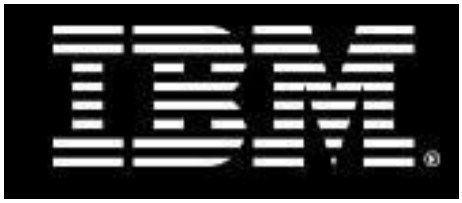


SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

IBM PROJECT REPORT

TEAM ID – PNT2022TMID48694

SUBMITTED BY

PORKODI S(**920819106043**)
AMIZHTHAA B(**920819106004**)
ASMA ROSHAN T(**920819106007**)
DIVYA DHARSHINI S(9**20819106017**)

FINAL DELIVERABLES REPORT

Date	20.11.2022
Team ID	PNT2022TMID48694
Project Name	Signs with Smart Connectivity for Better RoadSafety

TEAM MEMBER AND THEIR CONTRIBUTIONS

NAME	ROLL NO	CONTRIBUTION
Asma Roshan T	920819106007	Created source code for the wokwi simulator and MIT app code.
Divya Dharshini S	920819106017	Created node red and IOT Watson platform.
Amizhthaa B	920819106004	Project report making process and gathering ideas for creating project.
Porkodi S	920819106043	Working in node red flow and IBM cloud deployment.

CONTEXT

1.INTRODUCTION

PROJECT OVERVIEW

- To replace the static signboards, smart connected signboards are used. These smart connected sign boards get the speed limitations from a web app using weather API and update automatically.

- Based on the weather changes the speed may increase or decrease. Based on the traffic and fatal situations the diversion signs aredisplayed.

- Guide (Schools), Warning and Service (Hospitals, Restaurants) signs are also displayed accordingly. Different modes of operations can be selected with the help ofbuttons.

PURPOSE

- Smart Traffic Management is a system to monitor and control traffic signals using sensors to regulate the flow of traffic and to avoid congestion for a smooth flow of traffic.

- Prioritizing traffic like ambulances, police etc. is also one application comes under smart traffic management.

2.LITERATURE SURVEY
EXISTING PROBLEM

- Andrzej Czyżewski in his paper titled "Development of Intelligent Road Signs with V2X Interface for Adaptive Traffic Controlling", IEEE 2019, developed IOT based intelligent road

signs capable of interacting with both the vehicles and other neighbouring sign boards using LORA. These sign boards were capable of communicating with one another and changing the speed limit based on traffic and weather.

- Muhammed O. Sayin, Chung-Wei Lin, Eunsuk Kang, Shinichi Shiraishi & Tamer Basar in their paper titled "Reliable Smart Road Signs", IEEE 2019, proposed a game theoretical adversarial intervention detection mechanism for reliable smart road signs. A future trend in intelligent transportation systems is “smart road signs” that incorporate smart codes (e.g., visible at infrared) on their surface to provide more detailed information to smart vehicles. Such smart codes make road sign classification problem aligned with communication settings more than conventional classification .

- L.F.P. Oliveira, L.T. Manera, P.D.G. Luz in their paper titled "Smart Traffic Light Controller System", IEEE 2019, developed smart traffic lights capable of traffic accident detection enabling the enhancement of traffic light management systems, blocking and creating alternative routes to not only avoid the traffic jams, but also avoid new accidents.

- Dariusz Grabowski & Andrzej Czyzewski in their paper titled "System for monitoring road slippery based on CCTV cameras and convolutional neural networks", Springer Publications 2020, made use of Convolutional Neural Networks to identify slippery roads using CCTV cameras.

PROBLEM STATEMENT DEFINITION

- The road signs and speed limit are static. So sometimes when there is extreme weather conditions it is very hard for

the riders to see the speed limit and directions.

- This project can be very useful for the riders. That is we can change the road signs accordingly if they are digitalized.

- When there is a rainfall the roads get very slippery which may lead to the accidents. So that can be prevented by using technology .

