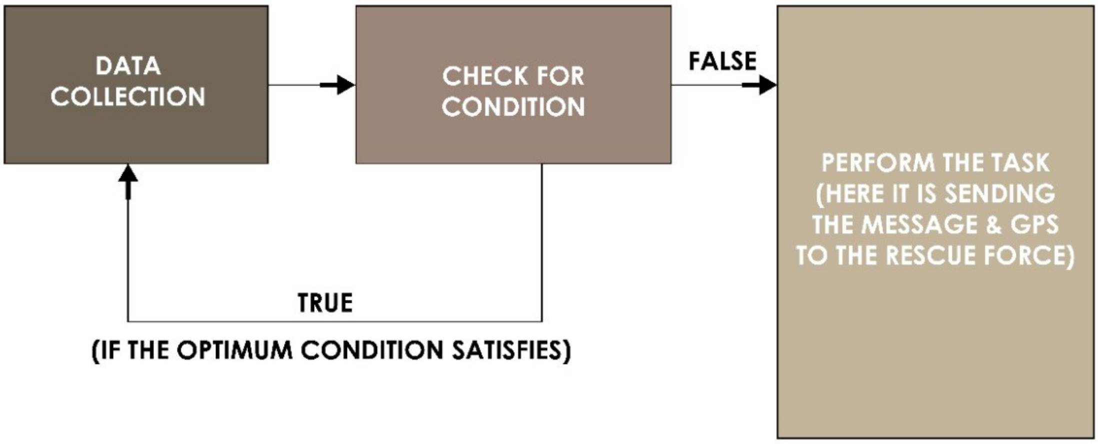


Figure 4.1: Process Diagram

The process includes collection of data through various sensors and this data is sent to the IoT platform via WiFi module. Then the conditions are checked, if they satisfy the conditions, the process is repeated again. If they don't satisfy the conditions, task is performed which includes sending message to the guardian and ambulance.

4.6 Algorithmic design

Prevention system

- The sensor values are initialized and are read
- The data collected from micro controller is processed
- Display alert message when anamoly is detected
- The microcontroller data is constantly monitored and alerts the user.

Detection system

- Detect changes in Accelerometer and Vibration sensor
- The gps location of accident is traced using gps module
- GPS location generated is sent to the Wifi module

- The sent data is observed in IoT platform
- The accident location is sent to the ambulance and guardians chosen

4.7 User Interface Design

Internal components
The sensors namely Temperature sensor, vibration sensor and ir sensor input pins are given to Arduino, Once the Microcontroller is powered then the data from these sensor is collected in the Microcontroller performing necessary tasks.
External components
In this case everything is automatic,the only thing the user must make sure is to keep the hotspot "ON" for the internet access of Bolt Wifi module.

Figure 4.2: Connecting your Bolt WiFi module to Bolt IoT App

- Step 1: Take out your WiFi Module.
- Step 2: Power it on.Connect a 5V 1A micro USB cable. These are often used to charge an Android phone.
- Step 3: Check your WiFi network frequency
- If you have created an account before on the mobile app or via the website cloud.boltiot.com, use the same credentials to log in.

- Step 4: Login into the Bolt Cloud
- If you have created an account before on the mobile app or via the website cloud.boltiot.com, use the same credentials to log in.
- If you have not created an account before, then open the Bolt App and click on ’REGISTER to create a new account.
- Step 5: Connecting Bolt WiFi Module to the Internet
- The Bolt Mobile App will help you connect your WiFi module to the Internet as well as add it to your Bolt cloud account.
- To add a device to your cloud account, click on the ’ADD DEVICE’ button.
- If you have not yet done, power ON the Bolt device by inserting the Micro USB cable into the USB port provided on the Bolt and connecting the other end of the cable to the Android charger or your laptop.
- Once you have powered ON the Bolt device, the blue LED on the Bolt will begin to blink slowly, and the green LED will be OFF.

4.8 Security problems

Security problems are : The board has to be kept very safe as it could be subjected to physical damage.

Advantages of spi interface are: It has complete flexibility for the bits transferred i.e. not limited to 8 bit word. It has very simple hardware interfacing. Not limited to any maximum clock speed, enabling potentially high speed. It’s faster than asynchronous serial. It also has some disadvantages It requires more pin on IC packages as compare to I2C. It does not support hot swapping (dynamically adding nodes). It supports only one master device.

