PROJECT DOCUMENTATION

Date	12 November 2022
Team ID	PNT2022TMID03169
Project Name	Signs with Smart Connectivity for Better Road Safety

1. INTRODUCTION

1.1 Project Overview:

The project aims to replace the static signboards with smart connected sign boards which can be changed frequently according to the purposes like weather, traffic condition, particular zones, etc.., These smart connected sign boards get the speed limitations for a particular city from a web app using weather API and update automatically. Based on the weather changes the speed may increase or decrease. So, the signboards will display “slow down”, if the weather is bad. Based on the traffic and fatal situations the diversion signs are displayed. The driver can change the location pins in the map to their current location and destination location. The app shows the route and navigation instructions like “turn left on abc road”, “take U turn”. Using the location sensor, it can sense the speed of the vehicle. Sign boards near school zone, hospital zone, construction zone, uneven and narrow roads, animal zone should display appropriate signs according to the zone. Different modes of operations can be selected with the help of buttons.

1.2 Purpose:

- 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.

2. LITERATURE SURVEY

2.1 Existing Problem:

Increased traffic can increase carbon emissions and other pollution.
Land use for roads can damage built and natural environment, impose mortality on wildlife if habitats are severed.

construction has associated environmental costs

2.2 References:

W.E.Marshall,"Understanding international road safety disparities:Why is Australia so much safer than the United States?" Accident Analysis & Prevention, vol. 111, pp. 251–265, 2018

Digital signage is evolving to smart signage which provides personalized service by adaptively changing contents according to the user context. However, previous smart signage services have difficulty to expend their service because it is not easy to connect additional sensing devices. Furthermore, previous smart signage systems only consider single signage for a service. In this paper, we propose IoT based smart signage platform. The proposed platform provides IoT based connectivity between sensors and signage platform for flexible service extension. Also, we suggest IoT based signage connection, status sensing, and controlling. Therefore, our platform can make a service group of signages dynamically and enables signages to collaborate for a service in wide area. To show the performance of proposed platform, we implemented smart nursing home service. The service shows that IoT devices and signages can be connected to the platform dynamically and collaborate together for a service in wide area.

This paper presents an approach to detect traffic signs using You Only Look Once version 4 (YOLOv4) model. The traffic sign detection and recognition system (TSDR) play an essential role in the intelligent transportation system (ITS). TSDR can be utilized for driver assistance and, eventually, driverless cars to reduce accidents. When driving an automobile, the driver's attention is usually drawn to the road. On the other hand, most traffic signs are situated on the side of the road, which may have contributed to the collision. TSDR allows drivers to view traffic sign information without having to divert their attention. Due to the existence of a large background, clutter, fluctuating degrees of illumination, varying sizes of traffic signs, and changing weather conditions, TSDR is an important but difficult process in intelligent transport systems. Many efforts have been made to find answers to the major issues that they face. The objective of this study addresses road traffic sign detection and recognition using a technique that initially detects the bounding box of a traffic sign. Then the detected traffic sign will be recognized for usage in a speeded-up process. Since safe driving necessitates real-time traffic sign detection, the YOLOv4 network was employed in this research. YOLOv4 was evaluated on our dataset, which consisted of manual annotations to identify 43 distinctive traffic signs classes. It was able to achieve an average recognition accuracy of 84.7%. Overall, the work adds by presenting a basic yet effective model for real- time detection and recognition of traffic signs.

Ubiquitous nature of smart cities requires multiple technologies to be implemented in this area. To develop the smart cities in practice, there is huge need of “Smart Traffic Management”. Smart Traffic Management is a system to monitor and control the traffic signals using sensors to regulate the flow of traffic and to avoid the congestion for smooth flow of traffic. Prioritizing the traffic like ambulance, police etc. is also one application comes under smart traffic management. Traffic sign board plays important

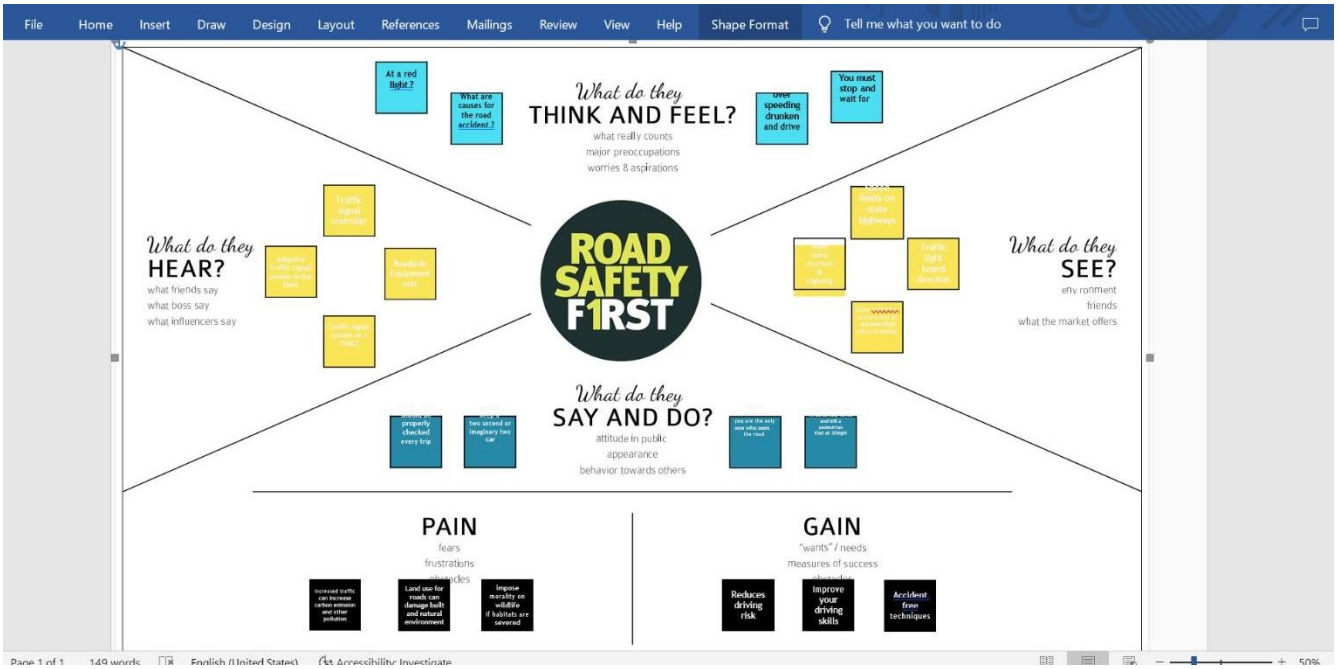
role to make the traffic in shape and to control and manage the traffic on roads. Many at times the driver misses the sign boards while driving due to various reasons like insufficient light, fog, rain, traffic etc. In this paper, a framework of the Smart Traffic Sign Boards (STSB) is proposed, which can communicate with the system deployed in all the vehicles to make the drivers of those vehicles aware of speed breakers, speed limits, schools, or 'U' turn ahead, etc. beforehand, to avoid the mishap due to sudden appearing of such unusual features of the road during the road journey.

2.3 Problem Statement Definition:

- A driver who wants to drive safely on road but there are many obstacles because of heavy traffic, weather condition, etc..,
- A driver who wants to avoid the heavy traffic roads but they are unpredictable because they change from time to time.
- A passenger who wants to travel safely but there are many road accidents because of some drivers who drive very fast and carelessly.
- A driver who wants to reach the destination but unable to choose the route and turn in wrong direction because there are no navigation instructions.

