IBM PROJECT REPORT

Team Id	NM2023TMID01923
Project Name	Drowsiness Detection And Alerting System.

S.NO	TEAM MEMBERS NAME	MAIL ID
1.	S.SANGAVI	ssangavi0701@gmail.com
2.	S.SWETHA	swethaselvaraj2405@gmail.com
3.	D.TEENA	teenateena253@gmail.com
4.	E.YAZHINI	yazhinielansezhian@gmail.com

INDEX:

S.NO	TITLE
1.	INTRODUCTION
2.	LITERATURE SURVEY
3.	IDEATION&PROPOSED SOLUTION
4.	REQUIREMENT ANALYSIS
5.	PROJECT DESIGN
6.	CODING&SOLUTIONING
7.	RESULTS
8.	ADVANTAGES & DISADVANTAGES
9.	CONCLUSION
10.	FUTURE SCOPE
11.	APPENDIX

1.INTRODUCTION:

1.1PROJECT OVERVIEW:

The purpose of the drowsiness detection system is to aid in the prevention of accidents passenger and commercial vehicles. The system will detect the early symptoms of drowsiness before the driver has fully lost all attentiveness and warn the driver that they are no longer capable of operating the vehicle safely.

1.1.1 HUMAN PSYCHOLOGY WITH CURRENT TECHNOLOGY:

Humans have always invented machines and devised techniques to ease and protect their lives, for mundane activities like traveling to work, or for more interesting purposes like aircraft travel. With the advancement in technology, modes of transportation kept on advancing and our dependency on it started increasing exponentially.

It has greatly affected our lives as we know it. Now, we can travel to places at a pace that even our grandparents wouldn't have thought possible. Immodern times, almost everyone in this world uses some sort of transportation every day. Some people are rich enough to have their own vehicles while others use public transportation. However, there are some rules and codes of conduct for those who drive irrespective of their social status. One of them is staying alert and active while driving.

Neglecting our duties towards safer travel has enabled hundreds of thousands of tragedies to get associated with this wonderful invention every year. It may seem like a trivial thing to most folks but following rules and regulations on the road is of utmost importance.

While on road, an automobile wields the most power and in irresponsible hands, it can be destructive and sometimes, that carelessness can harm lives even of the people on the road. One kind of carelessness is not admitting when we are too tired to drive. In order to monitor and prevent a destructive outcome from such negligence, many researchers have written research papers on driver drowsiness detection systems. But at times, some of the points and observations made by the system are not accurate enough.

Hence, to provide data and another perspective on the problem at hand, in order to improve their implementations and to further optimize the solution, this project has been done.

1.1.2FACTS & STATISTICS:

Our current statistics reveal that just in 2015 in India alone, 148,707people died due to car related accidents. Of these, at least 21 percentwere caused due to fatigue causing drivers to make mistakes.

- ✓ This can be a relatively smaller number still, as among the multiple causes that can lead to an accident, the involvement of fatigue as a cause is generally grossly underestimated.
- ✓ Fatigue combined with bad infrastructure indeveloping countries like India is a recipe for disaster.
- ✓ Fatigue, in general, is very difficult to measure or observe unlike alcohol and drugs, which have clear key indicators and tests that are available easily. Probably, the best solutions to this problem are awareness about fatigue-related accidents and promoting drivers to admit fatigue when needed.
- ✓ The former is hard and much more expensive to achieve, and the latter is not possible without the former as driving for long hours is very lucrative.

2.LITERATURE SURVEY:

PAPER 1:

PAPER TITLE	Driver Drowsiness Detection and Alert System.
PROBLEM DEFINITION	The project employs a CNN model to determine whether or not a person is drowsy based on whether or not their eyes are closed or open the idea has direct application in the vehicle sector, making driving safer and lowing number of people killed in car accidents caused by drowsy driving.
METHOROGY ALGORITHM	Open CV, DLIb , EAE(eye aspect ratio) , face recognition.

ADVANTAGE	Driver drowsiness detection is a car safety technology which prevent accident when driver is getting drowsy.
DISADVANTAGE	Mainly, using of two cameras in the system one monitoring the head movement and the other one facial expression.

PAPER 2:

PAPER TITLE	Driver Drowsiness Prediction Based on Multiple Aspects Using Image ProcessingTechniques.
PROBLEM DEFINITION	The fixed either in the front mirror or dashboard. Images of people driving have been collected in color 24-b (RGB) with resolution 640 X 480 from the 30 frames/sec.
METHOROGY ALGORITHM	Convolution neural network(CNN)
ADVANTAGE	It depends on the eye and mouth closure status along with new vector FAR .This help to find the status of the closure eye or opened mouth like yawning etc..,
DISADVANTAGE	An automated non – contact system that identify driver drowsiness early is the need of the hour.

PAPER 3:

PAPER TITLE	A Review of Recent Developments in Driver Drowsiness Detection Systems.

PROBLEM DEFINITION	The proposed system detects drowsiness if the eyes have been closed for a period of four or more frames. The detection system differentiates the normal eye blink from drowsiness. The developed is an non-invasive system.
METHOROGY ALGORITHM	biological-based measures; driver drowsiness detection; hybrid-based measures; image-based measures; vehicle-based measures.
ADVANTAGE	The number of times the mouth opens over a specific period of time.
DISADVANTAGE	EAR reflects the eye's openness degree. The EAR value drops down to zero when the eyes are closed. On the other hand, it remains approximately constant when the eye is open. Thus, the EAR detects the eye closure at that time.

PAPER 4:

PAPER TITLE	IOT-Based Smart Alert System for Drowsy Driver Detection.
PROBLEM DEFINITION	An IOT-based system is designed to avoid countless mishaps due to drowsy driver's behavioural and psychological changes by focusing on driver's eye movements.
METHOROGY ALGORITHM	Pi camera model V2, Raspberry Pi3, Crash Sensor, Speaker, . GPS Module, Force Sensitive Resistor Sensor (FSR).
ADVANTAGE	It is basically based on image-formed or pictorial-based steering movement and the CNN algorithm for proper classification of

	drowsiness, which can also reduce false drowsy detection rates.
DISADVANTAGE	Driver Drowsiness is one of the most factor of road accidents, leading. Drowsiness means difficulty staying awake which can lead to falling asleep.

PAPER 5:

PAPER TITLE	Drowsiness Detection and Alert System.
PROBLEM DEFINITION	Study shows that accidents occur due to sleepy drivers in need of a rest, which means that road accidents occurs more due to drowsiness rather than drink-driving.
METHOROGY ALGORITHM	GSM SIM 800A Modem, Buzzer ,IR sensor, Goggle, battery.
ADVANTAGE	Whenever the driver feels drowsy and closes his eyes for more than a second, the buzzer is blown. As a result, it alerts the driver. It also warns the owner of the truck driver by sending him text messages.
DISADVANTAGE	Other truck driver errors are similar to those that anyone can make, such as not paying attention to surrounding, speeding, not knowing routes, exhaustion and driving under the influence of alcohol or drug.

3.IDEATION & PROPOSED SOLUTION:

3.1 PROBLEM STATEMENT DEFINITIONS:

The car accident is the leading cause of death, killing around 1.3 million people each year. Most of these accidents are caused by driver distraction or drowsiness. Drowsiness decreases the driver's concentration, activity, alertness, and alertness, and causes the driver to make slow decisions and sometimes not makedecisions.

- Lack of sleep or distractions such as talking on the phone, talking to the passenger, etc. can cause anaccident.
- To avoid these accidents, we propose a system that will warn the driver if they are distracted ordrowsy.
- The Drowsiness Alert feature may alert you if you're drowsy ansuggested you take a break when it's safe to do so.
- Drowsiness alerts are designed to warn you that you havebecome

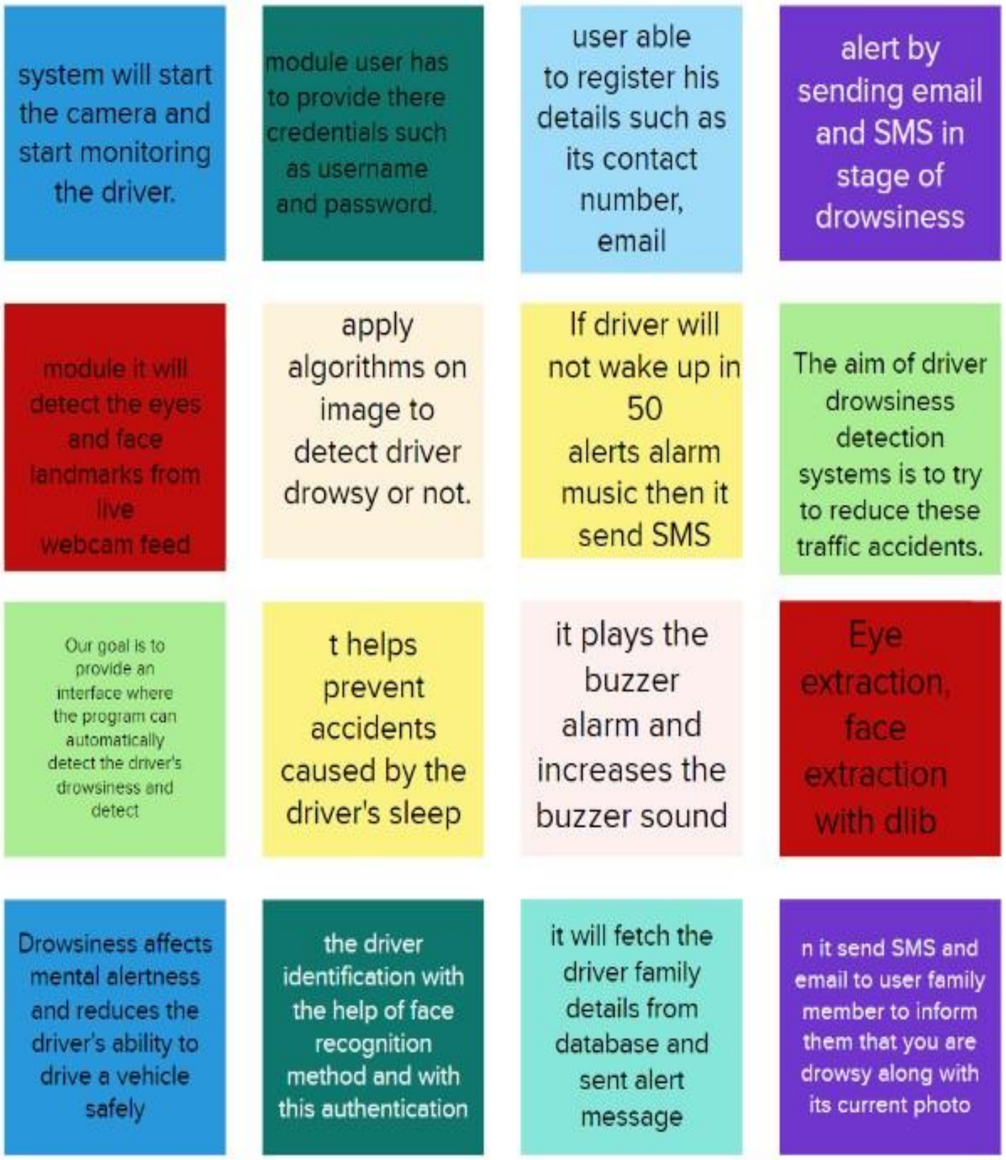
drowsy *after* you have already begun driving; you shouldn't get

