

# **TEAM ID –PNT2022TMID41135 Project Report**

## **1. INTRODUCTION**

- 1.1 Project Overview
- 1.2 Purpose

## **2. LITERATURE SURVEY**

- 2.1 Existing problem
- 2.2 References
- 2.3 Problem Statement Definition

## **3. IDEATION & PROPOSED SOLUTION**

- 3.1 Empathy Map Canvas
- 3.2 Ideation & Brainstorming
- 3.3 Proposed Solution
- 3.4 Problem Solution fit

## **4. REQUIREMENT ANALYSIS**

- 4.1 Functional requirement
- 4.2 Non-Functional requirements

## **5. PROJECT DESIGN**

- 5.1 Data Flow Diagrams
- 5.2 Solution & Technical Architecture
- 5.3 User Stories

## **6. PROJECT PLANNING & SCHEDULING**

- 6.1 Sprint Planning & Estimation
- 6.2 Sprint Delivery Schedule
- 6.3 Reports from JIRA

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

- 7.1 Feature 1
- 7.2 Feature 2
- 7.3 Database Schema (if Applicable)

## **8. TESTING**

- 8.1 Test Cases
- 8.2 User Acceptance Testing

## **9. RESULTS**

- 9.1 Performance Metrics

## **10. ADVANTAGES & DISADVANTAGES 11. CONCLUSION 12. FUTURE SCOPE 13.**

## **APPENDIX Source Code**

GitHub & Project Demo Link

## **1. INTRODUCTION**

## **1.1 Project Overview**

The project implemented using IOT that predict traffic , rainfall ,ambulance detection ,human present in road , any alert sound . Using arduino UNO interface with sensors, buzzer, LED etc. Sensor such as Ultrasonic sensor , Temperature sensor , PIR sensor. The user will see the display on SMART SIGNS on the road . The device we designed UI (WEB based Application) given to the traffic controller. They display on SIGNS. The main advantages of the project is prevent the traffic ,give path to ambulance , in rainy days give instruction to the driver about weather using open weather app they go fast or slow.

## **1.2 Purpose**

The purpose of the project is make easy road travel with IOT devices. It also save lives. In this UI we have information of temperature , sound ,human detection . It makes travel easier.

## **2. LITERATURE SURVEY**

### **2.1 Existing Problem**

In present Systems the road signs and the speed limits are Static. But the road signs can be changed in some cases. We can consider some cases when there are some road diversions due to heavy traffic or due to accidents then we can change the road signs accordingly if they are digitalized.

The early effects to prevent road accidents and to ensure road safety includes the use of speed detection devices,CCTVs,speed limiters and emergency accident units as the first phase.Despite achieving the state-of-the-art performance, the existing systems suffer from two main problems.

- Over Speed : These systems cannot control speed at some specific zones.
- Exact location of accident occurred: These systems cannot give the precise location of accident .

### **2.2 References**

Authors	Type	Availability	Research	Findings
Houser ,Pierowicz ,& Fuglewicz (2005)	FMC SA report	Public	A report to provide a better understanding of the function of on-board safety systems and provide insight into the safety and efficiency benefits of using such systems.	Describe the concept of operations and the voluntary requirements for the use of VSS for large trucks greater than 10,000 pounds GVWR .
Berg ,Nie woh ner, Burke,& Morschh euser (2001)	Journal article	Public	An investigation of 109 real life truck crashes and a crash test involving a Mercedes-Benz Actros.	Safety belts in heavy trucks have a potential to save drivers and passengers. Ejected truck occupant have the greatest probability of being killed in a crash.
Trevorrow & Eady (2010)	Australian road s report	Public	A report to improve knowledge and understanding of heavy vehicle brakes safety on long steep and very steep roads. literature review ,review of crash data ,and a vehicle test	Advanced braking systems offer increased safety in an emergency on steep roads due to the automatic application of the service brakes preventing roll-over or run-off-road crashes. While brake failure crashes accounted for less than one quarter of fatal truck crashes ,break failure crashes were found to be more serious. Fatal break failure crashes were more likely on horizontal curves ,however brake failure crashes on a combination of horizontal curve and vertical grade were more serious than those occurring on vertical grade alone. The main safety issue highlighted was the drivers' interaction with the auxiliary braking system. Inadequate owner's manual information and a lack of real-time driver feedback regarding the performance (or lack thereof) of brakes were identified.
Lambert & Rechner (2002)	MUARC report	Public	A review and report of the Issue of rear and side under run crashes.	Two major effects of under run on the outcomes of crashes were identified: under run can expose light vehicle occupants to the rigid structures of the truck before the safety features of the light vehicle come into effect; and damage to the heavy vehicle component.