3.IDEATION AND PROPOSED SOLUTION

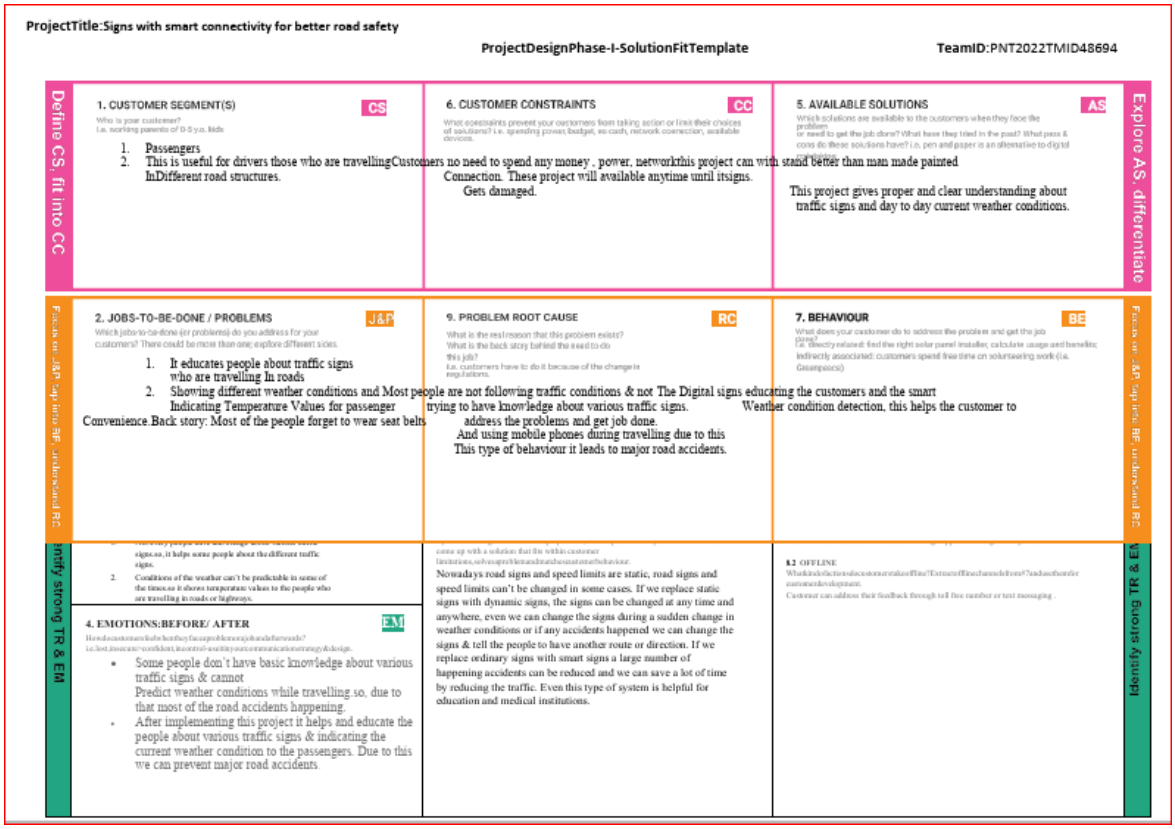
EMPATHY MAP CANVAS

PROPOSED SOLUTION

S.No.	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	Project - Signs with Smart Connectivity for Better Road Safety is used to educate the drivers digitally using IOT who do not have knowledge about traffic signs and weather indication for the drivers and passengers convenience.
2.	Idea / Solution description	Replacing the man made painted signs into digital as well as their name which is more visible compared to current signs and also indicating weather in the same sign boards for driver where weather is not predictable.
3.	Novelty / Uniqueness	Weather indication on sign boards is unique which will help mostly the two wheelers from unfortunate heavy rains and winds. Digital traffic signs also educates the drivers to follow traffic rules easily.
4.	Social Impact / Customer Satisfaction	It makes the people to know about traffic signs if they don't know ,it shows signs digitally to avoid the accidents and weather indication based on IOT to avoid accidents and it helps mostly for two wheeler passengers .

5.	Business Model (Revenue Model)	This project can make revenue by selling many equipments to the government sector and also private sectors(educational &medical institutions).Maintain services are also taken by the company.
6.	Scalability of the Solution	It makes the daily life of drivers and passengers better. The product can be scalable by adding new features to the product makes more revenue.