4.9 Test Cases

Table 4.1: Test Cases for Temperature, IR and Vibration Sensor

Sensor	Parameter	i/p Values	Threshold Check	Vehicle Status
Temperature Sensor	Temperature in °C	80°C 90°C 100°C 110°C	80<100 90<100 100=100 110>100	safe Safe Overheating Temperature Overheating Temperature
IR Sensor	Obstacle distance in mm	10mm 20mm 3mm	10>7 20>7 3<7	Safe Safe Warning Signal
Vibration Sensor	Duration	100s 150s 200s 250s	100<200 150<200 200=200 250>200	Saf Safe Warning Signal Warning Signal

Table 4.2: Test cases for Accelerometer

Sensor	Roll	Pitch	Condition Check	Vehicle Status
Accelerometer	0.3	0.08	Roll, Pitch < |4|	Stable
	0.34	1.09	Roll, Pitch < |4|	Stable
	1.86	.35	Roll, Pitch < |4|	Stable
	7.23	1.4	Roll>4, Pitch <|4|	Unstable

Chapter 5

Implementation and Testing

This chapter includes implementation and the standards that are being used for the implementations. Testing approach, Code details and Code efficiency are explained in a very easy way.The project is tested unit wise and integrated all these units to firm the actual working model. We have described how we started from simulation to performing the main task.Some modifications and improvements are included to optimize the code.

5.1 Approaches

In our project we have taken the data from sensor to know the safety status of that vehicle and depeding on that microcontroller performs certain tasks based on the sensor output for prevention as well as detection.For loading the data to bolt cloud, we will interface the pins of wifi module to the microcontroller , In the wifi module, Uart protocol is used,in which data format and transmission speeds are configurable,The features of uart are as follows.

- A clock generator, usually a multiple of the bit rate to allow sampling in the middle of a bit period.
- Input and output shift registers.
- Transmit/receive control
- Read/write control logic.
- Autobaud measurement.

- Transmit/receive buffers.

5.2 Interfacing Bolt With Arduino : Bolt UART

The Bolt has a single UART interface and supports the following baud rates for serial communication.

1. 2400
2. 4800
3. 9600
4. 19200

To initiate the UART with required Baud rate you will need to call the following URL from your preferred Browser. Where ip-address-of-bolt is the IP of the Bolt Unit on your local network and baud=n where ″n″ takes a value from 1 to 4 and it represents the respective baud rates as seen above.
For example if IP address of Bolt is 192.168.1.5 and you need to set 9600 baud rate the URL becomes:
http://192.168.1.5/serialBegin?baud=2 We will be using the same baud rate for this instructable

- Step 1: Send Values Over TX (Transmit Line) of Bolt UART
- Once you have initialized the Baud rate on Bolt , you can now begin to send and receive date over the Bolt UART.
  Serial Write Function (Transmit data)
  To transmit data over the Serial port of Bolt you need to type the call the following URL from your browser:
  http://ip-address-of-bolt/serialWrite?data=string-to-transmit For example if we wish to transmit the String ″hello″:
  http://192.168.1.5/serialWrite?data=hello This command if executed correctly return a JSON object with key ″success″ set to ″1″ and key ″value″ set to ″serial write successful″.

- Step 2: Receiving Data Over Rx (Receive Pin)This step will allow you to read contents of the Receive buffer of the Bolt UART Serial Read Function (Receive data)
  To receive data over the Serial port of Bolt you need to type the call the following URL from your browser:
  http://ip-address-of-bolt/serialRead Example:
  http://ip-address-of-bolt/serialRead

- Step 3:Flush Receive Buffer To empty the contents of the Rx buffer of the Bolt UART you need to call the following URL:
  http://ip-address-of-bolt/flushReceiver Example:
  http://192.168.1.5/flushReceiver This will return a JSON object with ″success″ key set to ″1″ and ″value″ set to ″Receive Buffer Flushed″
  This command is particularly useful when you feel that the buffer may contain some extra or garbage data that needs to be cleared before acquiring relative data.

- Step 4: Serial Read Command Response:An added functionality to Bolt is its ability to send a command to a system over the UART and if the response is received within 200ms it can fetch the same using a single command.
  Serial Write and Read Function (serialWR)
  To read response of a command you need to call the following URL from your Browser:
  http://ip-address-of-bolt/serialWR?data=command and till=decimalcode-for-ascii-char Example:
  http://192.168.1.5/serialWR?data=hello and till=10 The reply for this command will be same as that of Step 2, only difference being that if the command response is empty then the key ″value″ will be an empty string =¿ ″″ .

- Step 5: Interfacing Bolt With Arduino In this step we will make the necessary hardware connections to setup serial communication between Arduino Deumilanove and Bolt.
  This is pretty straight forward and you will need three maleto-male jumper cables for this step

  1. Connect Tx pin of Bolt to Rx pin of Arduino.
  2. Connect Rx pin of Bolt to Tx pin of Arduino.
  3. Connect Gnd(ground) pin of Bolt to Gnd(ground) pin of Arduino.

Next you can upload the following code on Arduno to test Serial communications.