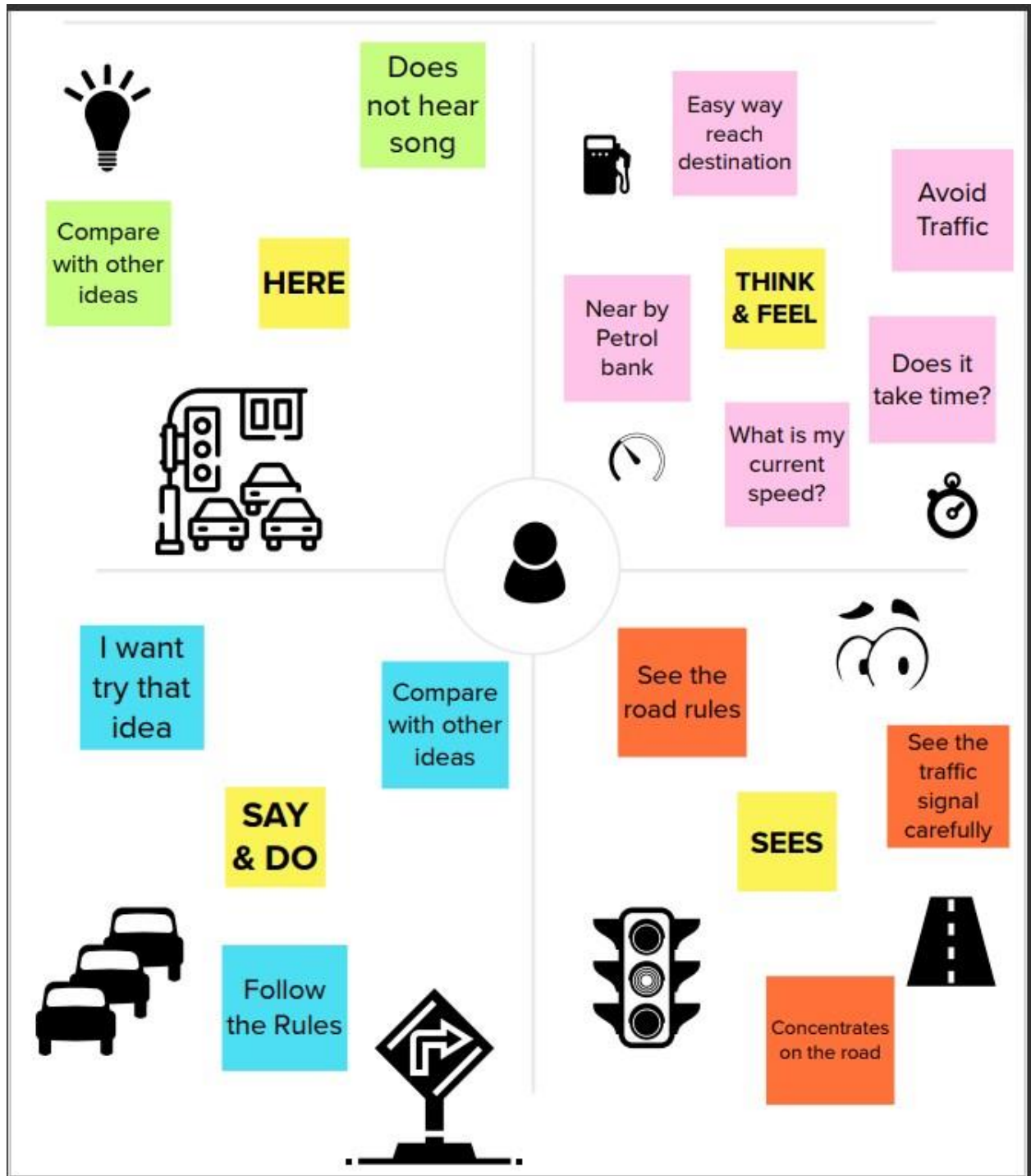
				<p>n ents(e.g.,steering,braking,etc.) can reduce the controllability of the truck during or after the crash.</p> <p>Thereislittleevidencesuggestingthatimprovementsintruckunderrunprotectioncan notbeachieved.</p> <p>Thereissomeevidencethatenforcementofunderrunrequirementsandstandardsisla king.</p> <p>Performanceoffrontbarriersmusthave asignificantlyhigherstandard,atleast twice that of rear under run barriers.</p> <p>The requirements of barriers should extend to vehicles of 3.0tonnesGVM.</p>
Hart(2010 )	Conf eren ce pape r	Pub lic	Describesthedevelopmentof theAustralianbrakebal ancecodeofpracticetoguidet heintermixingofbraket echnologiesonheavyvehicle combinationvehicles.	Awiderangeofbrakingtechnologiescannow beintermixedoncombinationvehicles,e. g.,advancedelectroniccontrolsarebeingco nnectedtobasicvehicles.Therecommen dedperformancelevelsetoutbythecodeisth atacominationvehiclebeabletoachieve aninstantaneousdecelerationlevelonase aled60km/h

### 2.3 Problem Statement Definition

The traffic problem is one of the common problems in metropolitan cities around the world. It causes accidents and loss of lives .We cannot control the occurrences of accidents but taking precautions to avoid life threatening injuries due to road accident is in our hands-by wearing helmet.

### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas



## 3.2 Ideation and Brain storming

**Define your problem statement**

What problem are you trying to solve? Frame your problem as a three-sentence statement. This will be the focus of your brainstorm.

15 minutes

**PROBLEM**

The traffic problem is one of the common problems in metropolitan cities around the world. It causes accidents and loss of lives.

**Brainstorm**

Write down any ideas that come to mind that address your problem statement.

15 minutes

**Methanize**

Brainstorming ideas and then writing them down in a structured manner.

**Nichitha**

Brainstorming ideas and then writing them down in a structured manner.

**Viji**

Brainstorming ideas and then writing them down in a structured manner.

**Priyadarshini**

Brainstorming ideas and then writing them down in a structured manner.

**Kalaivani**

Brainstorming ideas and then writing them down in a structured manner.

**Group ideas**

Now when sharing your ideas while clustering similar or related notes as you go. (Show all only notes have been grouped, give each cluster a sentence like label. If a cluster is bigger than an sticky note, try not use 7 pins and instead use the smaller sub-groups)

15 minutes

**People**

Brainstorming ideas and then writing them down in a structured manner.

**Prioritize**

Now open should all be on the same page about which's important meaning forward. Place your ideas on the grid to determine which ideas are important and which are feasible.

15 minutes

**Importance**

Brainstorming ideas and then writing them down in a structured manner.

**Feasibility**

Brainstorming ideas and then writing them down in a structured manner.

## 3.3 Proposed Solution

PROPOSED SOLUTION			
S.NO.	PARAMETER	DESCRIPTION	
1.	Problem Statement	The traffic problem is one of the common problems in metropolitan cities around the world. It causes accidents and loss of lives.	
2.	Solution	Traffic management Rules and regulations must be impressed upon everyone from drivers and passengers to pedestrians that safety comes first. Including road safety issue in textbooks. Finding alternative to road transportation and emphasizing on waterways and rail communication.	
3.	Uniqueness	An algorithm is given to predict the traffic solidity for future to minimize the traffic congestion. Development of IOT based traffic management system. Identify and penalize traffic violators and help officials identify unauthorized drivers. Reroute the ambulance to the low congestion roads to help get medical care at the earliest.	
4.	Social Impact	It helps the driver free from traffic jam also save lot of time. Minimize road accident by regulating the traffic also get free flow of traffic without unnecessary interruption and congestion. Promotes driver confidence.	
5.	Business Model	Drivers are under pressure to reach the destination in correct time due to traffic jams. To overcome this pressure, they can make use of predictive models which help them to ease the smart signs. Drivers free from congestion.	
6.	Scalability	Further to reduce the immense pressure faced by the drivers to travel on road, the model can also helped driver travel safely.	