PROBLEM SOLUTION FIT



4.REQUIREMENT ANALYSIS
FUNCTIONAL REQUIREMENT

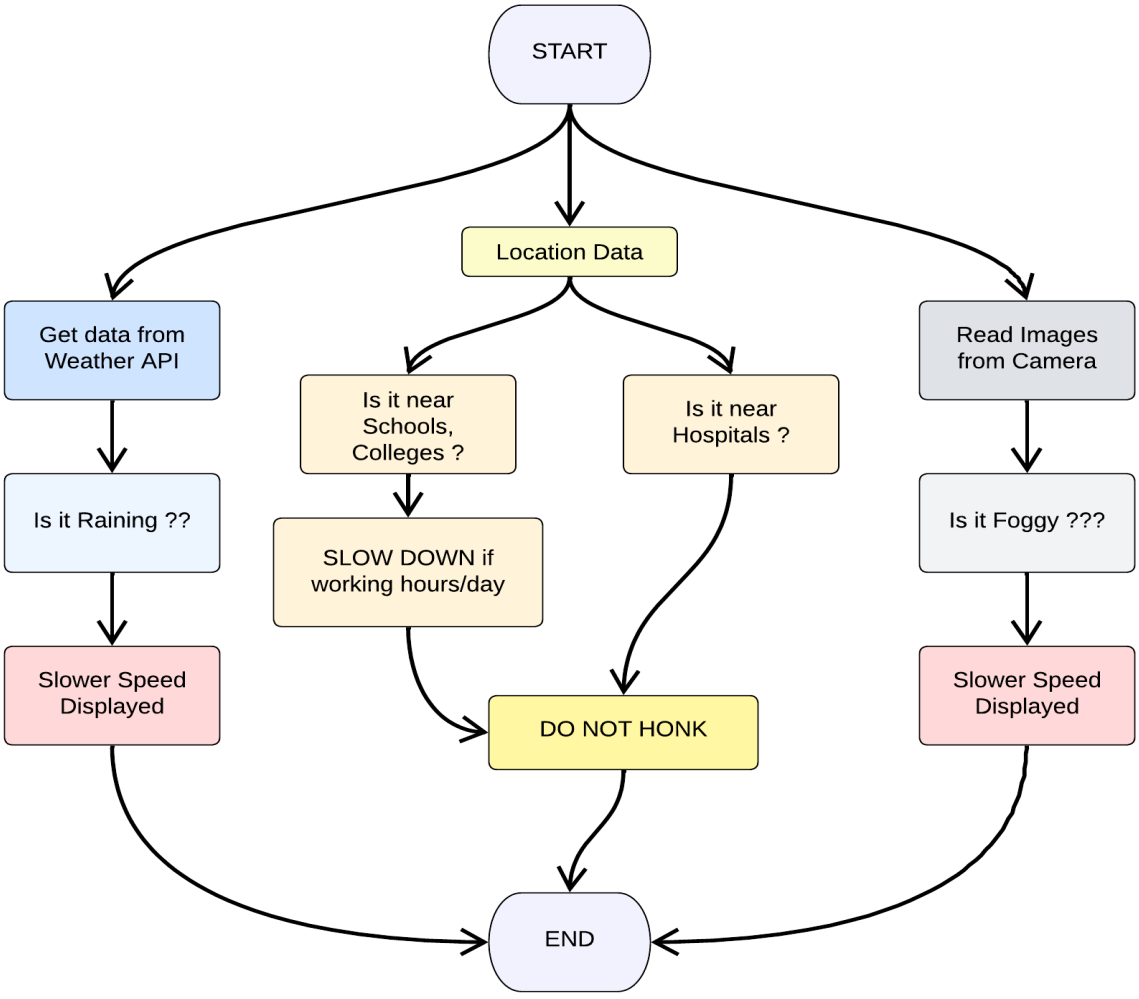
FN.NO	FUNCTIONAL REQUIREMENT	SUB REQUIREMENT
FR1	User Visibility	Sign Boards should be made of bright colored LEDs capable of attracting driver's attention Not too distracting to cause accidents.
FR2	User Understanding	Should display information through means like images/illustrations with text so that the user can understand the signs correctly.
FR2	User Convenience	Display should be big enough to display all the signs correctly so that it is visible even to far away drivers.