behind the wheel in the first place if you know you're drowsy.

- ✓ Choose a safe, secure location to take a break. Some cars with drowsiness alert may automatically inform you of nearby rest areas using the built-inGPS.
- ✓ On long trips, plan ahead periodic driving breaks – every two hours or so – to reduce the risk of becoming drowsy behind-the-wheel. If driving with other licensed drivers, consider takingturns.

REFERENCE:

https://app.mural.co/t/drowsinessdetectionandalerti6401/m/drowsinessdetectionandalerti6401/1682735404865/d0d7d27131b38d3b3d2f0c322b81a8702eda5ea2?sender=u6471a365b1fbe9f6bd790



system will start the camera and start monitoring the driver.

module user has to provide there credentials such as username and password.

user able to register his details such as its contact number, email

alert by sending email and SMS in stage of drowsiness

module it will detect the eyes and face landmarks from live webcam feed

apply algorithms on image to detect driver drowsy or not.

If driver will not wake up in 50 alerts alarm music then it send SMS

The aim of driver drowsiness detection systems is to try to reduce these traffic accidents.

Our goal is to provide an interface where the program can automatically detect the driver's drowsiness and detect

t helps prevent accidents caused by the driver's sleep

it plays the buzzer alarm and increases the buzzer sound

Eye extraction, face extraction with dlib

Drowsiness affects mental alertness and reduces the driver's ability to drive a vehicle safely

the driver identification with the help of face recognition method and with this authentication

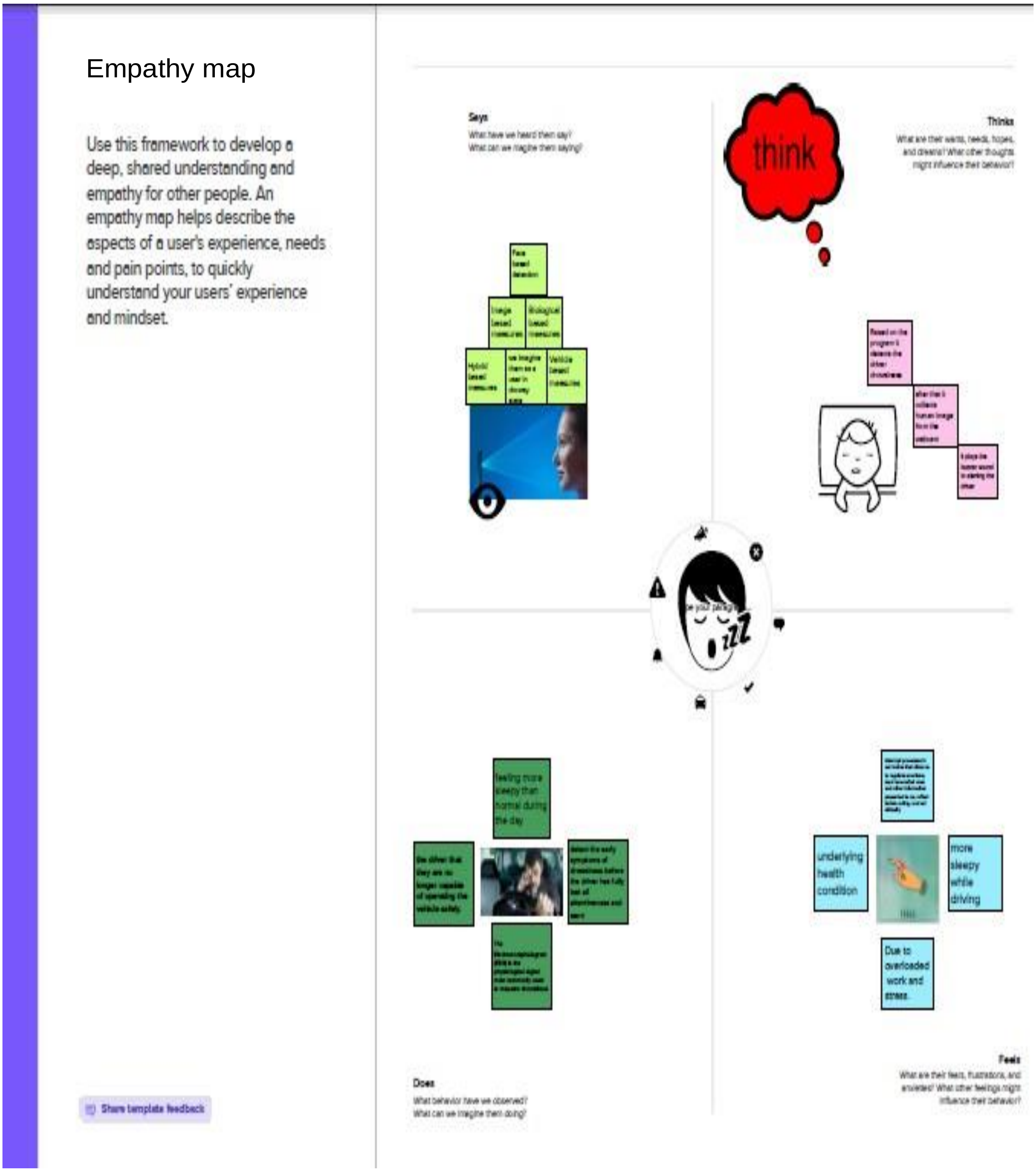
it will fetch the driver family details from database and sent alert message

n it send SMS and email to user family member to inform them that you are drowsy along with its current photo

3.1EMPATHY MAP CANVAS:

- ✓ An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviors and attitudes.
- ✓ It is a useful tool to helps teams better understand their users.
- ✓ Creating an effective solution requires understanding the true problem and the person who is experiencing.
- ✓ The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

EXAMPLE :



Empathy map

Use this framework to develop a deep, shared understanding and empathy for other people. An empathy map helps describe the aspects of a user's experience, needs and pain points, to quickly understand your users' experience and mindset.

Share template feedback

Says

What have we heard them say?
What can we imagine them saying?

Face based detection

Image based measures

Biological based measures

Hybrid based measures

we imagine them as a user in drowsy state

Vehicle based measures

Thinks

What are their wants, needs, hopes, and dreams? What other thoughts might influence their behavior?

Based on the program it detects the driver drowsiness

after that it collects human image from the webcam

it plays the buzzer sound in alerting the driver

feeling more sleepy than normal during the day

the driver that they are no longer capable of operating the vehicle safely

detect the early symptoms of drowsiness before the driver has fully lost all attentiveness and warn

[illegible]

Does

What behavior have we observed?
What can we imagine them doing?

[illegible]

underlying health condition

more sleepy while driving

Due to overloaded work and stress.

Feels

What are their fears, frustrations, and anxieties? What other feelings might influence their behavior?

3.2IDEATION & BRAINSTORMING

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving.

