

**An Independent Project Report submitted in partial fulfilment. of the regulations governing the award of the**

**Bachelor of Computer Science (Hons) 2021 Driver Drowsiness Recognization System**

KOMARINA DINESH SAI SUDARSHAN BABU 1001956216

## Bachelor of Computer Science (Hons)(ICSDI),

UCSI University March, 2021

# Authorship Declaration:

Except where reference is made in the references, this report contains no material published elsewhere or extracted in whole or in part from a dissertation or report presented by me for another degree or diploma.

No other person’s work has been used without due acknowledgement in the content of

the report.

|  |  |
| --- | --- |
| Signature: k.d.s babu | Date: 26/03/2022 . |

**Acknowledgement**

# I am overwhelmed in gratitude to acknowledge my depth to all those who have offered me assistance to make these ideas a reality.

I would like to thank my supervisor Asst.Prof. Dr. Mohd Fikree Bin Hassan who closely guided me throughout the project by providing me with all the information and advice I need to develop the system.

I would also like to thank my secondary supervisor Asst. Prof. Ts Dr. Raenu Al Kolandaisamy for providing me guidance in the project.

Thank you,

Komarina Dinesh Sai Sudarshan Babu

**Abstarct:**

Tiredness driving is the leading causes of car fatal accidents on the road. As a result, detecting and indicating driver tiredness is an important research subject. The majority of existing method are vehicle-based, behavioral-based, or physiological-based. Few approaches are disruptive and distract the driver, while others need the use of costly sensors and data processing. As a result, in this work, a low-cost, real-time driver tiredness detection system with satisfactory accuracy is built. A webcam captures the video in the designed system, and the driver's face is detected in each frame using image processing techniques. Facial landmarks on the identified face are highlighted, and afterwards the eye aspect ratio, mouth opening ratio, and nose length ratio are computed, and tiredness is recognised based on these values using established adaptive thresholding. Offline implementations of machine learning algorithms have also been made. Support Vector Machine-based classification attained a sensitivity of 95.58 percent and a specificity of 100 percent.

## TABLE OF CONTENTS:

Authorship Declaration **----------------------------------------------------------------------------> 2**

Acknowledgement ----------------------------------------------------------------------------> 3

Abstract ----------------------------------------------------------------------------> 4

## CHAPTER 1: INTRODUCTION

* 1. Background > 8,9
  2. Problem Statement > 9,10
  3. Aims > 11

1.4 Objectives ----------------------------------------------------------------------> 11

1.5 Scope ----------------------------------------------------------------------> **11,12**

## CHAPTER 2: INFORMATION GEATHERING

2.1 LITERATURE REVIEW **> 13**

2.2 Existing System ------------------------------------------------------> 14

* 1. Disadvantages of Existing system > 14,15
  2. Comaprision of Old systems and proposed system **>** 16,17
  3. Proposed System >17
  4. Advantages of proposed system >17

## CHAPTER 3: ANALYSIS

* 1. Research Methodology > 18
  2. Develop Methodology > 18

3.3 Deliverables ------------------------------------------------------------------------------> 19,20,21

3.4 Constraints and Assumptions -------------------------------------------------------> 21

* 1. Requirements Definition Stage > 22 ,23
  2. Resources > 23,24
  3. Functional Requirements > 24,25
  4. Nonfunctional Requirements > 25,26
  5. External Bodies > 26
  6. Survey Research > 27 to 33
  7. Gantt Chart > 34
  8. Project Plan > 34,35,36
  9. Algorithms used for classification > 37

## CHAPTER 4 : SYSNTHESIS

* 1. Design Stage > 40,41
  2. System Design > 42 to 48

Class Diagram > 42

Use case Diagram > 44

Sequence Diagram > 45

Collabration Diagram > 46

Project State Chart > 47

Project activity Diagram >48

Project component >49

Application Deployment Diagram >50

* 1. Implementation > 61 to 66
  2. Software Overview > 67,68

## CHAPTER 5 : EVALUATION

* 1. Integration & Test Stage > 69
  2. Testing techniques used > 70
  3. Checking All test Cases >70, 71
  4. Test Results and outputs > 71 to 74
  5. Installation & Acceptance Stage > 75
  6. User Acceptence Test and feedback > 76 to 80

## CHAPTER 6 : CONCLUSION > 81

REFERENCES > 81,82

APPENDICES------------------------------------------------------------------------------------> 83,84

**CHAPTER 1 : INTRODUCTION**

## Background

Now a days Drunk and driving is one of the leading problem of car accidents. Drivers who travel for long periods and long-distance are relatable to this condition. In every country passengers face nightmare of drowsy drivers. Every year Drowsiness related traffic accidents result in many injuries and deaths.

So, due to this detecting the drowsiness of driver is very important topic of research. There are three blocks of the basic drowsiness detection system . The Data Acquisition system , Processing System, and warning System.The acquisition system video of driver face is recorded, then it sent to The processing block, there it analyse online and there the drowsiness is detected.The warning system will send a warning to the driver if it detect the drowsiness. generally, there are three methods of detecting drowsy drivers; vehicle-based, physiological ,behavior-Based. In the Vehicle Based Method, a Number of things such as accelerator and braking patterns , steering wheel movement, lateral acceleration , vehicle speed, deviations from the position, etc. are continuously observed. we can considered these values as driver fatigue. As the sensors are not connected to the conductor we can consider this as a non-contact measure. [8-14] In the behavior Based Method, The Visual Behavior of the Driver, ie blinking of eyes, yawning,and Closing the eyes, Bending the head, etc. is considerd for drowsiness. We also consider this as a non-contact measurement as A **simple digital web-cam** is used to recognize the features. Method based on [15,16] on physiological signals such as electroencephalogram (EEG), electrooculogram (EOG),electrocardiogram (EKG), pulse rate and heartbeat, etc. they are monitored and the drowsiness or fatigue is recognized on the basis of these . we consider this as an intrusive measure this method distract the drivers as the sensors are connected to the controller . The cost and size of the system increase based on sensors used in the system. These elements Motivate me to develop an less cost, inexpensive, real-time driver tiredness recongnization system. Therefore, the proposed system has a web-cam Based System To recognize the Driver's drowsiness From The facial Image by Using

only techniques of machine learning and image processing. to develop the system portable and inexpensive.[8-14]

authors learned about a system to identify driver drowsiness with the help of eye

tracking . Here he first looks at the image of the face from the color camera to find the color of the skin and takes a gray image that does not contain a color image, only a white and black color image, then the Canny Edge Detection Algorithm is applied to find the Frames of eye. "After finding eyes, it calculate the white points at the edges every twenty images, and if drowsiness is detected, the driver is warned. [1]

authors presented evidence of non-intrusive drowsiness according to the SVM

classification. Using this method trace a training log with the attributes static values for nose, mouth, and eye and the value for the target column, such as drowsiness or false. it is a non-intrusive sleepiness detection. The application begin with a working color camera that will open and read image, recognize the facial marks from that image, incessantly get 3 thresholds and predict the sleep state, then it will warn . [2]

The authors have invented a video frame-based recognition method for drowsy drivers, in which facial features can be recognized even with glasses.[3]

In previous systems, glasses could not be used to detect facial features. Due to this, it would not be possible to perfectly detect the drowsiness of car drivers, but with this method it is possible to determine the drowsiness of drivers.

## Problem statement

**Main problem**: Drowsiness is the leading causes of death in traffic accidents. Heavy vehicle drivers who drive for more hours (particulary at nights) **with out** rest, bus drivers who travel far distances, or vehicles that travel one day are particularly susceptible to this problem.

**Problems in the Exsisting devices**:

**Problem - 1 : Sensors distract the driver and cost also high**

Systems carried out by primary physiological-based method:

. In the general physiological method, based mainly [8,9], physiological indicators such as the electrocardiogram are monitored (EKG), Electrooculogram (EOG), Electroencephalogram (EEG), heart beat, pulse rate, etc. and, with these things, the level of drowsiness is recognized. This is an intrusive dimension as the sensors are connected to the drive it can Distract the drive. Depending on sensors used within the device, the cost of the device will increase in addition to the length

**Problem - 2 :** Does not give us great results.

Systems performed by vehicle- based method :

In the general method based mainly on automobiles, some of the metrics such as steering accelerator

, brake pedal pattern or wheel movement, speed of the vehicle, Lateral acceleration, pinch roller Deviations, etc.continuously detected and monitered. if it find anything strange it will alert the driver

. As the sensors are not connected this is a nonintrusive measurement. This is also cost high and does not give us great results.

**Problem - 3: Using only eye blink we cannot get excellent success rate** Systems carried out by behavior method based :

In total method based on behavior , the visible behavior of the driver, that is, blink eye, closing eye

.are scanned for drowsiness .This is also a non-intrusive dimension as a simple digital camera is used to detect these characteristics. Only when a blink and an eye are detected for the last time, we may not get the best performance rate when detecting drowsiness. [8-14]

## Aim:

Using the machine learning techniques and visual behaviour methods Proposing a web-cam-based system is developed to identify driver’s tiredness from the face image only to make the system inexpensive and portable.

* 1. **Objective:** To resolve All the above troubles we've proposed a web-cam based device to come across driver's fatigue from the face photo simplest the use of photo processing and device studying

strategies to make the device

less cost portable and

don’t distract the driver.

The Video is recorded in a webcam on this system and image processing techniques are used to recognize the drivers face in each frame. After identifying the driver's face, it begins to detect facial features nose, eyes, mouth on the identified face to calculate the aspect ratio of eyes, the nose length ratio and mouth opening ratio, and also calculates its thresholds that allow it to be compared with the declared ones. Only our technology can locate driver drowsiness primarily on the basis of these threshold values.

## Scope

By using Webcam & machine learning to implement the real time driver drowsiness figuring out system with a appropriate accuracy.

**Purpose: -** To identify the **driver’s** drowsiness recognization using machine learning**:**

* + - Data Acquisition
    - Face Detection
    - Facial Landmark marking
    - Identifying the eye blinking
    - Identifying of eye closing
    - Identifying of yawning
    - Identifying of head bending

## CHAPTER 2: INFORMATION GEATHERING

* 1. **LITERATURE REVIEW**

In this research I am very much concern about the road accidents occurring due to the drowsiness of driver. Below are works done by others.

Mandalapu Sarada Devi and Dr. Preeti R. Bajaj (2008), In their research on driver fatigue, the number of traffic accidents per year was recorded according to internal sources. Therefore, a system that detects driver fatigue in advance and generates the warning may be necessary to prevent many traffic accidents, save money and reduce personal suffering . Therefore, in this system, the authors have developed a system in which the camera can be placed in front of the driver's seat to easily detect fatigue. If the driver is determined to be overly tired while driving , this system can immediately warn the driver. In this system they used video documents recorded by cameras, then frames were Extracted from those video documents and the eye area was tracked to compute the intensity distances between the open eyes and closed eyes. If closed eyes are identified back to back for a few frames, this system will decide that driver is asleep and warn the driver.[4]

Matthew Sacco and Reuben A. Farrugia (2012), Traffic accidents mainly occur when drivers get tired. Therefore, this system implemented the Driver Drowsiness Detection System, which can recognize the driver's facial expression to warn drowsy drivers. To obtain the driver's facial characteristics, this system was used with the ViolaJones method, as well as a template matching technique to obtain the status of the characteristics of Each frame. Furthermore, this system also implement an SVM classifier to categorise the driver 's facial appearance as fatigued or not-fatigued.[6]

R. Ahmad, and J. N. Borole (2015), In their study, the numerous safety and surveillance packages are used to generate signals when a theft occurs. Therefore, for nonintrusion, a machine-based approach is proposed to recognize driver fatigue by generating warning signals. This is where the webcam is placed, where the head movements of driver's can be recorded to detect drowsiness. This system works by recognizing the face, mouth and eyes from an important segmentation of the image. Also, drowsiness depends on the blinking of the driver's eyes. The author implemented a cascading object recognition algorithm to recognize the face, mouth, eyes, and nose from an image captured by a webcam. An ROI methodology was also developed to recognize rectangular shapes of facial descriptors such as the face,

mouth, eyes, and nose. Finally, driver drowsiness is recognized based on the open and closed state of the eye.[5]

Karchaniet and A. Mazloumi (2015) The detection of drowsiness was proposed using image processing techniques and the MLP neural network algorithm. This research was tested on 25 drivers and recorded 3,000 images of each driver with the camera positioned in front of the driver. These frames are enhanced as if converted to a grayscale image and perform the Viola and Jones cascading method to detect sleepy faces. Based on an image with a recognized face, you can extract the eye from this image and send it to the system. Subsequently, histograms and binary methods are used to obtain feature extraction. The MLP classifier is trained with these functions and tested on a 30 percent test data set. Ultimately, this research was evaluated with 93 percent accuracy.[7]

* 1. **Existing System :**

In the present system, it is noted that the recognition of driver drowsiness includes monitoring events due to unconsciousness by eye blinking. In this case, a blink sensor is connected to the vehicle. Also in some systems they used sensors , which alerts the driver in case of loss of consciousness via the buzzer, to prevent vehicle accidents. In the future, we can implement a drowsiness detection system on aircraft to be alert.

* 1. **Disadvantages of the existing system:**
     + When using the IR sensor or eye blink , we cannot get the results of the outputs
     + We are not using any algorithms

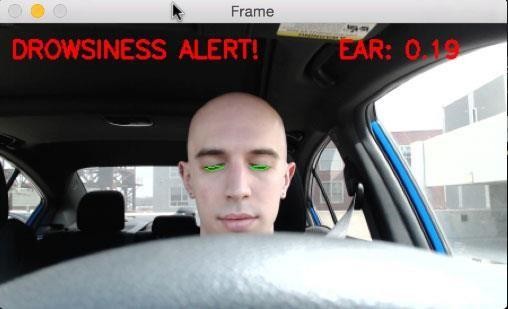


Fig -1 (15)

Existing system detecting eyes blink or eyes closed

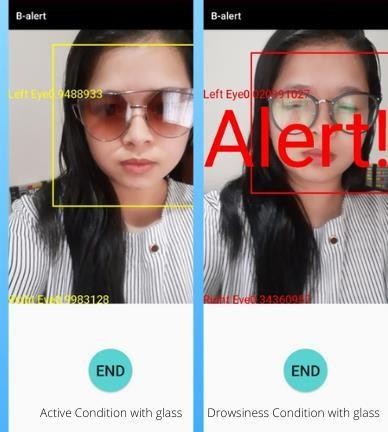


Fig -2(16) ;

Existing system detecting eyes blink or eyes closed even driver keep glasses

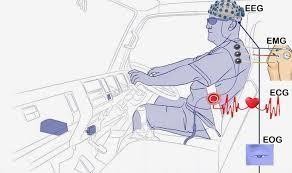


Fig-3(17) :

Existing system detecting drowsiness using the sensors

## Comaprision table of Old systems and proposed system:

|  |  |
| --- | --- |
| **Old system** | **Proposed system** |
| High cost | Low-cost |
| Distract driver | Don’t distract driver |

|  |  |
| --- | --- |
| Less success rate | Will be High success rate |

* 1. **Proposed System:**

Real-time,A less cost motive force drowsiness detection device with proper accuracy will be created in the proposed device. As a result, we have got offered a webcam- based device that makes use of photo processing and machine learning getting to discover fatigue from a face photo. To hold the device low-cost and portable, the digital digicam can be placed in the front of the driver to gather the the front face photo. To create 2-D pictures the frames are taken from the video. Eye factor ratio, mouth establishing ratio, and head function are all calculated from facial landmarks, and conclusion regarding the driver drowsiness is made utilizing these functions and a machine learning approach. If drowsiness is identified, an warning alarm can be sounded to inform the driver.

