IBM-Nalaiya Thiran 2022-23

Project Report

SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

Team ID	PNT2022TMID06179
Project Name	Signs with Smart Connectivity for Better Road Safety
Team Members	P.Swetha Team leader M.Mohanapriya V.Bairavi C.Abirami

1.INTRODUCTION:

1.1 Project Overview

To replace the static signboards, smart connected sign boards 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 are displayed. Guide (Schools), Warning and Service(Hospitals, Restaurant) signs are also displayed accordingly. Different modes of operations can be selected with the help of buttons.

1.2Purpose

To replace the static signboards, smart connected sign boards 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.
- Traffic diversion signs are displayed.
- Messages indicating school ,hospital, police station zones are also displayed.

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

TITLE	AUTHOR	METHODOLOGY	MERITS	DEMERITS	YEAR
IOT Based Smart Traffic Management System	❖ Rachana K P ❖ Aravind R ❖ Ranjitha M Spoorthi Jwanita Soumya	❖ Internet of Things ❖ Digital Image Processing ❖ MATLAB	❖ IOT based traffic management Easy penalize traffic violators and help officials identify unauthorized drivers. ❖ Reroute the ambulance to the low congestion roads tohelp get medical care at the earliest.	❖ Additionai security measures are required ❖ Require High-Tech network infrastructure	2021

TITLE	AUTHOR	METHODOLOGY	MERITS	DEMERITS	YEAR
IOT Road Safety	Sowparnika Manakkattil	❖ IoT ❖ Arduino Code ❖ Node MCU controller ❖ Blynk app .	❖ Alert the driver about the speed limits and reduce the speed of the vehicles in sensitive public zones without any interference of the drivers. ❖ Abnormal information is passed to the vehicles which entering the same zone to take diversion to avoid traffic congestion	❖ The system takes more time because of short range communication ❖ No vision system ❖ Doesn't provide solution during network unavailability	2020

TITLE	AUTHOR	METHODOLOGY	MERITS	DEMERITS	YEAR
Reliable Smart Road Signs	Muhammed O. Sayin, Chung-Wei Lin, Eunsuk Kang, Shinichi Shiraishi, and Tamer Basar.	❖Machine learning-to recognize the surroundings and can base its strategic decisions on the information learnt. ❖Dedicated short range communication (DSRC) radios ❖Game Theoretical Approaches	❖ Road-sign classification in adversarial environments ❖ The detection mechanism involves multiple performance metrics ❖ The cost associated with adversary induced decoding error or failure, the false alarm cost, and the ease of a deceptive perturbation	❖ Need state-of-the-art vision-based roadsign recognition algorithms for better reliability ❖ Relaxation to attacker's algorithm under ❖ Stackelberg equilibrium leads to trigger of false alarm.	2019

TITLE	AUTHOR	METHODOLOGY	MERITS	DEMERITS	YEAR
;oT traffic prediction using multi-step ahead prediction with neural network	Ali R. Abdellah, Omar Abdul Kareem Mahmood, Alexander Paramonov and Andrey Koucheryavy	❖ Digital Speed meter Boards ❖ Artificial neural networks ❖ Traffic prediction ❖ A re-routing algorithm - to deviate ambulances to low congestion position based on network of sensors and vehicles	❖ dynamic handling of traffic signals basedon traffic density. ❖ Provides a real-time dashboard to monitor the traffic updates ❖ Prevent the loss of human life who is need to reach hospital at time	❖ Software based solution ❖ Require training ❖ Accuracy is not 100%	₡019

2.3 Problem Statement Definition

To replace the static signboards with smart connected sign boards to get the speed limitations from a web app using weather API and update it automatically based on the weather conditions, set diversions through API and warn drivers for school zones and hospital zones.

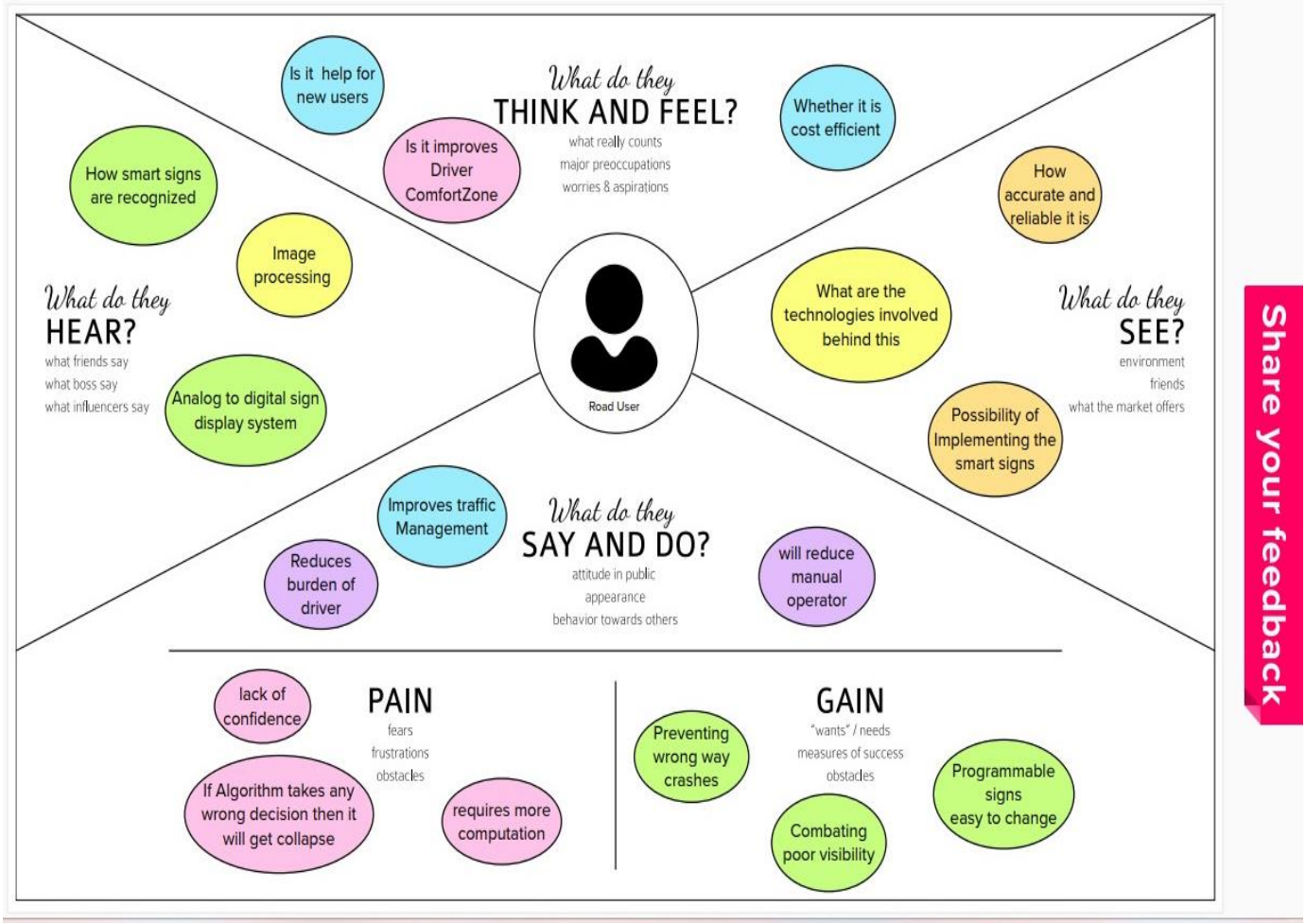
- Man behind the wheel plays an important role in most of the crashes. In most of the cases crashes occurs either due to carelessness.
- In this project reduce the human error and provide the safety roads.
- In this project we have to improve the road safety.
- Road safety refers to the methods and measures taken to prevent a person using the road from being killed or seriously injured.
- It focuses on preventing accidents that result in serious injury and death, keeping in mind that human error can occur.
- The system is monitoring for road so the drivers can not break the rules