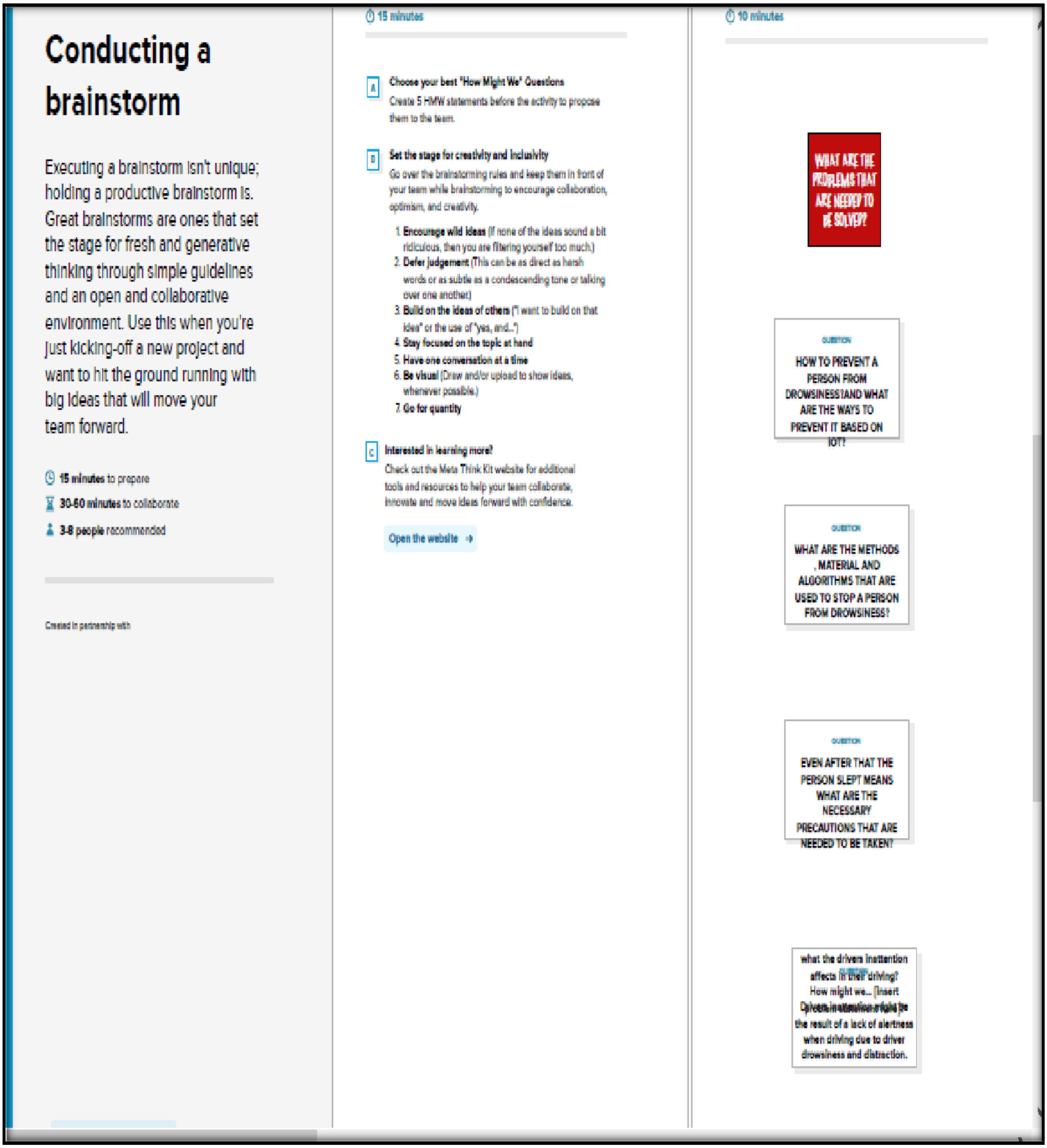
- Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.
- Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

REFERENCE:

https://app.mural.co/invitation/mural/drowsinessdetectionandalerti6401/1682688290980?sender=uc47b69cf07e75 094ae424043&key=68074430-daf0-4a96-a408-79b8664ace74

STEP 1: TEAM GATHERING , COLLABORATION AND SELECT THE PROBLEM STATEMENT:

STATEMENT



Conducting a brainstorm

Executing a brainstorm isn't unique; holding a productive brainstorm is. Great brainstorms are ones that set the stage for fresh and generative thinking through simple guidelines and an open and collaborative environment. Use this when you're just kicking-off a new project and want to hit the ground running with big ideas that will move your team forward.

15 minutes to prepare

30-60 minutes to collaborate

3-8 people recommended

Created in partnership with

15 minutes

A **Choose your best "How Might We" Questions**
Create 5 HMW statements before the activity to propose them to the team.

B **Set the stage for creativity and inclusivity**
Go over the brainstorming rules and keep them in front of your team while brainstorming to encourage collaboration, optimism, and creativity.

1. **Encourage wild ideas** (If none of the ideas sound a bit ridiculous, then you are filtering yourself too much.)
2. **Defer judgement** (This can be as direct as harsh words or as subtle as a condescending tone or talking over one another)
3. **Build on the ideas of others** ("I want to build on that idea" or the use of "yes, and...")
4. **Stay focused on the topic at hand**
5. **Have one conversation at a time**
6. **Be visual** (Draw and/or upload to show ideas, whenever possible.)
7. **Go for quantity**

C **Interested in learning more?**
Check out the Meta Think Kit website for additional tools and resources to help your team collaborate, innovate and move ideas forward with confidence.

Open the website →

10 minutes

WHAT ARE THE PROBLEMS THAT ARE NEEDED TO BE SOLVED?

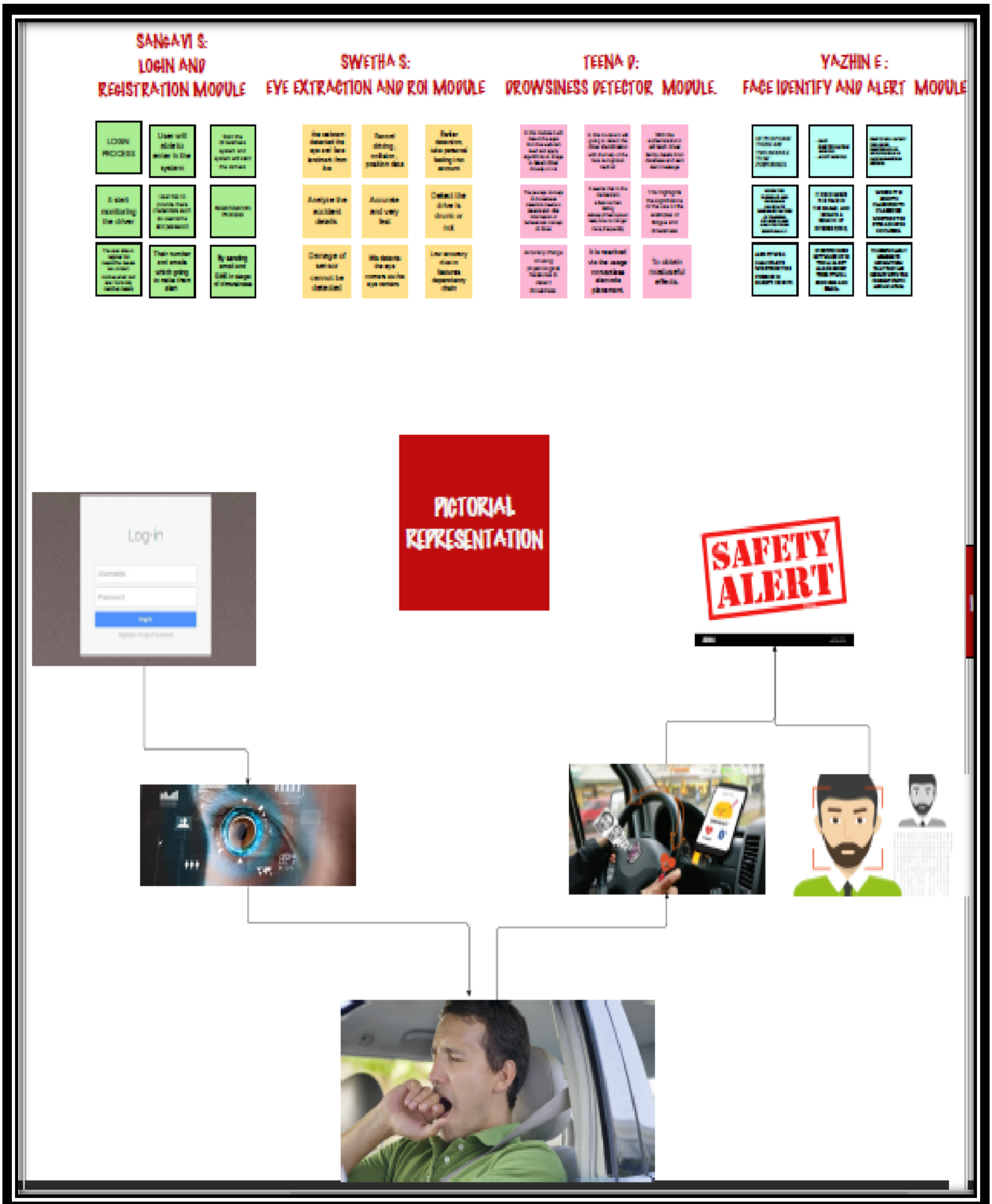
QUESTION
HOW TO PREVENT A PERSON FROM DROWSINESS?AND WHAT ARE THE WAYS TO PREVENT IT BASED ON IOT?

QUESTION
WHAT ARE THE METHODS , MATERIAL AND ALGORITHMS THAT ARE USED TO STOP A PERSON FROM DROWSINESS?