5.3 Code details

In the code we have separately defined sensor functions and the function is called when the data starts to read from each sensors and perform tasks when any anamoly is detected. We have used arduino ide to write our code,Also we have interfaced the arduino with the bolt cloud. For that we gave sensor output pins to the bolt cloud, so that whatever the data generated by the microcontroller is viewed in the bolt cloud.When the Accident is detected, the gps of the location is detected and ambulance/rescue team will get the information.

5.4 Algorithm

1. Define API key, Device ID, static constant variables and set the baud rate to 9600.
2. Create Tiny GPS Object and software serial port.
3. Declare all the functions for each sensor’s functionality.
4. Set the transmission line and initiate serial communication for printing the results on the serial monitor.
5. Initiate or setup Bolt IoT message transmission.
6. Initialize accelerometer sensor and setup other sensor with their respective I/O pin.
7. Define each function.

For Accelerometer

8. Set ADXL345 measuring mode.
9. Start communication with device and enable measurement.
10. Calibrate Offset using function from wire library.
11. Similarly calculate measurement for X, Y and Z axis.
12. Read Accelerometer data.
13. Read six registers.
14. Calculate X and Y values (original values) from raw data.

15. Calculate roll and pitch.

IR Sensor

16. Read the data.
17. Check if the data is within the desired range, if not then activate the buzzer.

Temperature Sensor

18. Read the input and convert it in to suitable format.
19. Check the condition, if it is within the desired range.

Vibration Sensor

20. Read the Vibration i/p and Vibration time.
21. For Vibration time check when the input pulse is high.

GPS Module

22. Read the latitude and longitude of the location with the help of tiny GPS module such as gps.location.lat().
23. Call all the functions infinite times.
24. Print the sensor values.
25. If all condition are satisfied (values are greater than threshold), then pass the current location to the saved contact using bolt IOT.

5.5 Code Efficiency

We managed to keep our code small and precise and defined the function of each sensor in the code,Also there is faster software runtime execution because of the used procedure. In the used code, the sensitivity of the sensor can be varied directly by making variation in the code which makes the code highly optimisable.Due to the used procedure our code can be reusable which improves the code efficiency.

5.6 Testing Approach

We have used various steps to check the functionality of the code by simulating and testing the code in proteus software,where the code is compiled in Arduino IDE software and its hex file is transferred to the virtual arduino board in the proteus where the circuit is built and tested before its implementation .The output of the circuit can viewed via the serial monitor.For the physical implementation we make use of bolt cloud ,where all the data from microcontroller is sent and inorder to warn the driver there is a buzzer and an led during overheated engine situation and when the foriegn objects comes in contact with the vehicle.

5.7 Unit Testing

By observing the figure we come to know how the sensors gives out the output,starting from temperature sensor, where we check for overheating in the vehicle,where the threshold temperature is set and gives results accordingly for gps module, longitude and latitude of the location is displayed.for ir sensor, a buzzer signal will show the warning sign,when there is an obstacle in front of the vehicle,In vibration sensor, the monitor shows whether the vibration of high frequency is detected or not. finally for accelerometer ,the roll and pitch of the vehicle is constantly monitored and also detects the accident due to vehicle rollover.

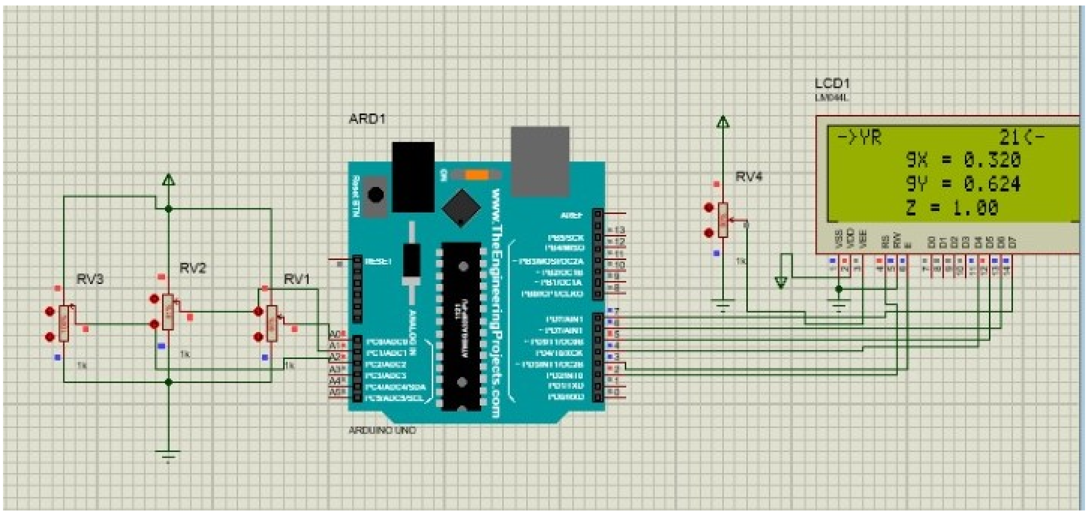


Figure 5.1: Accelerometer Simulation

In the simulation of the accelerometer sensor, we use a potentiometer to change the x, y, and z-directiona which is displayed in a 16*2 LCD. These potentiometers of resistance 1k ohms provide variable resistance as their position are changed. The output pins are connected to A0, A1, A2 pins respectively to the Arduino UNO board which can be seen in the *figure* 5.1