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

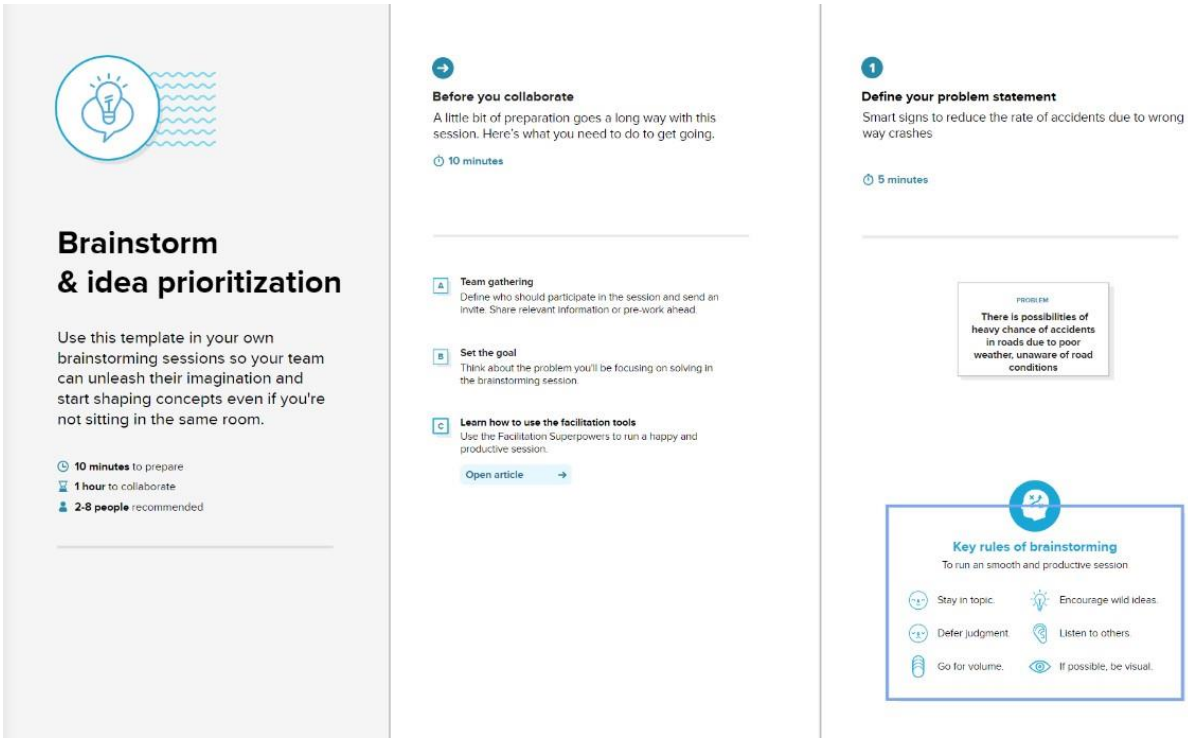
1

Build empathy and keep your focus on the user by putting yourself in their shoes.



Share your feedback

3.2 Ideation & Brainstorming



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article

Define your problem statement

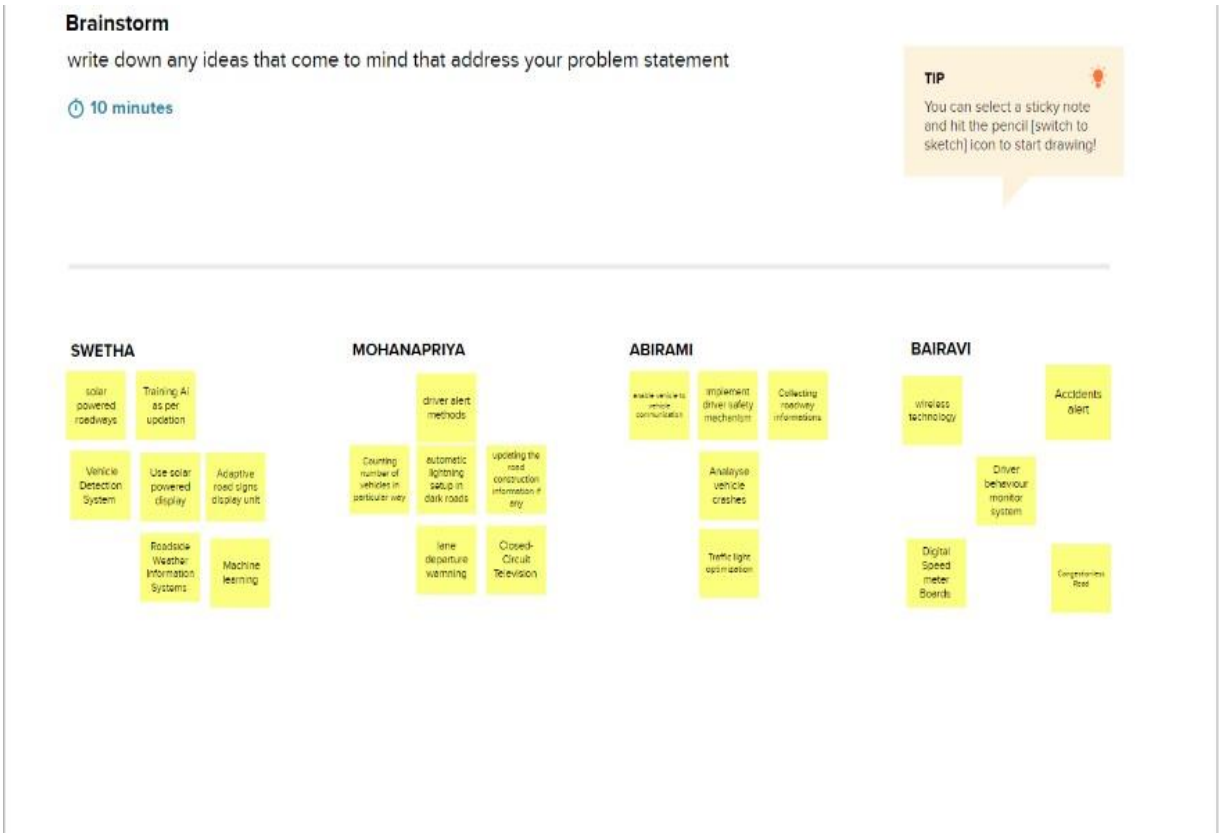
Smart signs to reduce the rate of accidents due to wrong way crashes

5 minutes

PROBLEM
There is possibilities of heavy chance of accidents in roads due to poor weather, unaware of road conditions

Key rules of brainstorming
To run an smooth and productive session

- Stay in topic.
- Defer judgment.
- Go for volume.
- Encourage wild ideas.
- Listen to others.
- If possible, be visual.



Brainstorm

write down any ideas that come to mind that address your problem statement

10 minutes

TIP
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

SWETHA

- solar powered roadways
- Training AI as per updation
- Vehicle Detection System
- Use solar powered display
- Adaptive road signs display unit
- Roadside Weather Information Systems
- Machine learning

MOHANAPRIYA

- driver alert methods
- Counting number of vehicles in particular way
- automatic lightning setup in dark roads
- updating the road construction information if any
- lane departure warning
- Closed-Circuit Television

ABIRAMI

- enable vehicle to vehicle communication
- implement driver safety mechanism
- Collecting roadway informations
- Analayse vehicle crashes
- Traffic light optimization

BAIRAVI

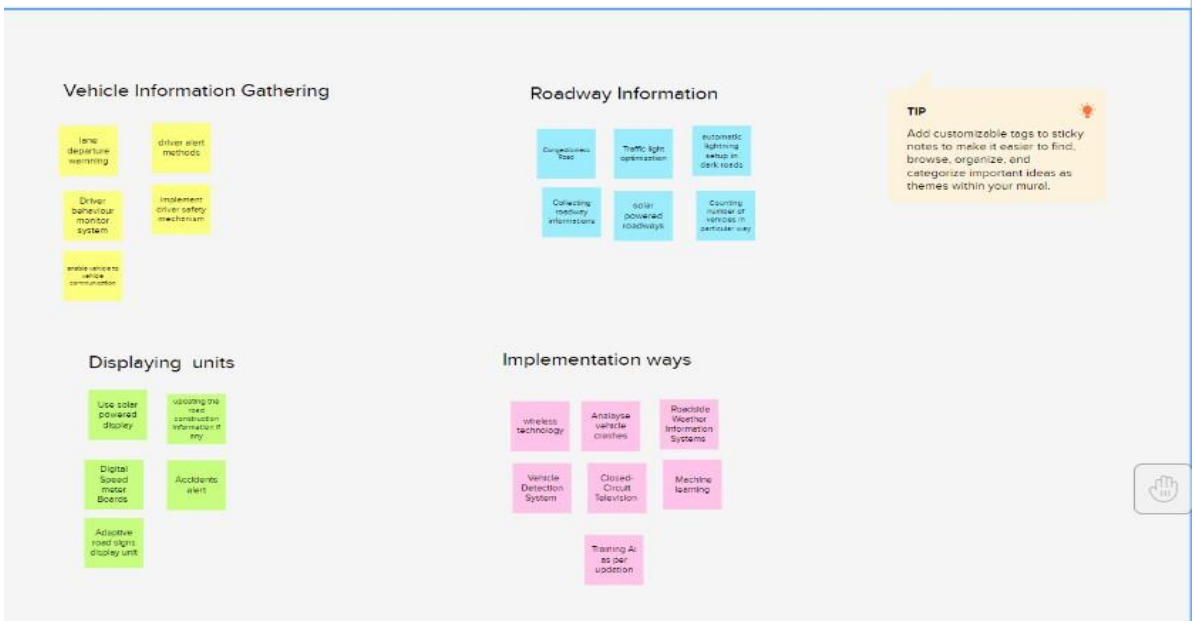
- wireless technology
- Accidents alert
- Driver behaviour monitor system
- Digital Speed meter Boards
- Congestionless Road