## 3.4 Problem Solution Fit

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> Who is your customer? i.e. working parents of 0-5 y.o. kids <div>CS</div> <ol style="list-style-type: none"> <li>Passengers</li> <li>This is useful for drivers those who are travelling Different road structures.</li> </ol>	<b>6. CUSTOMER CONSTRAINTS</b> What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available services. <div>CC</div> <p>Customers no need to spend any money , power, network Connection. These project will available anytime until it gets damaged.</p>	<b>5. AVAILABLE SOLUTIONS</b> Which solutions are available to the customers when they face the problem? or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital navigation. <div>AS</div> <p>This project can with stand better than man made painted in signs. This project gives proper and clear understanding traffic signs and day to day current weather condition.</p>	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides. <div>J&amp;P</div> <ol style="list-style-type: none"> <li>It educates people about traffic signs who are travelling in roads.</li> <li>Showing different weather conditions and Indicating Temperature Values for passenger Convenience.</li> </ol>	<b>9. PROBLEM ROOT CAUSE</b> What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations. <div>RC</div> <p>Most people are not following traffic conditions &amp; not trying to have knowledge about various traffic signs. Back story: Most of the people forget to wear seat belts And using mobile phones during travelling due to this This type of behaviour it leads to major road accidents.</p>	<b>7. BEHAVIOUR</b> What does your customer do to address the problem and get the job done? i.e. Directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace) <div>BE</div> <p>The Digital signs educating the customers and the smart Weather condition detection, this helps the customer to address the problems and get job done.</p>	
<b>3. TRIGGERS</b> What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. <div>TR</div> <ol style="list-style-type: none"> <li>Not every people have knowledge about various traffic signs so, it helps some people about the different traffic signs.</li> <li>Conditions of the weather can't be predictable in some of the times so it shows temperature values to the people who are travelling in roads or highways.</li> </ol>	<b>10. YOUR SOLUTION</b> If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. <div>SL</div> <p>Nowadays road signs and speed limits are static, road signs and speed limits can't be changed in some cases. If we replace static</p>	<b>8. CHANNELS of BEHAVIOUR</b> ONLINE What kind of actions do customers take online? Extract online channels from #7. Customers can address their feedback through app or mail to get their job done. OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. Customer can address their feedback through toll free number or text messaging. <div>CH</div>	Identify strong TR & EM	
<b>4. EMOTIONS: BEFORE / AFTER</b> How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure -> confident, in control - use it in your communication strategy & design. <div>EM</div> <ul style="list-style-type: none"> <li>Some people don't have basic knowledge about various traffic signs &amp; cannot Predict weather conditions while travelling, so, due to that most of the road accidents happening.</li> <li>After implementing this project it helps and educate the people about various traffic signs &amp; indicating the current weather condition to the passengers. Due to this we can prevent major road accidents.</li> </ul>	signs with dynamic signs, the signs can be changed at any time and anywhere, even we can change the signs during a sudden change in weather conditions or if any accidents happened we can change the signs & tell the people to have another route or direction. If we replace ordinary signs with smart signs a large number of happening accidents can be reduced and we can save a lot of time by reducing the traffic. Even this type of system is helpful for education and medical institutions.	Identify strong TR & EM		

## 4. REQUIREMENT ANALYSIS

### 4.1 Functional Requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR No.1	Drivers and number of passenger on the vehicle	Number of passenger on the vehicle are noted
FR No.2	Predicting vehicle speed using sensors	IR sensor, Proximity sensor etc
FR.No.3	Pre-processing the speed of vehicle	Determination of Moving Vehicle speed using Image Processing

FR.No.4	Classification of sensor	A few examples of analog sensors are: accelerometers, pressure sensors, light, and sound sensors. Digital Sensors (also known as electronic or electrochemical sensors) convert
		the data transmission, digitally. Examples include digital accelerometers, pressure, and temperature sensors
FR.No.5	Building and training the system	The proposed system uses a set of ultrasonic sensors and has two modules: one for vehicle monitoring and other for priority management.
FR.No.6	Testing the model	In this phase, we tested the accuracy of the models with the test dataset that was formed in previous phase and the most accurate model is figured out.

## 4.2 Functional Requirement

### Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	It is simpler and possible to predict speed of the vehicle, accident at an earlier stage. Because that it benefits all kinds of people, it is a life saving option.
NFR-2	<b>Security</b>	Predict speed of the vehicle, accident helps to saves the life.
NFR-3	<b>Reliability</b>	This approach offers excellent performance and scalability, making it more dependable.