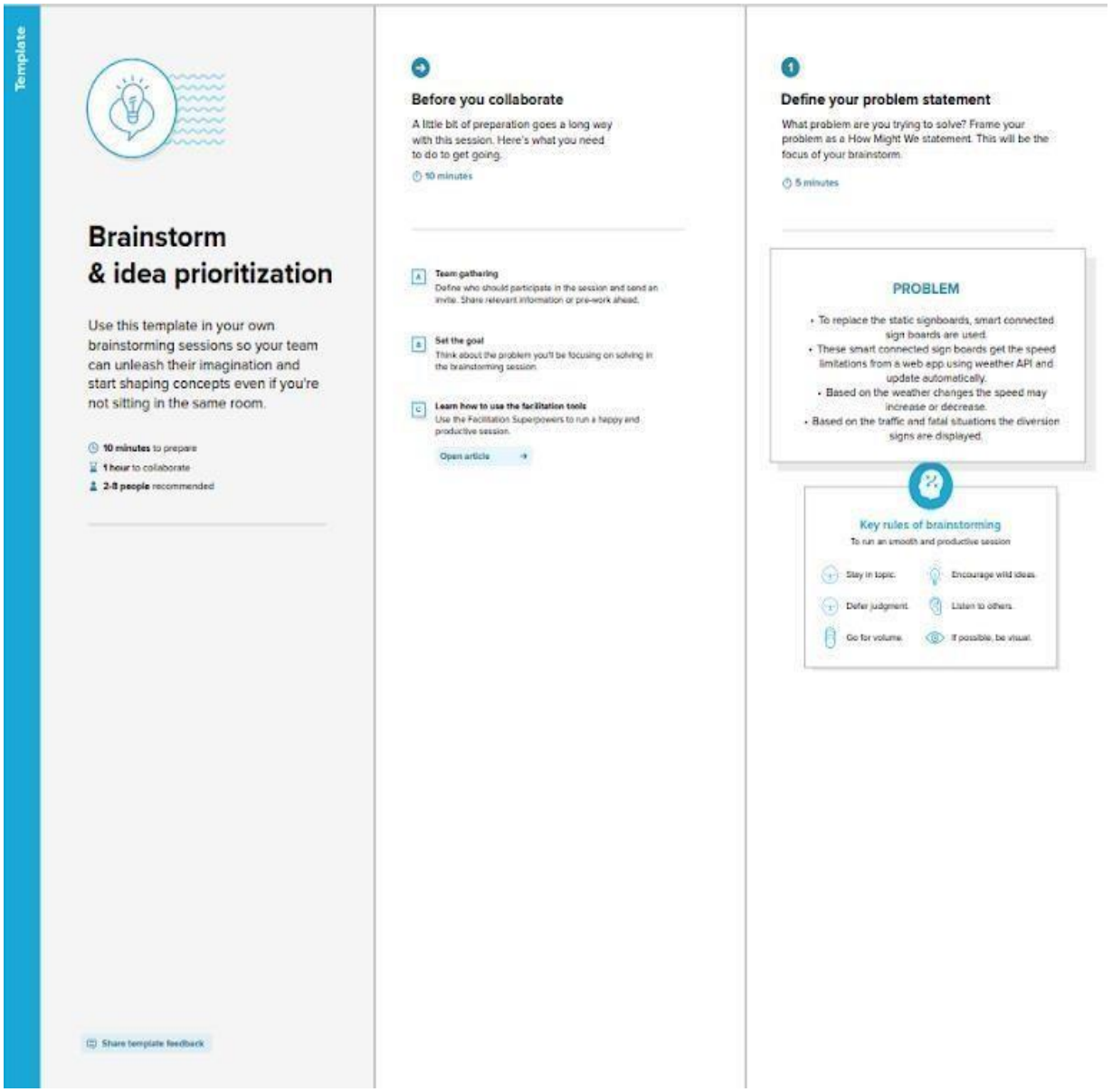
3. IDEATION AND PROPOSED SOLUTION:

3.1 Empathy Map Canvas:

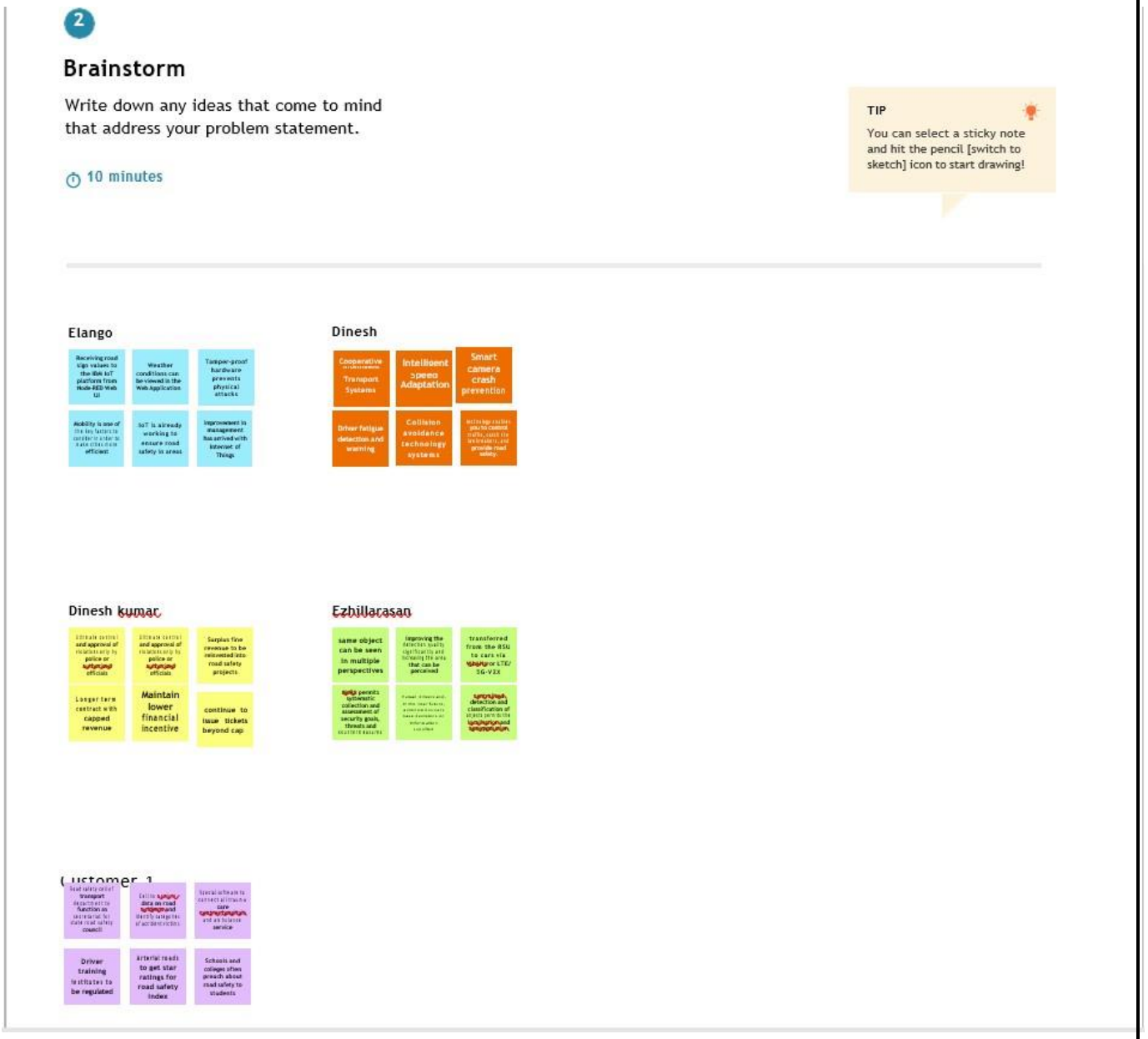

File
Home
Insert
Draw
Design
Layout
References
Mailings
Review
View
Help
Shape Format
Tell me what you want to do
ROAD SAFETY F1RST
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
At a red light ?
What are causes for the road accident ?
You must stop and wait for
What do they HEAR?
what friends say
what boss say
what influencers say
What do they SEE?
env ronment
friends
what the market offers
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
PAIN
fears
frustrations
GAIN
"wants" / needs
measures of success
Reduces driving risk
Improve your driving skills
Accident free techniques
Land use for roads can damage built and natural environment
Impose morality on wildlife if habitats are severed

3.2 Ideation & Brainstorming:

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



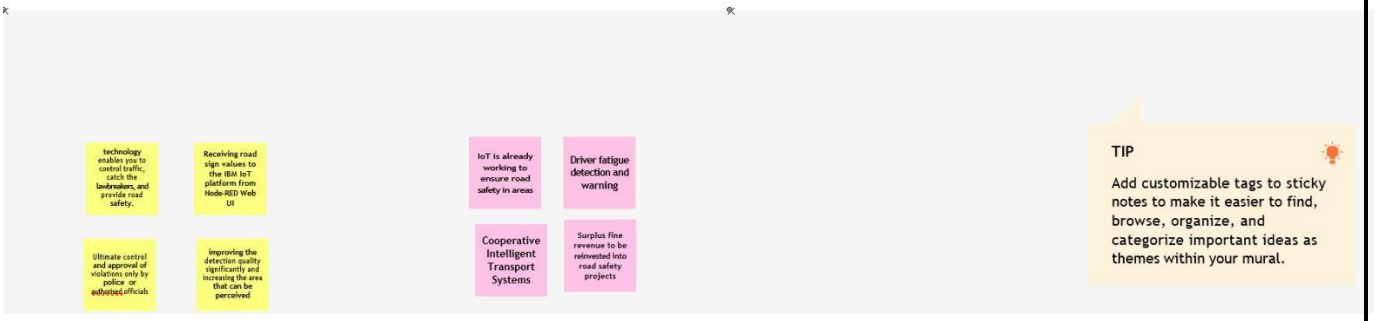
Step-3: Idea Prioritization

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, 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



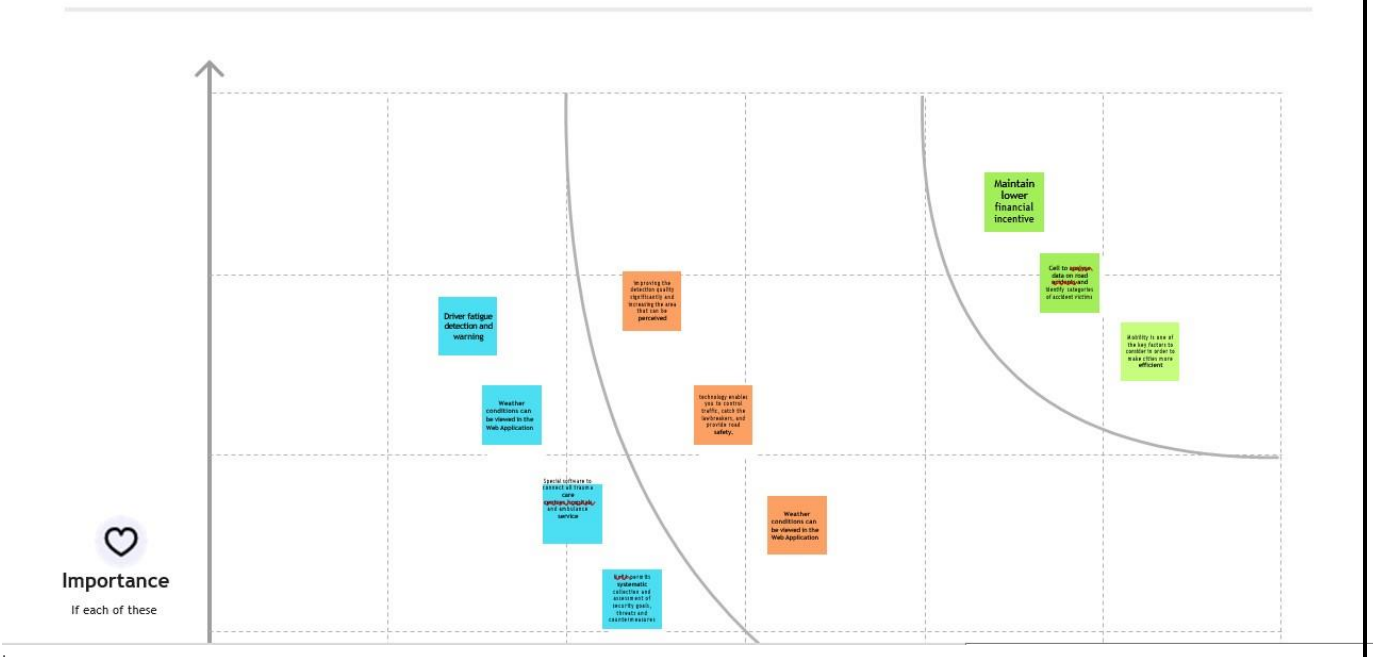
Prioritize:

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

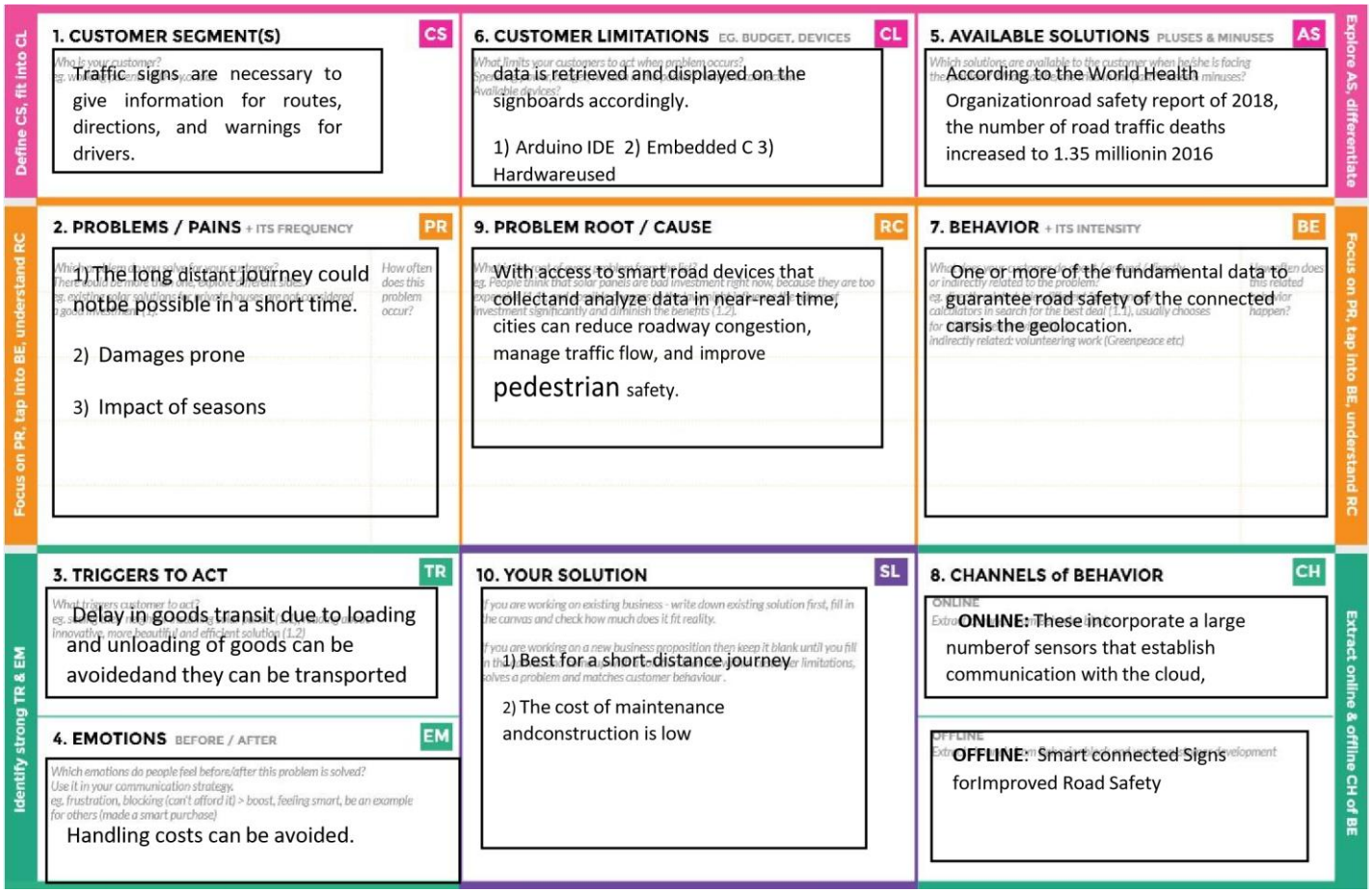


3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Fire Brigade dealing with fire incidents related to accidents on the roads Hospitals dealing with attending to accident victims Drivers related to driving behavior mind set, driving licenses, offences, and penalties
2.	Idea / Solution description	Road safety involves the management of roads, cars, and drivers and key actors are involved in this process which is traffic police, Tanzania Roads Agency ANROAD, Fire Brigade, TRA, SUMATRA, and Traffic Police interact in one way or another in their daily routine handling of traffic offences
3.	Novelty / Uniqueness	Enforcement has therefore been a problem since information on accident incidents rely on that collected by a sole police officer manually when they arrive at the scene of the accident physically The existing systems hosted by different stakeholders do not see each other since most of their specification were vendor-driven without integration focus even for database systems under one entity
4.	Social Impact / Customer Satisfaction	Multimodal sensors and edge computing help speed up the flow of traffic with real-time processing, reducing congestion and emissions. Smart road technology can assist in optimizing traffic flow and managing road conditions, creating a more sustainable environment within cities
5.	Business Model (Revenue Model)	Connected vehicles could dramatically reduce the number of fatalities and serious injuries caused by accidents on our roads and highways potential of connected vehicles, they also promise to increase transportation options and reduce travel times

6.	Scalability of the Solution	The study has shown that every part of the community is affected in one way or another by road safety, be it in the developed or in developing countries Close observations made in some of the areas that are prone to violations that result in accidents has shown that enforcement has a much higher impact in addressing road accident problems than increasing fines several folds.

3.4 Problem Solution Fit:



4. REQUIREMENT ANALYSIS:

4.1 Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)

FR-1	USER REGISTRATION	• Through google forms • Through mail • Through linkedin • Through facebook
FR-2	USER CONFIRMATION	• Through verification mails • Through OTP
FR-3	USER APPROVAL	• Through mails • Through phone calls • Through SMS
FR-4	USER TRANSACTION	• Through net banking • Through UPI
FR-5	TESTING	• Testing through components • Testing through API and UI
FR-6	END RESULT	• End result through product features • By using the technology

4.2 Non-Functional Requirements:

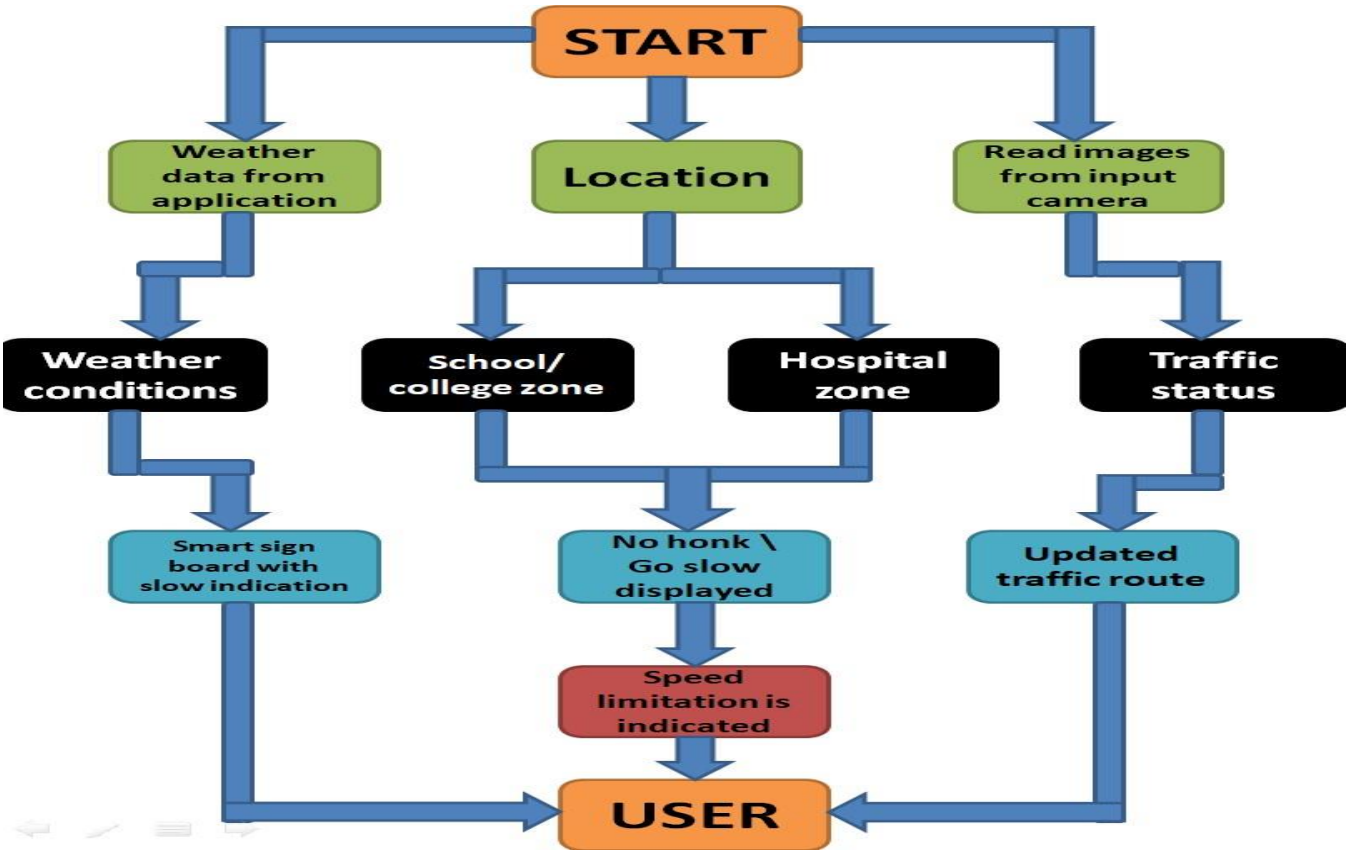
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	• Situations never remain the same. Therefore there must be a constant check of the conditions prevailing and accordingly there must be changes made in these boards. • Sign boards with caution or alerts must be placed well in advance so that the drivers could be more alert with the journey. • The text content must be available in different languages to help the drivers.
		• Boards must be large and clear for better visibility.