## Advantages of proposed system

* + - Using Webcam we can do live
    - We are using drowsiness detection , nose length ratio , visual behaviour, mouth opening ratio, eye aspect ratio
    - Easely we can detect the driver drowsiness and we can alert a driver through alarm sound effect. So with the proposed system we save lives of people.

## CHAPTER 3 : ANALYSIS

* 1. **Research Methodology :**

Research methodology in this project is quantitative methodology. In quantitative data collection, the survey will be spread to popular social media to collect information and generalized the information collected. How and why questions from a group of people should be obtained. The survey will be carry out using Google Form and will spread via the WhatsApp, FaceBook, Instagram, TheCN, and WeChat.

## Devlopement Methodology

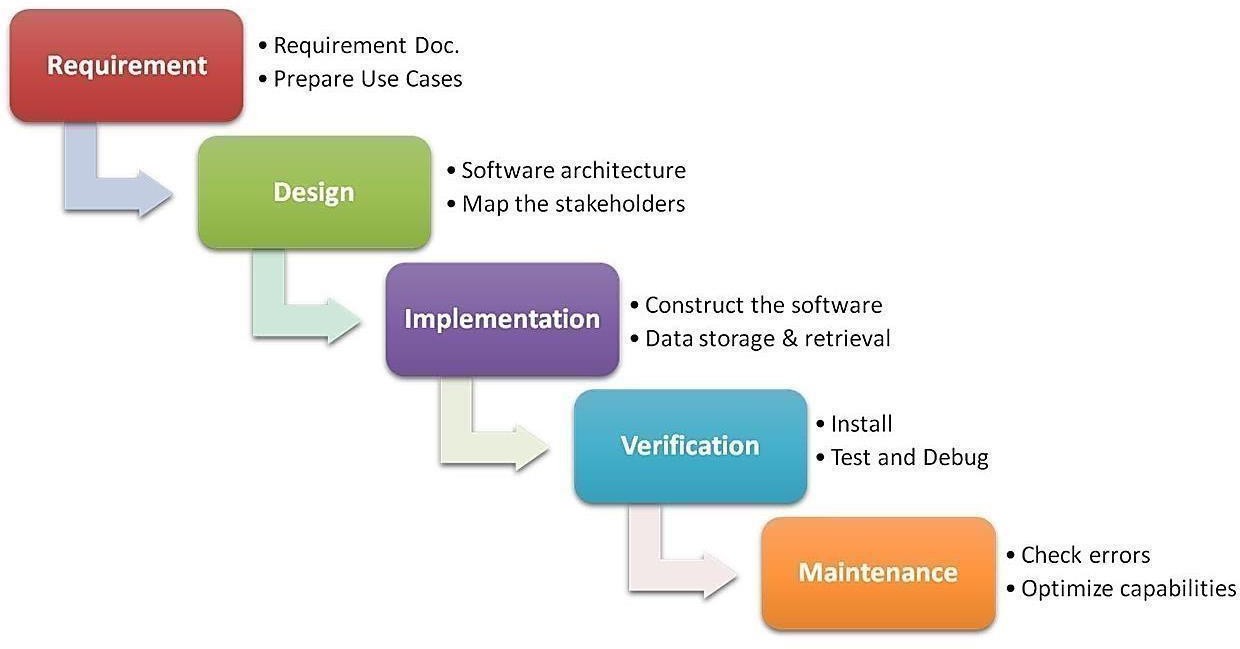


Fig 4 : Development Methodology

**Description**

The waterfall model is a linear sequential flow where the software implementation progress is viewed as a steady flow (like a waterfall) through the phases, indicating that each step in the development process cannot begin until the previous one is completed. The method does not specify how to revert to

an earlier stage to accommodate the changes in requirements. The cascade method was one of the first approaches in software development.

## Deliverable

**Planning Phrase**

* + - Determine the project title, scope, aims, and objectives
    - Gantt chart
    - Research on similar applications
    - Project Proposal

## Requirement Analysis Phrase

* + - Survey form
    - Survey result
    - All possible requirements

## system Design Phrase

* + - UML diagrams
    - System area model
    - System analysis and outline

## Development Phrase

* + - System GUI
    - Construct system
    - Test each unit & integrate
    - Product prototype

## Testing Phrase

**Unit test :**

Unit testing is a software development process that examines the Applications smallest testable parts

, known as units, independently and individually for proper operation. The software developers during the development process use this testing methodology.

**MANUAL Testing :**

Since we are doing is a academic level project , we are unable to perform any automated testing; therefore, we must rely on manual testing through trial and error. methods.

* + - Conduct user acceptance test
    - Manual Testing
    - Test the requirements and system design
    - Fix all issues found
    - Present the prototype

## Deployment Phrase

Once the global project, we can get to the distribution of consumer gadgets in international realities during his academic leave, we have made the distribution in our university laboratory more efficient with all the necessary software with the Windows operating system

* + - Available the installation of application
    - Deploy application to live environment

## Maintenance

Maintaining our project is a one-time process handiest

* + - Receive and analyze feedbacks from the users
    - Fix the bugs reported by user
    - Release patch & update the application

## Constraints and Assumptions

*Constraints*

1. The product is only a prototype.
2. There is no an actual server and database.
3. All the data is made for the presentation of prototype only.

## Assumptions

1. Most of the people using technology.
2. Most Accidents occurring due to drowsiness
3. In this world most people use the cars
4. Every driver must drive safe
5. Every driver need a technology to help him in difficult situation.

At End we can help the driver to drive safely with this technology

## Requirements Definition Stage:

The objectives defined in the project plan's high-level requirements section are fed into the requirements gathering process. Each goal will be broken down into one or even more requirements. These requirements outline the anticipated application's principal functions, as well as the operational and relatable content areas and the first data entities. The vital operations to be handled, and also mission-critical intakes, outcomes, and reports, are all key functions. These primary functionalities, data fields, and entity data are coupled with a system of user classes. A requirement is the name given to each of these definitions. Needs are specified by specific need identifiers and must include a title and a textual description at the very least.

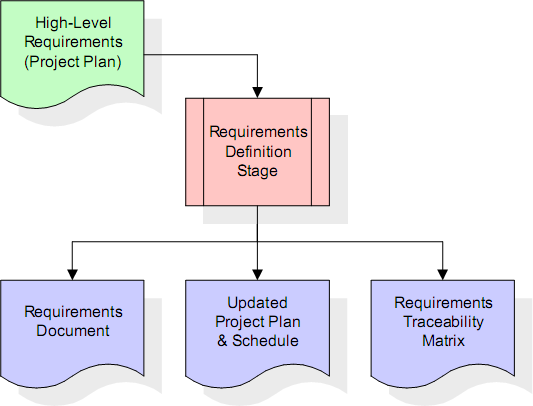


Fig 5 : Requirements Definition Stage

The requirements paper and the requirements traceability matrix are the key deliverables of this phase, and they detail these needs (RTM). The requirements document includes detailed explanations of each need, as well as diagrams and, if necessary, hyperlinks to external publications. The requirements document does not offer specific listings of database tables and fields. The title of each need, as well as the name of each planning phase objective, is contained in the first version of the RTM. RTM's purpose is to demonstrate that product components created throughout each step of the software development process are formally related to components created during prior phases.

The RTM for the requirements phase contains a list of high-level requirements, or goals, organised by title, as well as a list of accompanying requirements for each objective, organised by title. Each need produced during this phase is officially tied to a particular product objective, as shown in this hierarchical list by the RTM. Each demand may be linked to a specific product goal in this style, which is why the phrase requirements traceability was coined. The requirements document, gtr, and a revised draught plan are among the results of the needs definition phase.

## Resources

**SYSTEM CONFIGURATION:**

**Hardware requirements:**

Processer : Any Update Processer

Ram : Min 1 GB

Hard Disk : Min 100 GB

## Software requirements:

Operating System : Windows family Technology : Python 3.6

IDE : PyCham

UML : Star UML

DFD : DFD Drawer

## Functional requirements

* **On webcam**
* **Data Acquisition**
* A webcam is used to record the video, and the images are retrieved and processed on a laptop. Image processing techniques are performed to these 2D images after the frames have been extracted. Data for synthetic drivers is being created. Participants are asked to blink, close their eyes, yawn, and tilt their heads while watching the webcam. The video has been recorded.

## Face Detection

Human faces are discovered initially after the frames have been extracted.

## Facial Landmark marking

* The following step is to locate the various facial features, such as the corners of the eyes and mouth, the tip of the nose, and so on, after detecting the face. Face photographs should first be normalised to eliminate the effects of camera distance, inconsistent lighting, and changing image resolution.

## Alarm Module

The acquisition system, processing system, and alert system are the three blocks/modules of the basic sleep detection system. A front face live camera of the driver is taken in the capture mode and sent to the processing block, where it is analysed online to detect tiredness. The alarm system will issue a warning or alarm to the driver if drowsiness is detected.

## Nonfunctional requirements Usability:

**Based on usage trends, it prioritises key system functions. Complex and important functions, as well as frequently used ones, should be tested for usability. Make sure you include this as a prerequisite.**

**Reliability:**

Even after utilising the system for a long period, people must have faith in it. without the system tampering with the data It's a good idea to incorporate requirements that make system performance easier to track.

## Performance

What will be the system response times, evaluated from every point and under what conditions? Are there any specified peak hours when the system's load will be unusually high? Consider moments of stress, such as the month ending or when it's time to pay payslips.

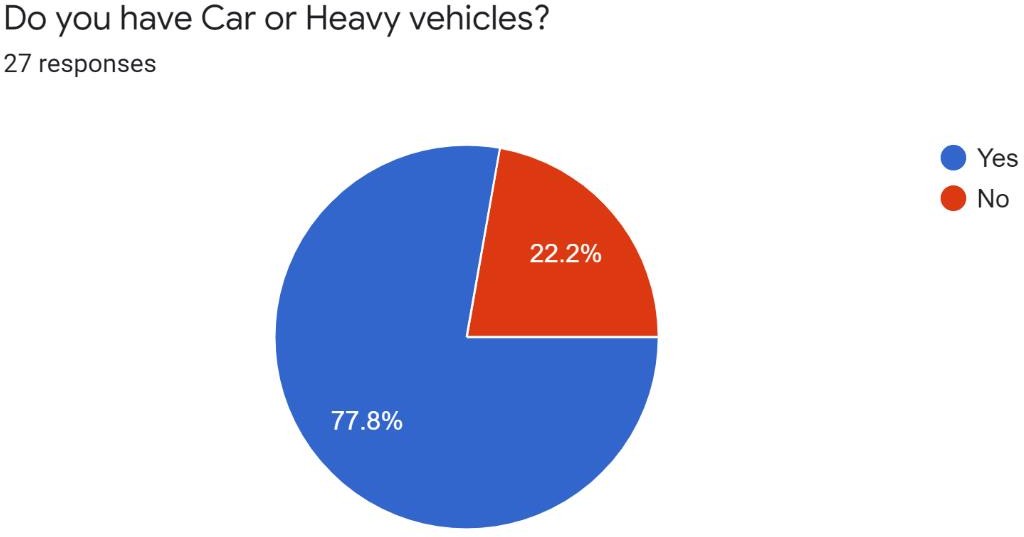
## Supportability

The cost of maintaining system must be reasonable. Various levels of document, such as system documentation and test documentation, such as what tests and testing process will follow the system, can be covered by maintenance requirements.

## External Bodies

There is no external body involved in this project currently.

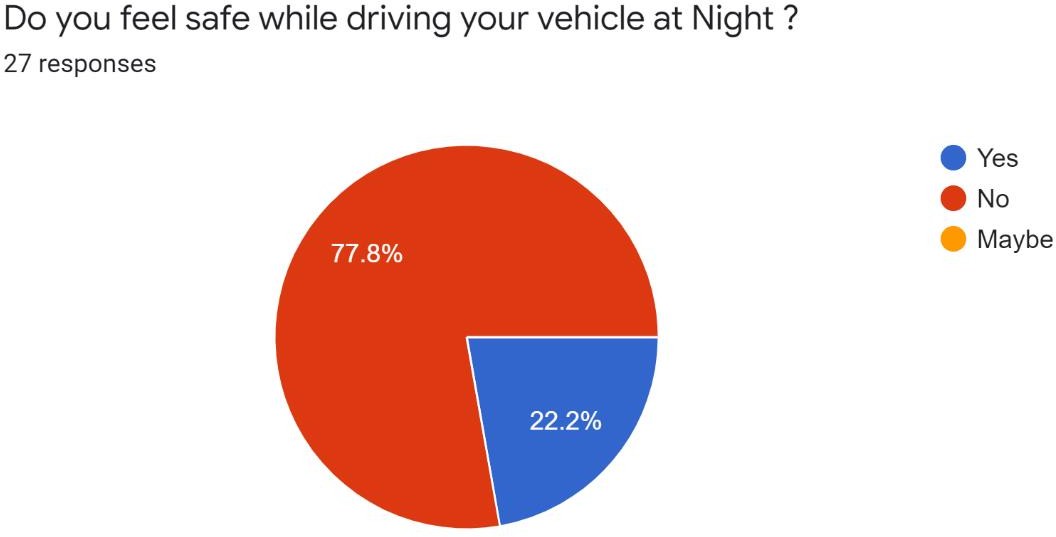
## Survey has been conducted Among students and other members:



In the Total respondends of the survey we have get 77.8% people said that they are having car/heavy vehicles. So I finally by this question I

concluded that implementing the low-cost drowsiness detection system using webcam.Is very important for the people who are driving cars/other heavy vehicles.