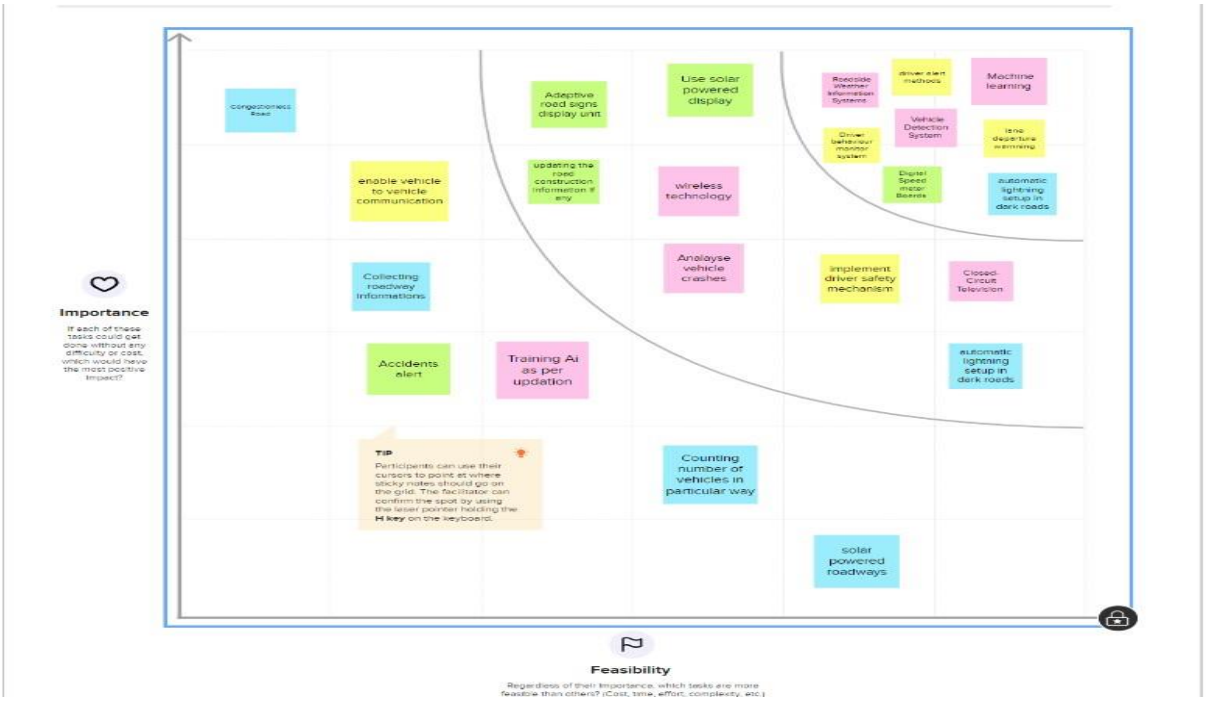
3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes



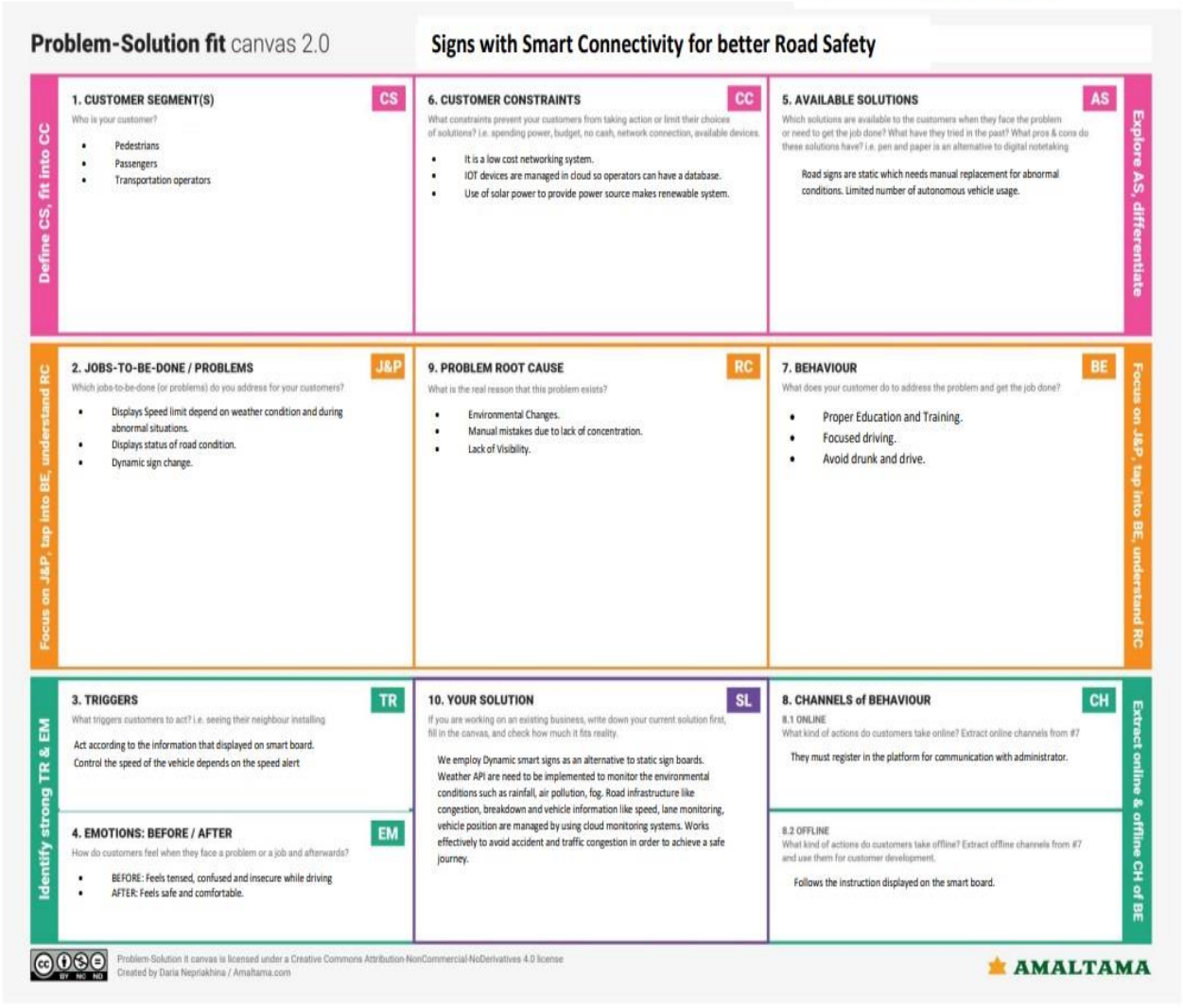


3.3 Proposed Solution

S.No	Parameter	Description
1.	Problem Statement	To provide a solution to incautious driving and poor weather conditions causes major road accidents
2.	Idea / Solution description	Roadway sensors are need to be implemented to monitor the environmental conditions such as rainfall, air pollution, fog and to monitor road infrastructure like congestion, breakdown and to monitor vehicle information like speed alert, lane monitoring, vehicle position. Adaptive Digital board displays the signs as per the conditions. Communication from road infrastructure to vehicle provides high level safety, the driver will get updated information every time sensors detect any abnormal situations in the road or weather conditions. Data are stored in cloud.
3.	Novelty / Uniqueness	Adaptive digital display updates each and every information to both commonly in road and also to the driver via specific cloud platform where all the data were stored. Monitoring data, collecting data, storing data of each vehicle in the cloud. Every commuters get the information when they enter into the specific road.
4.	Social Impact / Customer Satisfaction	Commuters feel comfort while driving, reduce number of accidents, avoid traffic congestion and knowledge about the road they travel.
5.	Business Model (Revenue Model)	• Road contractors and vehicle manufacturer are the main partners. • Providing cloud platform. • Make commuters to subscribe for updating information. • Providing reliable services • Commuters comfort level.
6.	Scalability of the Solution	The scalability of the solution will fit for even heavy load of vehicles are using the road and also it will withstand all environmental conditions. The analysis process should be done and produce result within fraction of seconds so that it avoid accidents.

3.4 Problem Solution fit

TEAM ID: PNT2022TMID06179



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Travelers Registration	Registration in the platform needs for communicating with customer through their mobile
FR-2	Transport Agency Registration	Register for getting approval to implement the smart sign boards for better road safety
FR-3	Weather Monitoring	Open weather API implemented to monitor weather reports and update in database
FR-4	Sensor implementation	Monitoring traffic density and road condition, pedestrian monitoring and controls traffic signals.
FR-5	Database Management	Updating information in the database to intimate the users about the abnormal situations
FR-6	Information Sharing	Once the situation detected the user get information via the digital display who travels along the road also it will update in the platform, so others plan accordingly

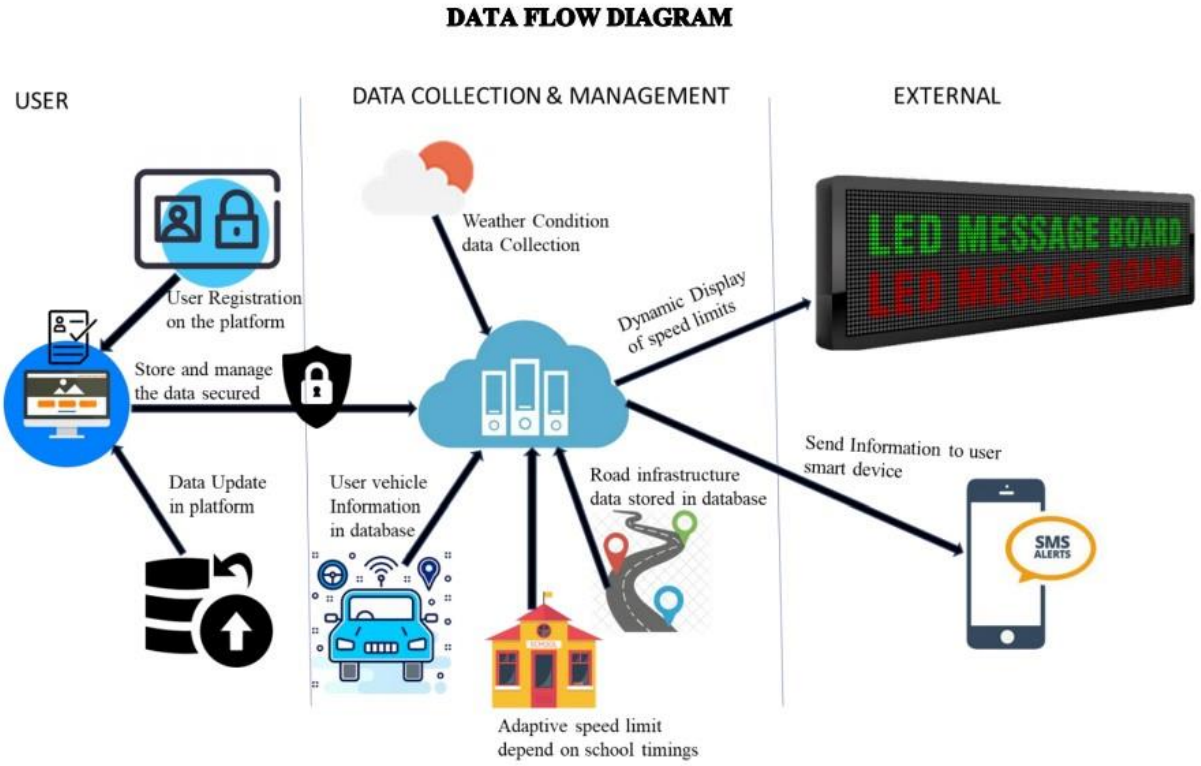
4.2 Non-Functional requirements

Following are the non-functional requirements of the proposed solution.