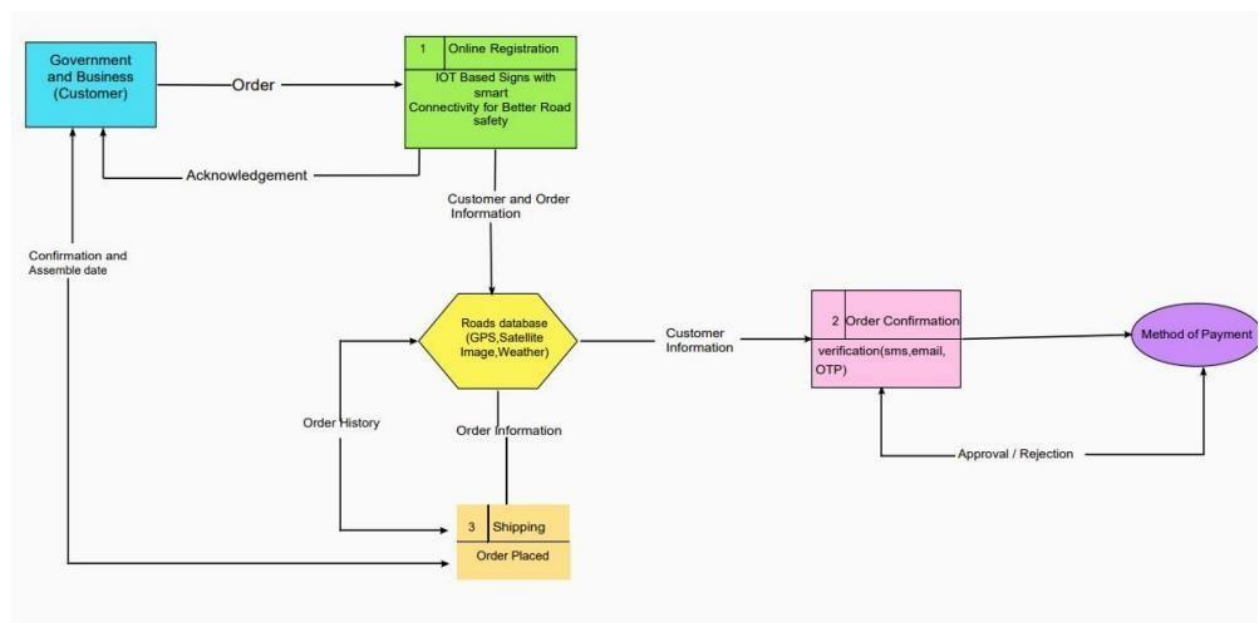
NFR-4	<b>Performance</b>	It provides accuracy of over 90%. Thus, it has a high performance rate.
NFR-5	<b>Availability</b>	By having few basic data set of people we can predict the accident ,speed of the vehicle
NFR-6	<b>Scalability</b>	It has more efficiency in detecting speed of the vehicle ,accident than any other models.

## 5. PROJECT DESIGN

### 5.1 Data Flow Diagram

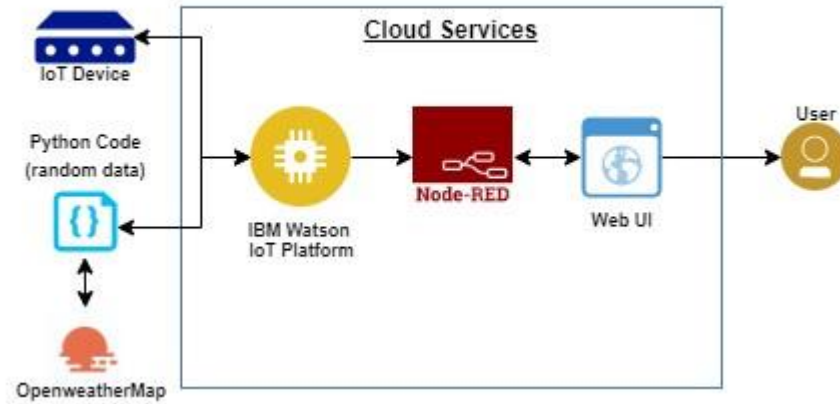
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.

A neat and clear DFD can depict the light amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 Solution and Technical Architecture

### Technical Architecture:



## 5.3 User Stories

### User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	Access my account / dashboard	High	Spint-1
Weather	Open weather map	USN-2	As a user, I want to check the weather of that location	Get the weather of that location	High	Spint-1
IoT devices	Automation	USN-3	As a user, I want to use IoT devices for automation purposes	Get the work done without manual effort	High	Spint-2
Python code	Random data	USN-4	As a user, I want to give some input to the devices for performing some action to complete the tasks very easily	Get the data Work flow	Medium	Spint-1
IBM Cloud	Cloud services	USN-5	As a user, I want to deploy these application for public version	Useful for all domain users	High	Spint-1
Node-Red	Integration	USN-6	As a user, I want to integrate the applications with hardware	IoT precise for linear work flow	Medium	Spint-3
Web UI	Interaction	USN-7	As a user, I want to interact with the digital products	IOT interact with the users	Medium	Spint-2

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning and Estimation

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Resource	Create a account in Tinkercad ,Open weather map app, MIT etc	1	Low	R.Mathaniza M.Viji K.Nishitha
Sprint-2	Software	Python IDLE	1	Medium	B.Priyadharshini R.Kalavani
Sprint-3	Interface	Interface wokwi with IBM lot Watson. Interface MIT with IBM lot Watson	2	High	R.Mathaniza M.Viji
Sprint-3	Interface	Interface Node-red with cloud	2	Medium	K.Nishitha M.Viji
Sprint-4	Hardware	Integrate the hardware components to the IBM cloud(IOT Watson)	2	High	M.Viji R.Mathaniza
Sprint-4	Road Safety	User has better road safety	1	Low	R.Mathaniza