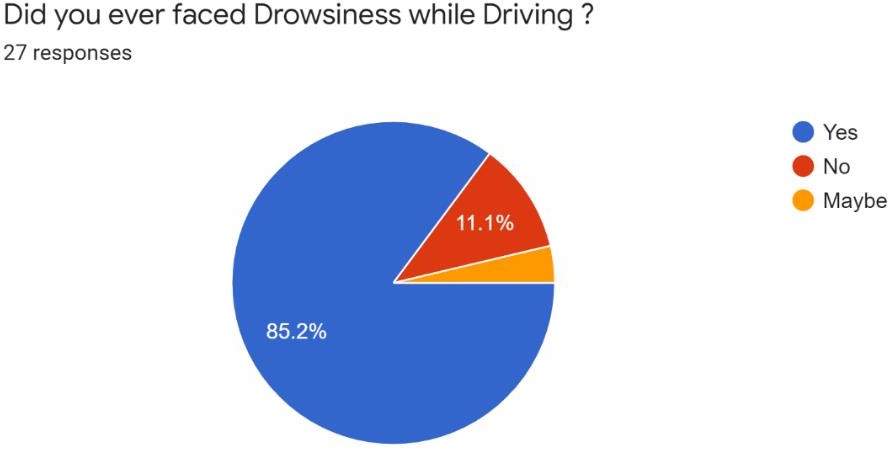
Question 2



Second question that was asked that will they feel safe while driving the car at night/late night times majority of they said that they won’t feel safe while driving car/other vahicles at night/late night time.What ever maybe

the reason the drowsiness is the common reason for accidents and in other unsafe conditions occurred while driving.So I finally by this question I concluded that implementing the low-cost drowsiness detection system using web cam. Is very important for the people.

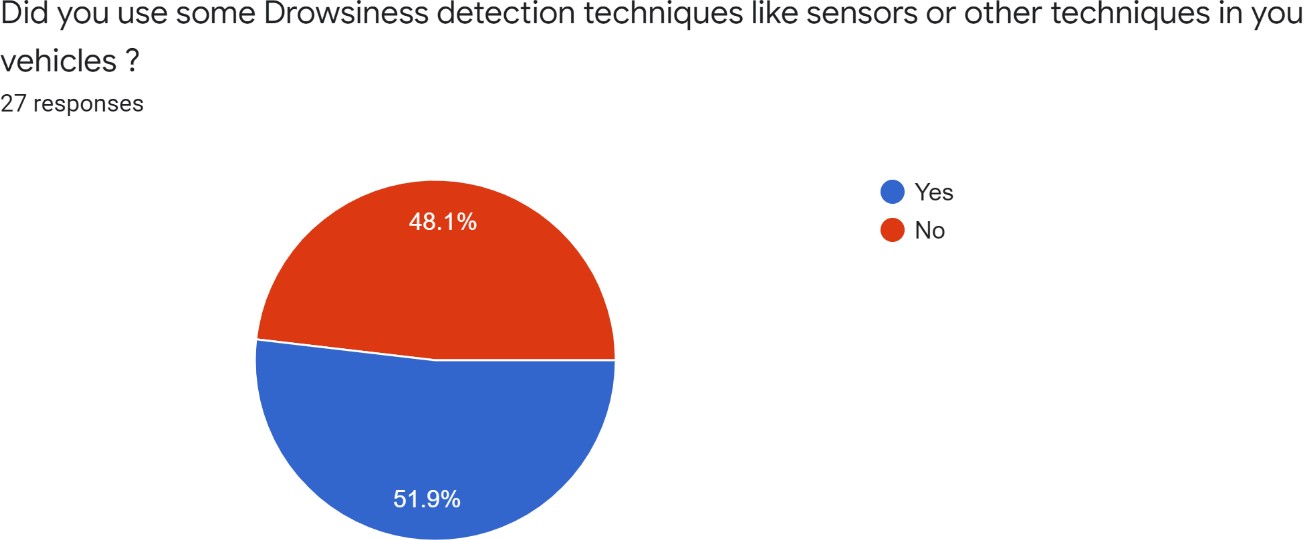
Question 3:



So most of the respondents faced the drowsiness while Driving so now a days this is a big issue for people expecially for late night drivers and heavy drivers so I finally by this question I concluded that implementing

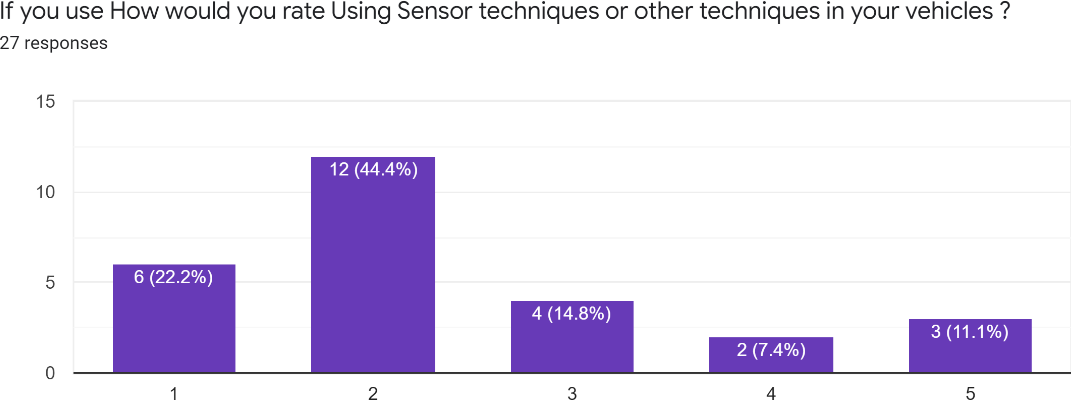
the low-cost drowsiness detection system using webcam. Is very important for the people.

Question 4 :



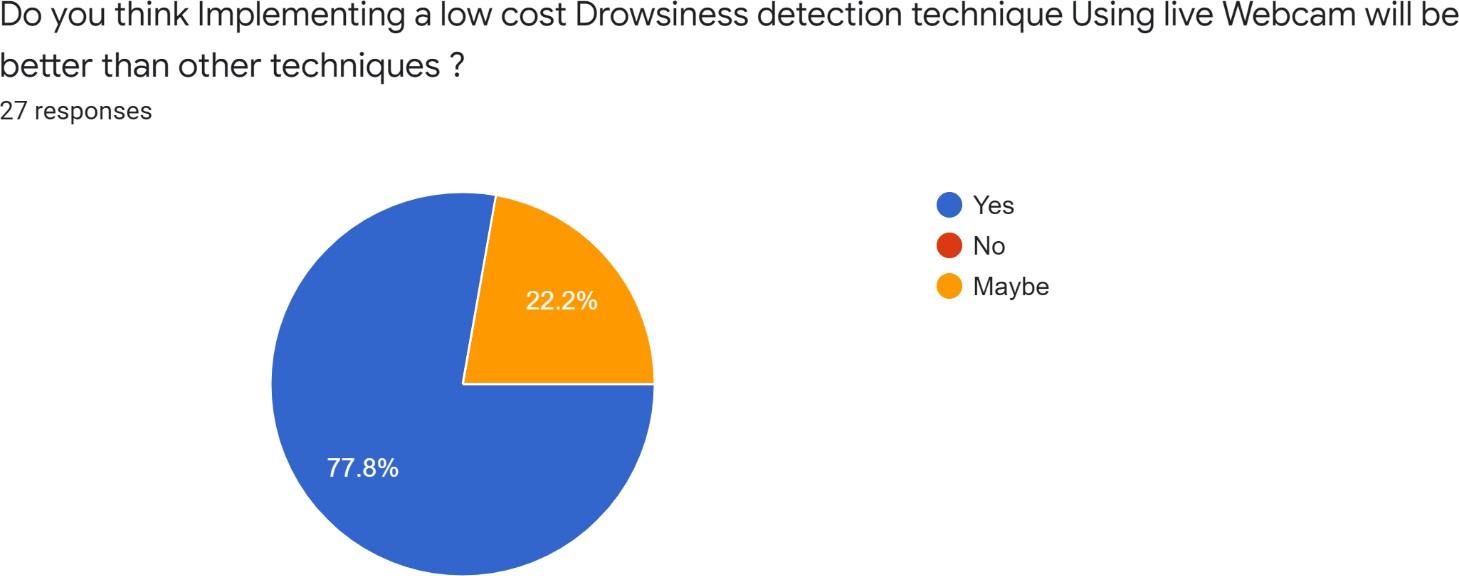
For this question Some people tell that they use other old systems to detect drowsiness.

Quetion 5



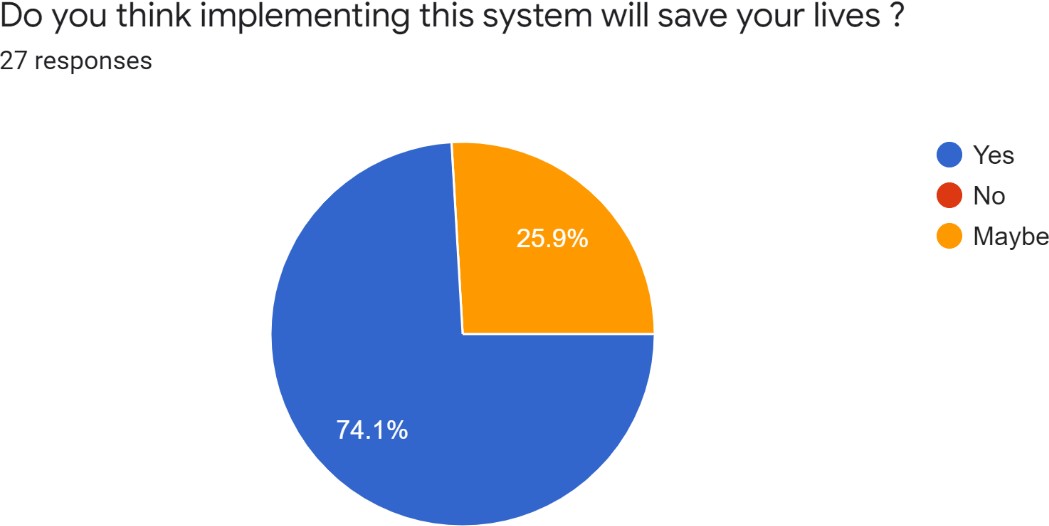
Respondents said that using old drowsiness detection techniques have some drawbacks.Majority of the respondents given “2” rating for old systems they used. So I finally by this question I concluded that implementing the low-cost drowsiness detection system using webcam. Is very important for the people.

Question : 6



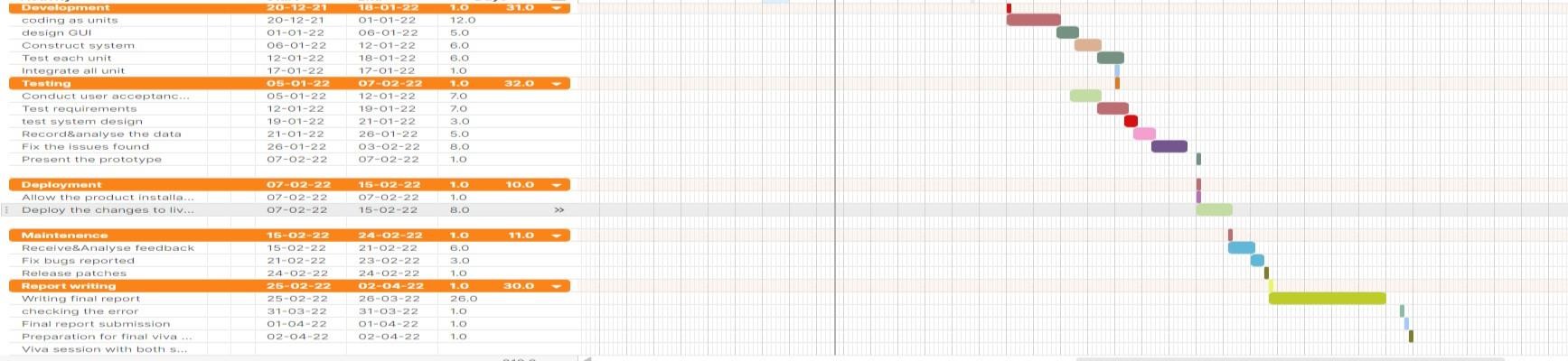
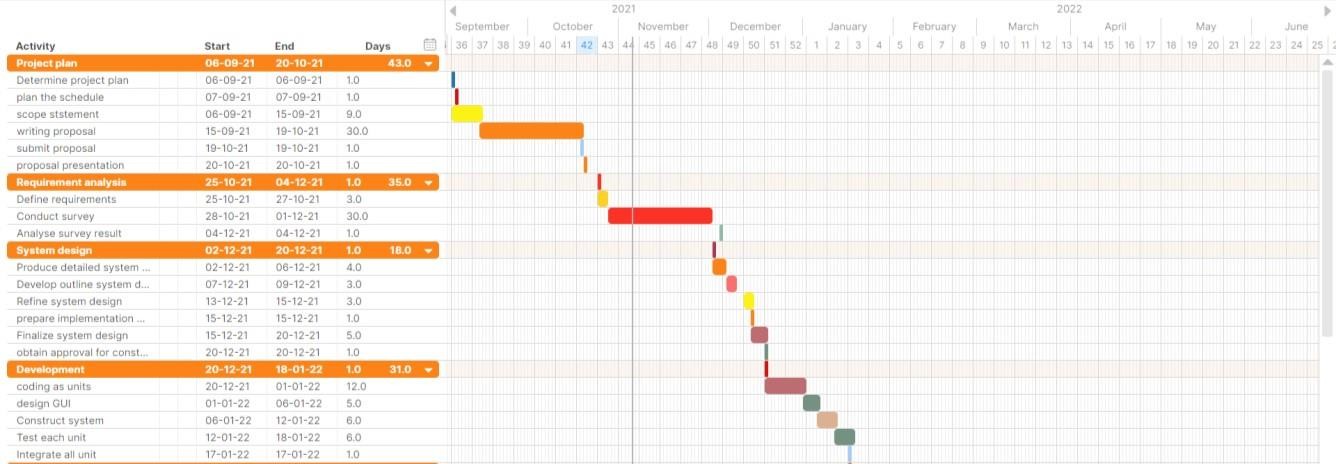
Most people said yes to this question they are believing that implementing low-cost drowsiness detection technique using webcam will be a good techniques to detecting drowsiness. So I finally by this question I concluded that implementing the low-cost drowsiness detection system using webcam. Is very important for the people.

Question 7 :



Most people belive this system will save humans lives. So I finally by this question I concluded that implementing the low-cost drowsiness detection system using webcam. Is very important for the people.