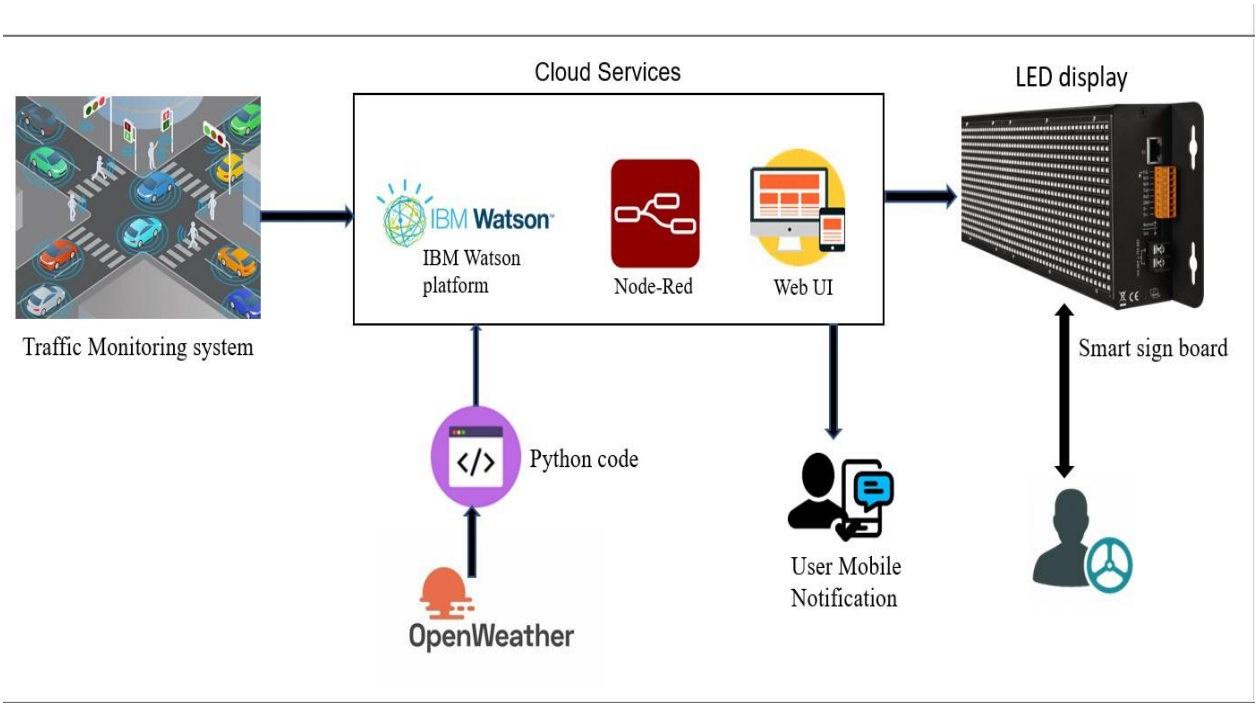
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Easy to follow instructions displays on the board. Understanding the signs should be clear.
NFR-2	**Security**	Provide better security, any other third party can't able to display information in the board, Users data are kept confidential.
NFR-3	**Reliability**	It can able to withstand in any weather condition and the hardware parts require periodic monitoring to avoid any damage. It is dynamic in nature and reduce traffic congestion.
NFR-4	**Performance**	The smart display improves the safety and it makes user tense free and keep them in a comfort zone. Also quality of service is improved.
NFR-5	**Availability**	The solution is available 24X7 and also withstand any climate changes.
NFR-6	**Scalability**	It can be implemented efficiently in anywhere and data execution will be faster. Provides better safety

5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



5.3 User Stories

Functional Requirement (Epic)	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**
Dynamic Speed Limit	USN-1	As a traveller, it is essential for me to know the speed limit.	7	High
Sensor implementation	USN-2	As a traveller, I should concern in traffic density and road condition, pedestrian monitoring and controls traffic signals.	7	Low
Weather speed limit	USN-3	As a user, I should be aware of weather influence on speed limit of safer ride. Open weather API has to implement to monitor weather reports.	6	Medium
Safer Ride	USN-4	As a traveller, I should have a hustle free journey.	12	Medium
Transport Agency Registration	USN-4	Register for getting approval to implement the smart sign boards for better road safety	8	Medium
Login	USN-6	As an administrator, I should have an account on the website.	7	Low
Dashboard	USN-7	As an admin, I should be able to monitor and add sign nodes.	13	Medium

Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Monitoring	USN-8	As an admin, I must control and monitor the proper functioning of the sign through rarely required.	9	Low
More accurate indications	USN-9	As a user, as days pass by, more accurate guidance is needed.	4	Low
Information Sharing UI	USN-10	Once the situation detected the user get information via the digital display who travels along the road also it will update in the platform, so others plan accordingly	7	High

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dynamic Speed Limit	USN-1	As a traveller, it is essential for me to know the speed limit.	7	High	Swetha P Mohanapriya M Bairavi V

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Sensor implementation	USN-2	As a traveller, I should concern in traffic density and road condition, pedestrian monitoring and controls traffic signals.	7	Low	Bairavi V Abirami C
Sprint-1	Weather speed limit	USN-3	As a user, I should be aware of weather influence on speed limit of safer ride. Open weather API has to implement to monitor weather reports.	6	Medium	Mohanapriya M Abirami C
Sprint-2	Safer Ride	USN-4	As a traveller, I should have a hustle free journey.	12	Medium	Bairavi V Swetha P Mohanapriya M
Sprint-2	Transport Agency Registration	USN-4	Register for getting approval to implement the smart sign boards for better road safety	8	Medium	Abirami C Swetha P
Sprint-3	Login	USN-6	As an administrator, I should have an	7	Low	Swetha P Abirami C

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			account on the website.			
Sprint-3	Dashboard	USN-7	As an admin, I should be able to monitor and add sign nodes.	13	Medium	Bairavi V Mohanapriya M
Sprint-4	Monitoring	USN-8	As an admin, I must control and monitor the proper functioning of the sign through rarely required.	9	Low	Swetha P Abirami C
Sprint-4	More accurate indications	USN-9	As a user, as days pass by, more accurate guidance is needed.	4	Low	Mohanapriya M Swetha P
Sprint-4	Information Sharing UI	USN-10	Once the situation detected the user get information via the digital display who travels along the road also it will update in the platform, so others plan accordingly	7	High	Bairavi V Mohanapriya M Swetha P

Velocity:

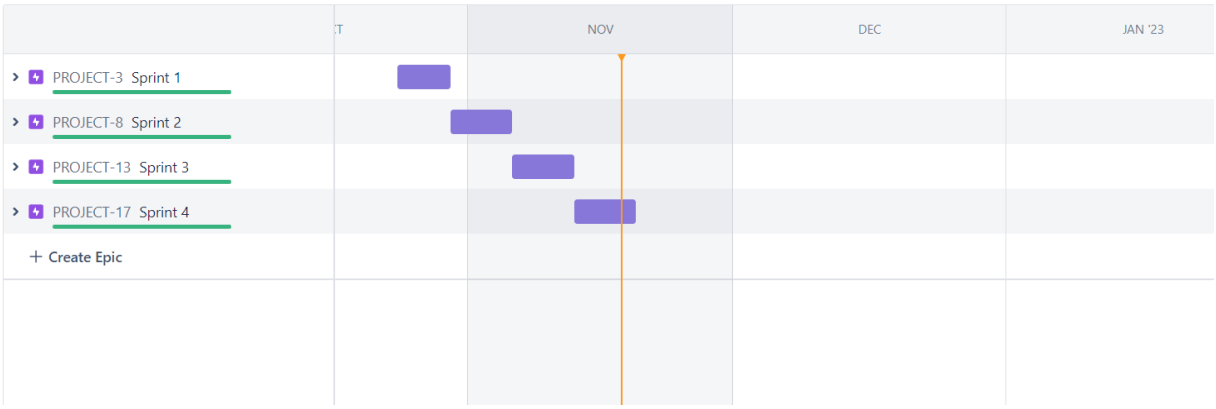
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's averagevelocity (AV) per iteration unit (story points per day).