NON FUNCTIONAL REQUIREMENT

NFR .No.	NON FUNCTIONAL REQUIREMENT	DESCRIPTION
NFR 1	Security	Should be secure enough that only the

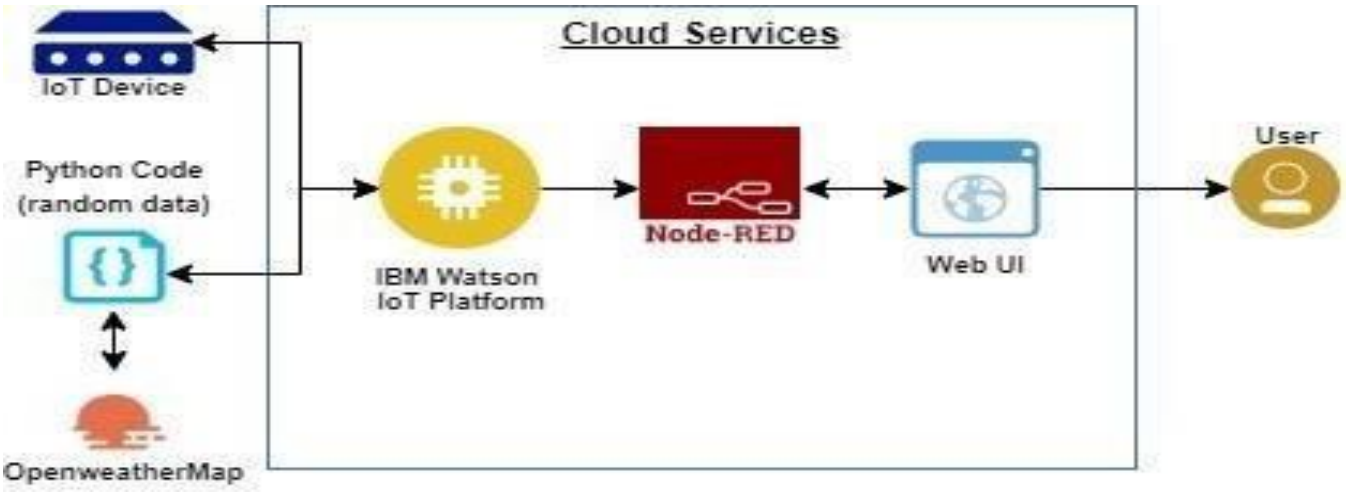
		intended messages are displayed in the display.
NFR 2	Reliability	Should convey the traffic information correctly.
NFR3	Performance	Display should update dynamically whenever the weather or traffic values are updated
NFR 4	Availability	Should be on service 24/7
NFR 5	Scalability	Should be modular and hence able to scale on servers horizontally.
NFR 6	Usability	Should be able to dynamically update withrespect to time.

5.PROJECT DESIGN
DESIGN FLOW DIAGRAM



SOLUTION & TECHNICAL ARCHITECTURE

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2.



- To replace the static signboards, smart connected sign boards are used.
- These smart connected sign boards get the speed limitations from a web app usingweather API and update automatically.
- Based on the weather changes the speed may increase or decrease.
- Based on the traffic and fatal situations the diversion signs are displayed.
- Guide(Schools), Warning and Service(Hospitals, Restaurant) signs are alsodisplayed accordingly.
- **6.** Different modes of operations can be selected with the help of buttons.

TABLE-1: COMPONENTS & TECHNOLOGIES

SI.No	COMPONENTS	DESCRIPTION	TECHNOLOGY
1.	User Interface	How user interacts with application e.g.	HTML, CSS, JavaScript / Angular Js / React

		Web UI, Mobile App, Chatbox etc.	Js etc.
2.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
3.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
4.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
5.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.

TABLE-2: APPLICATION CHARACTERISTICS

SI.NO	CHARACTERISTICS	DESCRIPTION	TECCHNOLOGY
1.	Security Implementations	Strong security system that anyone without login credentials and hackers are not allowed to enter the network.	Firewall,Firebase,cyber resiliency strategy.
2.	Scalable Architecture	Easy to expand the operating range by increasing the	IOT, internet.

		bandwidth of the network.	
3.	Availability	Available anytime and everywhere 24/7 as long as the user is signed into the network.	IBM Cloud
4.	Performance	Supports a large number of users to access the technology simultaneously.	IBM cloud