## Ghant chart



* 1. Project Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Name** | **Start Date** | **End Date** | **Duration** |
| **Planning** | | | |
| Determining the Project Title | 6/9/2021 | 6/9/2021 | 1 day |
| Plan the schedule | 7/9/2021 | 7/9/2021 | 1 day |
| Identify the scope | 6/9/2021 | 14/9/2021 | 7 days |
| Writing the Proposal | 15/9/2021 | 21/10/2021 | 25 days |

|  |  |  |  |
| --- | --- | --- | --- |
| Submitting the Finalized Proposal | 22/10/2021 | 22/10/2021 | 1 day |
| Proposal Presentation | 25/10/2021 | 25/10/2021 | 1 day |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirements Analysis** | | | |
| Define Requirements | 26/10/2021 | 28/10/2021 | 3 days |
| Conduct Survey | 29/10/2021 | 30/11/2021 | 32 days |
| Analyse Survey Result | 1/12/2021 | 1/12/2021 | 1 days |
| **System Design** | | | |
| Produce Detailed System Area Model | 2/12/2021 | 5/12/2021 | 2 days |
| Develop Outline System Design | 7/12/2021 | 8/12/2021 | 2 days |
| Refine System Design | 9/12/2021 | 12/12/2021 | 2 days |
| Prepare Implementation Strategies | 12/12/2021 | 26/12/2021 | 4 days |
| Finalize System Design | 15/12/2021 | 20/12/2021 | 5 days |
| Obtain Approval for Construction | 20/12/2021 | 21/12/2021 | 2 days |
| **Development** | | | |
| Coding as units | 20/12/2021 | 17/1/2021 | 20 days |
| Design GUI | 1/1/2021 | 7/1/2021 | 5 days |
| Construct system | 5/1/2021 | 11/1/2021 | 5 days |
| Test each unit | 7/1/2021 | 14/1/2021 | 6 days |
| Integrate all unit | 14/1/2021 | 17/1/2021 | 2 days |
| **Testing** | | | |
| Conduct user acceptance test | 17/1/2021 | 25/1/2021 | 7 days |

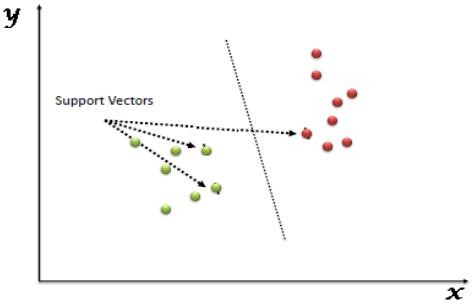
|  |  |  |  |
| --- | --- | --- | --- |
| Test requirements | 17/1/2021 | 18/1/2021 | 2 days |
| Test system design | 19/1/2021 | 20/1/2021 | 2 days |
| Record & Analyse the data | 21/1/2021 | 25/1/2021 | 3 days |
| Fix the issues found | 26/1/2021 | 8/2/2021 | 6 days |
| Presenting the prototype of the application to supervisor | 3/2/2021 | 3/2/2021 | 1 days |
| **Deployment** | | | |
| Allow the product installation | 4/2/2021 | 4/2/2021 | 1 days |
| Deploy the changes to live environment | 4/2/2021 | 14/2/2021 | 7 days |
|  |  |  |  |
| **Maintenance** | | | |
| Receive & Analyse feedback | 15/2/2021 | 22/2/2021 | 6 days |
| Fix the bugs reported | 15/2/2021 | 22/2/2021 | 6 days |
| Release patches | 23/2/2021 | 23/2/2021 | 1 days |

|  |  |  |  |
| --- | --- | --- | --- |
| **Report writing** | | | |
| Writing of the report | 24/2/2021 | 26/3/2021 | 23 days |
| Modification and error-checking of the report | 26/3/2021 | 29/3/2021 | 4 days |
| Submission of final report | 31/3/2021 | 2/4/2021 | 1 days |
| Preparation for final Viva session | 1/4/2021 | 2/4/2021 | 1 days |
| Viva session with both supervisors | 2/4/2021 | 2/4/2021 | 1 days |

## Algorithms Used for Clasification :

**Support Vector Machine (SVM):**

Support Vector Machine (SVM) is a widely used machine learning technology that may be used as both a classification and a prediction (given a predefined target variable). Find a hyperplane in the characteristic space that separates the classes for classification. In an SVM model, training data points are represented as points in feature space, which are mapped so that units belonging to various classes that are separated by as wide a range as practicable. After that, the test data points are projected into the same space and categorised based on which part of the boundary they fall on.



## Fig-6: Support Vector Machine (SVM):

**NAÏVE BAYES ALGORITHM**

The Naïve Bayes classifier is based on Bayes' theorem. It has a strong assumption of independence. It is also known as the independent feature model. It assumes that the presence or absence of a particular feature of a class is not related to the presence or absence of any other feature in the given class. The naive bayes classifier can be trained in a supervised learning environment.Use the maximum similarity method. It was worked in complex real-world situations. Requires a small amount of training data. Estimate the classification parameters. You need to determine only the variance of the

variable for each class and not the entire array.Naïve berries are mainly used when the entries are high. It provides output in a more sophisticated form. The probability of each input attribute is indicated by the predicted state. Machine learning and data mining methods are based on naive Bayes classification.

## FEASIBILITY STUDY

In this phase, the project's feasibility is assessed, and a commercial proposal is developed, complete with a high-level project plan and cost estimates. A feasibility analysis of the system that was proposed must be conducted as part of the system analysis. It is to guarantee that the planned system will not be a financial burden to the company. A basic understanding of the primary system needs is required for the feasibility analysis. The following are three important factors to consider while doing a feasibility analysis:

* economic feasibility
* technical feasibility
* social feasibility

## Economic feasibility:

This study is being conducted to ascertain the device's economical impact on the organisation. The corporation has a limited amount of money to spend in the system's research and development. Justification for spending is required. As a result, the finished system came in under budget, thanks to the fact that the vast majority of the innovations used were freely available. The only things that required to be purchased were the customised ones.

## Technical feasibility:

This research is being carried out to determine the system's technological feasibility, or technical requirements. Any system that is created shouldn't even have a large demand on technological resources. As a result, there will be a strong demand for technical resources. Set high expectations for the customer.

The designed system should really have a low demand because it will only require little or no changes to be implemented.

social feasibility

The study's goal is to determine the user's acceptance level of the system. This involves the procedure of instructing the user on how to utilise the system effectively. The users should not perceive the system as a danger, but rather as a requirement. The approaches utilised to educate and acquaint the user with the system determine the acceptence level of users ’ . His self-assurance must be high in order for him to be able to provide constructive critique, which is desirable given that he is the system's ultimate user.

## Chapter 4 : SYNTHESIS

* 1. **Design Stage:**

The criteria mentioned in the authorized requirements specification document are used as an initial input in the design process. As a consequence of interviews, workshops, and/or prototyping activities, a collection of one or even more design features will be developed for each requirement. Functional hierarchy diagrams, screen layout diagrams, business rules tables, diagrams of business process, pseudocode, and a comprehensive entity relationship diagram with a complete data dictionaries are examples of design elements that explain the required software features in depth. These design features are meant to define the software in such depth that it can be developed by experienced developers with little extra input.

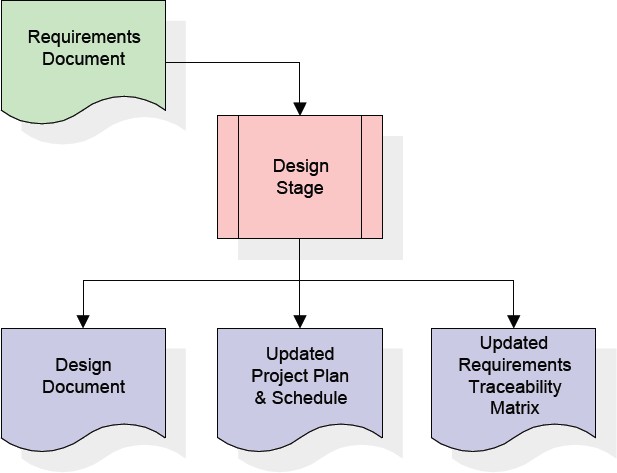


Fig-7 : Design Stage

The RTM is updated once the design document is finalised and accepted to demonstrate that every design element is officially linked to a specific requirement. A design specification, an updated RTM, and an updated design planning process are the deliverables of the design phase.

## Development Stage:

The design features specified in the authorized design specification document are used as major input in the development phase. A set of one or even more software artefacts is created for each design element. Menus, dialogue boxes, and information management forms are examples of software artefacts, as are data reporting formats and specific work steps and functionalities. For each collection of functionally linked software artefacts, appropriate test cases are created, and an internet support system is created to guide users through their interaction with the product.

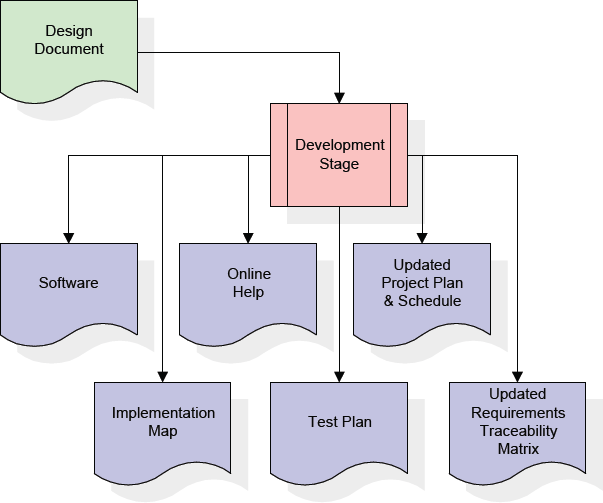


Fig 8: **Development Stage:**

The RTM has been updated into reflect that every created artefact was associated with a specific design feature and that each created item contains one or more legal test elements. The RTM is now in its final configuration. Among the development deliverables are a perfectly functioning combination of programs that meets previous research findings specifications and design features, an internet support network that explains the software's procedure, an implementation chart that identifies important code points of entry including all major structural features, a work determines that explains the test scenarios used to confirm the completeness and accuracy of the software, an up to date RTM, and an updated project plan.

## System Design:

**Class Diagram:**

|  |
| --- |
| **VideoStream** |
| +left\_eye  +right\_eye  +mouth  +nose |
|  |

|  |
| --- |
| **webcam.** |
| +SetUp loading |
| +start() |

|  |
| --- |
| **DROWSINESS** |
| +Head Bending  +Yawning |
| +ALERT() |

**Fig 9:-Project Class Diagram**

**Description:**

In object-oriented modelling, class diagrams are the most important component. They're used to display the numerous objects in a system, as well as their relationships , Operations, and Attributes. We have a system. the upper rectangle has the class's name; the center rectangle contains the class's properties; and the bottom rectangle contains the class's methods, commonly known as operations.In our system webcam class consist of setUploading as a attribute and method to start(). Videostream class consist of lefteye,

righteye, mouth , nose as a attributes and in Drowsiness class consist of Headbending , Yawning , eyeclosing as a attributes and method of Alert().

In the unified modelling language we employed in our project development, a class diagram is used in pc coding designs. A star (UML) is a static shape outline that depicts the shape of a device, displaying the system's characterizations, attributes, activities (or techniques), as well as associations and groupings.

Explain how statistics are improved by elegance.

Our Application on Use Case :

System

**webcam**

**loading face**

**landmark predictor**

**left\_eye**

<<include

e>>

**right\_eye**

>

**nose**

**DROWSINESS ALERT**

**mouth**

<<include>

<<include>>

<<includ

>>

**video stream**



**user**

## Fig 10:- Project User Case

**Description:**

Use case diagrams depict a system's high-level functionality and breadth. The interconnections between both the system and its actors are also depicted in these diagrams. In use case diagrams, use cases and actors define what the system performs and how the actors interact with it. When the user activates the

camera in our system, the system automatically begin recognising facial landmarks and, if drowsiness is detected, it will send us an alert.

A use case profile in the Star(UML) is the unified modelling language that we used in the development of our project.can be a kind of behavior diagram represented and produced using a case review of use. Its thinking is to provide a graphical representation of the presence of machine-delivered thoughts to volume-acting professionals, their goals (treated as use cases), and any conditions between those instances of use. Most of the explanation on the back of a Use Case Diagram is To reveal which form boundaries execute which character on the screen. The parts of the artist in the gadget can be a Diagram.

## Sequence:



webcam loading face

landmark predictor

video stream mouth

right\_eye

left\_eye

nose DROWSINESS ALERT

: user

1 : ON()

2 : Auto()

3 : Auto()

4 : Auto()

5 : Auto()

7 : Result()

9

6

8

10

11

12

**Fig 11:-Project Sequence**

**Description:**

A sequence diagram is similar to an interaction diagram in that it describes in what order and how a group of items interact with one another. diagrams are used by software programme developers and business specialists to identify requirements for a new system. When person activate the Camera robotically it's going to begin detecting the facial landmarks and while drowsiness located in mouth it's going to go through inner system and it's going to deliver us a few alert and identical with eye and head bending.

## Collaboration diagram

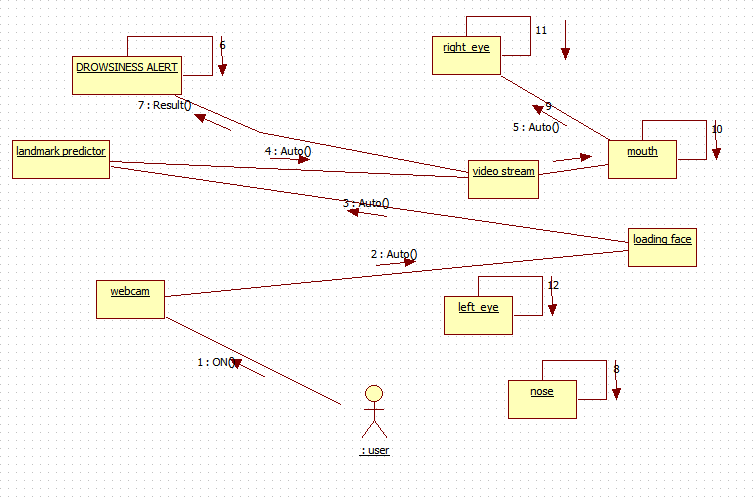


Fig 12: **Collaboration diagram**

Description:

A Collaboration Diagram, **additionally referred to as** a **communique** Diagram,it is an **example** of the

relationships and Interconnections **amongst software program gadgets withinside the** unified modelling language (UML).

## Fig 13:- Project State Chart

**Description:**

The state diagram graphic depicts the flow of control through one state to the next. The state of an object is the state in which it exists and change as an event happens.

## Activity of Project :



webcam

loading face

landmark predictor

video stream

nose

right\_eye

mouth

left\_eye

DROWSINESS ALERT

imlane1

Sw

**Fig 14:-Project activity Diagram**

**Description:**

An activity diagram represents a series of actions or a flow of control in a system similar to a data flow diagram or flowchart.This is the Activity diagram of our System. At first when we enter into the vehicle we will switch on the web camera attached Infront of driver and when we start camera it will start detecting driver lefteye , righteye , mouth , nose and if it detect any drowsiness like eyeclosing , yawning

, or head bending it will give us some Alert by Alarm Sound.