		• Sign boards should be bright coloured so that it catches the drivers' sight. • The illustrations or the symbols used in the boards must be easily understandable.
NFR-2	Security	• The security system should be strong enough that no one can modify it other than the authority. • No one should be able to enter into the network to change, delete or manage the intimations or messages delivered through the sign boards.
NFR-3	Reliability	• There should not be any miscommunications or confusions regarding the messages displayed. • Maximum accuracy must be ensured. • All the information displayed must be checked periodically and updated if any changes are needed for error-free intimation.
NFR-4	Performance	• The efficiency and the accuracy of the information hence calculated should be maximum. • It should be ensured that minimum amount of energy, time and cost is required for the operation.
NFR-5	Availability	• These should be available anytime and everywhere that is 24/7. • Sign boards should be located in places which has direct view from the road. It should not present amidst bushes, trees, building etc • It should be properly monitored that no sign boards are damaged, repaired or malfunctioning at any time.

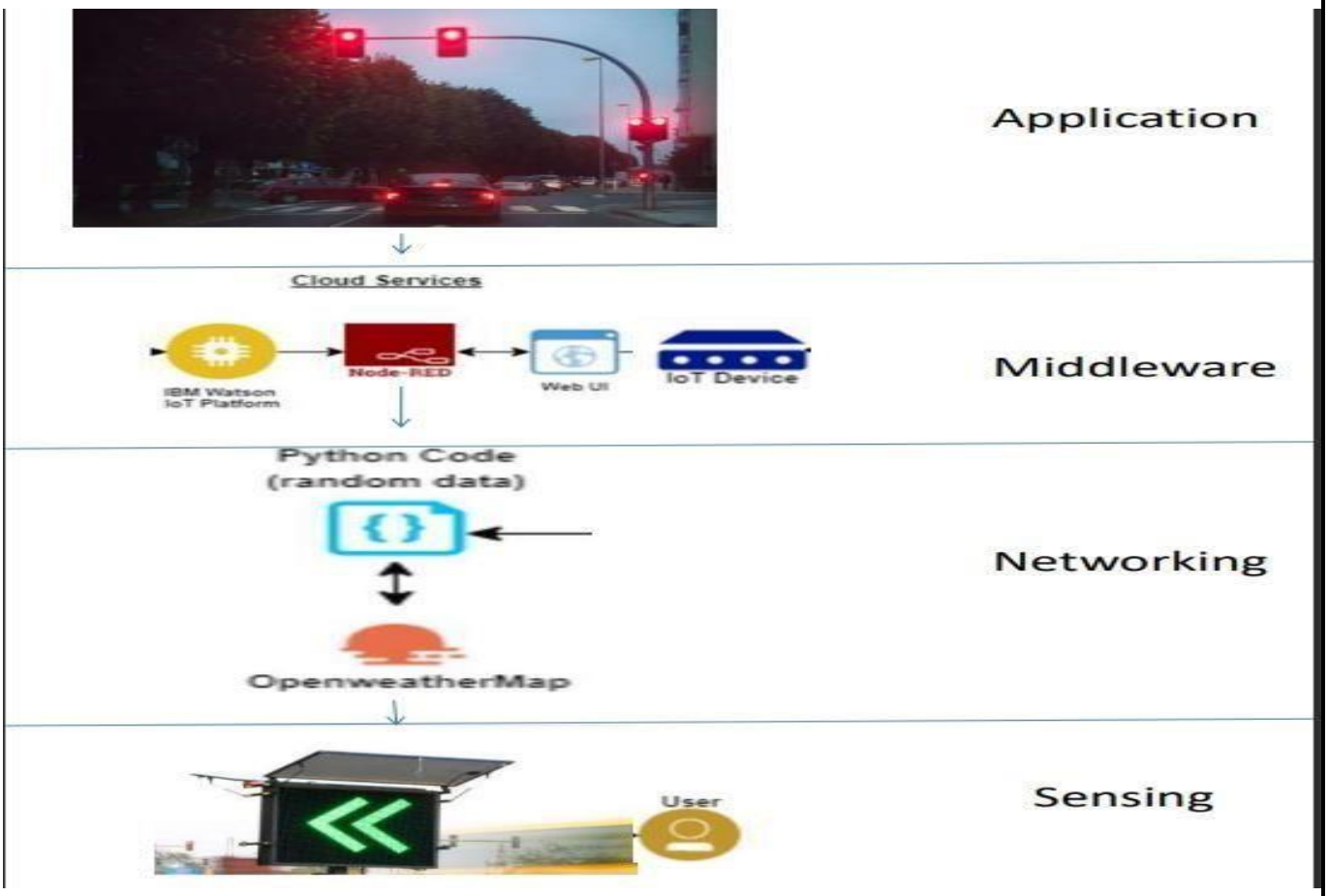
		• The sign boards should be made available only in places where they are required the most. Frequent availability of boards may lead to confusion and mistakes.
NFR-6	Scalability	• It should be easy to scale according to the requirement. • It should be in such a way that the network at any time of period should be ready to be expanded and implemented on a wider scale.

5. PROJECT DESIGN:

5.1 Data Flow Diagrams:



5.2 Solution & Technical Architecture:



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Visibility**	Sign Boards should be made with LED's whichare bright colored and are capable of attractingthe drivers attention but it should also not be too distracting or blinding cause it may lead to accidents.
FR-2	**User Need**	The smart sign boards should be placed frequentlyin places it is needed and less in places where it isnot needed much to avoid confusion for the user during travel.
FR-3	**User Understanding**	For better understanding of the driver, the signs should be big, clear and legible and it can also include illustrations which will make it easily understandable to the driver.
FR-4	**User Convenience**	The display should be big enough that it should even be visible from far distance clearly.