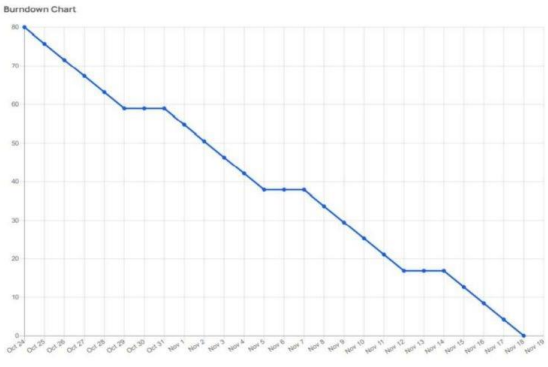
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

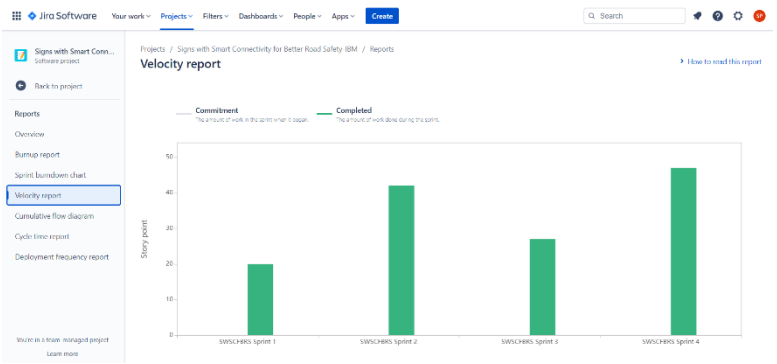
6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	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	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA







7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1 -GET WEATHER DETAILS FOR GIVEN LOCATION

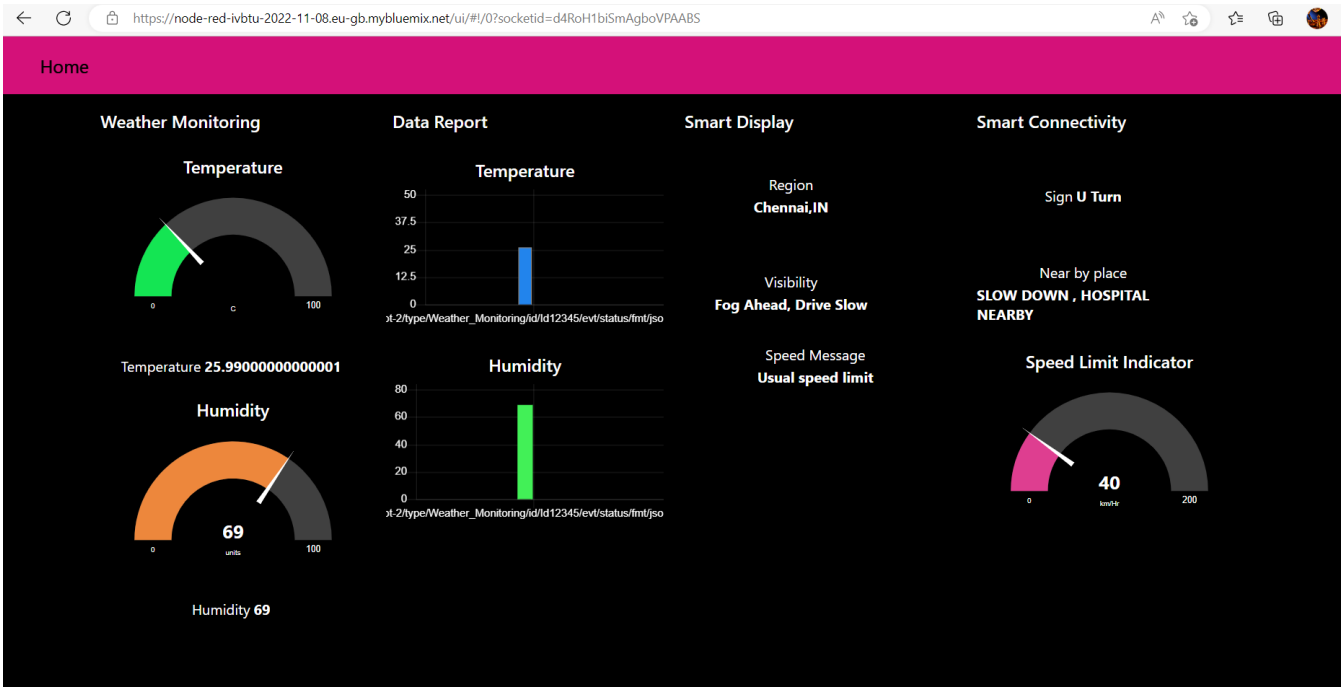
This part of Node RED flow accepts an http GET end point, from which the location, uid, info are passed. Message parser sets the required APIKEY for OpenWeatherAPI for the next block. This data is then passed onto Decision Maker which makes all the decisions regarding the message to be output at the display and sends it as a http response. This data is displayed at the microcontroller. Thus a lot of battery is saved due to lesser processing time.

Event Payload in Watson:


```
{
                    "temperature": 25.99000000000001,
                    "speedlimit": 40,
                    "humidity": 69,
                    "Message": "",
                    "Sign": "Left Diversion <-",
                    "Speed": "Moderate Speed",
                    "Visibility": "Fog Ahead, Drive Slow",
                    "location": "Chennai,IN"
}
```

7.2 Feature 2- GET SPEED LIMITATIONS, MESSAGES, SIGNS

The Node RED flow obtains the data published to the cloud such as speed limitations, messages such as warnings about the zones (schools, hospitals, police stations), signs such as diversions, U-turns etc, and displays them in the dashboard.



8. TESTING

Test Cases

```
1.{temperature: 25.99000000000001
speedlimit: 40
humidity: 69
Message: ""
Sign: "Right Diversion ->"
Speed: "SLOW DOWN , Speed Limit Exceeded"
Visibility: "Rain, Drive Slow"
location: "Chennai,IN"}

2.{temperature: 25.99000000000001
speedlimit: 40
humidity: 69
Message: "SLOW DOWN , HOSPITAL NEARBY"
Sign: "U turn"
Speed: "Usual speed limit"
Visibility: "Clear Weather"
location: "Chennai,IN"}

3.{temperature: 25.99000000000001
speedlimit: 60
humidity: 69
Message: ""
Sign: "Left Diversion <-"
Speed: "Moderate Speed"
Visibility: "Fog Ahead, Drive Slow"
location: "Chennai,IN"}

4.{temperature: 25.99000000000001
speedlimit: 40
humidity: 69
Message: "SLOW DOWN , HOSPITAL NEARBY"
Sign: "Up right"
Speed: "Moderate Speed"
Visibility: "Fog Ahead, Drive Slow"
location: "Chennai,IN"}
```

8.2 User Acceptance Testing

Dynamic speed & divertion 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. RESULTS

9.1 Performance Metrics

The performance of the website varies based on the software chosen for implementation . Built upon NodeJS, a light and high performance engine, NodeRED is capable of handling upto 10,000 requests per second. Moreover, since the system is horizontally scalable, a 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 be used since processing requirements are reduced. Longer lasting systems. Dynamic Sign updation. School/Hospital Zone alerts .

IoT-enabled systems can quickly respond to changing traffic patterns and return real-time data to help drivers plan their journeys better with better congestion monitoring. Reducing congestion and energy consumption has a positive impact on the environment.

DISADVANTAGES

The size of the display determines the requirement of the micro controller . Dependent on Open WeatherAPI and hence the speed reduction is same for a large area in the scale of cities.

The implementation of IoT technology in the long term reduces costs and optimizes the operations of any organization. But the initial integration requires investment. These are the costs associated with the network infrastructure, the modernization of vehicles directly connected to the IoT solution, and the planning, implementation, management, and security of IoT systems. Therefore, many Customers immediately submit IoT development on outsourcing to Ukraine, known for its pool of talented programmers and quality solutions at a fair price

11. CONCLUSION

Our project is capable of serving as a replacement for static signs for a comparatively lower cost and can be implemented in the very near future. This will help reduce a lot of accidents, traffics and maintain a peaceful environment

12. FUTURE SCOPE

Introduction of intelligent road sign groups in real life scenarios could have great impact on increasing the driving safety by providing the end-user 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 a driver's awareness of the road

13. APPENDIX

Source Code:

Main.py:

```
# IMPORT SECTION STARTS
import brain
# IMPORT SECTION ENDS
# USER INPUT SECTION STARTS
myLocation = "Chennai,IN"
APIKEY = "0dde980b5ae5e4ab1c49d8f04d51507e"
localityInfo = {
   "schools" : {
      "schoolZone" : True,
      "activeTime" : ["7:00","17:30"] # schools active from 7 AM till 5:30 PM
      },
   "hospitalsNearby" : False,
   "usualSpeedLimit" : 40 # in km/hr
}
# USER INPUT SECTION ENDS
# MICRO-CONTROLLER CODE STARTS
while True :
   print(brain.processConditions(myLocation,APIKEY,localityInfo))
'''
MICRO CONTROLLER CODE WILL BE ADDED IN SPRINT 3 AS PER
OUR PLANNED SPRINT SCHEDULE
'''
# MICRO-CONTROLLER CODE ENDS
```