USER STORIES

6.PROJECT PLANNING & SCHEDULING

SPRINT PLANNING & ESTIMATION

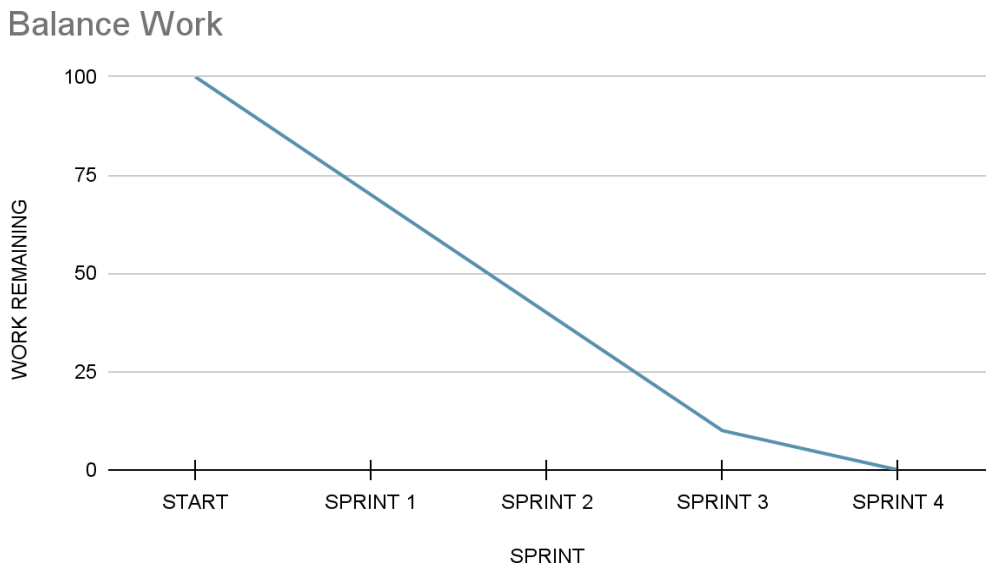
SPRINT	FUNCTIONAL REQUIREMENT	USER STORY(TASK)	STORY POINTS	PRIORITY	TEAM MEMBERS
Sprint-1	Resources Initialization	Create and initialize accounts in various public APIs like Open Weather API.	1	LOW	PORKODI S AMIZHTHAA B ASMA ROSHAN T DIVYA DHARSHINI S
Sprint-1	Local Server/Software Run	Write a Python program that outputs resultsgiven the inputs like weather and location.	1	MEDIUM	PORKODI S AMIZHTHAA B ASMA ROSHAN T DIVYA DHARSHINI S
Sprint-2	Push the server/software tocloud	Push the code from Sprint 1 to cloud so it can be accessed from anywhere	2	MEDIUM	PORKODI S AMIZHTHAA B ASMA ROSHAN T DIVYA DHARSHINI S
Sprint-3	Hardware initialization	Integrate the hardware to be able to access thecloud functions and provide inputs to the same.	2	HIGH	PORKODI S AMIZHTHAA B ASMA ROSHAN T DIVYA DHARSHINI S
Sprint-4	UI / UX Optimization & Debugging	Optimize all the shortcomings and provide better user experience.	2	LOW	PORKODI S AMIZHTHAA B ASMA ROSHAN T DIVYA DHARSHINI S

SPRINT DELIVERY SCHEDULE

PROJECT TRACKER, VELOCITY & BURNDOWN CHART

Sprint	Total Story Points	Duration	Sprint StartDate	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	31 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	07 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	14 Nov 2022