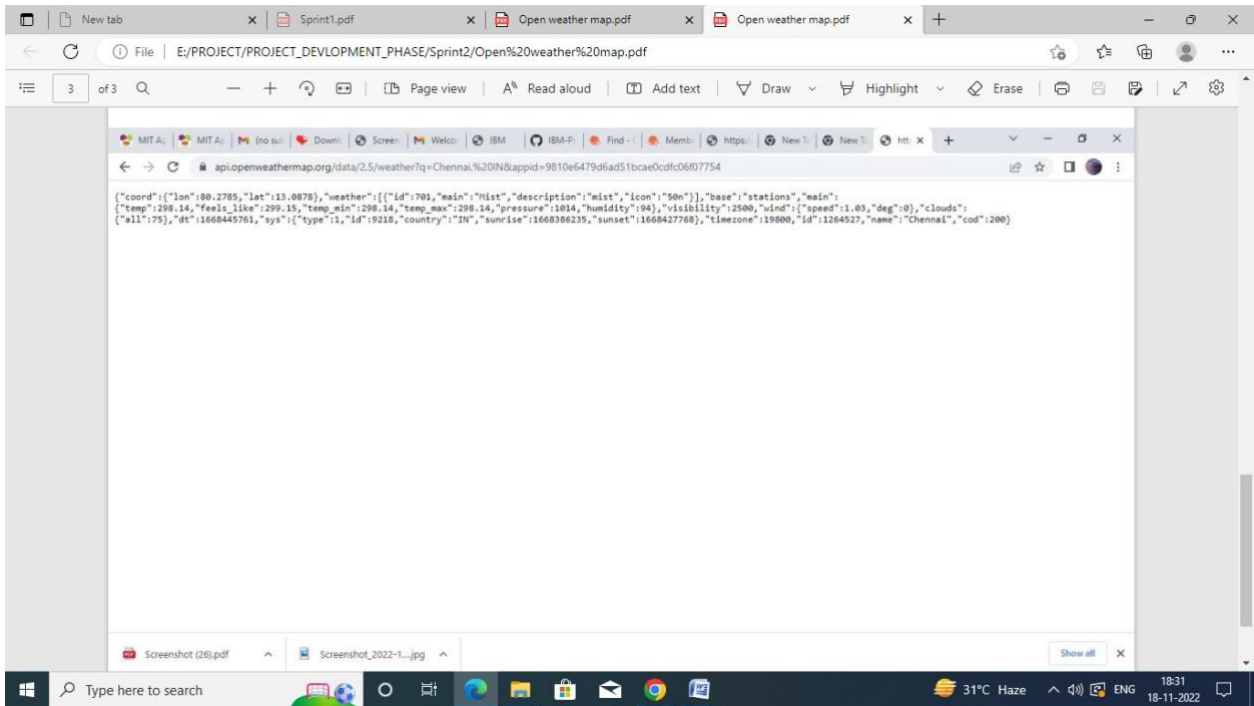
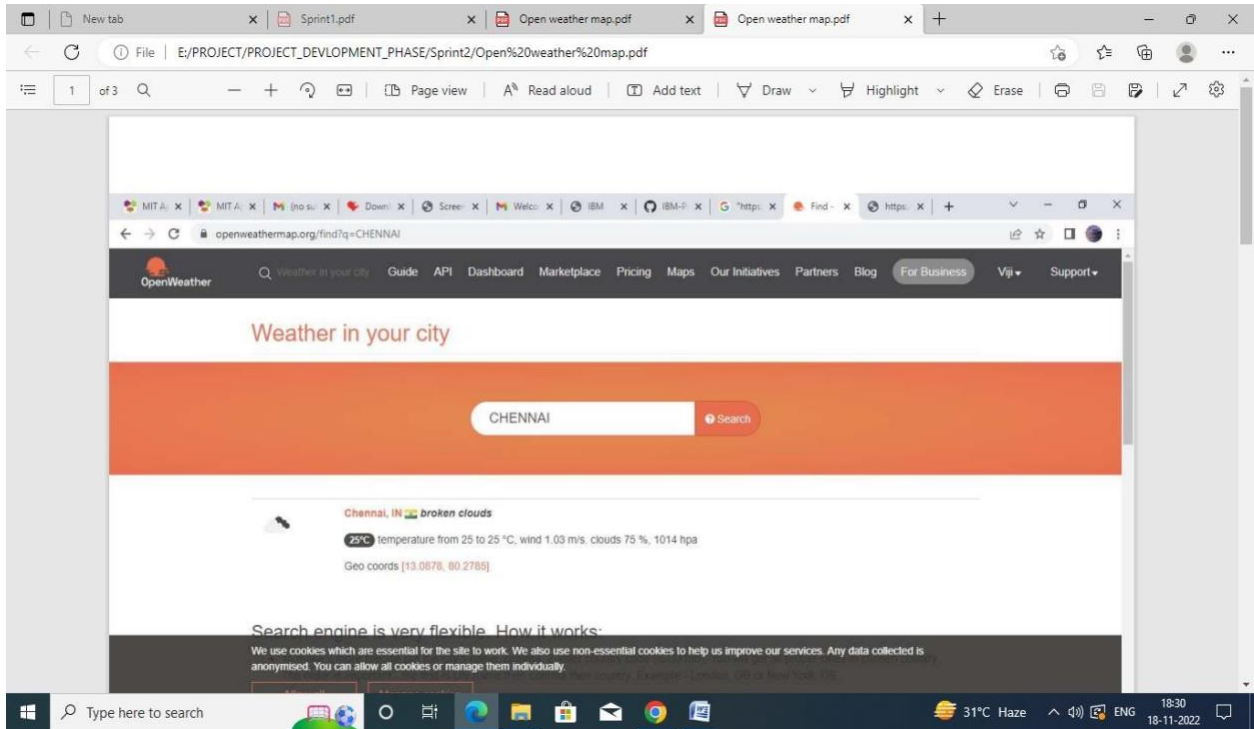
### 6.2 Sprint Delivery Schedule

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	5 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