Weather.py:

```
# Python code


import requests as reqs


def get(myLocation,APIKEY):
    apiURL = f"https://api.openweathermap.org/data/2.5/weather?q={myLocation}&appid={APIKEY}"
    responseJSON = (reqs.get(apiURL)).json()
    returnObject = {
        "temperature" : responseJSON['main']['temp'] - 273.15,
        "humidity" : responseJSON['main']['humidity'],
        "weather" : [responseJSON['weather'][_]['main'].lower() for _ in range(len(responseJSON['weather']))],
        "visibility" : responseJSON['visibility']/100, # visibility in percentage where 10km is 100% and 0km is 0%
    }
    if("rain" in responseJSON):
        returnObject["rain"] = [responseJSON["rain"][key] for key in responseJSON["rain"]]
    return(returnObject)
```

Brain.py:

```
# IMPORT SECTION STARTS
import weather
from datetime import datetime as dt
from publishData import logData2Cloud as log2cloud

# IMPORT SECTION ENDS
# -----------------------------------------
# UTILITY LOGIC SECTION STARTS
def processConditions(myLocation,APIKEY,localityInfo):
    weatherData = weather.get(myLocation,APIKEY)
log2cloud(myLocation,weatherData["temperature"],weatherData["visibility"],w
eatherData["humidity"])

    finalSpeed = localityInfo["usualSpeedLimit"] if "rain" not in weatherData
else localityInfo["usualSpeedLimit"]/2
    finalSpeed = finalSpeed if weatherData["visibility"]>35 else finalSpeed/2

    if(localityInfo["hospitalsNearby"]):
        # hospital zone

     doNotHonk = True
    else:
        if(localityInfo["schools"]["schoolZone"]==False):
            # neither school nor hospital zone
            doNotHonk = False
        else:
```

```
        # school zone
        now = [dt.now().hour,dt.now().minute]
        activeTime = [list(map(int,_.split(":"))) for _ in
localityInfo["schools"]["activeTime"]]
        doNotHonk = activeTime[0][0]<=now[0]<=activeTime[1][0] and
activeTime[0][1]<=now[1]<=activeTime[1][1]


  return({
      "speed" : finalSpeed,
      "doNotHonk" : doNotHonk
   })

# UTILITY LOGIC SECTION ENDS
```

publishData.py:

```
# Python code

# IMPORT SECTION STARTS

import wiotp.sdk.device # python -m pip install wiotp
import time
import random

# IMPORT SECTION ENDS
# -----------------------------------------
```

```
# API CONFIG SECTION STARTS
myConfig = {
    "identity" : {
        "orgId" : "b7lu7v",
        "typeId" : "Weather_Monitoring",
        "deviceId" : "Id12345"
    },
    "auth" : {
        "token" : "12345678"
    }
}
# API CONFIG SECTION ENDS
# ---------------------------------------------
# FUNCTIONS SECTION STARTS

def myCommandCallback(cmd):
    print("recieved cmd : ",cmd)

def logData2Cloud(location,temperature,speedlimit,humidity):
    client =
wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
    client.connect()
    repo=random.randint(1,3
  speedlimit=40;
    if repo==1:
        prt="SLOW DOWN , SCHOOL IS NEAR"
    elif repo==2:
        prt="SLOW DOWN , HOSPITAL NEARBY"
```

```
elif repo==3:
    prt="NEED HELP, POLICE STATION NEARBY"
else:
    prt=""
speed=random.randint(0,150)
if speed>=100:
    prt3="SLOW DOWN , Speed Limit Exceeded"
elif speed>=60 and speed<100:
    prt3="Moderate Speed"
else:
    prt3="Usual speed limit"
sign=random.randint(1,3)
if sign==1:
    prt2="Right Diversion ->"
elif sign==2:
    prt2="Left Diversion <-"
elif sign==3:
    prt2="U Turn"
else:
    prt2=""
if temperature<=30:
    prt4="Fog Ahead, Drive Slow"
else:
    prt4="Clear Weather"

client.publishEvent(eventId="status",msgFormat="json",data={
    "temperature" : temperature,
```

"speedlimit" : speedlimit,"humidity":humidity,"Message":prt, "Sign":prt2, "Speed":prt3, "Visibility":prt4,

"location" : location

},qos=0,onPublish=None)

client.commandCallback = myCommandCallback

client.disconnect()

Simulation:

```
#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+=String(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+=String(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-nwmrt-2022-11-04.eu-
gb.mybluemix.net/getSpeed?";
  url += "location="+myLocation+"&";
  url += "schoolZone="+(String)digitalRead(schoolZone)+(String)"&";
  url += "hospitalZone="+(String)digitalRead(hospitalZone)+(String)"&";
  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(payload);
  }
  else {
    Serial.print("Error code: ");
    Serial.println(httpResponseCode);
  }
  http.end();
}

void myPrint(String contents) {
  tft.fillScreen(ILI9341_BLACK);
  tft.setCursor(0, 20);
  tft.setTextSize(4);
  tft.setTextColor(ILI9341_RED);
  //tft.println(contents);

  tft.println(stringSplitter1(contents));
  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_WHITE);
  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);
}
```

Diagram json:

```
{
  "version": 1,
  "author": "Swetha P ECE",
  "editor": "wokwi",
  "parts": [  { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -4.95,
"left": -39.33, "attrs": {} },
    {
      "type": "wokwi-ili9341",
      "id": "lcd1",
      "top": -65.91,
      "left": 214.72,
      "rotate": 90,
      "attrs": {}
    },
    {
      "type": "wokwi-slide-switch",
      "id": "sw1",
      "top": 108.52,
      "left": -129.35,
      "rotate": 270,
      "attrs": {}
    },
    {
```

```
      "type": "wokwi-slide-switch",
      "id": "sw2",
      "top": 32.54,
      "left": -127.36,
      "rotate": 270,
      "attrs": {}
    } ],
  "connections": [  [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "sw2:3", "esp:D32", "green", [ "h18.07", "v27.04" ] ],
    [ "sw1:3", "esp:D26", "green", [ "h18.63", "v-20.06" ] ],
    [ "sw2:2", "esp:3V3", "white", [ "h7.96", "v196.62", "h160.9", "v-94.83" ]
],
    [ "sw1:2", "esp:3V3", "white", [ "h17.9", "v104.03", "h146.45", "v-78.22"
] ],
    [ "lcd1:GND", "esp:GND.1", "black", [ "h-20.15", "v81.46" ] ],
    [ "lcd1:CS", "esp:D15", "violet", [ "h-47.36", "v71.85", "h-61.37" ] ],
    [ "lcd1:RST", "esp:D4", "magenta", [ "h-71.41", "v45.18" ] ],
    [ "lcd1:D/C", "esp:D2", "#8f4814", [ "h-34.23", "v43.98", "h-74.5" ] ],
    [ "lcd1:MOSI", "esp:D23", "green", [ "h-62.03", "v-70.03" ] ],
    [ "lcd1:SCK", "esp:D18", "gray", [ "h-86.57", "v-21.93" ] ],
    [
      "lcd1:VCC",
      "esp:VIN",
      "red",
      [ "h-19.42", "v-90.92", "h-206.34", "v201.11", "h9.27", "v-0.03" ]
    ],
    [ "lcd1:MISO", "esp:D19", "gold", [ "h-26.5", "v-73.21", "h-60.8",
"v21.04" ] ] ]
}
```

Libraries:

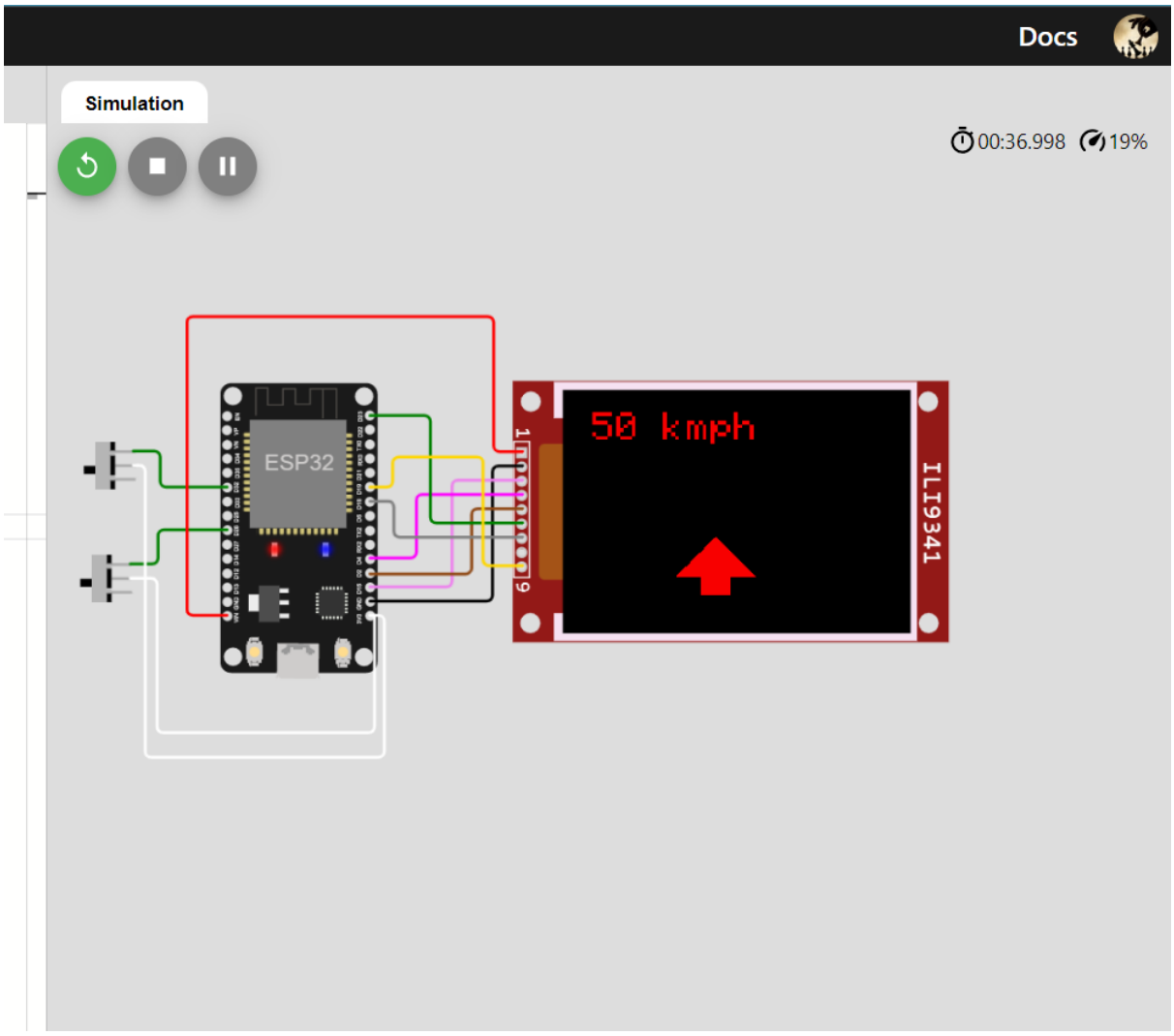
Wokwi Library List

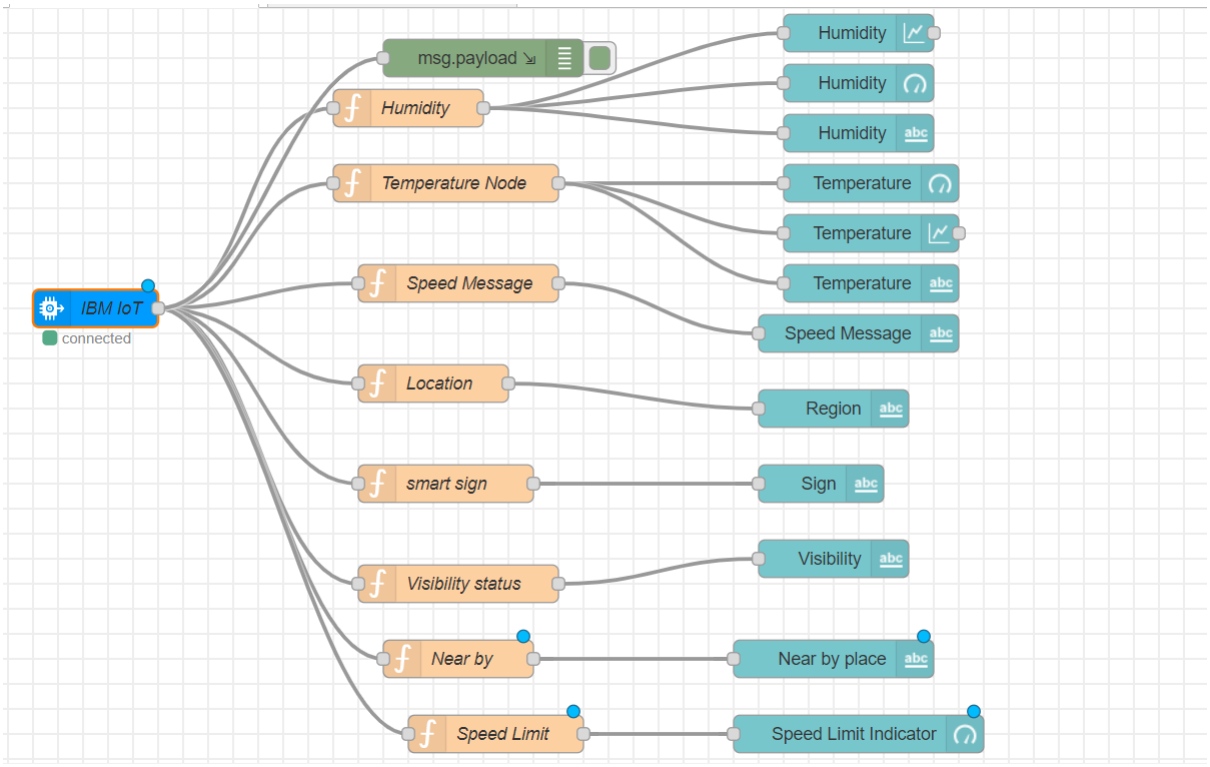
See https://docs.wokwi.com/guides/libraries

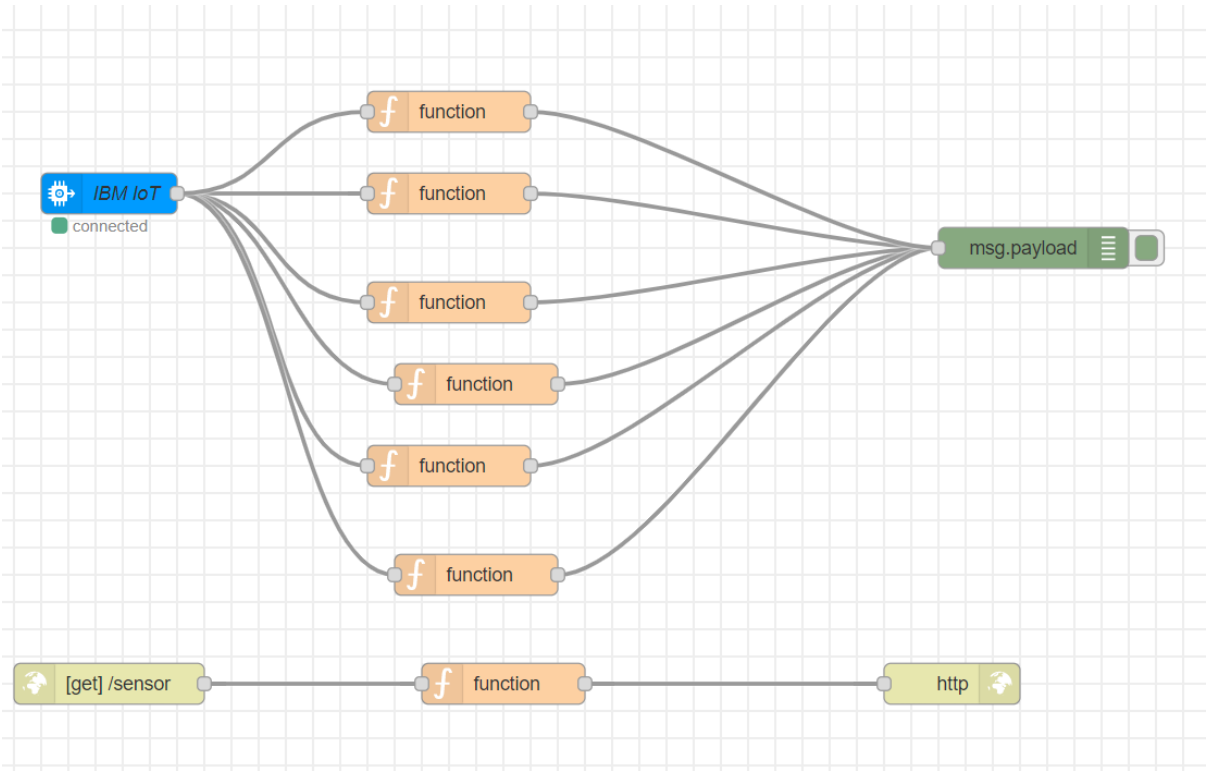
Adafruit GFX Library

Adafruit ILI9341

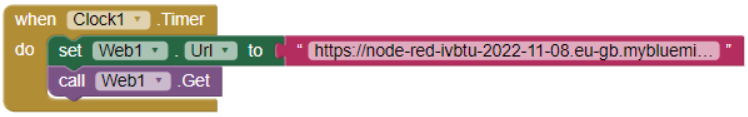
Output :