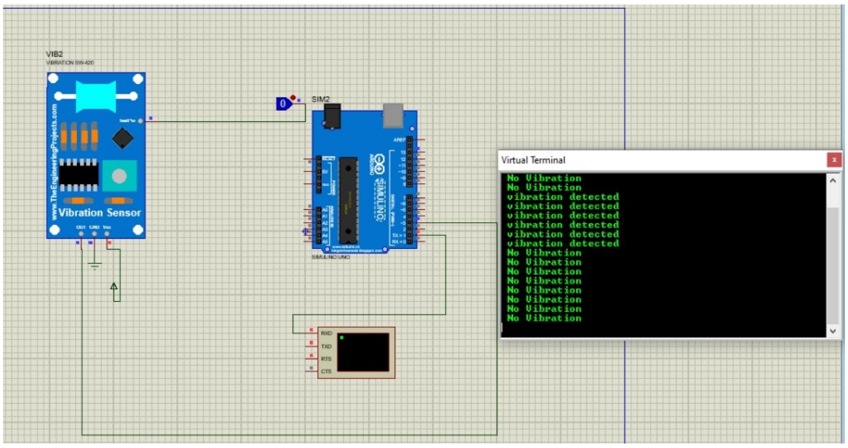


Figure 5.2: Vibration Sensor Simulation

In the simulation of vibration sensor, a logic state block is used to control the state of the vibration sensor, if the logic state goes high, which means that the vibration is detected and vice versa. The status of the vibration sensor is displayed in the virtual terminal as shown in the *figure*5.2

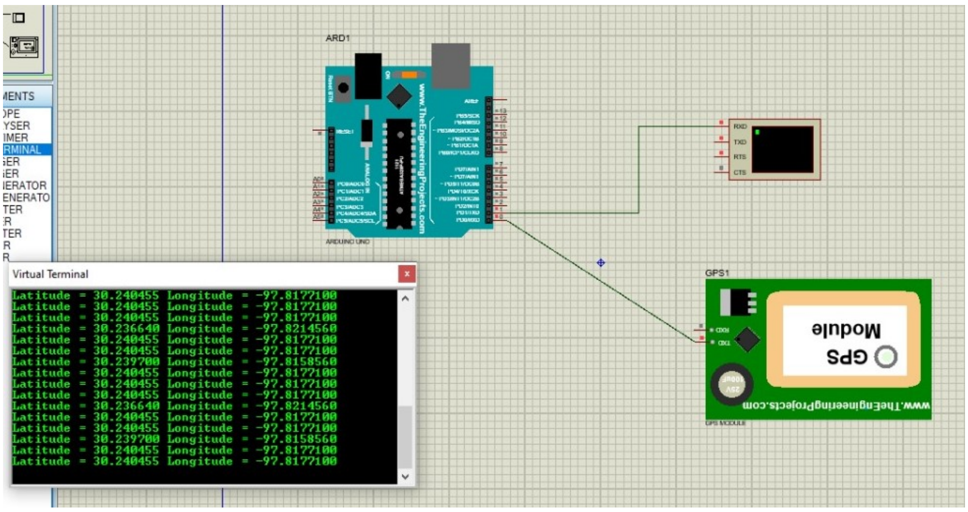


Figure 5.3: GPS module Simulation

In the simulation of GPS sensor, we connect the transmission pin TXD of the GPS module to the receiving pin of the Arduino UNO and observe the location through a virtual terminal. This is demonstrated in the *figure*5.3