## component:

numpy

imutils

imutils.video

scipy.spatial

**Fig 15: project component**

**Description**

A parts diagram, also known as an AN UML section diagram, depicts the body division affiliation and wiring in a complete picture. Phase graphs are generally intended to facilitate the exposition of leisure utilization factors and to verify that each part of the frame's desired capabilities is guaranteed by a prepared development. Within the main fashion of UML, the elements incorporated into these graphs were bodily: data, tables of records, files and executables, each bodily element having a place. In the world of UML, a few of these elements are not maximal bodily, but rather a number of solitary calculated structural components, for example, a business process that provides or expects interfaces to communicate in totally unique ways. develops in the frame.

## Deployment Diagram :

**System**

**Pycham**

**Fig 16:- Application Deployment Diagram\**

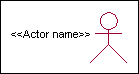
DIAGRAMS IN UML

The requirements of the System, architecture of System and Subsystem , Operating environment , formats of input, database design and files, layouts of output, detailed design, interfaces of human-machine, logic processing , and in the Document of System Design all external interfaces are described.

## Diagrams of Global Use Cases:

**ACTOR's identification:**

**ACTOR:** The role a user plays with respect to the system that is represented by Actor. Actor interferes with the use cases , however, it had no Control on them.

GRAPHICAL illustration::

<<Actor name>>



Actor

## Actor is the one that:

Interacts with or it makes use the system.

* Data is entered into the system and received from it.
* Is not a component of the system and had no impact on the use cases Actors are identified by inspecting:
* Who is the primary user of the system?
* Who is responsible for maintaining the system up - to - date?
* Any external hardware used by the system; and
* Some another systems that must interact with the system.

## To identify actors, ask the following questions:

* Who is utilising the system? To put it differently, who is the system affecting? Or, what group need the support of the system to execute a task?
* Who has a say in the system? Alternatively, which user groups are necessary by the systems to perform its functions? Administrative functions, for example, can be direct or indirect.
* Which external systems or hardware (if any) do you use to perform system tasks?
* Which issues does this application answer (and for whom)?
* And, finally, how do users engage with the system (use case)? what and How do they want to achieve with this system?

In this system, following Actors have been identifid:

## Administrator of system

1. **customer**
2. **customer Care**

Identification of usecases:

**Usecase:** A use-case is a specific manner of utilising the system from the standpoint of a actor(user).

## Diagrammatic represent:



**A detail description of a use-case might be as following:**

* The general pattern of behaviour of the system
* A collection of interrelated transactions performed by an Actor as well as the system.
* Provides something of value to the performer

## Use cases make it possible to

* provide the system requirements
* communicate with-end customers also with specialists
* keep the system through its paces

Evaluating the actors and specifying whatever the actor would be able to accomplish with that system is the good way to find use-cases.

## The following are some guidelines for identification of use cases:

* Identifying the duties and responsibility that every actor must be capable of doing or that the system needs of the actor. The use-case must describe a series of events that lead to a specific goal.
* Assign a name to each use cases.
* Using terminology that the user is familiar with, explain a use case briefly.

This lessens the ambiguity inside the description.

The following are some questions to ask when looking for use cases:

* What were each actor's responsibilities?
* Can everyone in this system create, store, update, delete, or read data?
* Under what conditions will this information be retained, modified, erased, or read?
* Will any actor be required to notify the system of unexpected external events?
* Would any actor have to be informed about certain system events?
* Which use-cases will the system support and maintain?

## The Sequence of Activities

A flow of events is a succession of actions (events) performed by the system. They typically contain a large amount of information that is presented in terms of what the system should do rather than how it actually accomplishes it. The action sequence is created as a separate file or document in your preferred word document, then connected or linked to a use-case via the Files page of a data structure.

## The following items should be included in a sequence of events:

* When does the use case begin and conclude, and how does it do so?
* Use the interplay between the case and the Actors to your advantage.
* Data needed by the use case
* The use-case's typical sequence of events
* Abnormal or divergent flows

## The following steps are involved in the construction of usecase diagrams:

Use-case diagrams graphically show a system's behaviour (use cases). These Diagrams provide a high-level view of how well system is being utilised from the standpoint of an outsiders (actor).A Use-Case diagram can depict all or a subset of the use cases in a system.

## The following items may be included in a use-case diagram:

* Actors
* Use Cases
* Connections or linkages between actors and use cases in the system, including associations, dependency, and assumptions.

## In use cases, there are a variety of relationships to consider.

* 1. **Communication is number one.**

A solid path links the actor sign to the use-case sign to represent an actor's relations in a use-case. The actor, as per the actor, interacts to the use-case.

## Uses:

The usecase's generalisation arrow displays a Uses link among use-cases. 3.**Extends a concept:**

We employ the extended relationship if we have one usecase is similar to the other but accomplishes some more. In essence, it's the same as a subclass.

## DIAGRAMS OF SEQUENCES

A sequence diagram is a graphically depiction of a scene in which item interactions are depicted in a moment sequence, like what happens first and next and what happens next. Sequence diagrams illustrate the duties of objects and aid in the determination of class roles and interfaces. The differences between sequence and collaboration drawings are as follows: sequence diagrams depict moment item connections, whereas collaborative diagrams depict way objects interact each other. A sequence diagram has two major aspects: time is shown vertical, and distinct things are depicted horizontally.

## Object:

The three qualities of an object are its state, behaviour, and identity. The architecture and behaviour of comparable items are defined by the shared class. A class instance is represented by each element in a diagram. A classes instances is an object without a name.

The objects symbol is similar to a class icon, except that the name is underscored: An object's concurrency is determined by the concurrency of its own class.

## Message:

A message is a sort of communication that occurs between two items and cause an event to happen. A message is a data packet that passes from of the voice of authority to target of command. The messaging

specifications could be used to adjust a message's synchronization. A communication in which the transmitting item awaits for answers is referred to as synchronisation.

## Link:

Two objects, including class functions, should be linked only when there's a connection among their respective classes. The construction of a connection among these two classes signifies the existence of a communication link between the instances of the classes: one item could transmit information to another. A cooperation diagram depicts the link as a single direction between items or between objects and class instances.

## Class Diagram:

**CLASS DIAGRAM ANALYSIS**: A class is a collection of items with similar structure also behaviour. A class is a graphical representation of anything that exists in reality.

## Class identification can be accomplished in four ways:

* + 1. Noun phrase-based approach:
    2. Methodology depends on similar class patterns.
    3. Implement a case-driven sequencing strategy or a collaboration strategy.
    4. Responsibilities, classes, and Associates The Approaches to Noun Phrases:

## The guidelines for identifying the class are as follows:

* Look for a nouns and noun phrases in the usecases.
* Some classifications are believed or obtained from shared knowledge;
* All categories should make logical sense of the software system; computer implementation classes should be postponed till the design phase.
* Choose and define class labels with caution. After recognising the classes, we should eliminate the following categories of programmes:
* Adverbial Adjectives are subsets of classes. Approach to similar class patterns:

## The following are the methods for locating applicant classes:

* This is a class about concepts.
* An event class.
* Organizational skills training
* People's social class
* This class organises the class.
* This class is for tangible things and systems.

## Approach based on use-cases:

The cooperation diagram or design pattern must be created. If certain classes are needed to reflect certain features, a new class should be created to fulfil those tasks.

## The CRC method is as follows:

**The steps in the procedure are as follows**:

* Determine the duties of each class.
* Key responsibilities to every person.
* Make a collaborators list .

## Responsibilities of each class are defined as follows:

**In determining a classes properties and behaviours, the essential questions will be addressed:**

1. Which information about an object do we need to keep track of?
2. What services must a class provide?

## The following are the relationships between classes:

Object relations are classified into three types:

What is the connection in-between items?

In a super-sub structure, how do items organised in super classes and child classes? What is the make-up of the complex classes?

What was structure of complicated class's in terms of aggregation?

## Association:

A few of the questions that will help us determine the relationships are as follows:

a. Is really class capable of completing the job on its own?

c. If not, what is it in more need?

c. In which other categories may it receive what it needs?

## Some guidelines for determining initial associations are as follows:

* An connection is a relationship between two classes or more classes. Association is typically associated with a verb or a postfix.
* An connection is a connection from one category to another. Some links are implied or based on general knowledge.

Location associations such as part of, next to, and contained in are examples of prevalent association patterns.

Talk to, order to are examples of communication associations.

We must eliminate any unnecessary ties, including such implementing connections, tertiary or n-ary links, and derived connections.

## The following is a generalisation of the principles for determining the super-sub relationship.

1. **Top-down**: From the top down, look for noun phrases made up of many adjectives in a class name. Excess polishing must be prevented. Only specialized when the subclasses exhibit notable behaviour.
2. **Bottom-up**: From the bottom up, look for classes that have comparable properties or methods. To group the similar features and functions, assign them to an abstract data type. It is probable that the definitions will have to be adjusted***.***
3. **Reusability**: Move the traits and techniques as higher up in the organization as possible.
4. **Several inheritances**: Do not even rely too heavily on numerous inherited. One method for reaping the benefits of multiple inheritances is to inherit from the most appropriate class and add an item from another class as a feature.

Aggregate, also known as just a relationship, refers to a situation in which a class is composed of several component classes. A class composed of multiple other classes behaves differently than its components. It has a really difficult attitude. The primary properties of this connection are transitivity and antisymmetry.

## The following questions will determine the distinction between part and whole relationships:

* Is the part class included in the problem domain?
* Does the part class fall under the system's obligations?
* Can the part class capture higher than 1 value?
* Is this an appropriate abstraction for dealing with particular issue?

## There are three types of aggregation relationships. These are their names:

**Assembly:**

It is composed of parts, and there's a physical components scenario.

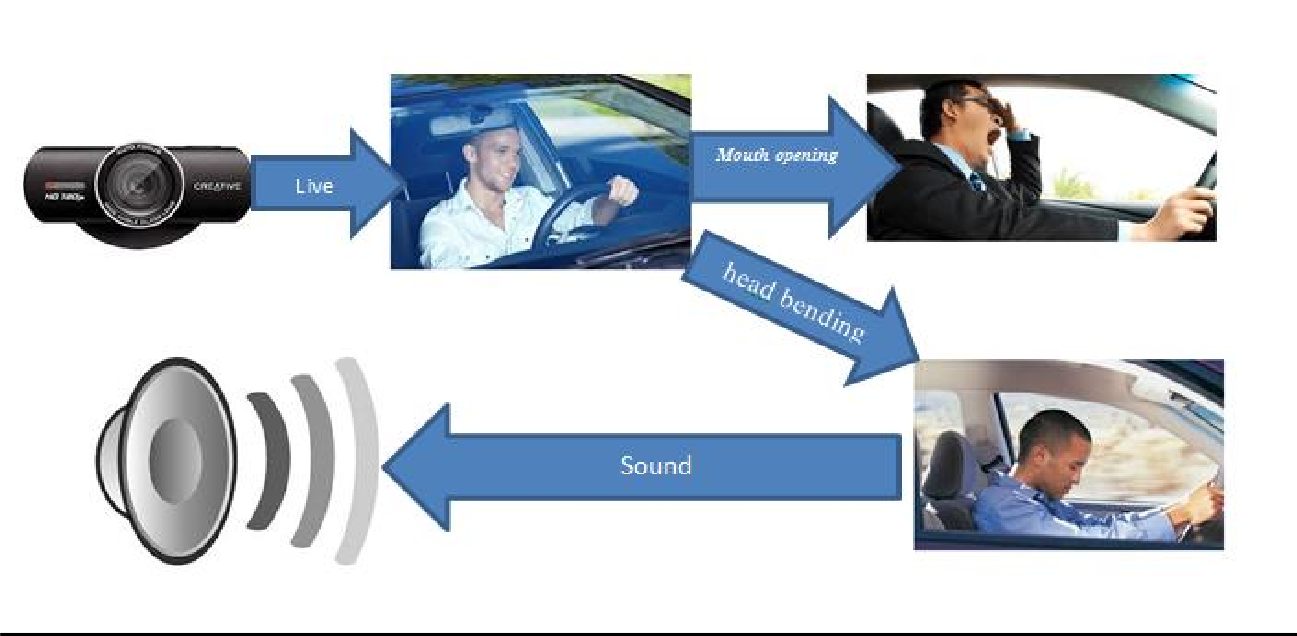
## Container:

A physically whole contains, but is not composed of, physical pieces.

## A collection member is:

A conceptual totality is composed of both physical and conceptual components. The containers and collecting are represented by hollow diamonds, whilst the mixture is represented by solid diamonds.