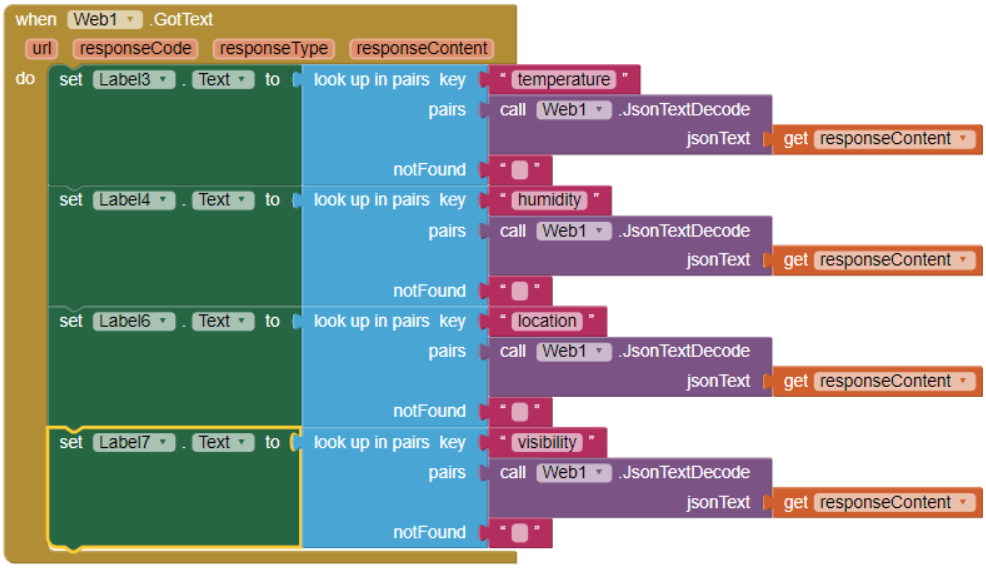


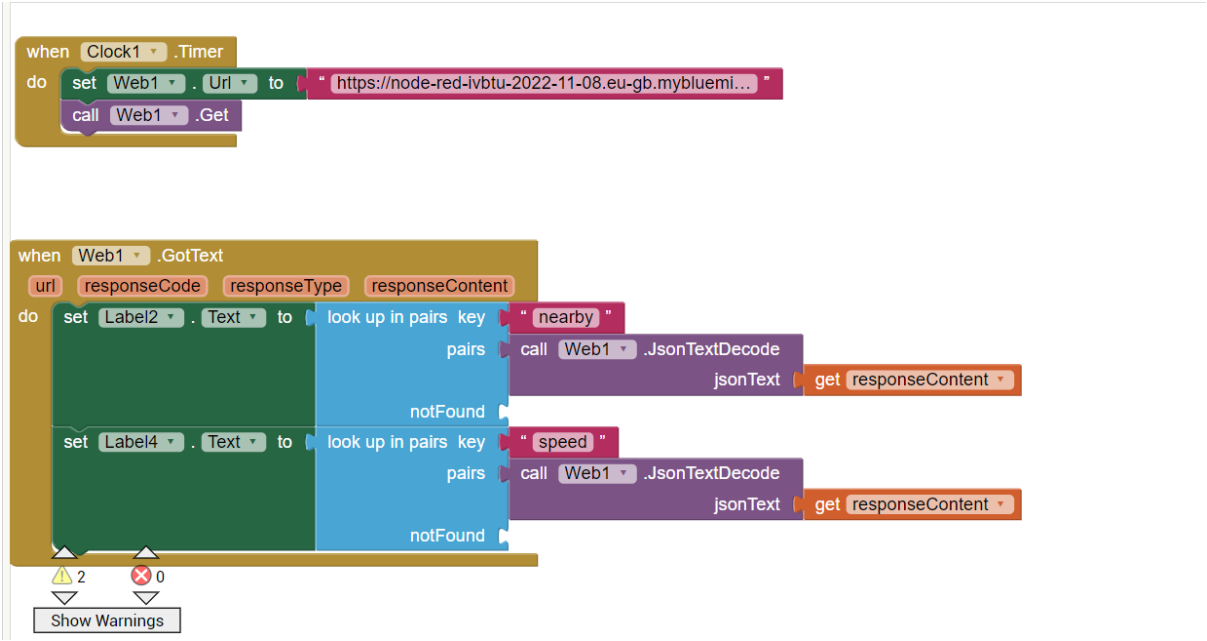

msg.payload
Humidity
Temperature Node
Speed Message
Location
smart sign
Visibility status
Near by
Speed Limit
IBM IoT
connected
Humidity
Humidity
Humidity
Temperature
Temperature
Temperature
Speed Message
Region
Sign
Visibility
Near by place
Speed Limit Indicator


function
function
function
function
function
function
IBM IoT
connected
msg.payload
[get] /sensor
function
http


Home
Weather Monitoring
Temperature
Temperature 28.99000000000001
Humidity
84
Humidity 84
Data Report
Temperature
Humidity
Smart Display
Region
Chennai, IN
Visibility
Fog Ahead, Drive Slow
Speed Message
Usual speed limit
Smart Connectivity
Sign Right Diversion ->
Near by place
SLOW DOWN , HOSPITAL NEARBY
Speed Limit Indicator
50

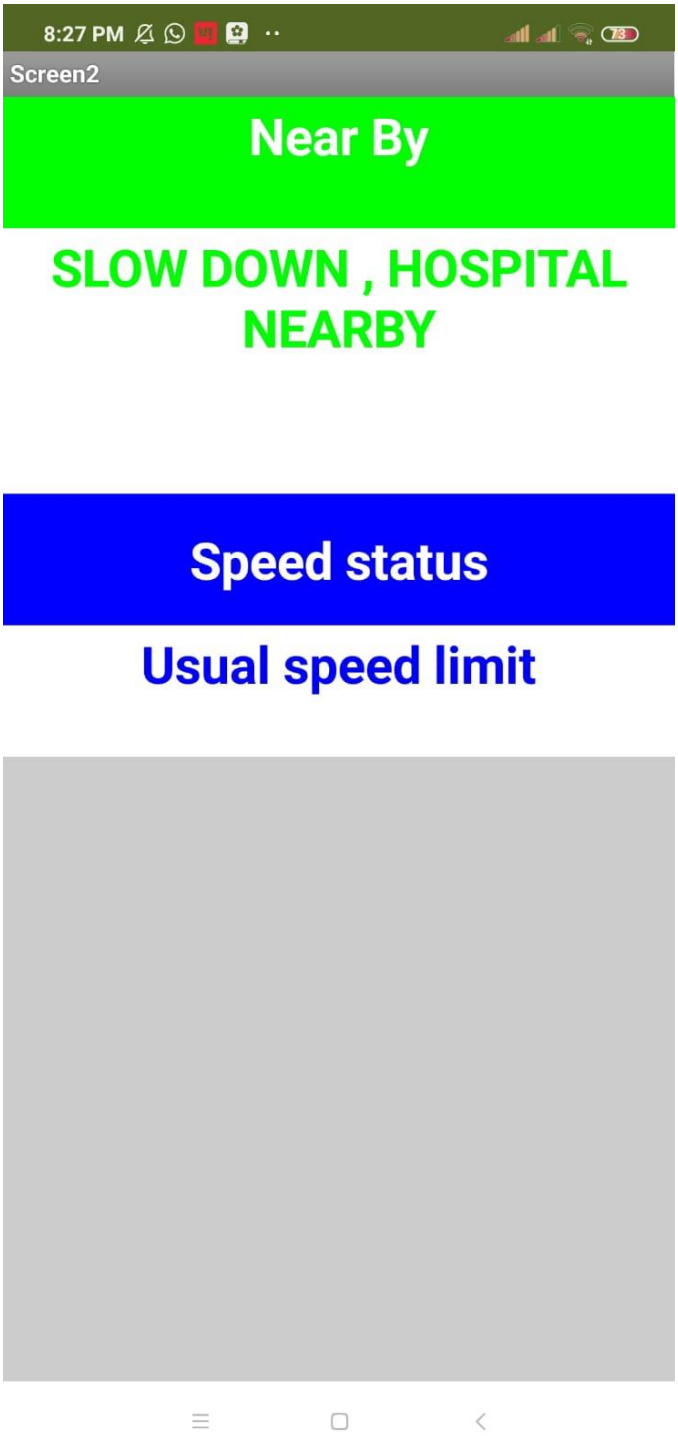

when Clock1 .Timer
do set Web1 . Url to " https://node-red-ivbtu-2022-11-08.eu-gb.mybluemi... "
call Web1 .Get


when Web1 .GotText
url responseCode responseType responseContent
do set Label3 . Text to look up in pairs key " temperature "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " "
set Label4 . Text to look up in pairs key " humidity "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " "
set Label6 . Text to look up in pairs key " location "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " "
set Label7 . Text to look up in pairs key " visibility "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " "


when Clock1 .Timer
do set Web1 . Url to " https://node-red-ivbtu-2022-11-08.eu-gb.mybluemi... "
call Web1 .Get
when Web1 .GotText
url responseCode responseType responseContent
do set Label2 . Text to look up in pairs key " nearby "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound
set Label4 . Text to look up in pairs key " speed "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound
2
0
Show Warnings


12:55 PM


9:07 PM
70
Screen1
Location
Chennai, IN
Temperature : 25.99
Humidity : 89
Fog Ahead, Drive Slow


8:27 PM
Screen2
Near By
SLOW DOWN , HOSPITAL NEARBY
Speed status
Usual speed limit

LINKS:

GIT HUB:

https://github.com/IBM-EPBL/IBM-Project-15603-1659601449

JIRA:

https://team-16678857772307.atlassian.net/jira/software/projects/SWSCFBRSI/boards/2

NODERED:

https://node-red-ivbtu-2022-11-08.eu-gb.mybluemix.net/red/#flow/c87be140779687c0

NODERED UI

https://node-red-ivbtu-2022-11-08.eu-gb.mybluemix.net/ui/#!/0?socketid=d4RoH1biSmAgboVPAABS

MIT

http://ai2.appinventor.mit.edu/#5205888143851520

https://node-red-ivbtu-2022-11-08.eu-gb.mybluemix.net/temperatureData

WOKWI

https://wokwi.com/projects/348844845571244627

OPEN WEATHER:

https://api.openweathermap.org/data/2.5/weather?q=Chennai,IN&appid=0dde980b5ae5e4ab1c49d8f04d51507e

DEMO LINK:

https://drive.google.com/file/d/1DYvHXS81wpN6sMxRB18JIffotczUekmJ/view?usp=drivesdk