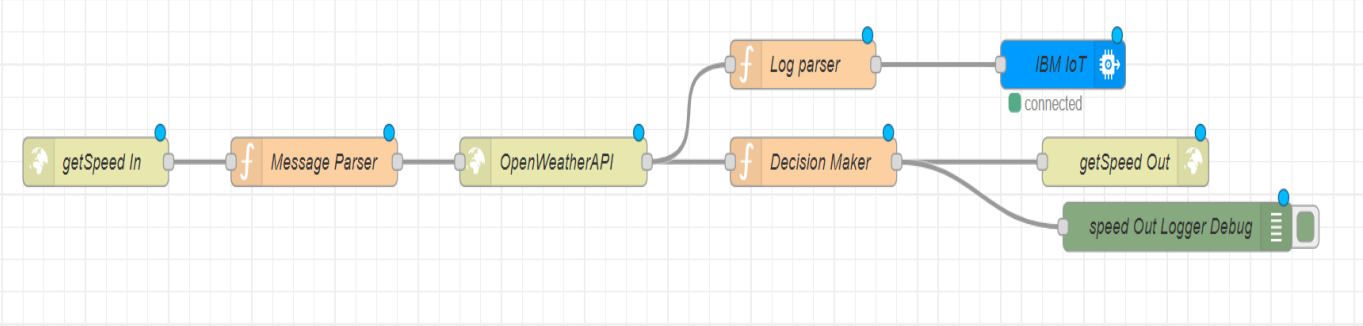
BURN DOWN CHART



7.CODING & SOLUTIONING

FEATURE - 1

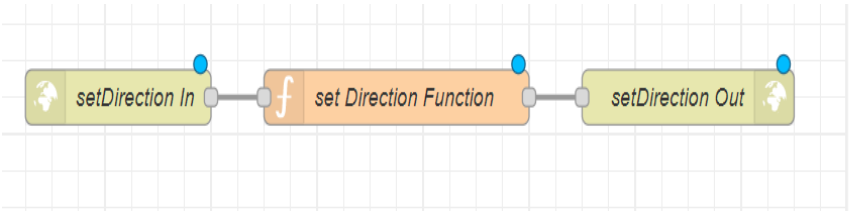
GET SPEED FOR GIVEN LOCATION & CLIMATE



- This part of Node RED flow accepts an http GET end point at **"/get Speed"** from which the location, uid, hospital/school zone infare passed.
- Message parser sets the required APIKEY for **Open Weather API** for the next block.
- This data is then passed onto Decision Maker which makes all the decisions regarding the message to be output at the display andsends it as a http response.
- This data is displayed at the microcontroller. Thus, a lot of battery is saved due to lesser processing time.

FEATURE – 2

SET DIRECTION REMOTELY FOR A GIVEN SIGN BOARD



- This part of Node RED flow accepts an **http GET** end point at **"/setDirection"** from which the uid and direction information are passed by the respective authorities**.**

- **Set Direction** Function block adds the direction information to the database and returns the sameas an http response. This data is sent to the microcontroller along with the **"/getSpeed"** path and the microcontroller displays it.

8.TESTING

TEST CASES

- **TEST CASE 1**

  Clear weather - Usual Speed Limit.

- **TEST CASE 2**

  Foggy Weather - Reduced Speed Limit.

- **TEST CASE 3**

  Rainy Weather - Further Reduced Speed Limit.

- **TEST CASE 4**

  School/Hospital Zone - Do not
  Honk sign is displayed.

USER ACCEPTANCE TESTING

Dynamic speed & diversion variations based on the weather and traffic helps user to avoid traffic and have a safe journey home. The users would welcome this idea to be implemented everywhere.

9.RESULT

PERFORMANCE METRICES

Based on the IBM pack we chose, the performance of the website varies. Built upon NodeJS, a light and high performance engine, Node RED is capable of handling up to 10,000 requests per second. Moreover, since the system is horizontally scalable,an even higher demand of customers can be served.

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Lower battery consumption since processing is done mostly by Node RED servers in the cloud.
- Cheaper and low requirement micro controllers can beused since processing requirements are reduced.
- Longer lasting systems
- Dynamic sign updating.
- School/Hospital zone alerts.

DISADVANTAGES

- The size of the display determines the requirement of themicro controller.
- Dependent on Open Weather Map API and hence the speed reduction is same for a large area in the scale of cities.

11.CONCLUSION

Our project is capable of serving as a replacement for static signs for a comparatively lower cost and can be implementedin the very near future. This will help reduce a lot of accidents and maintain a more peaceful traffic atmosphere in the country.

12.FUTURE SCOPE

- Introduction of intelligent road sign groups in real life scenarios could have great impact on increasing the driving safety byproviding the end-user (car driver) with the most accurate information regarding the current road and traffic conditions.

- Even displaying the information of a suggested driving speed and road surface condition (temperature, icy, wet or dry surface) could result in smoother traffic flows and, what is more important, in increasing driver's awareness of the road situation.