## 7. CODING & SOLUTIONING (Explain the features added in the project along with code) 7.1

### Feature1

## OPEN WEATHER MAP



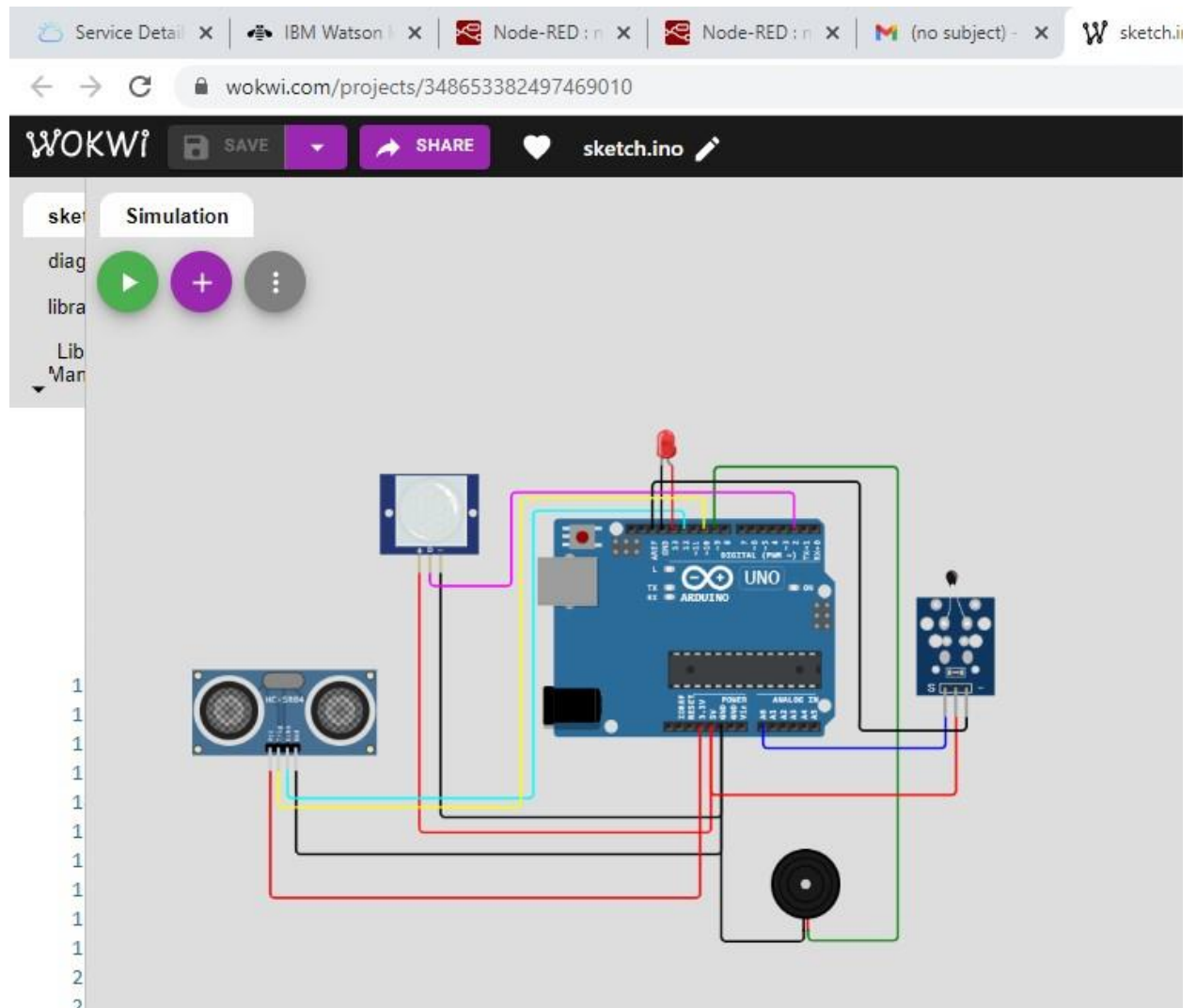
## Coding

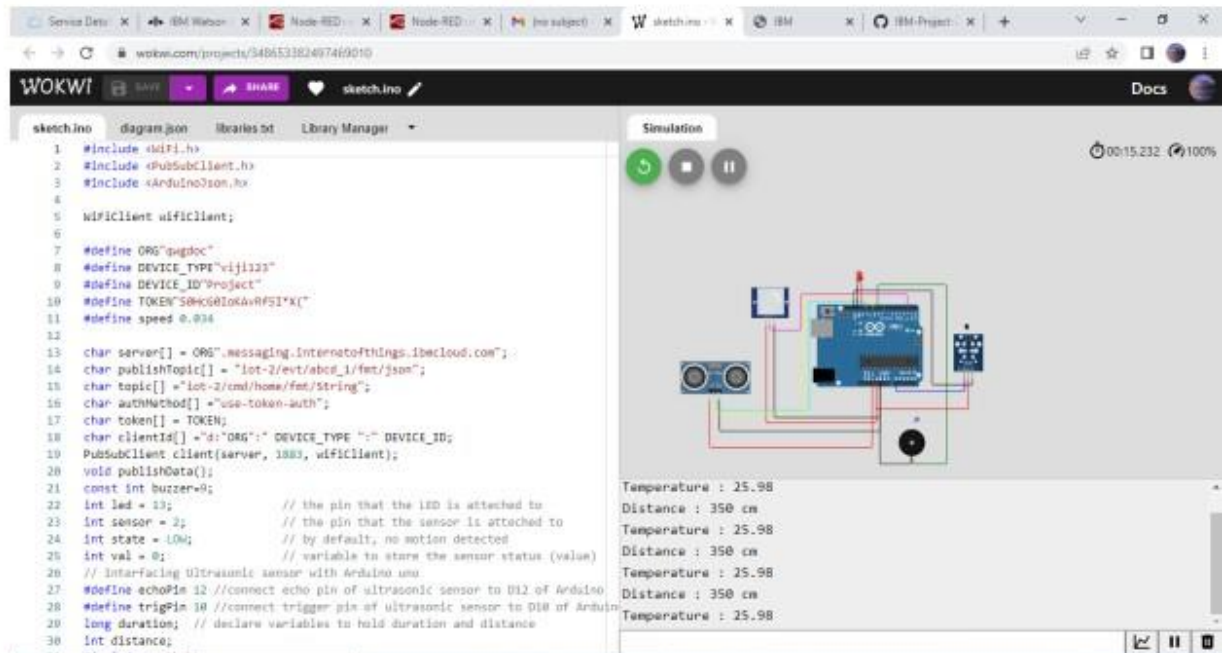
```
import requests #importing a library #replace
the url and it should be in ""
a="https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=bc453a0b339cb9ee1ad10
d2dd64d0bc0"
r=requests.get(url=a)

print(r)
```