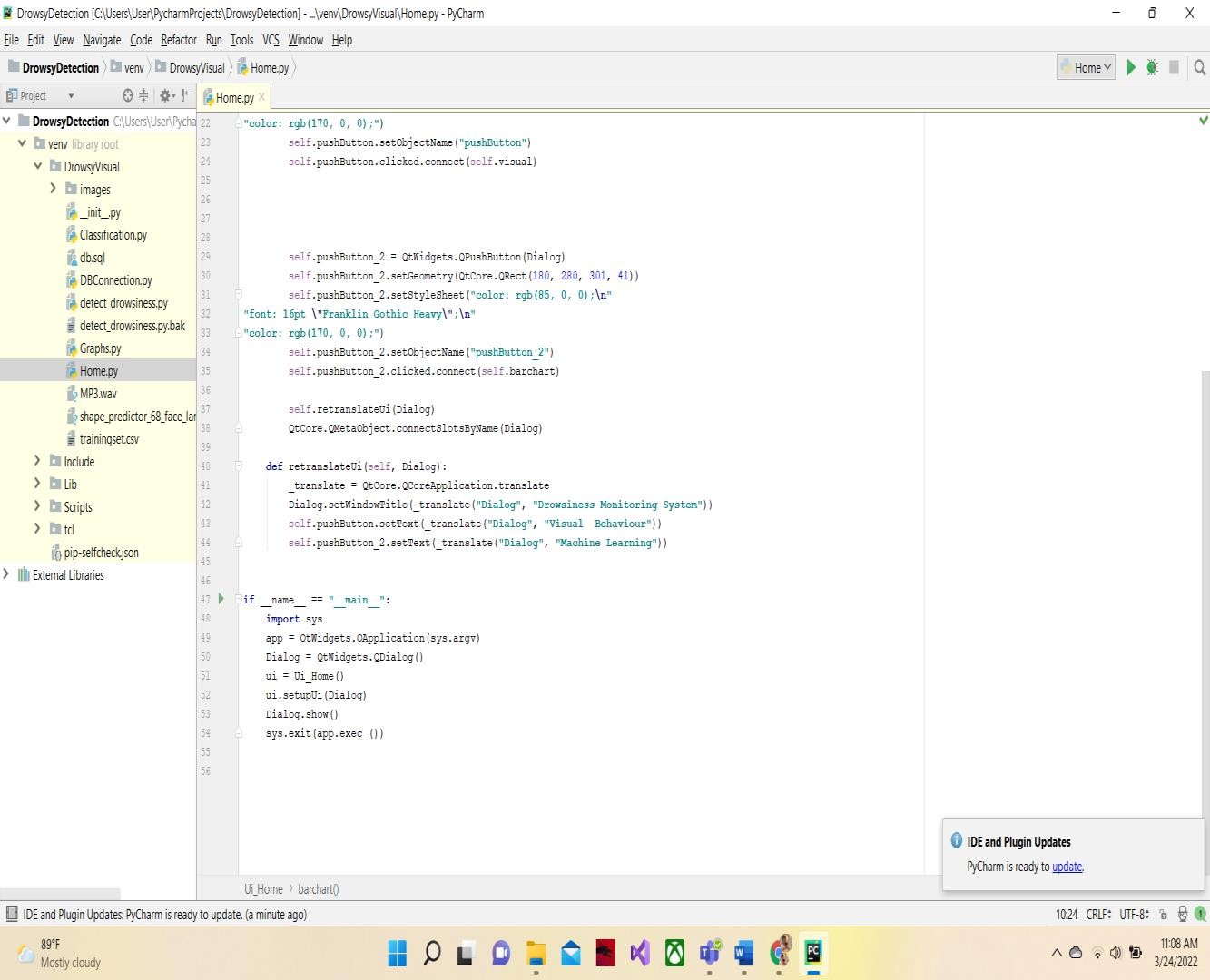
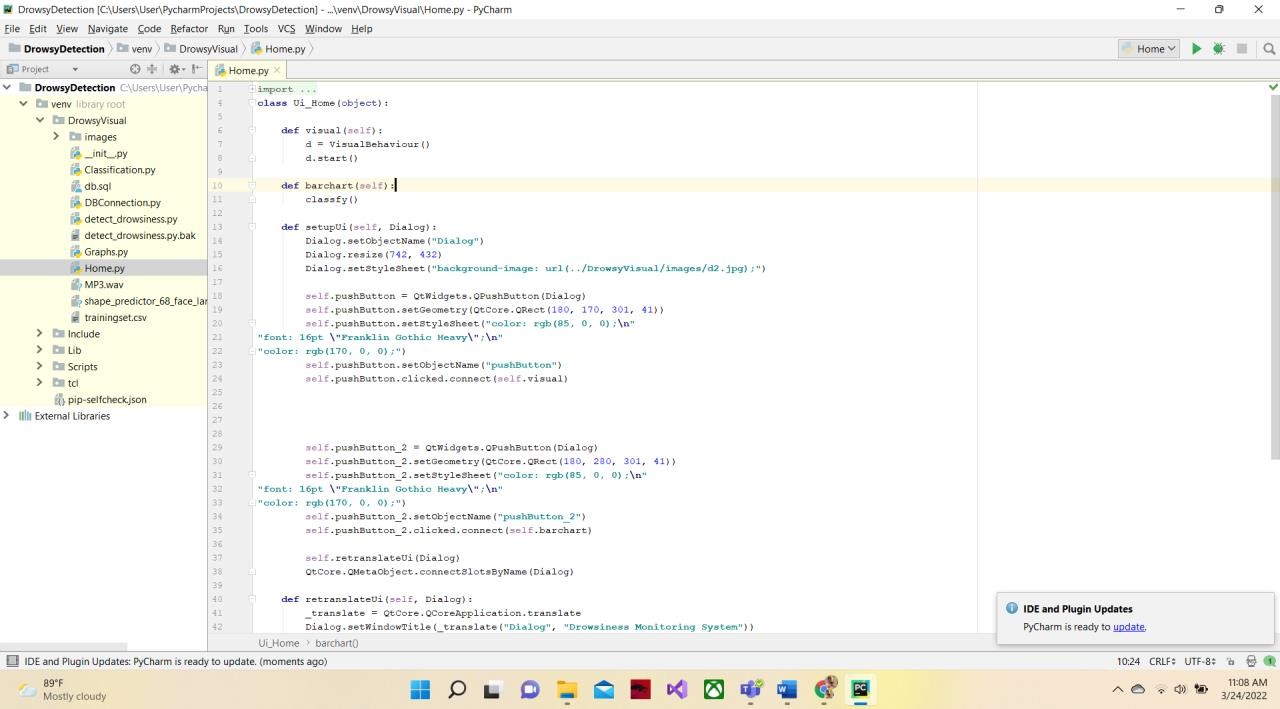
## Implementation:



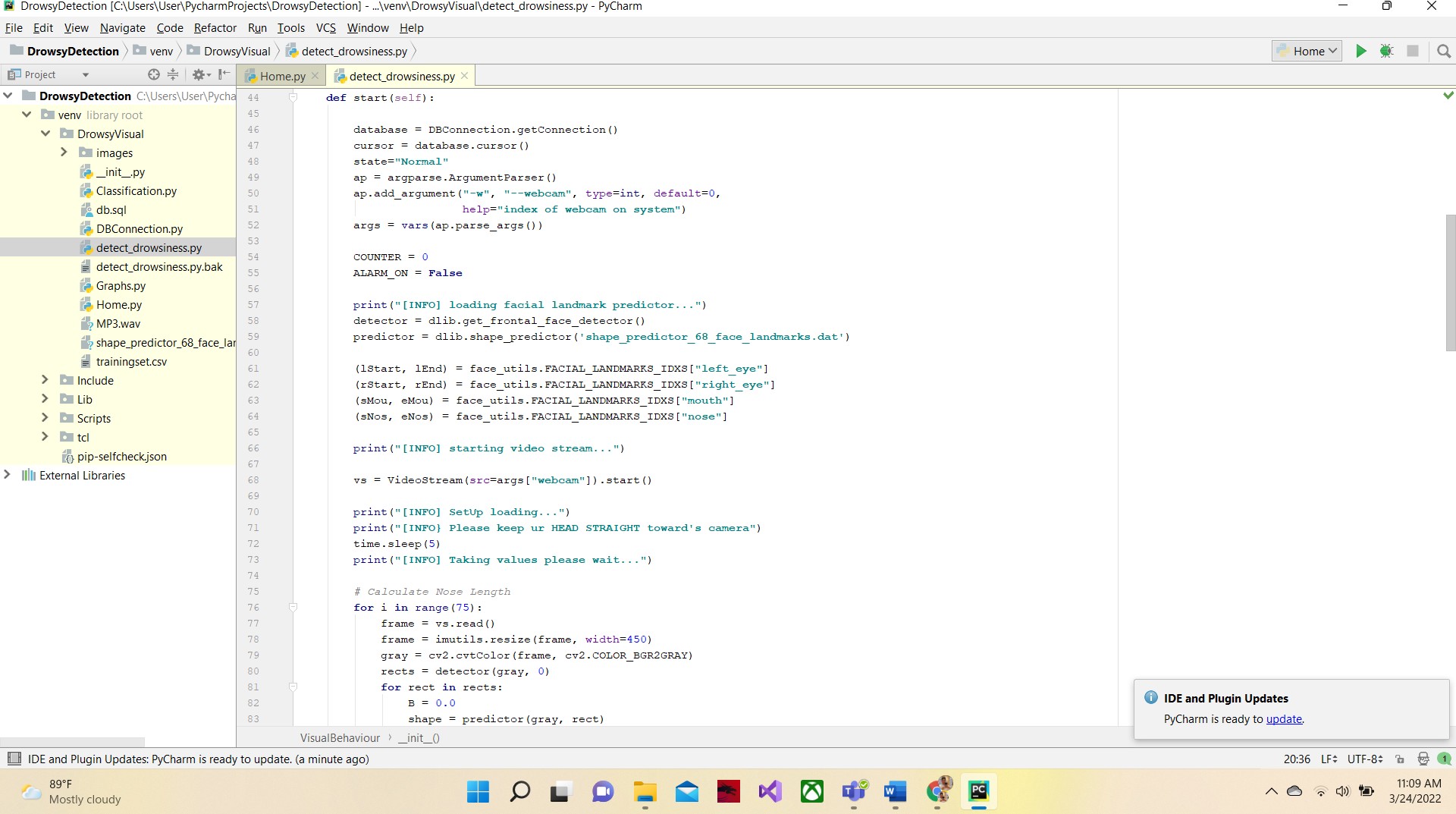
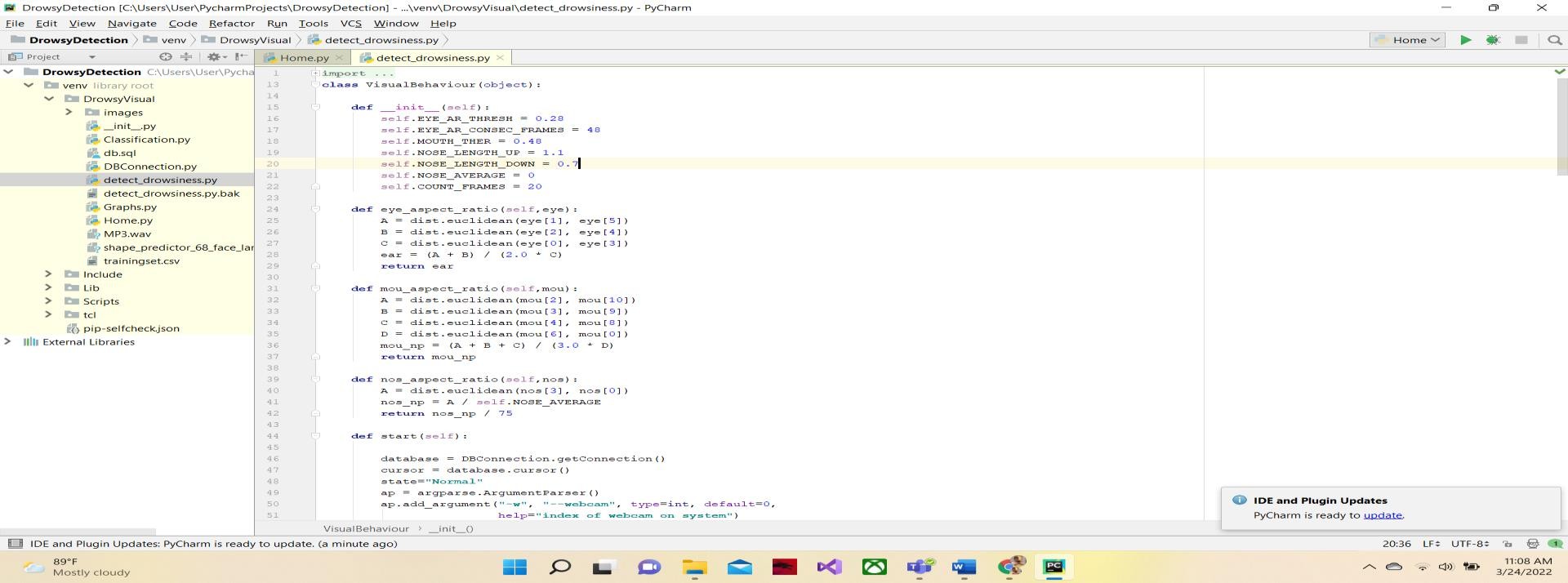
**Fig 17 : Process of system working**

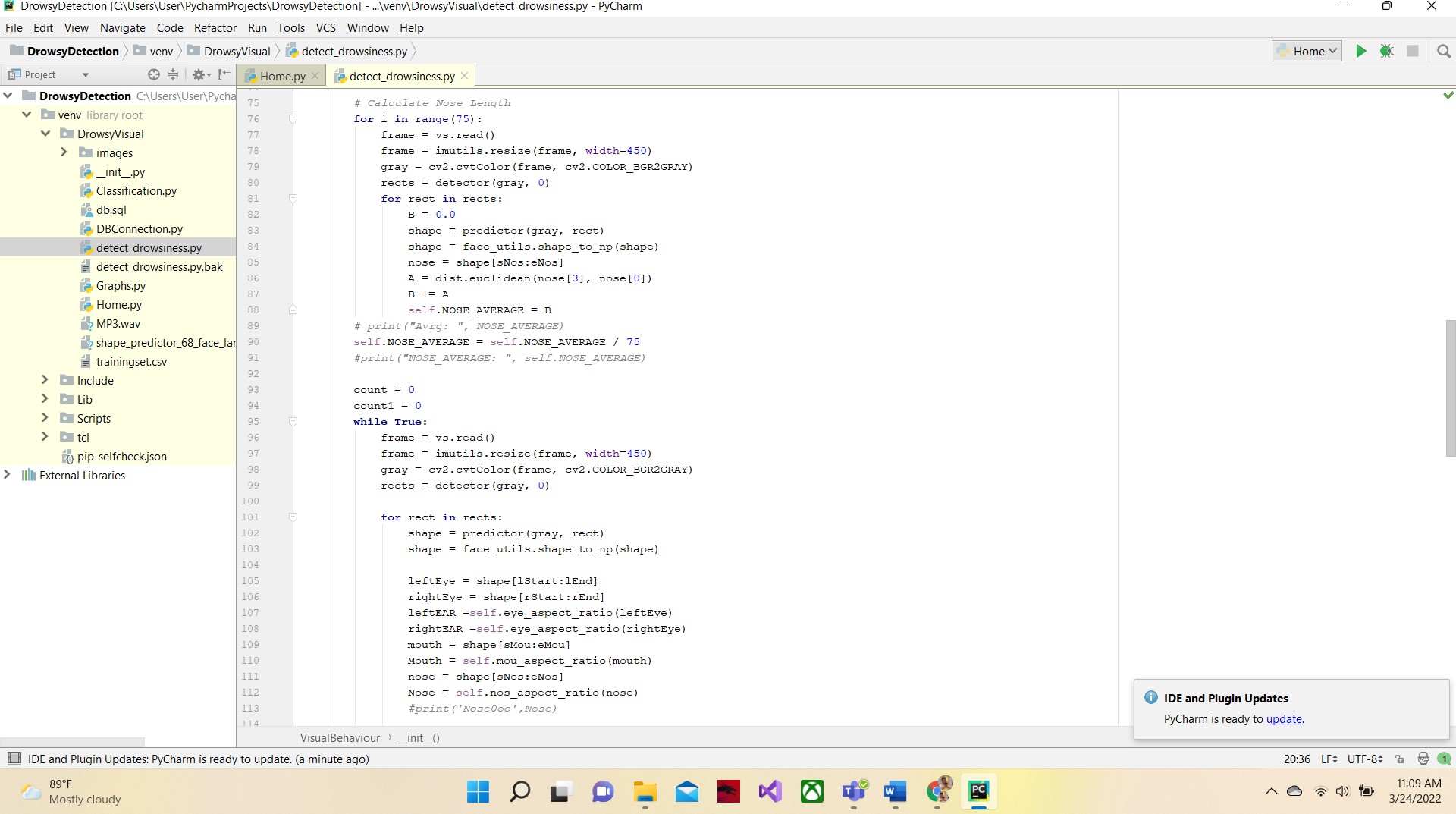
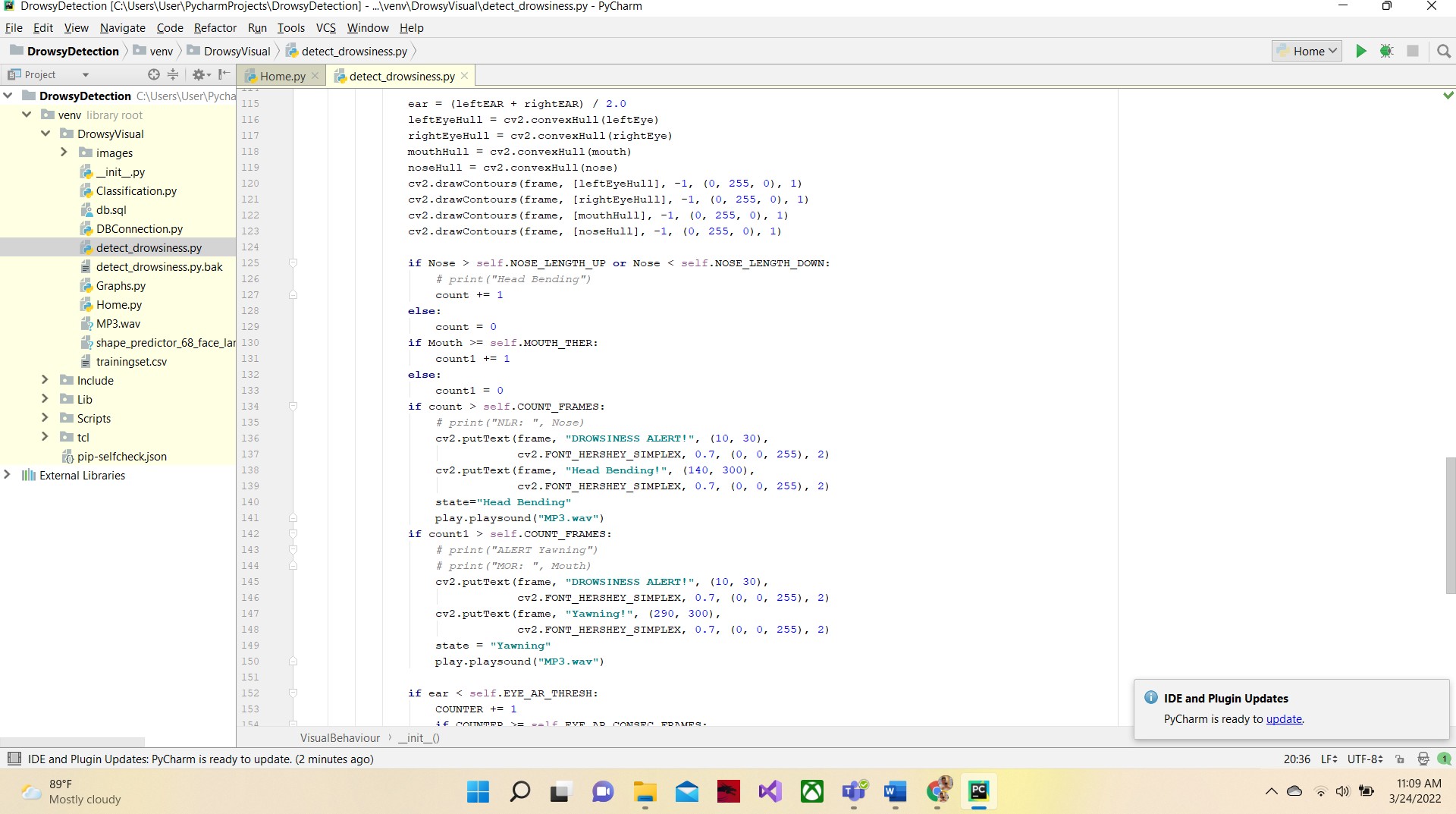
* + - In our program we used Dlib, using the pre-defined landmarks.we detect human faces.After passing our video feed to the dlib frame by frame, we are able to do detection of left eye and right eye features of the face.
    - . Using Euclidean function, we will perform calculations