13.APPENDIX

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-46319-1660745135

SOURCE CODE

```
#include <WiFi.h>

 #include <HTTPClient.h>

 #include <Adafruit_GFX.h>

#include <Adafruit_ILI9341.h>

#include <string.h>


 const char* ssid =
 "Wokwi-GUEST";const
 char* password = "";

 #define TFT_DC 2
 #define TFT_CS 15
 Adafruit_ILI9341 tft = Adafruit_ILI9341(TFT_CS, TFT_DC);

 String myLocation =
 "Chennai,IN"; String
 usualSpeedLimit = "70"; //
 kmph

 int schoolZone =
 32; int
 hospitalZone =
 26;

 int uid = 2504;

 String getString(char x)
 {
     String s(1,
     x);return
     s;
```

```
}

String stringSplitter1(String fullString,char delimiter='$')
{
    String returnString = "";
    for(int i = 0;
        i<fullString.length();i++) {
        char c = fullString[i];
        if(delimiter==c)
            break;
        returnString+=Stri
        ng(c);
    }
    return(returnString);
}

String stringSplitter2(String fullString,char delimiter='$')
{
    String
    returnString = "";
    bool flag = false;
    for(int i = 0;
        i<fullString.length();i++) {char
        c = fullString[i];
        if(flag)
            returnString+=Str
        ing(c);
        if(delimiter==c)
            flag = true;
    }
    return(returnString);
}

void rightArrow()
{
int refX = 50;
```

```
  int refY = tft.getCursorY() + 40;

  tft.fillRect(refX,refY,100,20,ILI9341_RED);
  tft.fillTriangle(refX+100,refY-
  30,refX+100,refY+50,refX+40+100,refY+10,ILI9341_RED);
   }

   void leftArrow()
   {
  int refX = 50;
  int refY = tft.getCursorY() + 40;

  tft.fillRect(refX+40,refY,100,20,ILI9341_RED);
  tft.fillTriangle(refX+40,refY-
  30,refX+40,refY+50,refX,refY+10,ILI9341_RED);
   }

   void upArrow()
   {
  int refX = 125;
  int refY = tft.getCursorY() + 30;

  tft.fillTriangle(refX-
  40,refY+40,refX+40,refY+40,refX,refY,ILI9341_RED);
  tft.fillRect(refX-15,refY+40,30,20,ILI9341_RED);
   }

String APICall()
  {HTTPClient
  http;

  String url = " https://node-red-eaicw-2022-11-08.au-
  syd.mybluemix.net/getSpeed?";
  url += "location="+myLocation+"&";
  url += "schoolZone="+(String)digitalRead(schoolZone)+(String)"&";
  url +=
```

```
  "hospitalZone="+(String)digitalRead(hospitalZone)+(Stri
  ng)"&";url +=
  "usualSpeedLimit="+(String)usualSpeedLimit+(String)"&
  ";
  url +=
  "uid="+(String)uid;
  http.begin(url.c_str())
  ;
  int httpResponseCode = http.GET();

  if (httpResponseCode>0) {
        String payload = http.getString();
        http.end();
        return(pay
        load);
  }
  else {
        Serial.print("Error code: ");

        Serial.println(httpResponseCode);
  }
  http.end();
   }

void myPrint(String
  contents) {
  tft.fillScreen(ILI9341_BLA
  CK);tft.setCursor(0, 20);
  tft.setTextSize(4);
  tft.setTextColor(ILI9341_
  RED);
  //tft.println(contents);

  tft.println(stringSplitter1(conten
  ts));String c2 =
  stringSplitter2(contents);
```

```
if(c2=="s") // represents
Straight
{
      upArrow();
}
if(c2=="l") // represents left
{
      leftArrow();
}
if(c2=="r") // represents right
{
      rightArrow();
}
 }

 void setup() {
WiFi.begin(ssid, password, 6);

tft.begin();
tft.setRotation(1
);

tft.setTextColor(ILI9341_W
HITE);tft.setTextSize(2);
tft.print("Connecting to
WiFi");

    while (WiFi.status() !=
      WL_CONNECTED) {
      delay(100);
      tft.print(".");
}

tft.print("\nOK! IP=");
tft.println(WiFi.localIP());
 }
```

```
      void loop()
      {
myPrint(APICall());

delay(100);
 }
```