## 7.2 Feature2

### Wokwi





## Coding

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

```
WiFiClient wifiClient;
```

```
#define ORG"qwgdcc"
#define DEVICE_TYPE"viji123"
#define DEVICE_ID"Project"
#define TOKEN"S0HcG0IoKAvRf5I*X("
#define speed 0.034
```

```
char server[] = ORG".messaging.internetofthings.ibmcloud.com"; char
publishTopic[] = "iot-2/evt/abcd_1/fmt/json"; char topic[] = "iot-
2/cmd/home/fmt/String"; char authMethod[] = "use-token-auth"; char
token[] = TOKEN; char clientId[] = "d:"ORG":" DEVICE_TYPE ":"
DEVICE_ID; PubSubClient client(server, 1883, wifiClient); void
publishData(); const int buzzer=9; int led = 13; // the
pin that the LED is attached to int sensor = 2; // the
pin that the sensor is attached to int state = LOW; // by
default, no motion detected
int val = 0; // variable to store the sensor status (value)
// Interfacing Ultrasonic sensor with Arduino uno
#define echoPin 12 //connect echo pin of ultrasonic sensor to D12 of Arduino
#define trigPin 10 //connect trigger pin of ultrasonic sensor to D10 of Arduino
long duration; // declare variables to hold duration and distance int
distance;
#include <math.h>
```



```

int sensorPin = A0; // select the input pin for the potentiometer

double Thermistor(int RawADC) {
    double Temp;
    Temp = log(10000.0*((1024.0/RawADC-1)));
    Temp = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 * Temp * Temp ))* Temp );
    Temp = Temp - 273.15; // Convert Kelvin to Celcius
    //Temp = (Temp * 9.0)/ 5.0 + 32.0; // Convert Celcius to Fahrenheit
    return Temp;
}

// the setup function runs once when you press reset or power the board void
setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(buzzer, OUTPUT);
    // PIR Sensor    pinMode(led, OUTPUT); // initialize LED
    as an output    pinMode(sensor, INPUT); // initialize
    sensor as an input    Serial.begin(9600); // initialize
    serial
    pinMode(trigPin,OUTPUT); //set trigPin as output pin of Arduino
    pinMode(echoPin,INPUT); //set echoPin as output pin of Arduino

}

// the loop function runs over and over again forever void
loop() {    val = digitalRead(sensor); // read sensor
value    if (val == HIGH) { // check if the sensor
is HIGH    digitalWrite(led, HIGH); // turn LED ON
        delay(500); // delay 100 milliseconds

        if (state == LOW) {
            Serial.println("Motion detected!");    state =
HIGH; // update variable state to HIGH
        } } else {    digitalWrite(led, LOW); //
turn LED OFF    delay(500); // delay 200
milliseconds
        if (state ==
HIGH){
            Serial.println("Motion stopped!");    state =
LOW; // update variable state to LOW
        }
    }

    digitalWrite(trigPin,LOW); //generate square wave at trigger pin
    delayMicroseconds(2);    digitalWrite(trigPin,HIGH);
    delayMicroseconds(10);    digitalWrite(trigPin,LOW);
    duration=pulseIn(echoPin,HIGH); //calculation of distance of obstacle
    distance=(duration*0.034/2);    Serial.print("Distance : ");
    Serial.print(distance);
    Serial.println(" cm ");
    delay(1000);    int
    readVal=analogRead(sensorPin);
    double temp = Thermistor(readVal);
    Serial.print("Temperature : ");
    Serial.println(temp); // display tempature

```

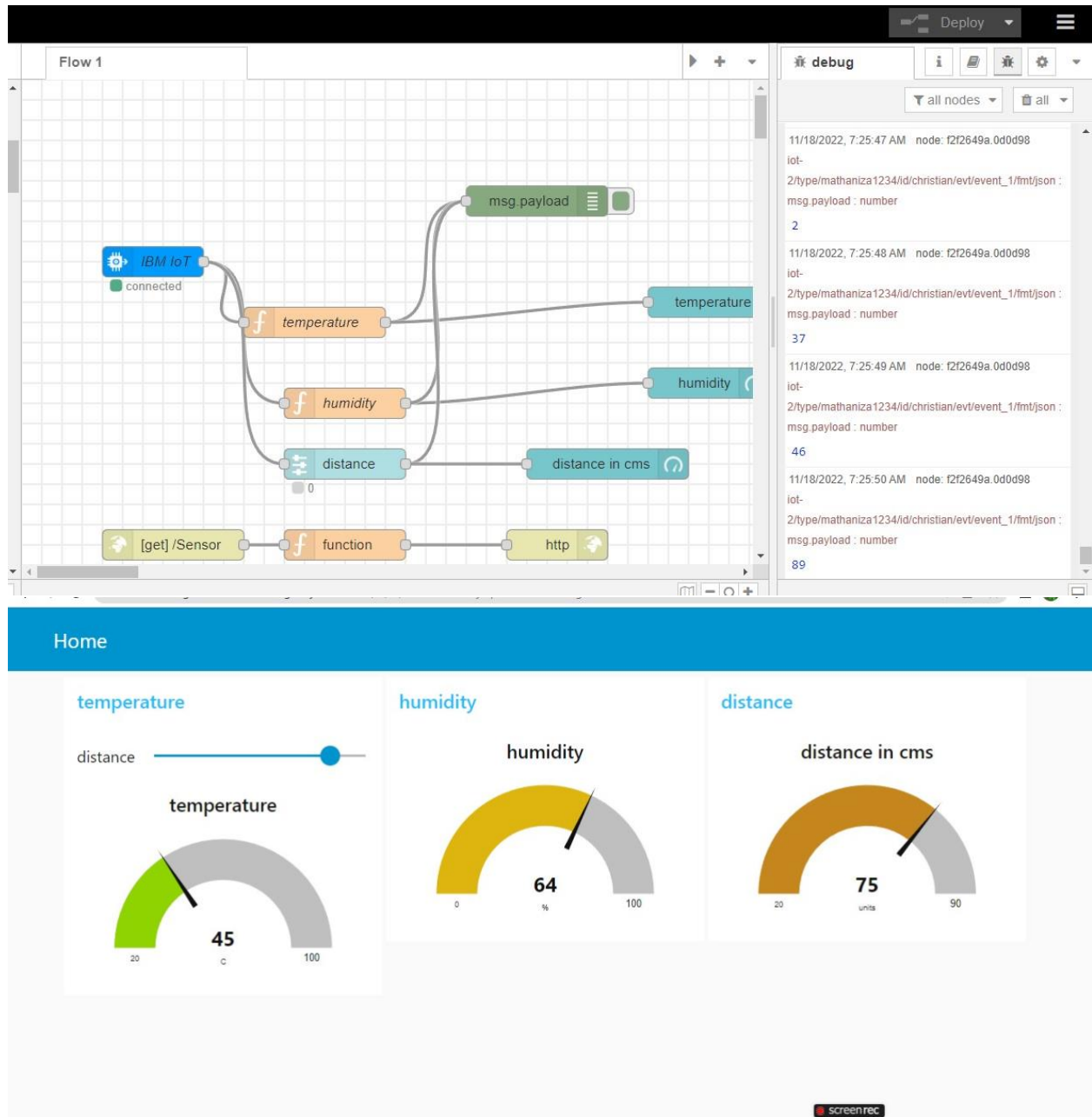
```

//Serial.println(readVal); // display temperature
    delay(500); digitalWrite(buzzer, HIGH); // turn the LED on (HIGH is the
voltage level) delay(1000); // wait for a second
digitalWrite(buzzer, LOW); // turn the LED off by making the voltage LOW
delay(1000); // wait for a second
}

```

## 8. TESTING

### 8.1 Test Cases





## IBM Cloud output

The screenshot displays the IBM Watson IoT Platform dashboard. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. The 'Recent Events' tab is selected, showing a table of live data streams. The table has four columns: 'Event', 'Value', 'Format', and 'Last Received'. Five rows of data are visible, each representing an event with a unique ID and a JSON payload containing temperature, humidity, and distance values. A status bar at the bottom indicates '1 Simulation running'. The Windows taskbar at the bottom shows the system time as 7:06 PM on 11/18/2022.