5.3 User Stories:

PROJECT PLANNING & SCHEDULING:

5.4 Sprint Planning & Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As a weather data controller, I log into my profile and start monitoring the weather updates	3	High	MANOJ
Sprint-1	dashboard	USN-2	I receive all the information about weather at a particular city from web from	2	High	MANOJ

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			weather API. Whenever there is change in weather, corresponding updates about speed limits are made on sign boards.			
Sprint-2	Login	USN-1	As a image controller, I keep note of all the images received from various areas and detect traffic in that particular area.	3	High	SAMUAL
Sprint-2	Dashboard	USN-2	With traffic, distance between the vehicles is detected by ultrasonic sensor and the vehicle will be automatically stopped if the distance is below the limit.	2	Medium	SAMUEL
Sprint-3	Login	USN-1	As a traffic controller, I keep note of all the vehicle's speed received from various areas using location sensor.	2	High	SABARISH
Sprint-3	Dashboard	USN-2	I ensure that the boards display "slow down" if high speed is detected.	2	Medium	SABARISH
Sprint-3	Login	USN-3	As a user, I move the marker to my current location and the destination location.	1	Medium	SABARISH
Sprint-3	Dashboard	USN-4	I receive the fastest route to the destination and navigation instructions like "Turn left", "Turn right" will be displayed.	1	Medium	JEYAN VIGNESH

Sprint-4	Login	USN-1	As a zonal officer, I ensure that boards near school display “slow down” and near hospitals display “no horn”.	3	High	MANOJ

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Login	USN-2	As an administrator, I ensure that the boards display “drive carefully” near construction site, narrow and uneven roads.	2	Medium	MANOJ

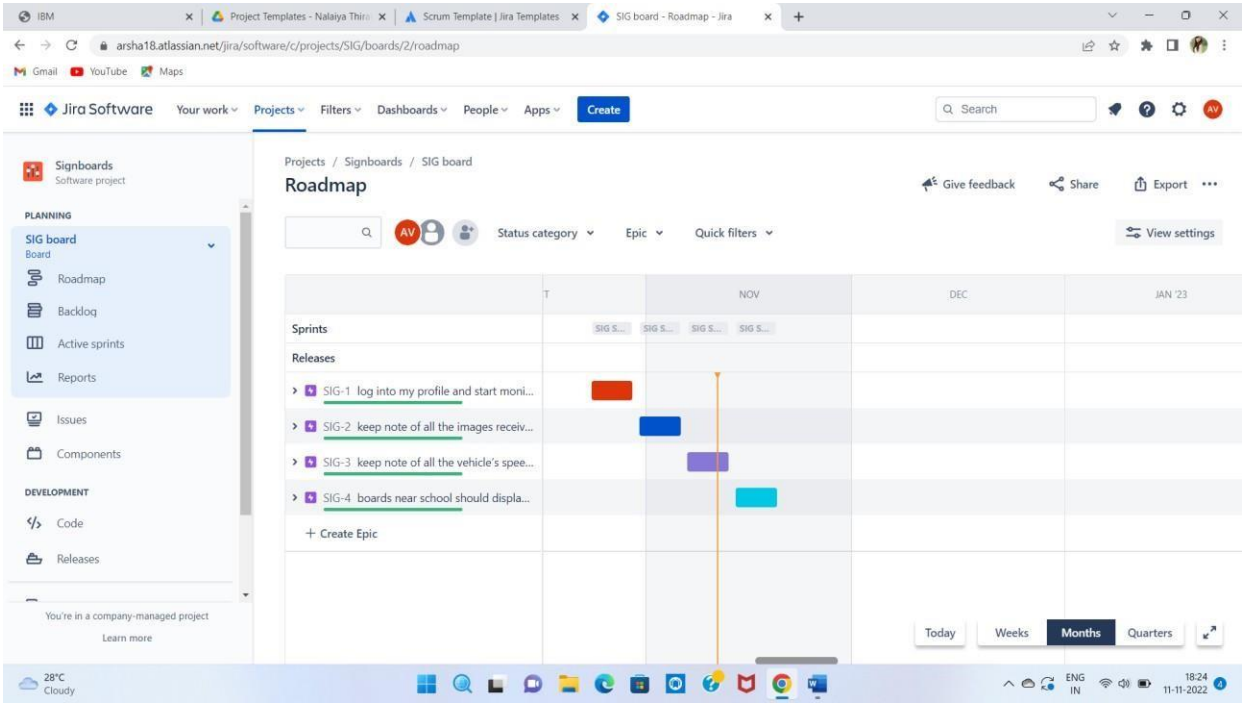
5.5 Sprint Delivery Schedule:

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

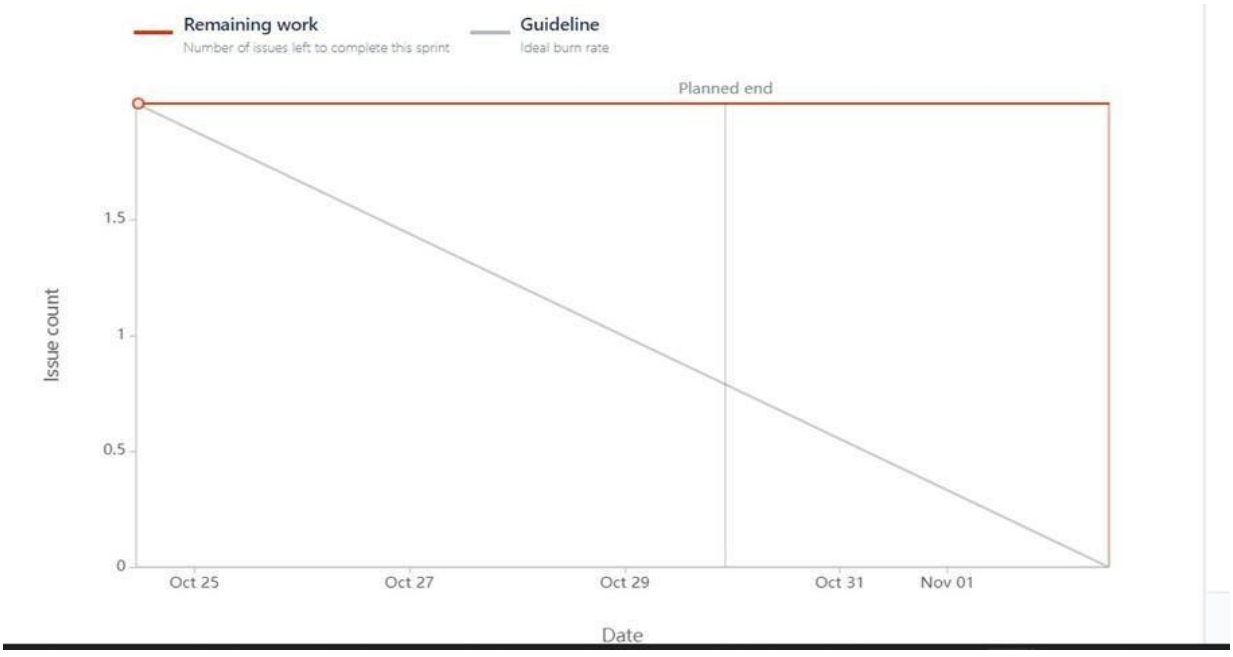
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

5.6 Reports From JIRA:

ROADMAP:



BURNDOWN CHART:



6. CODING & SOLUTIONING:

Code Explanation:

Libraries:

Including all libraries like json, random, time, sys, ibmiotf etc.