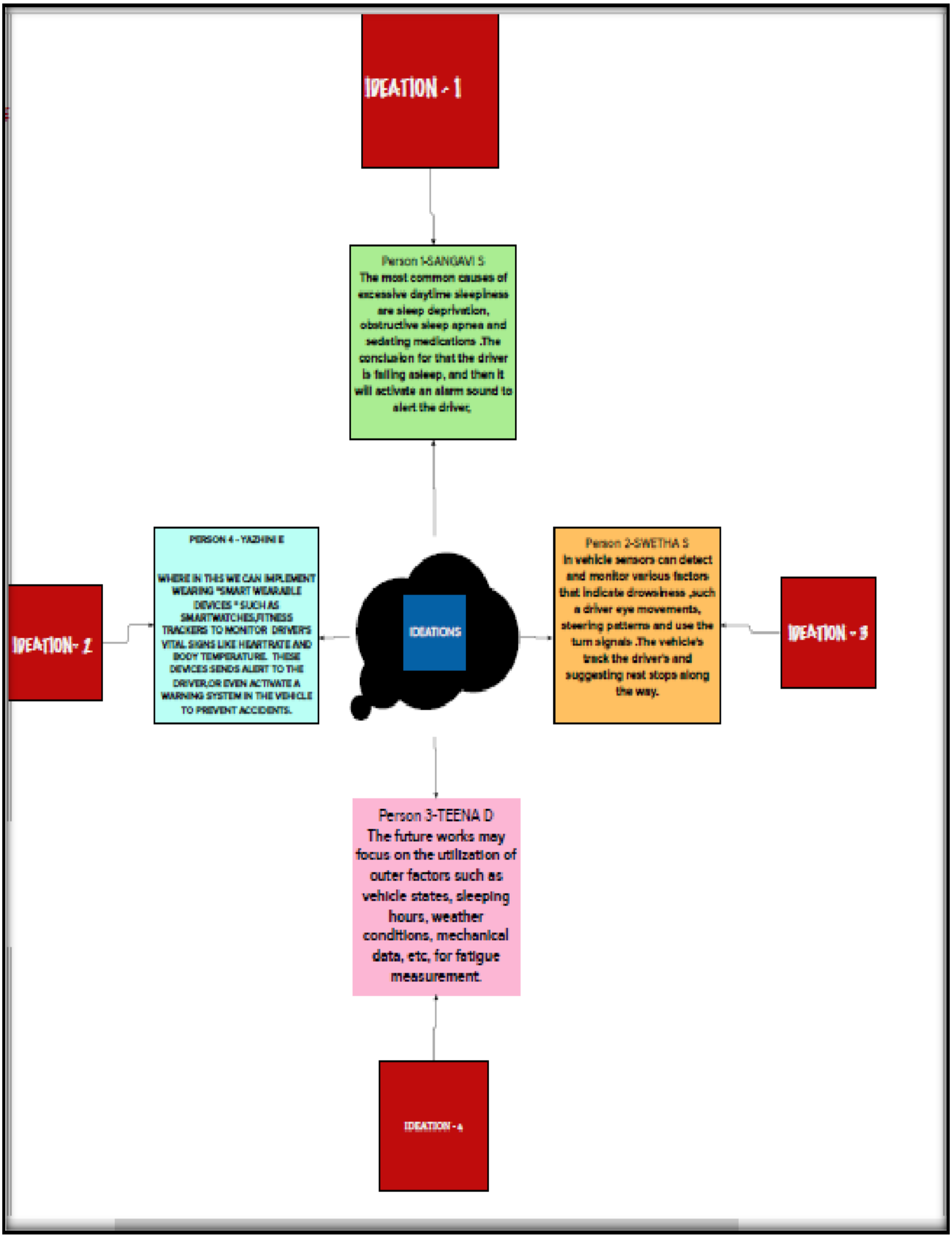
QUESTION
EVEN AFTER THAT THE PERSON SLEPT MEANS WHAT ARE THE NECESSARY PRECAUTIONS THAT ARE NEEDED TO BE TAKEN?

what the drivers inattention affects in their driving? How might we... [Insert Drivers inattention might be the result of a lack of alertness when driving due to driver drowsiness and distraction.

STEP2:BRAINSTORM,IDEALISTINGANDGROUPING:


SANGAVI S:
LOGIN AND
REGISTRATION MODULE
SWETHA S:
EYE EXTRACTION AND ROI MODULE
TEENA P:
DROWSINESS DETECTOR MODULE.
YAZHINI E :
FACE IDENTIFY AND ALERT MODULE
LOGIN PROCESS
User will able to enter in the system
It start monitoring the driver
Analyse the accident details
Accurate and very fast
Detect the driver is drunk or not
Damage of sensor cannot be detected
To obtain ... effects.
PICTORIAL REPRESENTATION
Log-in
Username
Password
login
SAFETY ALERT

STEP 3: BRAINSTORM AS GROUP (IDEATIONIMPLEMENTATIONBASEDONINDIVIDUALCREATIVITY):



Person 1-SANGAVI S
The most common causes of excessive daytime sleepiness are sleep deprivation, obstructive sleep apnea and sedating medications .The conclusion for that the driver is falling asleep, and then it will activate an alarm sound to alert the driver,

PERSON 4 - YAZHINI E

WHERE IN THIS WE CAN IMPLEMENT WEARING "SMART WEARABLE DEVICES " SUCH AS SMARTWATCHES,FITNESS TRACKERS TO MONITOR DRIVER'S VITAL SIGNS LIKE HEARTRATE AND BODY TEMPERATURE. THESE DEVICES SENDS ALERT TO THE DRIVER,OR EVEN ACTIVATE A WARNING SYSTEM IN THE VEHICLE TO PREVENT ACCIDENTS.

Person 2-SWETHA S
In vehicle sensors can detect and monitor various factors that indicate drowsiness ,such a driver eye movements, steering patterns and use the turn signals .The vehicle's track the driver's and suggesting rest stops along the way.

Person 3-TEENA D
The future works may focus on the utilization of outer factors such as vehicle states, sleeping hours, weather conditions, mechanical data, etc, for fatigue measurement.

3.4 PROPOSED SOLUTIO

3.4 PROPOSED SOLUTION:

S. No.	**Parameter**	**Description**
1	Problem Statement (Problem to be solved)	➢ The problem that causes “**Major Road accident** ” is drowsiness, which is a serious issue that needs to be addressed. ➢ The drowsiness can be caused by various factors such as sleep deprivation, medications, medical conditions and so on
2	Idea / Solution description	➢ Driver drowsiness detection systems can use cameras, eye tracking sensors and other hardware to monitor visual cues. ➢ Where drowsiness can be detected through eye aspect ratio algorithm that sends the data of “**Movements of eyelids over the cloud** ”. The driver is alerted by an alarm sound and a message through a mobile application that wakes them up.
3	Novelty / Uniqueness	➢ Rather than sending alert through mobile app we can use “**Wearable sensor**” such as EEG headbands or heart rate monitors. ➢ And we can also installed “**Steering wheel sensor**” to detect the changes in driving behavior, such as sudden jerks or swerves which indicates drowsiness
4	Social Impact / Customer Satisfaction	➢ The system will detect the early symptoms of drowsiness before, which “**Prevents Accidents** ”. ➢ By preventing this we can ➢ “ **Save lives**” and reduce the number of fatalities on the road. ➢ Which will “ **Lower healthcare cost**” which requires medical attention.

5	Business Model (Revenue Model)	➢ **Development and Manufacturing Cost**: The cost of developing and manufacturing a drowsiness detector in IOT can impact its profitability. Lower production costs can result in a higher profit margin for each unit sold. ➢ **Pricing Strategy**: The pricing strategy can impact profitability by a higher price point may result in a higher profit margin per unit sold, but may also limit adoption. ➢ **Competition**: The presence of competitors in the market can impact the profitability of a drowsiness detector in IOT. If there are many similar products on the market, the price point may need to be lower to remain competitive. **Marketing and distribution costs**: The cost of marketing and distribution can impact the profitability of the product. The marketing strategy should be designed to reach the target audience effectively, while keeping the cost of advertising and promotion under control.
6	Scalability of the Solution	➢ **Cloud computing**: Cloud computing can provide a scalable solution for drowsiness detection by allowing data to be processed and analyzed in the cloud. This can offload the processing requirements from the device and provide greater scalability by allowing multiple devices to connect to the cloud-based system. ➢ **Distributed architecture**: A distributed architecture can provide scalability by allowing data to be processed across multiple devices.

		This can help distribute the load and prevent performance bottlenecks. ➢ **Load balancing**: Load balancing can help distribute the workload across multiple servers or devices. This can prevent overloading of any one server or device and improve system performance and scalability. ➢ **Data compression**: Data compression techniques can reduce the amount of data that needs to be transmitted and processed. This can help reduce the processing requirements and improve system scalability. ➢ **Edge computing**: Edge computing can provide a scalable solution for drowsiness detection by allowing data to be processed locally on the device. This can offload the processing requirements from the cloud-based system and provide greater scalability by allowing multiple devices to connect to the edge-based system.

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 No.1	**Image Acquisition:**	For demonstration purposes, a webcam will be used to capture images in an infinite loop

FR No.2	**Region of interest detection eye:**	The face will be detected first, followed by the face, and finally our region of interest, the eyes
FR No.3	**Observation of the eye:**	We will be looking at a series of pictures to see if there are blinking quickly, concentrating their eyes to one side, or if they have closed their eyes
FR No.5	**Make a call to an emergency number:**	If the driver hasn't woken up or is asleep, the software willterminate