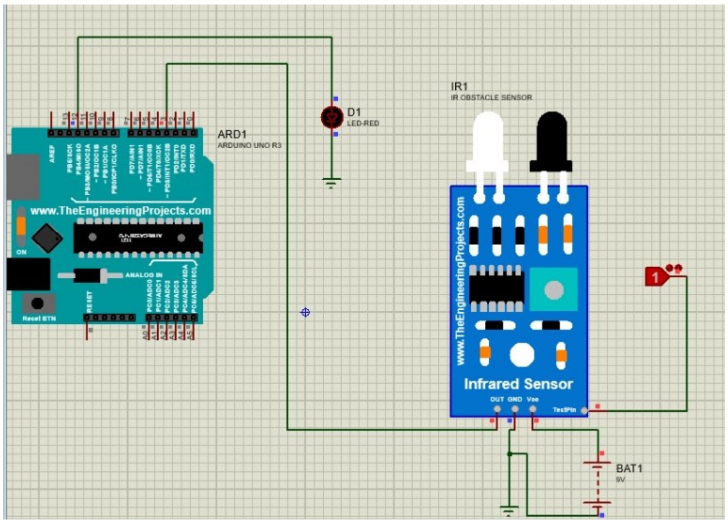


Figure 5.4: IR Sensor Simulation

In the simulation of IR sensor, a logic state block controls the state of the IR sensor, if the logic state goes high, which means that there is an obstacle and an led starts to glow as a sign of warning.If the logic state goes low, which means that there is no obstacle and an led does not glow. as shown in the *figure*5.4

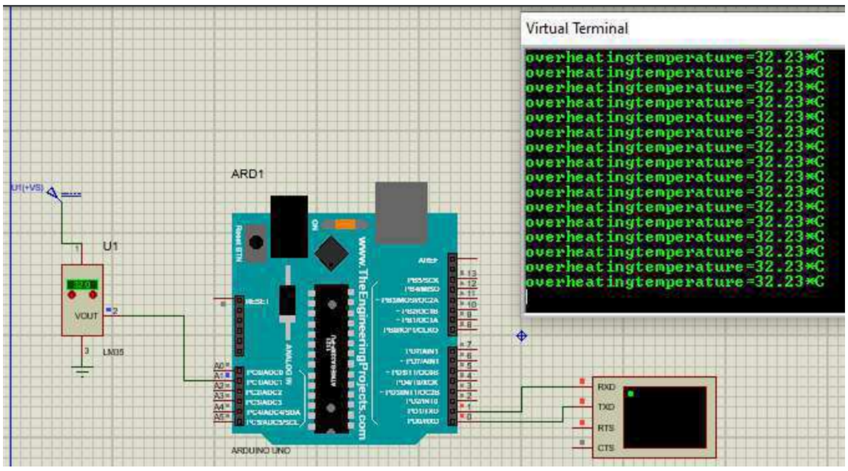


Figure 5.5: Temparature Sensor Simulation

In the simulation of temperature sensor, the output pin of lm35, connected to the A1 pin of Arduino UNO, gives temperature as an input to the microcontroller. It displays whether the temperature is above or below the threshold temperature as shown in the *figure*5.5

5.8 Integrated Module Testing

Here all the sensors are integrated and connected to the microcontroller, here all the sensors must work independent of each other.For the integrated cicuit we have given input to different pins of the microcontroller and in the code we have given the order to display the data from the sensors.

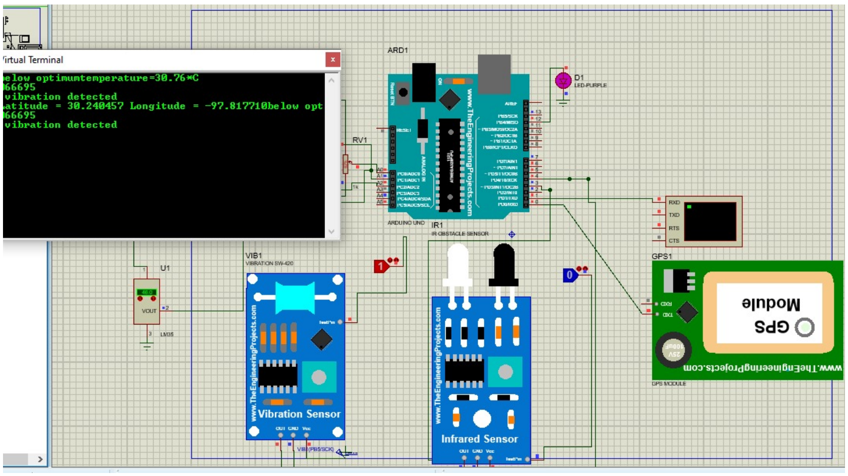


Figure 5.6: Integrated Module Simulation

Chapter 6

Results and Discussion

6.1 Results

Data collected from embedded sensors

1. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.03 pitchF = 0.08 Temperature Sensor : Below Optimum Temperature = 56.64 *C

2. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.06 pitchF = 0.17 Temperature Sensor : Below Optimum Temperature = 57.62 *C

3. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.10 pitchF = 0.25 Temperature Sensor : Below Optimum Temperature = 56.64 *C

4. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.12 pitchF = 0.33 Temperature Sensor : Below Optimum Temperature = 60.55 *C

5. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.15 pitchF = 0.41 Temperature Sensor : Below Optimum Temperature = 57.62 *C

6. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.17 pitchF = 0.48 Temperature Sensor : Below Optimum Temperature = 56.64 *C

7. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.19 pitchF = 0.55 Temperature Sensor : Below Optimum Temperature = 61.52 *C

8. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.22 pitchF = 0.62 Temperature Sensor : Below Optimum Temperature = 57.62 *C

9. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.24 pitchF = 0.68 Temperature Sensor : Below Optimum Temperature = 56.64 *C

10. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.26 pitchF = 0.73 Temperature Sensor : Below Optimum Temperature = 54.69 *C

11. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.29 pitchF = 0.78 Temperature Sensor : Below Optimum Temperature = 61.52 *C

12. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.31 pitchF = 0.83 Temperature Sensor : Below Optimum Temperature = 58.59 *C

13. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.33 pitchF = 0.87 Temperature Sensor : Below Optimum Temperature = 56.64 *C

14. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.34 pitchF = 0.92 Temperature Sensor : Below Optimum Temperature = 58.59 *C

15. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.32 pitchF = 0.96 Temperature Sensor : Below Optimum Temperature = 58.59 *C

16. Vibration Sensor : Below Optimum Vibration =71 Accelerometer Sensor : rollF = 0.34 pitchF = 1.00 Temperature Sensor : Below Optimum Temperature = 56.64 *C

17. Vibration Sensor : Below Optimum Vibration =0 Accelerometer Sensor : rollF = 0.35 pitchF = 1.04 Temperature Sensor : Below Optimum Temperature = 57.62 *C

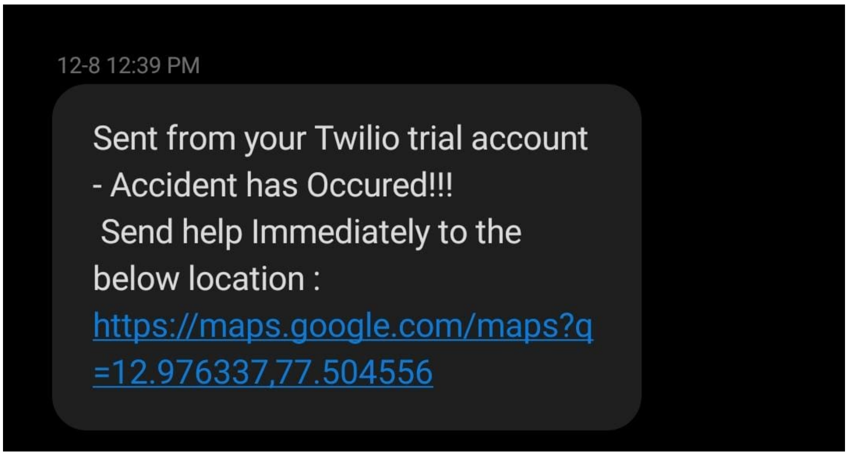


Figure 6.1: Alert SMS

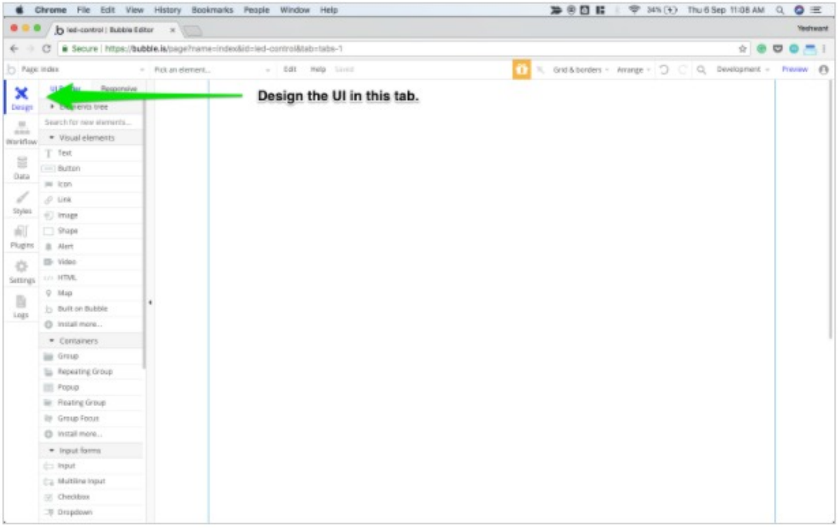
The triggering of the alerting system will generate the following message and sends as SMS to the chosen guardian and the nearest ambulance as shown in the *figure* 6.1