PROJECTFINALDND.py - D:/1ibm/PROJECTFINALDND.py (3.7.0)

File Edit Format Run Options Window Help

```
import requests #importing a library
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
```

Credentials:

Entering all the credentials corresponding to IoT watson device in order to publish data to it.

```
# watson device details

organization = "2s7yy7"
devicType =  "project"
deviceId = "projectid"
authMethod= "token"
authToken= "projecttoken"
```

MIT Inventor Interruption:

Receiving commands as inputs when buttons are pressed in MIT inventor in order to perform separate functions.

```
def myCommandCallback(cmd):
    global a
    #print("command recieved:%s" %cmd.data['command'])
    #status=cmd.data['command']
    print("command recieved:%s" %cmd.data['command'])
    control=cmd.data['command']
    print(control)


try:
        deviceOptions={"org": organization, "type": devicType,"id": deviceId,"auth-method":authMethod,"auth-token":authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
        print("caught exception connecting device %s" %str(e))
        sys.exit()
```

Exception Handling:

To handle exception if occurs while connecting with IBM IOT WATSON device

```
try:
        deviceOptions={"org": organization, "type": devicType,"id": deviceId,"auth-method":authMethod,"auth-token":authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
        print("caught exception connecting device %s" %str(e))
        sys.exit()
```

Main Body:

- Connecting to IBM IoT device.
- Getting temperature and humidity values in json format from openweathermap as inputs.
- Accessing the values using their corresponding keys.
- Generating random values for distance since hardware sensors are not implemented.
- Passing a warning “stating please slow down” when humidity is less than 100 in order to promote safe driving experience.
- Passing instruction when distance is less than 20 in order to avoid accidents and clashes.

PROJECTFINALDND.py - D:/1ibm/PROJECTFINALDND.py (3.7.0)

File Edit Format Run Options Window Help

```
#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for every 10 seconds
deviceCli.connect()

while True:

#get sensor data from DHT11



    a = "https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=e2bea247ed9ad643a04d9a8e55499d5f"
    r=requests.get(url=a)
    data=r.json()

    Temp= data['main']['temp']
    Humd= data['main']['humidity']
    data= {'temp':Temp,'humid':Humd}
    dist=random.randint(0,50)
    dis={'dista':dist}

    if(Humd<100):
        warn={'alert':'PLEASE SLOW DOWN!!!!!!'}
    if(dist<20):
        insta={'inst':'stop'}
```

Publish Data To IBM IOT WATSON Platform:

Passing all the data(temperature, humidity, warning, instruction) to ibm iotwatson.

Disconnecting the connection established with IoT Watson device.

PROJECTFINALDND.py - D:/1ibm/PROJECTFINALDND.py (3.7.0)

File Edit Format Run Options Window Help

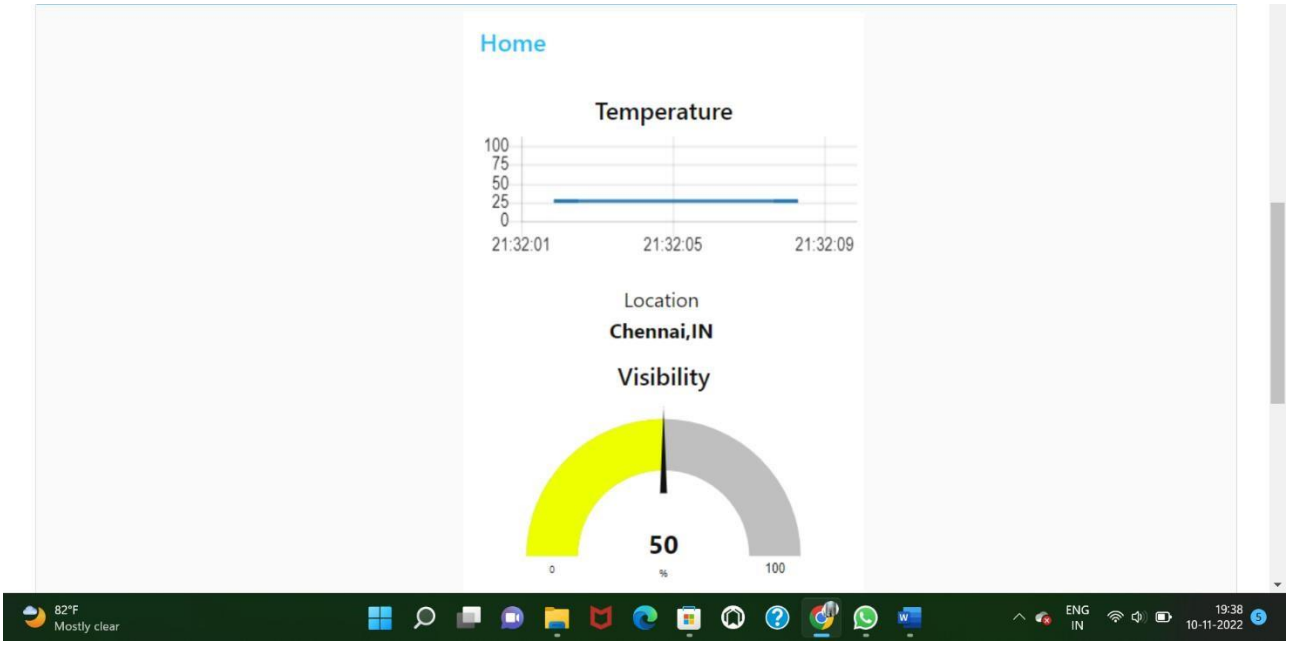
```
    data= {'temp':Temp,'humid':Humd}
    dist=random.randint(0,20)
    dis={'dista':dist}

    if(Humd<100):
        warn={'alert':'PLEASE SLOW DOWN!!!!!!'}
    if(dist<20):
        insta={'inst':'stop'}


    def myOnPublishCallback():

        print("published Temperature = %s c" %Temp,"humidity:%s %%" %Humd)
        print(warn)
        print(dis)
        print(insta)







    success=deviceCli.publishEvent ("IoTSensor","json",insta,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",data,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",warn,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",dis,qos=0,on_publish= myOnPublishCallback)

    if not success:
        print("not connected to ibmiot")
    time.sleep(5)



    deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect()
```

Ln: 87 Col: 16

6.1 Feature 1:

WEATHER UPDATE AND CORRESPONDING COMMAND:

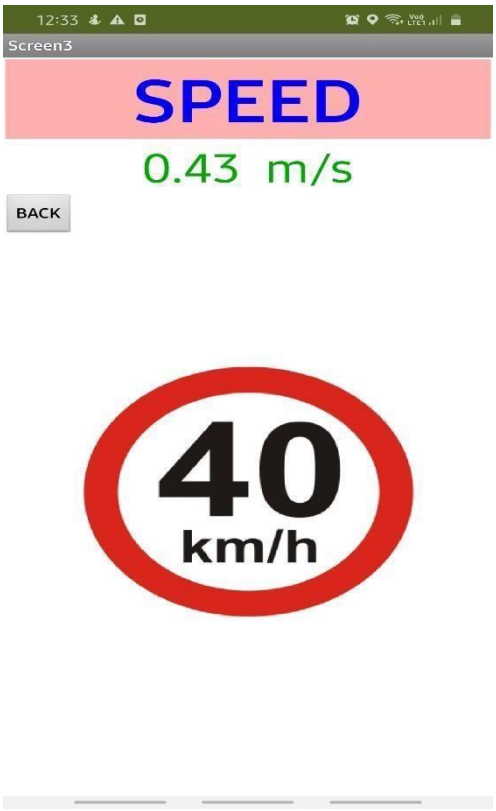
Getting temperature and humidity from OpenWeatherMap for a particular city and displaying warning regarding the speed



when humidity is below 100.

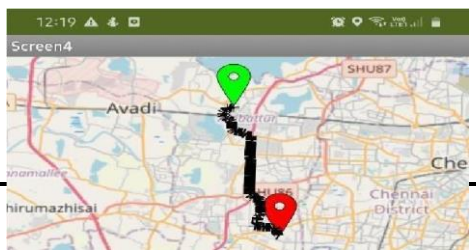
6.2 Feature 2:
SPEED DETECTION:

- By implementing a location sensor in MIT APP INVENTOR, with changes in the location with respect to time, speed can easily be detected and displayed in the app to the user.
- This requires location settings from user's phone to be active.
- An image of normal speed limit is also displayed which means that, travelling within that range would be safe.