4.2 NON-FUNCTIONAL REQUIREMENT:
Following are the non- functional requirements of the proposed solution.

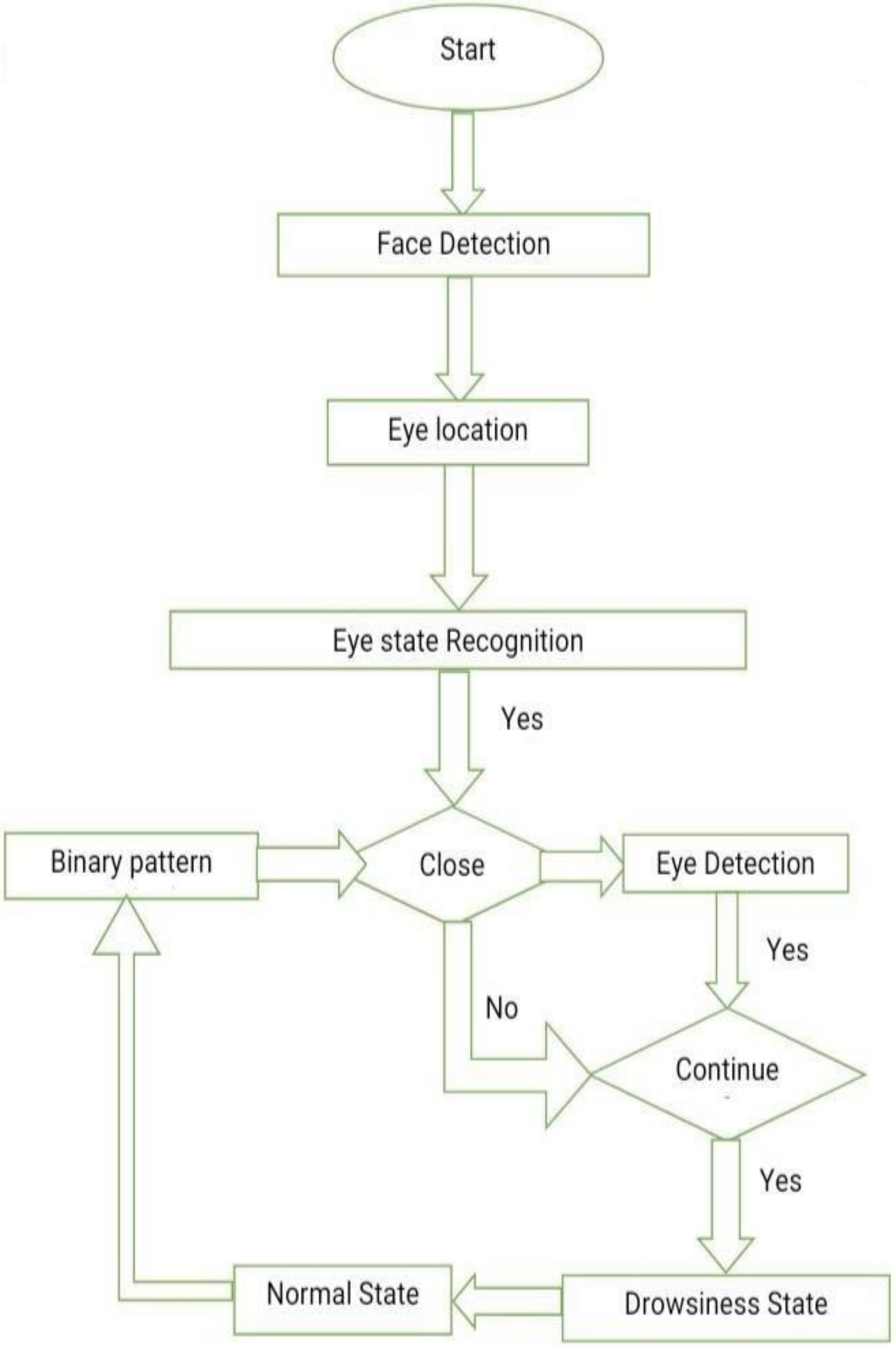
NFR No.	**Non-Functional Requirement**	**Description**
NFR No.1	**Usability**	Indicates how effectively and easy users can learn and use a system
NFR No.2	**Security**	Assures all data inside the system or it's part will be protected against malware attacks or unauthorized access

NFR No.3	**Reliability**	Specifies the probability of the software performing without failure for a specific number of uses or amount of time
NFR No.4	**Performance**	Deals with the measure of the system’s response time under different load conditions
NFR No.5	**Availability**	Describes how likely the system is accessible for a user at a given point in time
NFR No.6	**Scalability**	Assesses the highest workloads under which the system will still meet the performance requirements.

5.PROJECT DESIGN:

5.1 DATA FLOW DIAGRAMS:

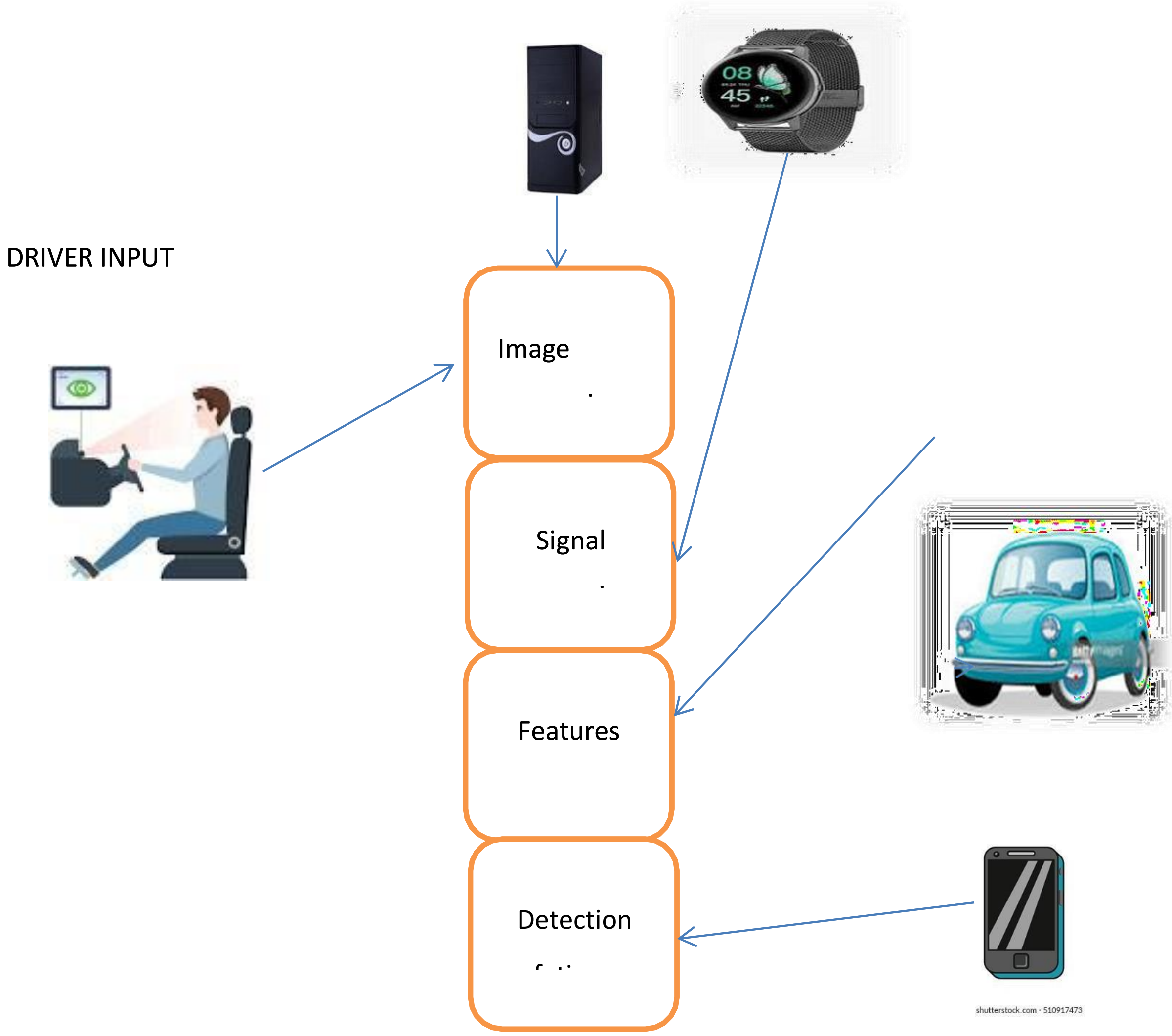
- A Data Flow Diagram (DFD) is a traditional visual representation information flows within a system.
- A neat and clear DFD can depict the right amount of the system requirement graphically.
- It shows how data enters and leaves the system, what changes the information, and where data is stored.


Start
Face Detection
Eye location
Eye state Recognition
Yes
Binary pattern
Close
Eye Detection
Yes
No
Continue
Yes
Normal State
Drowsiness State

5.2 SOLUTION & TECHNICAL ARCHITECTURE:

Solution architecture is a complex process – with many sub- processes that bridges the gap between business problems and technology solutions. Its goals are to:

- Find thebest tech solutiontosolveexisting business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to projectstakeholders.
- Definefeatures,developmentphases,and solution requirements.
- Provide specifications according to which the solution is defined, managed, anddelivered.



5.3 USER STORIES:

User type	Functional requirement	User story Number	User Story/ Task	Acceptance criteria	Priority	Release
Driver	To aid in the prevention of accidents passenger and commercial vehicles	USN-1	Place the drowsiness detection and alerting system in driver’s vehicle so it sends a alert and avoid accidents caused by drowsiness, fatigue.	Sends an alert to driver.	High	Sprint-1
Safety Enhancement	As a long-haul truck driver, I want a drowsiness detection and alerting system installed in my vehicle to ensure my safetyduring longjourney .	USN-2	The system should monitor the driver’s eye movements, head position, and facial expressions.	It should continuously analyse these factors and provide accurate drowsiness detection without causing discomfort to the driver	High	Sprint-2

IBM Cloud	Data transfer	USN-2	It sends collected data to the IBM cloud	Transfer data between the user through network	High	Sprint-3

6.CODING & SOLUTIONING

(Explain the features added in the project along with code)

6.1FEATURE 1:

```
Import cv2
import dlib
import requests
import pyttsx3
from scipy.spatial import distance

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "ocokk7"
deviceType = "Drowsy"
deviceId = "7339616"
authMethod = "token"
authToken = "12345678"
```

```
def ibmstart(x):


    def myCommandCallback(cmd):
        print("Command received: %s" % cmd.data['command'])
        print(cmd)


    try:
      deviceOptions = {"org": organization, "type": deviceType, "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()


    deviceCli.connect()
    data = { 'Status' : x}
    #print data
    def myOnPublishCallback():
        print ("Published Status = %s" % x, "to IBM Watson")


    success = deviceCli.publishEvent("DD", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")



    deviceCli.commandCallback = myCommandCallback
    deviceCli.disconnect()
```