6.2 User Documentation

Arduino IDE: The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

Writing Sketches: Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

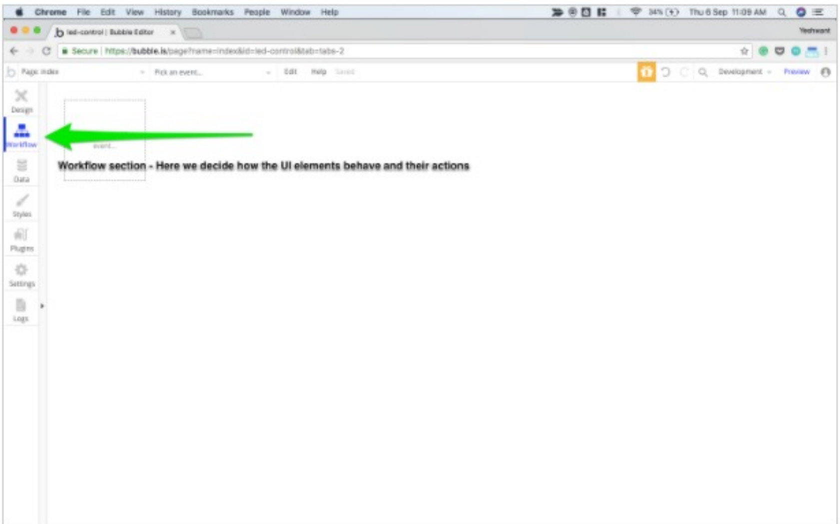
Bolt Iot
Designing the User Interface: We shall start with a simple button and use it to get the status of the device and check whether it is online or offline. Let us begin with an overview of how to use the Bubble editor.
Design section: In this section, we create UI elements like buttons, shapes, slide bars etc. The Design tab is shown in the *figure* 6.2



Design tab

Figure 6.2: Design tab

Workflow section: In this section, we decide how our UI elements should behave and what actions they should perform. The Bolt plugin provides actions related to the device which can be accessed in this section. The Workflow tab is shown in the *figure* 6.3

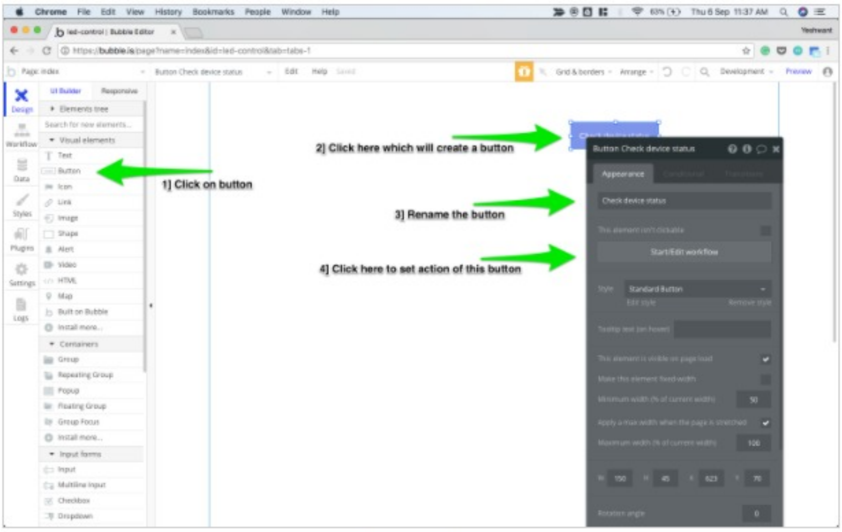


Workflow tab

Figure 6.3: Workflow tab

Creating a button and getting the device status

We shall now create a button in the design section and set its action to get device status

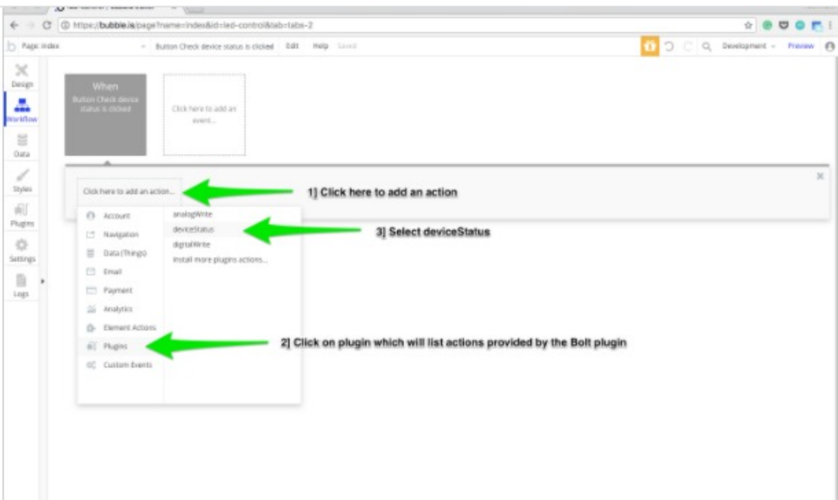


Creating the button

Figure 6.4: Creating the button

in the workflow section.