Event	Value	Format	Last Received
event_1	{"temperature":97,"humidity":95,"distance":63}	json	a few seconds ago
event_1	{"temperature":81,"humidity":74,"distance":86}	json	a few seconds ago
event_1	{"temperature":36,"humidity":59,"distance":74}	json	a few seconds ago
event_1	{"temperature":33,"humidity":41,"distance":21}	json	a few seconds ago
event_1	{"temperature":32,"humidity":25,"distance":72}	json	a few seconds ago

## 8.2 User Acceptance Testing

### 9.1 Performance Metrics

**Mobile view of user**

10:04

2.00 KB/S 4G 54

## SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

SIGNS WITH SMART CONNECTIVITY BETTER ROAD SAFETY

TEMPERATURE

31

HUMIDITY

38

DISTANCE

44

Motor ON

Motor OFF

## 10.ADVANTAGES AND DISADVANTAGES

Advantages	Disadvantages
Minimizes the human work and effort	Increased privacy concerns
Saves time and effort	Increased unemployment rates
Good for personal safety and security	Highly dependent on the internet
Useful in traffic and other tracking or monitoring systems	Lack of mental and physical activity by humans leading to health issues.
Beneficial for the healthcare industry	Complex system for maintenance
Improved security in homes and offices	Lack of security
Reduced use of many electronic devices as one device does the job of a lot of other devices	Absence of international standards for better communication

## 11.CONCLUSION

We have presented a system, to alert the driver about the speed limits in specific areas and reduce the speed of the vehicles in sensitive public zones without any interference of the drivers where controls are taken automatically by the use of a wireless local area network. In the initial phase, we designed the basic block and circuit diagram for the system. In the implementation phase, we executed the Stimulation with the help of IoT connecting technologies such as MIT APP. Extensive experiments conducted on IoT and other connecting technologies.

## 12.FUTURE SCOPE

We can be enhanced this system by implementing camera using Raspberri pi, GSM module in case of network unavailability and low RAM module/zigbee module for long range communication.

## 13.APPENDIX Source code

```
import time
import sys
import ibmiotf.application
import ibmiotf.device

#provide your ibm watson Device
credentials    organization = "vd4xhx"
devicetype="mathaniza1234"
deviceid="christian" authmethod="token"
authtoken="JLJdQ8p?5Rizji2Xa"
```

```

#Initialize GPIO
def
myCommandCallBack(cmd):
    print("Command received: %s"% cmd.data('command'))
print(cmd)

/*
    PIR sensor tester
*/

(9600);
} void loop() {int ledPin = 13;           // choose the pin for the
LED int inputPin = 2;                     // choose the input pin (for PIR
sensor) int pirState = LOW;              // we start, assuming no motion
detected int val = 0;                    // variable for reading the pin
status
    void setup() {    pinMode(ledPin, OUTPUT);    // declare
LED as output    pinMode(inputPin, INPUT);    // declare
sensor as input
    Serial.begin    val = digitalRead(inputPin); // read input
value    if (val == HIGH) {                // check if the input
is HIGH    digitalWrite(ledPin, HIGH); // turn LED ON
if (pirState == LOW) {
    // we have just turned on
    Serial.println("Motion detected!");
    // We only want to print on the output change, not state
pirState = HIGH;
    } } else {    digitalWrite(ledPin,
LOW); // turn LED OFF    if (pirState ==
HIGH) {
    // we have just turned of
    Serial.println("Motion ended!");
    // We only want to print on the output change, not state
pirState = LOW;
    }
    }
}
}

import
json
#replace the url and it should be in""
a="https://api.openweathermap.org/data/2.5/weather?q=chennai,%20IN&appid=bc453a0b339cb9ee1ad10d2dd64d0bc0"
r=requests.get(url=a) print(r) data =r.json () tem=data ('main')('h')          print(tem)

/*
    HC-SR04 Ultrasonic Sensor Example.

```

Turn the LED on when an object is within 100cm range.

```
Copyright (C) 2021, Uri Shaked
*/

#define ECHO_PIN 2
#define TRIG_PIN 3

void setup() {
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
} float readDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW); int
  duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
} void loop() { float distance =
  readDistanceCM();
  bool isNearby = distance < 100;
  digitalWrite(LED_BUILTIN, isNearby);
  Serial.print("Measured distance: ");
  Serial.println(readDistanceCM());

  delay(100);
}
```

DEMO using the server test.mosquitto.org

You can use any MQTT client with the following  
settings Server : test.mosquitto.org no login / no  
password  
port: 1883 or 8081 for websocket  
Topic: /AnnexTest  
Subscribe: /AnnexTx

Or you can use the free MQTT online client

<https://www.cicciocb.com/MQTT/>  
this is already configured so  
just

- click on Connect
- write your message in "Publish Message"
- Press "Publish" to send your message that will be shown in the scrolling display
- Click on Subscribe to receive the temperature sensor data

```
# Disconnect the device and the application from the  
cloud devicecli.disconnect()
```

**GITHUB LINK:** <https://github.com/IBM-EPBL/IBM-Project-20790-1659763214> DEMO

**LINK:** [https://drive.google.com/file/d/1DTZk3AGML1iaFbhNa-Tm3wNQ6xPG6bbV/view?usp=share\\_link](https://drive.google.com/file/d/1DTZk3AGML1iaFbhNa-Tm3wNQ6xPG6bbV/view?usp=share_link)