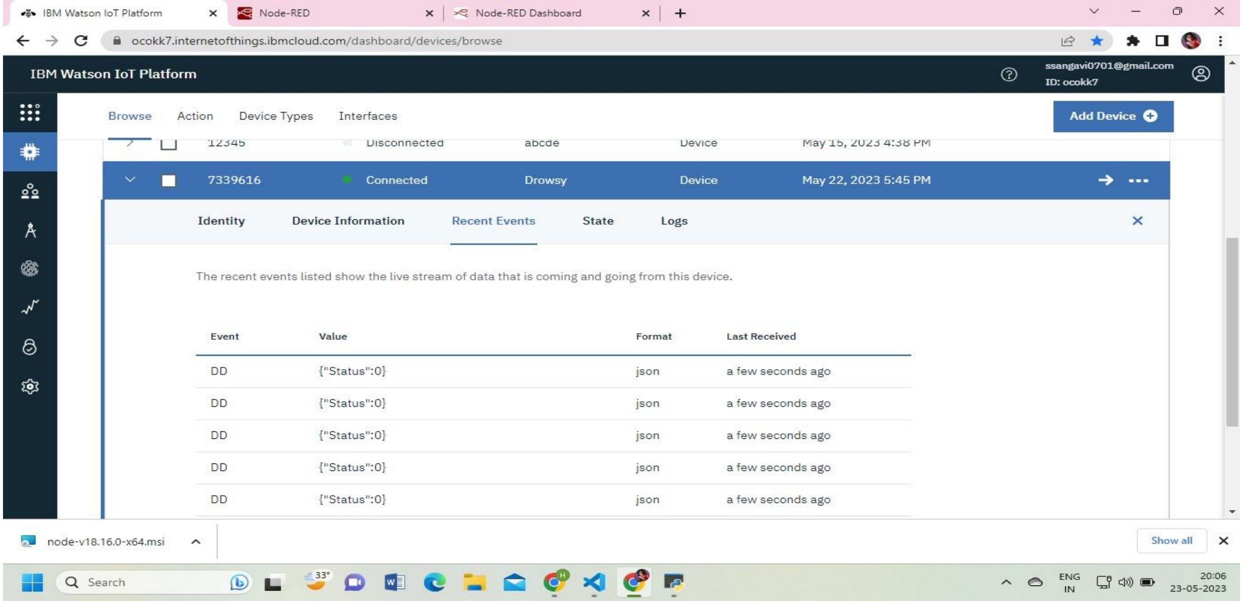
```
# INITIALIZING THE pyttsx3 SO THAT
# ALERT AUDIO MESSAGE CAN BE DELIVERED
engine = pyttsx3.init()

# SETTING UP OF CAMERA TO 1 YOU CAN
# EVEN CHOOSE 0 IN PLACE OF 1
cap = cv2.VideoCapture(0)

# FACE DETECTION OR MAPPING THE FACE TO
# GET THE Eye AND EYES DETECTED
face_detector = dlib.get_frontal_face_detector()

# PUT THE LOCATION OF .DAT FILE (FILE FOR
# PREDECTING THE LANDMARKS ON FACE )
dlib_facelandmark = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

# FUNCTION CALCULATING THE ASPECT RATIO FOR
# THE Eye BY USING EUCLIDEAN DISTANCE FUNCTION
def Detect_Eye(eye):
  poi_A = distance.euclidean(eye[1], eye[5])
  poi_B = distance.euclidean(eye[2], eye[4])
  poi_C = distance.euclidean(eye[0], eye[3])
  aspect_ratio_Eye = (poi_A+poi_B)/(2*poi_C)
  return aspect_ratio_Eye



# MAIN LOOP IT WILL RUN ALL THE UNLESS AND
# UNTIL THE PROGRAM IS BEING KILLED BY THE USER
while True:

  null, frame = cap.read()
  flag=0
```

```
gray_scale = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

faces = face_detector(gray_scale)

for face in faces:
  face_landmarks = dlib_facelandmark(gray_scale, face)
  leftEye = []
  rightEye = []

  # THESE ARE THE POINTS ALLOCATION FOR THE
  # LEFT EYES IN .DAT FILE THAT ARE FROM 42 TO 47
  for n in range(42, 48):
    x = face_landmarks.part(n).x
    y = face_landmarks.part(n).y
    rightEye.append((x, y))
    next_point = n+1
    if n == 47:
      next_point = 42
    x2 = face_landmarks.part(next_point).x
    y2 = face_landmarks.part(next_point).y
    cv2.line(frame, (x, y), (x2, y2), (0, 255, 0), 1)

  # THESE ARE THE POINTS ALLOCATION FOR THE
  # RIGHT EYES IN .DAT FILE THAT ARE FROM 36 TO 41
  for n in range(36, 42):
    x = face_landmarks.part(n).x
    y = face_landmarks.part(n).y
    leftEye.append((x, y))
    next_point = n+1
    if n == 41:
      next_point = 36
    x2 = face_landmarks.part(next_point).x
    y2 = face_landmarks.part(next_point).y
    cv2.line(frame, (x, y), (x2, y2), (255, 255, 0), 1)

  # CALCULATING THE ASPECT RATIO FOR LEFT
  # AND RIGHT EYE
  right_Eye = Detect_Eye(rightEye)
  left_Eye = Detect_Eye(leftEye)
  Eye_Rat = (left_Eye+right_Eye)/2

  # NOW ROUND OF THE VALUE OF AVERAGE MEAN
  # OF RIGHT AND LEFT EYES
  Eye_Rat = round(Eye_Rat, 2)

  # THIS VALUE OF 0.25 (YOU CAN EVEN CHANGE IT)
  # WILL DECIDE WHETHER THE PERSONS'S EYES ARE CLOSE OR NOT
  if Eye_Rat< 0.25:
    cv2.putText(frame, "DROWSINESS DETECTED", (50, 100),
       cv2.FONT_HERSHEY_PLAIN, 2, (21, 56, 210), 3)
    cv2.putText(frame, "Alert!!!! WAKE UP DUDE", (50, 450),
       cv2.FONT_HERSHEY_PLAIN, 2, (21, 56, 212), 3)

    # CALLING THE AUDIO FUNCTION OF TEXT TO
    # AUDIO FOR ALERTING THE PERSON
    engine.say("Alert!!!! WAKE UP DUDE")
    flag=1
    engine.runAndWait()



cv2.imshow("Drowsiness DETECTOR IN OPENCV2", frame)
print(flag)
ibmstart(flag)
```

```
  '''
  while True:

            data = { 'Status' : x}
            #print data
            def myOnPublishCallback():
                print ("Published Status = %s" % x, "to IBM Watson")

            success = deviceCli.publishEvent("DD", "json", data, qos=0, on_publish=myOnPublishCallback)
            if not success:
                print("Not connected to IoTF")
            time.sleep(1)

            deviceCli.commandCallback = myCommandCallback
  '''
  #r1 = requests.get('https://api.thingspeak.com/update?api_key=SEWZDEK7APG3P0P8&field1='+str(flag))
  #print(r1.status_code)
  key = cv2.waitKey(9)
  if key == 20:
    break

# Disconnect the device and application from the cloud
#deviceCli.disconnect()
cap.release()
cv2.destroyAllWindows()
```

FEATURE 2:

6.3 THE IBM WATSON EVENTS OUTPUT

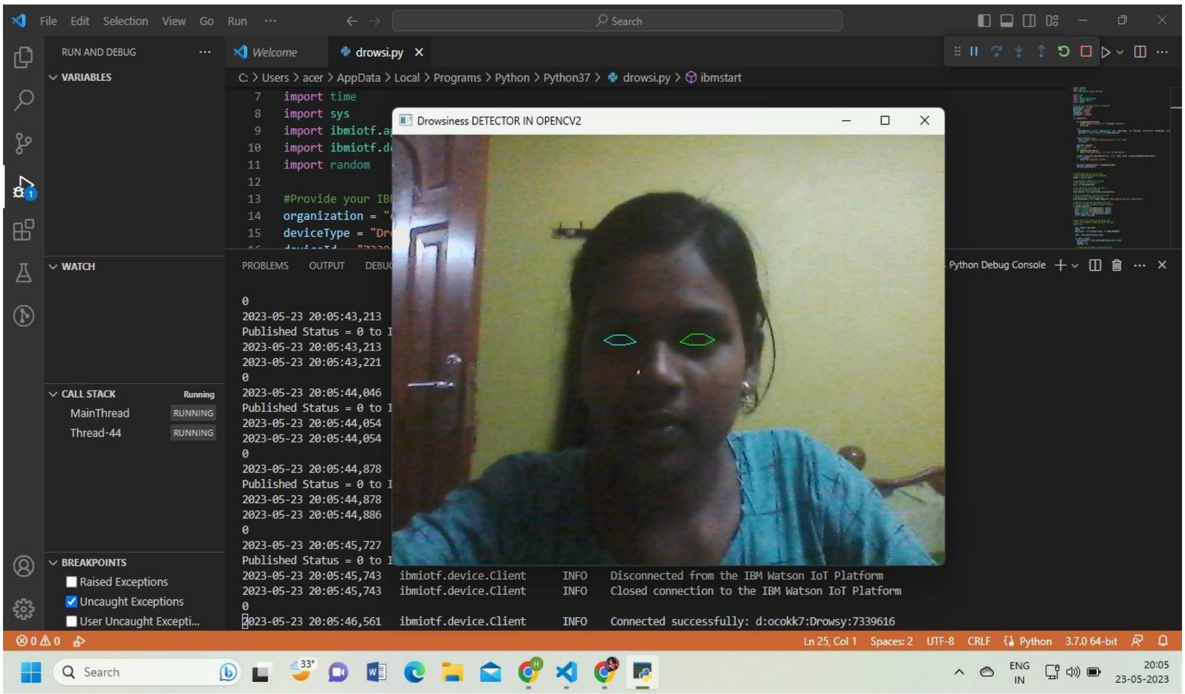


12345	Disconnected	abcde	Device	May 15, 2023 4:38 PM
7339616	Connected	Drowsy	Device	May 22, 2023 5:45 PM

The recent events listed show the live stream of data that is coming and going from this device.