- Go to the Design section, and click on ”Button”. This activates the pointer to create a button in the design area as shown in the *figure*6.4
- Click top right part of the design area which would create a button.
- Name the button. Preferably ”Check device status” since it will show device status.
- Click on start/edit workflow to set an action for this button.This will take you to the workflow section.

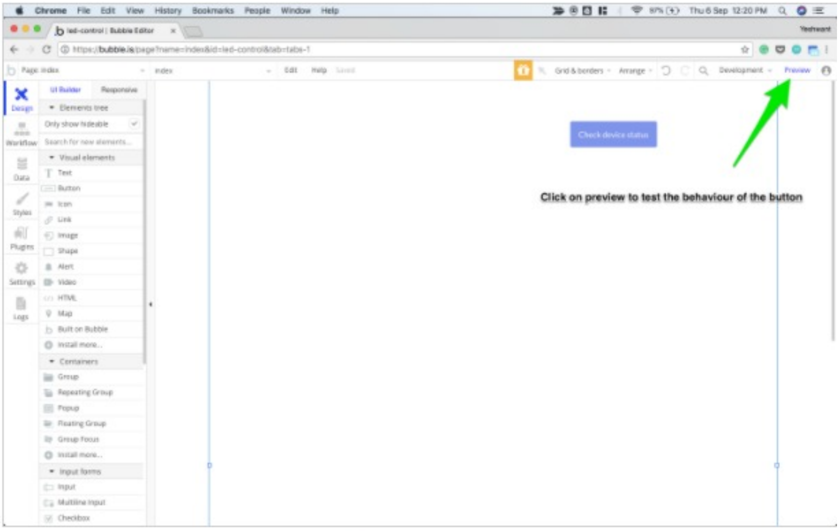


Adding an action

Figure 6.5: Adding an action

Once you are in the workflow section, set an action for this button as shown in the *figure* 6.5.

- Click on add an action.
- Select Plugins.
- Choose device Status.



Reviewing the app

Figure 6.6: Reviewing the app

Go back to the Design section and select preview from the top right corner as shown in the *figure* 6.6. This will open a preview of our app and we can test the behaviour of the button that we just created.

Click the button once you are in the preview mode. It will show you the status of your device as shown in the *figure* 6.7.

Response of the button

Figure 6.7: Response of the button

Chapter 7

Conclusion

The proposed methodology has the following advantages over the existing models. The model is more cost efficient than the existing systems. As the number of sensors are more than existing systems, it's more accurate in detecting an abnormality. Monitors all hazards and threats. Our safety system is cost efficient compared to the already existing safety mechanism which exists in a vehicle. Large Data collection helps to better analysis of the accident.

Also there are some limitations in the proposed moethodology. They are as follows, the sensors and microcontroller used can be damaged very. So utmost acre must be taken. The wifi module must be connected to the internet to stay connected to the cloud. Due to the delay in collection of data, therecan a slight delay in detecting an accident. System doesn't have quick responsiveness.

7.1 Future Scope

We can collet data of sensors from various vehicles caught in accident in a particular area and use ML and AL to those data. As a result we will observe a common pattern which lead to most obvious reasons of an accident. We can control the traffic according to ambulance movement as it will help ambulance to reach faster to the destination, here also we can use IoT or GSM technology. Allow the driver to call for an ambulance using voice assistance services like Google Assistant or Alexa. Data Visualization for better analysis.

References

[1] Overview of Road Accidents in India.

[2] Asad Ali and Mohamad Eid. An automated system for Accident Detection. pages 1608–1612, 05 2015.

[3] George Fernandez Savari, R. Palanisamy, and Vijayakumar Krishnasamy. GPS and GSM based accident detection and auto intimation. *Indonesian Journal of Electrical Engineering and Computer Science*, 11:356–361, 07 2018.

[4] Pranto Karmokar, Saikot Bairagi, Anuprova Mondal, Fernaz Narin Nur, Nazmun Nessa Moon, Asif Karim, and Kheng Cher Yeo. A Novel IoT based Accident Detection and Rescue System. In *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pages 322–327, 2020.

[5] Arsalan Khan, Farzana Bibi, Muhammad Dilshad, Salman Ahmed, Zia Ullah, and Haider Ali. Accident Detection and Smart Rescue System using Android Smartphone with Real-Time Location Tracking. *International Journal of Advanced Computer Science and Applications*, 9(6), 2018.

[6] K. Sangeetha, P. Archana, M. Ramya, and P. Ramya. Automatic Ambulance Rescue With Intelligent Traffic Light System. *IOSR Journal of Engineering*, 4:53–57, 2014.

[7] Tey Yee and Phooi Yee Lau. Mobile vehicle crash detection system. pages 1–4, 01 2018.