6.3 Feature 3:
MAP AND NAVIGATION:

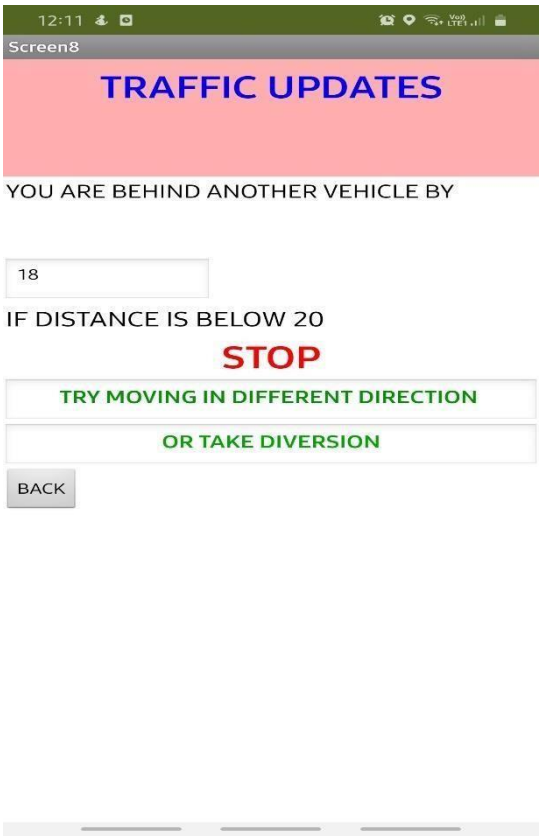
- By implementing same location sensor, current location can be detected. This



also requires location settings in user's phone to be active.

- By dragging the green marker to start location and red marker to the destination location to be reached and clicking on the navigate button, displays the street path that connects the start and end point specified.
- In addition to this, it also displays the directions to be followed to reach the destination.

- This helps the user by providing necessary details like current location and the best route to reach the destination.



6.4 Feature 4:

ZONAL CLASSIFICATION:

- Here, displays few sign boards indicating different zones like school zone, hospital zone, railway track etc. By clicking on the button below the sign displays the meaning and instruction to be followed in the particular region.

- This provides the user with better understanding about the sign boards and to act accordingly.

6.5 Feature 5:

DETERMINING TRAFFIC:

- Since hardware sensors are not implemented, we have used random function to generate values for the distance between the user and the vehicle ahead.

- If the distance is below 20, it instructs the driver or the user to stop immediately and try moving forward with different direction or to take diversion.
- This helps in avoiding accidents and clashes while driving.

```
dist=random.randint(0,50)
dis={'dista':dist}
```

```
if(dist<20):
    insta={'inst':'stop'}
```

7. ADVANTAGES:

- Signs with smart connectivity are an inexpensive and flexible medium that can help transmit information according to particular situation and entertain passengers.
- The digital signboards helps in reducing the air pollution due the emission of vehicles in heavy traffic area.
- The drivers can able to know about the weather condition and accordingly follow the speed limit displayed on the sign boards.
- The increased flexibility of these digital sign boards makes it easy for any private or government department to change the message as per the need of the hour.
- The driver can easily find the route and navigation instructions to reach the destination.
- The speed of the vehicle can be identified using location sensor.
- The digitals sign boards and the app are user-friendly.

DISADVANTAGES:

- The digital signboards involves high Installation Costs.
- Getting digital signboards up and running is a far more involved process than print media.
- If the people managing the screens are not graphic designers, it can be difficult to update the content regularly on the screen.
- The digital signboards are still new and developing technology in the road safety sector,

- While digital sign boards require power and therefore can't claim to be green, there is high energy use in the printing, erecting and replacement of traditional print media.

8. CONCLUSION:

Digital road signs are an important part of modern infrastructure and are becoming increasingly common. Digital road signs are becoming more common as technology improves and more states adopt them. The use of digital road signs is expected to continue to grow in the future as it would be observed user-friendly, economic, environment friendly, profitable promoting road safety. Digital road signs are designed to improve road safety and efficiency by providing real-time information to drivers. These signs can display a variety of information, including speed limits, traffic conditions, and weather warnings. Digital road signs can help drivers by providing information that is not always available from traditional signs.

9. FUTURE SCOPE:

One of the benefits of digital road signs is that they can be updated in real-time, which means that they can be used to provide motorists with up-to-the-minute information about conditions on the road ahead. This can be particularly useful in the case of accidents or other incidents that might cause delays. In the future, digital road signs could also be used to provide information about alternative routes that might be available in the event of a problem on the road. This could be particularly useful in the case of major incidents, such as road closures due to bad weather. Finally, digital road signs could be used to provide motorists with information about the best times to travel in order to

avoid traffic congestion. This could be particularly useful in areas where there is a lot of traffic.

10. APPENDIX:

Source Code:

```
import requests #importing a library
import json
import ibmiotf.application
import ibmiotf.device
import time import
random import sys

# watson device details

organization = "2s7yy7"
devicType = "project"
deviceId = "projectid"
authMethod= "token"
authToken= "projecttoken"

#generate random values for randomo variables (temperature&humidity)
def myCommandCallback(cmd): global
  a
  #print("command recieved:%s" %cmd.data['command'])
  #status=cmd.data['command']
  print("command recieved:%s"
  %cmd.data['command'])
  control=cmd.data['command'] print(control)
```

```
try:
      deviceOptions={"org": organization, "type": devicType,"id": deviceId,"auth-
method":authMethod,"auth-token":authToken} deviceCli
      = ibmiotf.device.Client(deviceOptions)
except Exception as e:
      print("caught exception connecting device %s" %str(e)) sys.exit()

#connect and send a datapoint "temp" with value integer value into the cloud as a type of
event for every 10 seconds deviceCli.connect()

while True:

#get sensor data from DHT11

   a =
"https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=e2bea247
e d9ad643a04d9a8e55499d5f" r=requests.get(url=a) data=r.json()

   Temp= data['main']['temp']
   Humd= data['main']['humidity'] data=
   {'temp':Temp,'humid':Humd}
   dist=random.randint(0,20)
   dis={'dista':dist}

   if(Humd<100): warn={'alert':'PLEASE SLOW
      DOWN!!!!!!'}
   if(dist<20):
      insta={'inst':'stop'}

   def myOnPublishCallback():
```

```
        print("published Temperature = %s c" %Temp,"humidity:%s %%"
        %Humd) print(warn) print(dis) print(insta)
    success=deviceCli.publishEvent ("IoTSensor","json",insta,qos=0,on_publish=
myOnPublishCallback) success=deviceCli.publishEvent
    ("IoTSensor","json",data,qos=0,on_publish=
myOnPublishCallback) success=deviceCli.publishEvent
    ("IoTSensor","json",warn,qos=0,on_publish=
myOnPublishCallback) success=deviceCli.publishEvent
    ("IoTSensor","json",dis,qos=0,on_publish=
myOnPublishCallback)

    if not success:
        print("not connected to ibmiot")
    time.sleep(5)

    deviceCli.commandCallback=myCommandCallback
#disconnect the device deviceCli.disconnect()
```