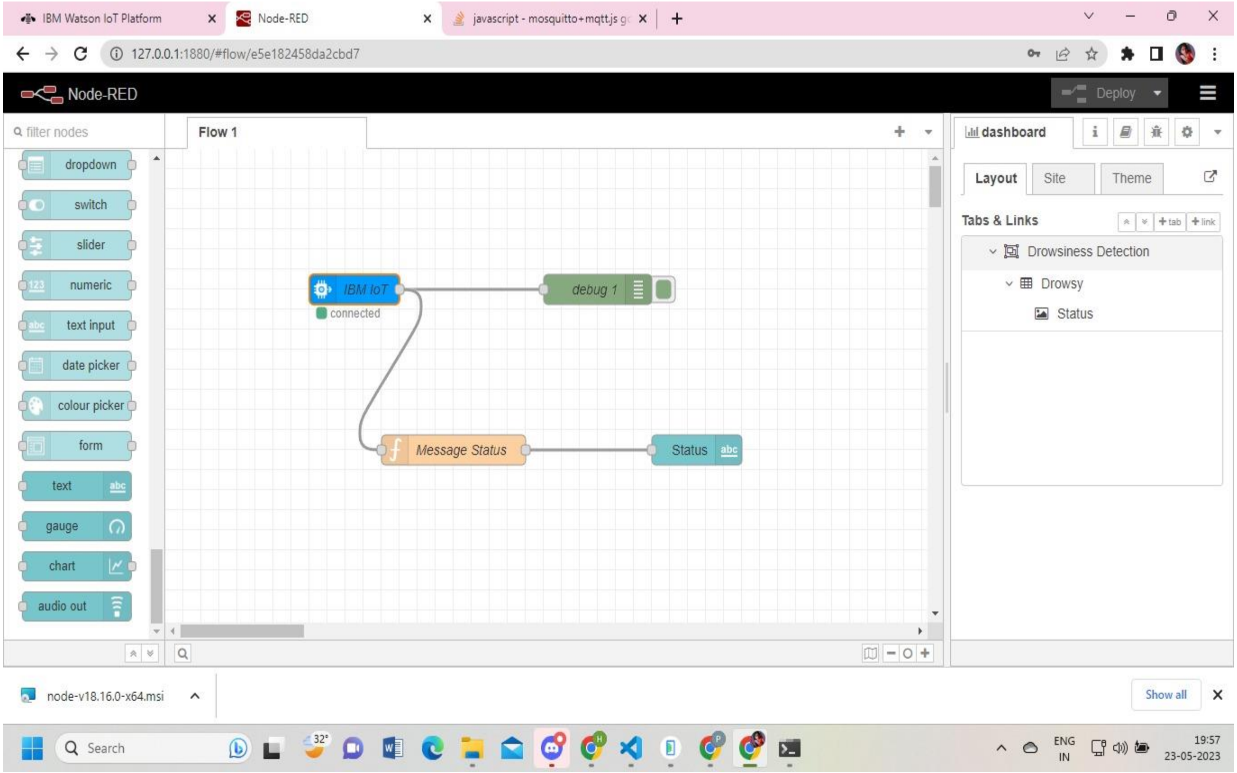
Event	Value	Format	Last Received
DD	{"Status":0}	json	a few seconds ago
DD	{"Status":0}	json	a few seconds ago
DD	{"Status":0}	json	a few seconds ago
DD	{"Status":0}	json	a few seconds ago
DD	{"Status":0}	json	a few seconds ago

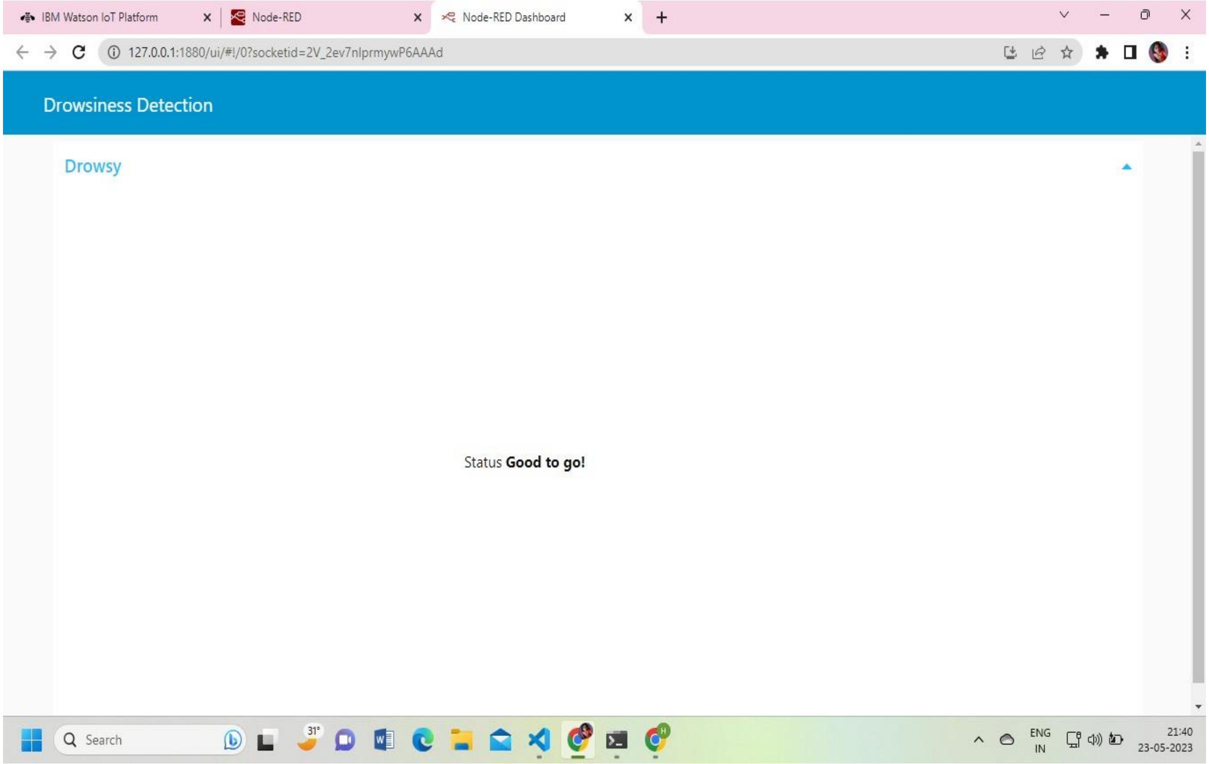
6.4 THE NODE RED OUTPUT



```
import time
import sys
import ibmiotf.a
import ibmiotf.d
import random

#Provide your IB
organization = "
deviceType = "Dr
```

```
0
2023-05-23 20:05:43,213
Published Status = 0 to I
2023-05-23 20:05:43,213
2023-05-23 20:05:43,221
0
2023-05-23 20:05:44,046
Published Status = 0 to I
2023-05-23 20:05:44,054
2023-05-23 20:05:44,054
0
2023-05-23 20:05:44,878
Published Status = 0 to I
2023-05-23 20:05:44,878
2023-05-23 20:05:44,886
0
2023-05-23 20:05:45,727
Published Status = 0 to I
2023-05-23 20:05:45,743   ibmiotf.device.Client    INFO    Disconnected from the IBM Watson IoT Platform
2023-05-23 20:05:45,743   ibmiotf.device.Client    INFO    Closed connection to the IBM Watson IoT Platform
0
2023-05-23 20:05:46,561   ibmiotf.device.Client    INFO    Connected successfully: d:ocokk7:Drowsy:7339616
```





Drowsiness Detection

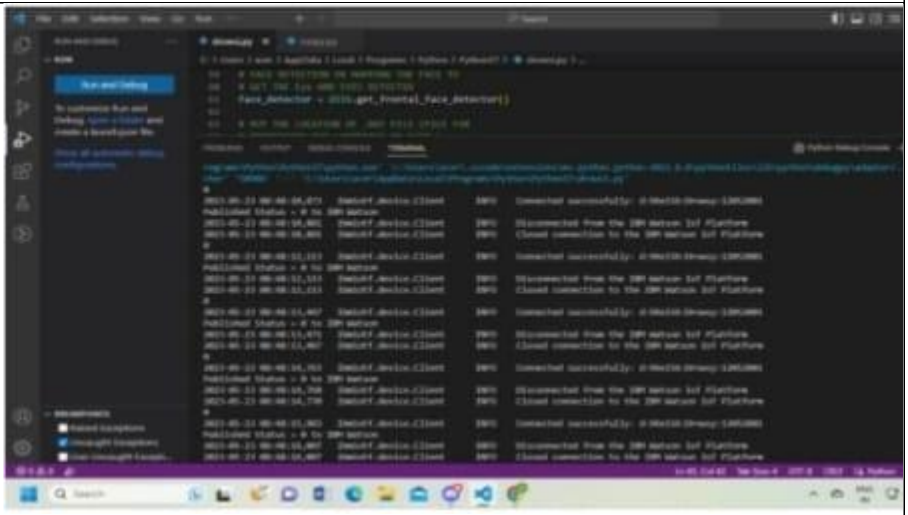
Drowsy

Status **Good to go!**

7.RESULTS:

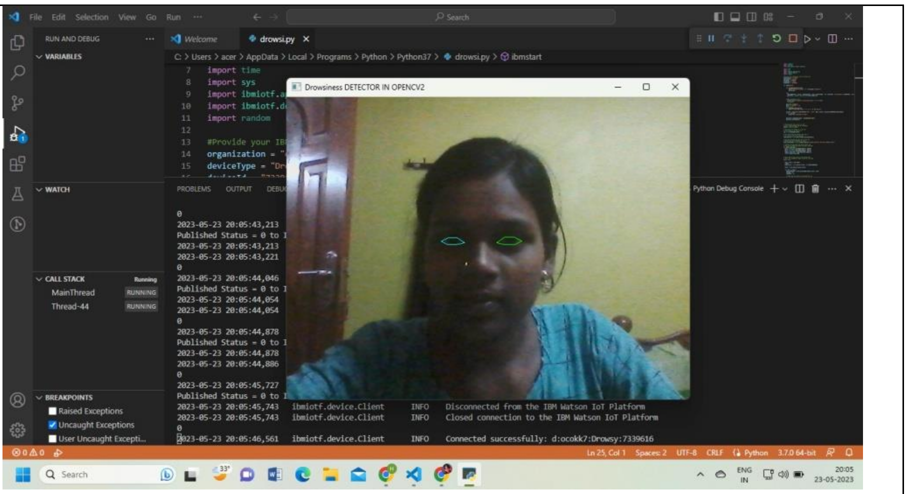
7.1PREFORMANCE METRICS:

Project team shall fill the following information in the performance testing template.

Param eter	Values	Screenshot
Metics	Wokwi Execution time and output screenshot or python accuracy of prediction and output screenshot	

```
face_detector = dlib.get_frontal_face_detector()
```

```
[illegible]    ibmiotf.device.Client    INFO    Connected successfully: [illegible]
Published Status = [illegible] to IBM Watson
[illegible]    ibmiotf.device.Client    INFO    Disconnected from the IBM Watson IoT Platform
[illegible]    ibmiotf.device.Client    INFO    Closed connection to the IBM Watson IoT Platform
```

☐ Raised Exceptions
☑ Uncaught Exceptions
☐ User Uncaught Excepti...

C: > Users > acer > AppData > Local > Programs > Python > Python37 > drowsi.py > ibmstart

```
import time
import sys
import ibmiotf.a
import ibmiotf.d
import random

#Provide your IB
organization = "
deviceType = "Dr
```

```
0
2023-05-23 20:05:43,213
Published Status = 0 to I
2023-05-23 20:05:43,213
2023-05-23 20:05:43,221
0
2023-05-23 20:05:44,046
Published Status = 0 to I
2023-05-23 20:05:44,054
2023-05-23 20:05:44,054
0
2023-05-23 20:05:44,878
Published Status = 0 to I
2023-05-23 20:05:44,878
2023-05-23 20:05:44,886
0
2023-05-23 20:05:45,727
Published Status = 0 to I
2023-05-23 20:05:45,743   ibmiotf.device.Client     INFO    Disconnected from the IBM Watson IoT Platform
2023-05-23 20:05:45,743   ibmiotf.device.Client     INFO    Closed connection to the IBM Watson IoT Platform
0
2023-05-23 20:05:46,561   ibmiotf.device.Client     INFO    Connected successfully: d:ocokk7:Drowsy:7339616
```

8. ADVANTAGES & DISADVANTAGES:

8.1 ADVANTAGES:

- The police can immediately trace the location where the accident has occurred and necessary action can be taken after receiving the emergency message.
- This system can prove to be a lifesaver in isolated areas where an accident has occurred and no one is around in order to report the accident.
- provides wide-area detection when information gathered at one camera location can be linked to another.
- Generally cost effective when many detection zones within the camera field-of-view or specialized data are required.
- It can be used while driving, so that an alarm or a notification can be sent if the driver's eye is closed for more than 2 seconds.
- Other applications, include scientific studies to measure fatigue or can be used for research purposes to count numberof eyeblinks in various different situations.

8.2DISADVANTAGES :

- It does not work with few users who wear contact lenses or have long eye lashes.
- It requires some calibration time before it gives satisfactory results.
- Eye movements of some users are often un-intentiona.
- The live system can't work if any of the following occur at the time of the crash Automatic or phone is disconnected or damaged. No GPS signal at the time of the crash
- Another limitation of SDLP is that it is purely dependent on external factors like road marking, climatic and lighting conditions. In summary, many studies have determined that vehicle-based measures are a poor predictor of performance error risk due to drowsiness.

9.CONCLUSION:

- It completely meets the objectives and requirements of the system.
- The framework has achieved an unfaltering state where all the bugs have been disposed .

- The framework cognizant clients who are familiar with the framework and comprehend it's focal points and the fact that it takes care of the issue of stressing out for individuals having fatigue-related issues to inform them about the drowsiness level while driving.

10. FUTURE SCOPE:

- The model can be improved incrementally by using other parameters like blink rate, yawning, state of the car, etc.
- If all these parameters are used it can improve the accuracy by a lot.
- We plan to further work on the project by adding a sensor to track the heart rate in order to prevent accidents caused due to sudden heart attacks to drivers.
- Same model and techniques can be used for various other uses like Netflix and other streaming services can detect when the user is asleep and stop the video accordingly.
- It can also be used in application that prevents user from sleeping.

11. APPENDIX:

11.1 SOURCE CODE:

```
import cv2
import dlib

import requests

import pyttsx3

from scipy.spatial import distance


import time

import sys

import ibmiotf.application

import ibmiotf.device

import random
```

```
#Provide your IBM Watson Device Credentials
organization = "ocokk7"
deviceType = "Drowsy"
deviceId = "7339616"
authMethod = "token"
authToken = "12345678"

def ibmstart(x):

    def myCommandCallback(cmd):
        print("Command received: %s" % cmd.data['command'])
        print(cmd)

    try:
      deviceOptions = {"org": organization, "type": deviceType, "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()

    deviceCli.connect()
    data = { 'Status' : x}
    #print data
    def myOnPublishCallback():
        print ("Published Status = %s" % x, "to IBM Watson")

    success = deviceCli.publishEvent("DD", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
    deviceCli.commandCallback = myCommandCallback
    deviceCli.disconnect()



# INITIALIZING THE pyttsx3 SO THAT
# ALERT AUDIO MESSAGE CAN BE DELIVERED
engine = pyttsx3.init()

# SETTING UP OF CAMERA TO 1 YOU CAN
# EVEN CHOOSE 0 IN PLACE OF 1
cap = cv2.VideoCapture(0)

# FACE DETECTION OR MAPPING THE FACE TO
# GET THE Eye AND EYES DETECTED
face_detector = dlib.get_frontal_face_detector()

# PUT THE LOCATION OF .DAT FILE (FILE FOR
# PREDECTING THE LANDMARKS ON FACE )
dlib_facelandmark = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

# FUNCTION CALCULATING THE ASPECT RATIO FOR
# THE Eye BY USING EUCLIDEAN DISTANCE FUNCTION
def Detect_Eye(eye):
  poi_A = distance.euclidean(eye[1], eye[5])
  poi_B = distance.euclidean(eye[2], eye[4])
  poi_C = distance.euclidean(eye[0], eye[3])
  aspect_ratio_Eye = (poi_A+poi_B)/(2*poi_C)
  return aspect_ratio_Eye
```

```
# MAIN LOOP IT WILL RUN ALL THE UNLESS AND
# UNTIL THE PROGRAM IS BEING KILLED BY THE USER
while True:

  null, frame = cap.read()
  flag=0
  gray_scale = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

  faces = face_detector(gray_scale)

  for face in faces:
    face_landmarks = dlib_facelandmark(gray_scale, face)
    leftEye = []
    rightEye = []

    # THESE ARE THE POINTS ALLOCATION FOR THE
    # LEFT EYES IN .DAT FILE THAT ARE FROM 42 TO 47
    for n in range(42, 48):
      x = face_landmarks.part(n).x
      y = face_landmarks.part(n).y
      rightEye.append((x, y))
      next_point = n+1
      if n == 47:
        next_point = 42
      x2 = face_landmarks.part(next_point).x
      y2 = face_landmarks.part(next_point).y
      cv2.line(frame, (x, y), (x2, y2), (0, 255, 0), 1)

    # THESE ARE THE POINTS ALLOCATION FOR THE
    # RIGHT EYES IN .DAT FILE THAT ARE FROM 36 TO 41
    for n in range(36, 42):
```

```
  x = face_landmarks.part(n).x
  y = face_landmarks.part(n).y
  leftEye.append((x, y))
  next_point = n+1
  if n == 41:
    next_point = 36
  x2 = face_landmarks.part(next_point).x
  y2 = face_landmarks.part(next_point).y
  cv2.line(frame, (x, y), (x2, y2), (255, 255, 0), 1)

# CALCULATING THE ASPECT RATIO FOR LEFT
# AND RIGHT EYE
right_Eye = Detect_Eye(rightEye)
left_Eye = Detect_Eye(leftEye)
Eye_Rat = (left_Eye+right_Eye)/2

# NOW ROUND OF THE VALUE OF AVERAGE MEAN
# OF RIGHT AND LEFT EYES
Eye_Rat = round(Eye_Rat, 2)

# THIS VALUE OF 0.25 (YOU CAN EVEN CHANGE IT)
# WILL DECIDE WHETHER THE PERSONS'S EYES ARE CLOSE OR NOT
if Eye_Rat< 0.25:
  cv2.putText(frame, "DROWSINESS DETECTED", (50, 100),
        cv2.FONT_HERSHEY_PLAIN, 2, (21, 56, 210), 3)
  cv2.putText(frame, "Alert!!!! WAKE UP DUDE", (50, 450),
        cv2.FONT_HERSHEY_PLAIN, 2, (21, 56, 212), 3)

  # CALLING THE AUDIO FUNCTION OF TEXT TO
  # AUDIO FOR ALERTING THE PERSON
  engine.say("Alert!!!! WAKE UP DUDE")
  flag=1
```

```
      engine.runAndWait()



  cv2.imshow("Drowsiness DETECTOR IN OPENCV2", frame)
  print(flag)
  ibmstart(flag)
  '''
  while True:

                data = { 'Status' : x}
                #print data
                def myOnPublishCallback():
                    print ("Published Status = %s" % x, "to IBM Watson")

                success = deviceCli.publishEvent("DD", "json", data, qos=0, on_publish=myOnPublishCallback)
                if not success:
                    print("Not connected to IoTF")
                time.sleep(1)

                deviceCli.commandCallback = myCommandCallback
  '''
  #r1 = requests.get('https://api.thingspeak.com/update?api_key=SEWZDEK7APG3P0P8&field1='+str(flag))
  #print(r1.status_code)
  key = cv2.waitKey(9)
  if key == 20:
    break

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

```
cap.release()
cv2.destroyAllWindows()
```

GITHUB LINK:

https://github.com/naanmudhalvan-SI/PBL-NT-GP--5668-1680797683

DEMO LINK:

https://youtu.be/4A2t6Y7eNXk