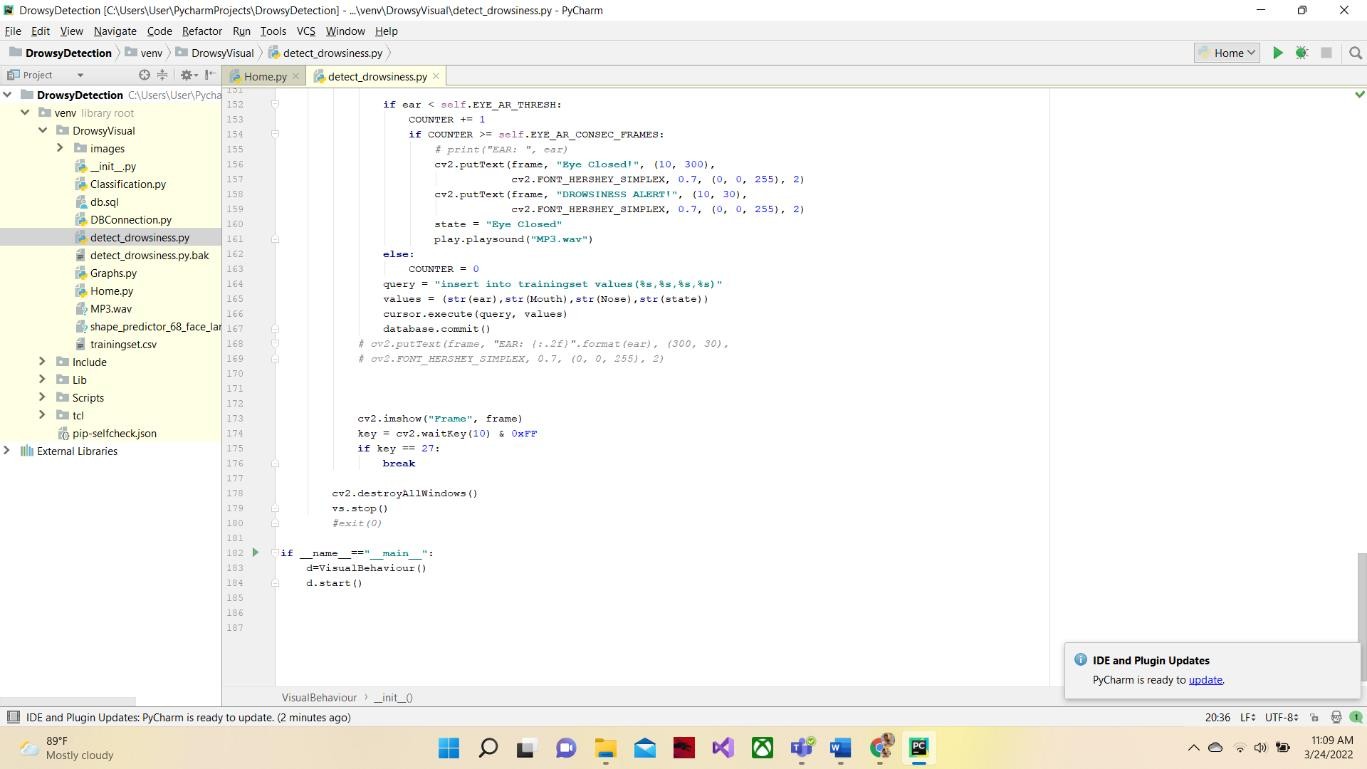
## Sample code Home Page:



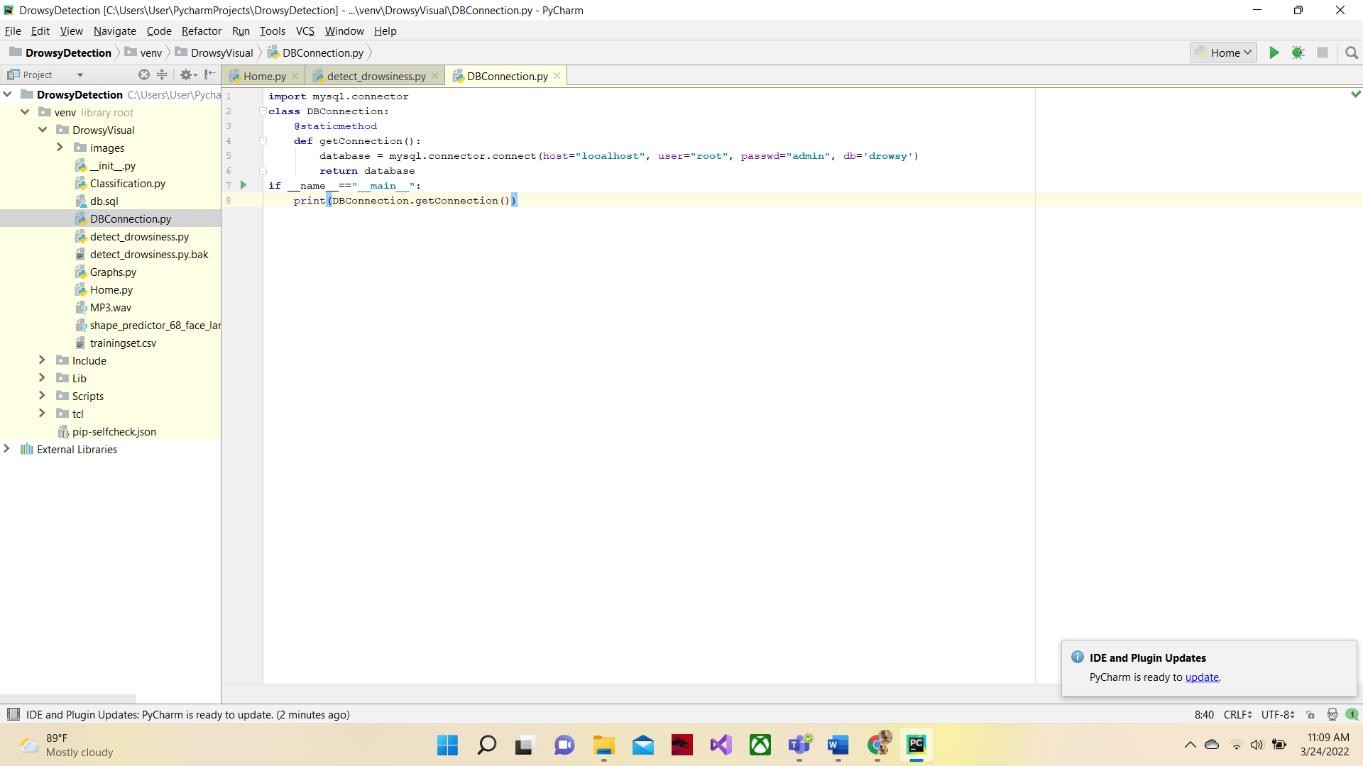
**Detect\_drowsiness :**



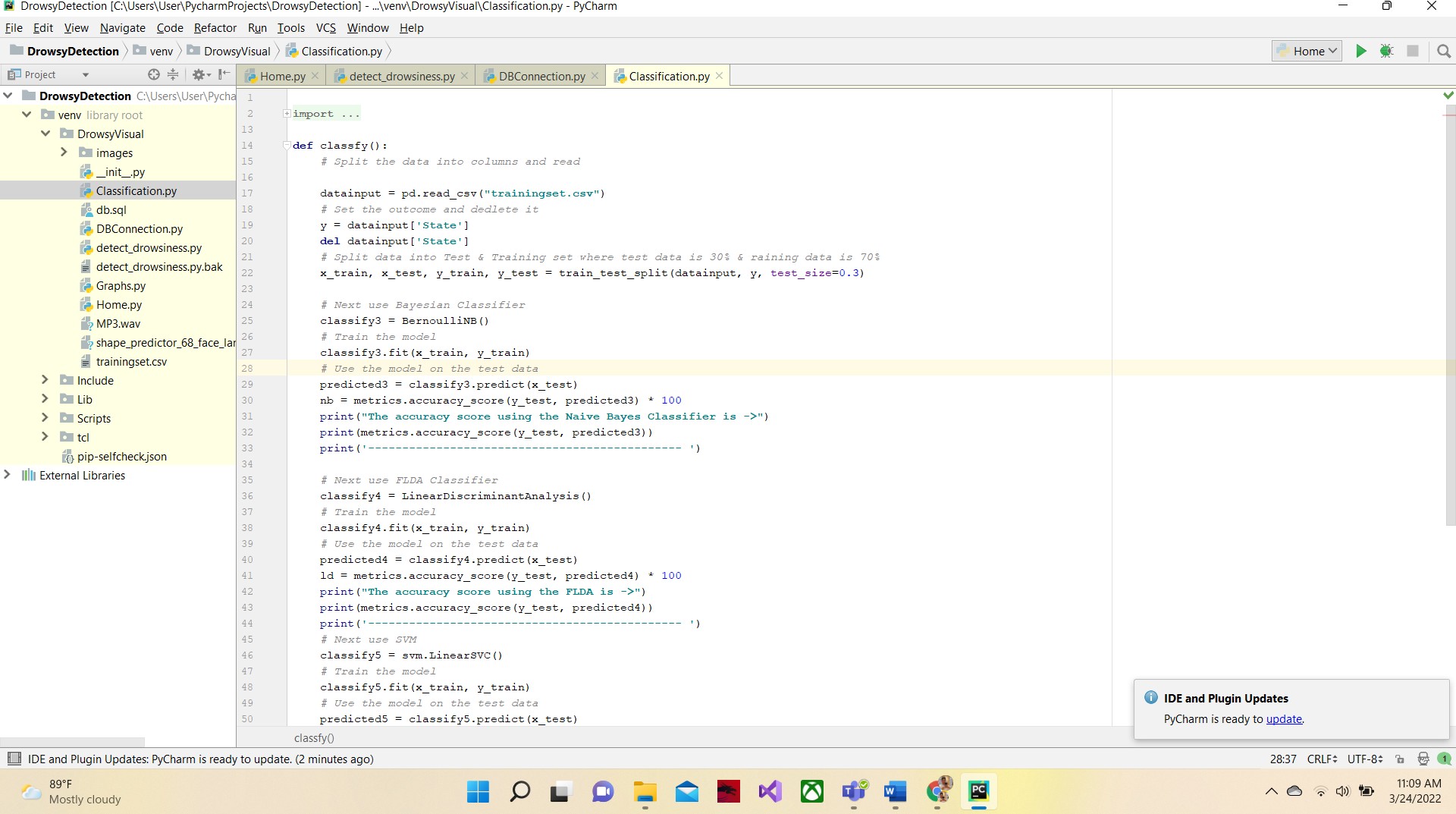


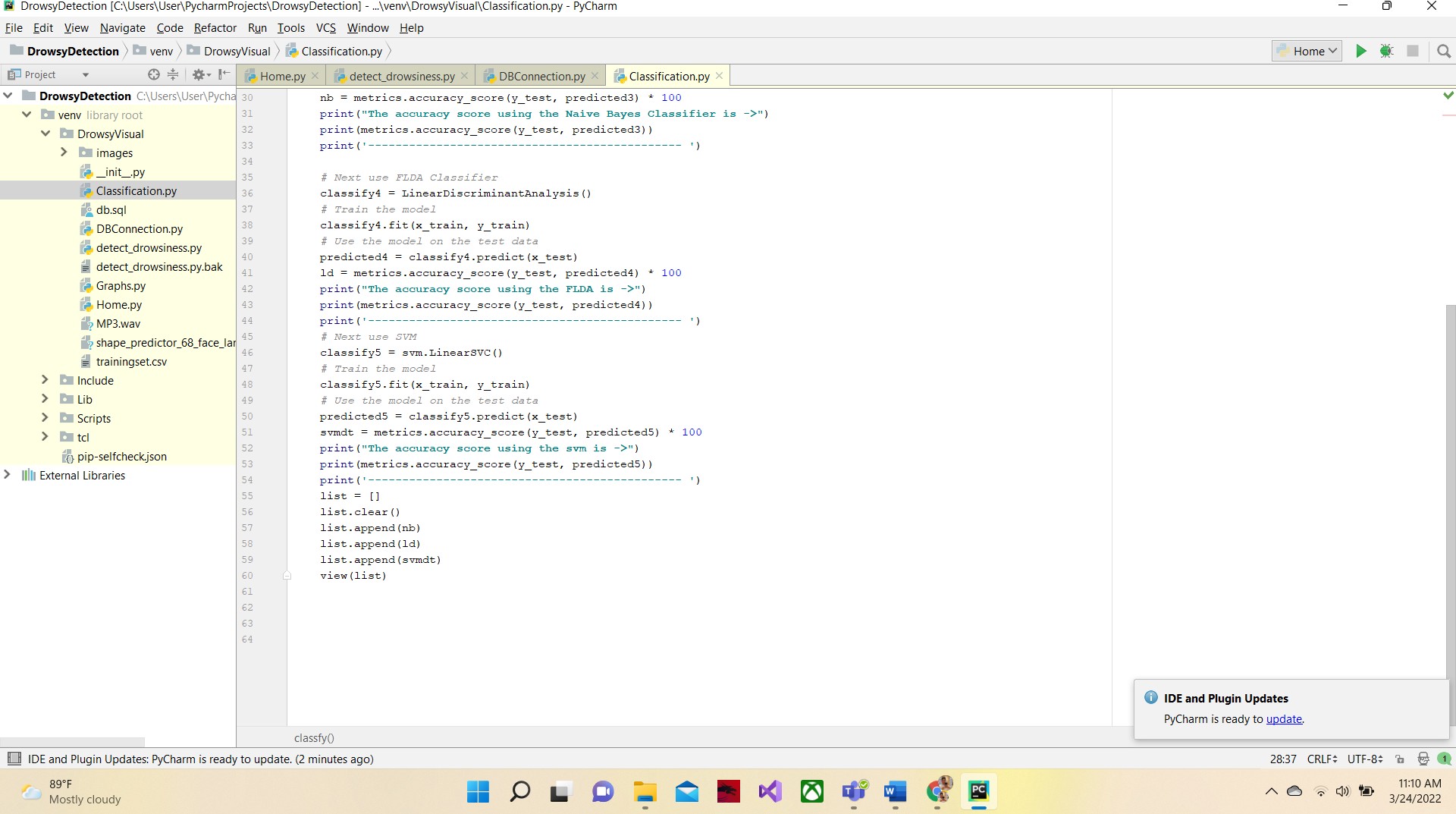


**DBConnection :**



**Classification :**





* 1. **SOFTWARE OVER VIEW:**

**History of Python**

Guido van Rossum created Python in the Netherlands' National Research Center for Computer science and mathematics in the late 1980s and early 1990s. Many other languages, such as ABC, Modula3, C, C++, Algol68, SmallTalk, and Shell Unix, as well as other scripting languages, have influenced Python.

Python is protected by copyright. Python's source code is currently available under the GNU General Public License, just like Perl's (GPL).

Python is now overseen by the institute's central development team, though Guido van Rossum continues to play a vital role in its direction.

## Input as CSV File

In Data Science, reading data from CSV (is a must. We frequently receive data from a variety of sources which can be exported in CSV format and used by other systems. The Panadas library allows you to read the entire CSV file or just a subset of it for a certain set of rows and columns.

The CSV file is a text file with commas separating the values in the columns. Consider the following information from the input.csv file. By copy pasting this data into Windows Notepad, you can produce this file. Using Notepad's Save As All Files (\*.\*) option, save the file as input.csv.

import pandas as pd

data = pd.read\_csv('path/input.csv') print (data)

## Operations using NumPy

NumPy stands for "Numerical Python" and is a Python package. It's a multidimensional array object library with a set of array processing routines.

## A developer can use NumPy to execute the following tasks:

* Array operations (mathematical and logical).
* Shape modification using Fourier transformations and algorithms.
* Linear Algebra-Related Operations NumPy includes linear algebra and random number generating functions.

**Pandaskeyfeatures:**

* + Fast and efficient DataFrame object with predefined and custom indexing.
  + Tools for loading data into memory data objects from various file formats.
  + Data alignment and integrated management of missing data.
  + Remodeling and rotation of date series.
  + Label-based slicing, indexing, and subsets of large datasets.
  + You can remove or insert columns from a data structure.
  + Group by data for aggregations and transformations.
  + High performance data merging and merging. • Time series functionality.

## Chapter 5 : Evaluation

* 1. **Integration & Test Stage:**

Software artefacts, online help, and test data are moved from the development platform to a distinct testing environment during the integration and testing process. All test cases are run at this step to ensure that the software is correct and complete. The test suite's successful completion confirms a robust and comprehensive migration capability. The master data for production use is finalised during this phase, and operational users are identified and linked to their proper roles. The production start plan contains the final master data (or links to master data source files) as well as a list of production users.

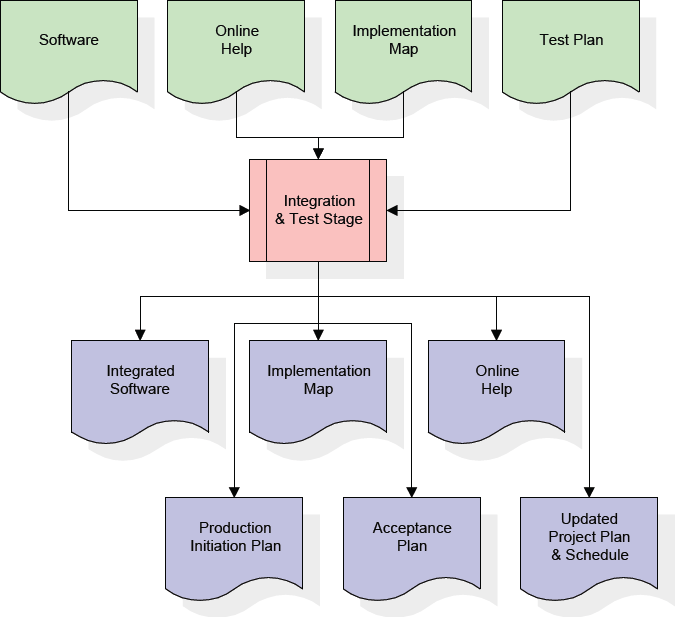


Fig 18: **Integration & Test Stage**

An integrated set of software, an online help system, an implementation map, a production start-up plan defining the baseline data and production users, an acceptance plan containing the final suite of test cases, and an updated project plan are the results of the integration and testing phase.

* 1. **Testing techniques** used

**Unit test :**

Unit testing is a software development process that examines the Applications smallest testable parts

, known as units, independently and individually for proper operation. The software developers during the development process use this testing methodology.

**MANUAL Testing :**

Since we are doing is a academic level project , we are unable to perform any automated testing; therefore, we must rely on manual testing through trial and error. methods.

* 1. Checking **All test cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SNO | Test Case | System Behaviour | Expected Result | Test Pass/fail |
| 1 | Cam On | Non Drowsiness | Non Drowsiness | Pass |
| 2 | Face Detection | Non Drowsiness | Non Drowsiness | Pass |
| 3 | Facial Landmark marking | Non Drowsiness | Non Drowsiness | Pass |

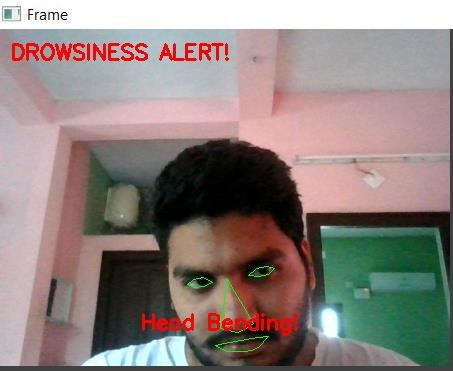
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | Detecting Head Beading and Notifying by Alarm  Sound | Drowsiness Alert | Drowsiness Alert | Pass |
| 5 | Detecting Eyes Closing  and Notifying by  Alarm Sound | Drowsiness Alert | Drowsiness Alert | Pass |
| 6 | Detecting Yawning and Notifying by Alarm Sound | Drowsiness Alert | Drowsiness Alert | Pass |

* 1. **Test Results and Outputs :**

**Facial landmark testing output screenshot :**



**Head Bending testing Screenshot**:



**Eyes Close testing screenshot :**



**Yawning Testing Screenshot**:



* 1. **Installation & Acceptance Stage**

The software artefacts, online support, and initial development data are put onto the production server during in the installation and acceptance step. At this step, all cases of test are executed to ensure that the software is correct and complete. The successful execution of the test suite is required for the customer to accept the programme. The customer formally accepts the delivery of the software once customer personnel have confirmed that the initial production data load is accurate and the testing procedure has been completed with good results.

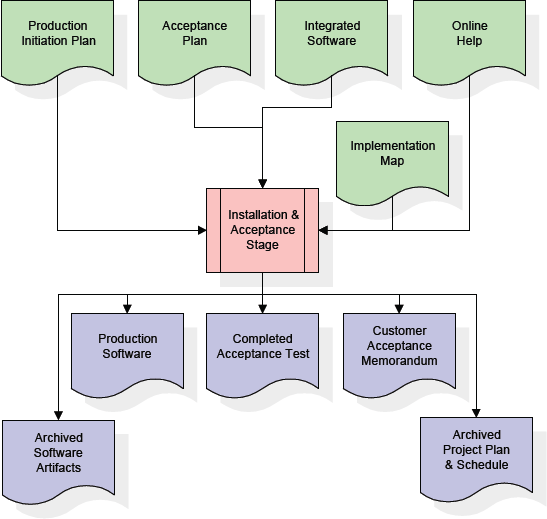


Fig 19: **Installation & Acceptance Stage**

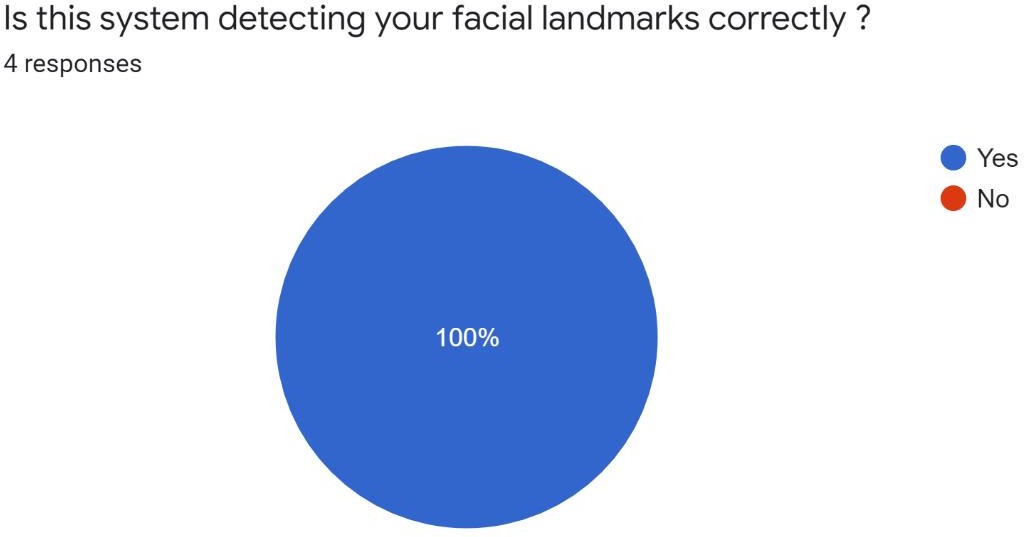
The installation and acceptance stage's key deliverables are an operational application, a complete acceptance test suite, and a statement confirming client acceptance of the software. Finally, the PDR adds the final real labour data into the project timeline and secures the project as a fixed project record. The PDR "locks" the project at this point by preserving all software components, the source code, implementation map, and the documentation for further use.

## User Acceptence Test & Feedback:

Acceptence test has been Conducted with 4 of my friends and feedback form has been given to them to get there feed back

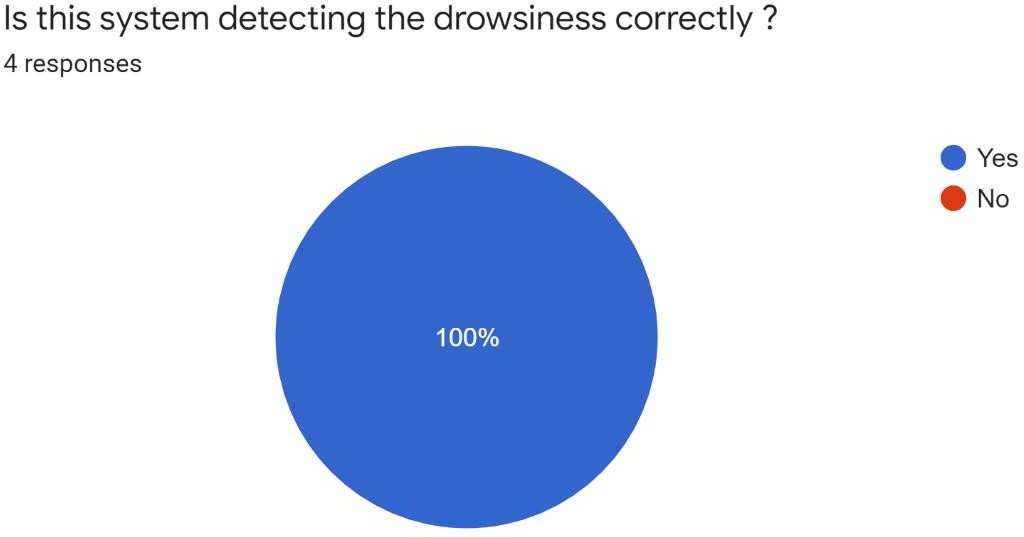
Responses as follow:

## Question 1:



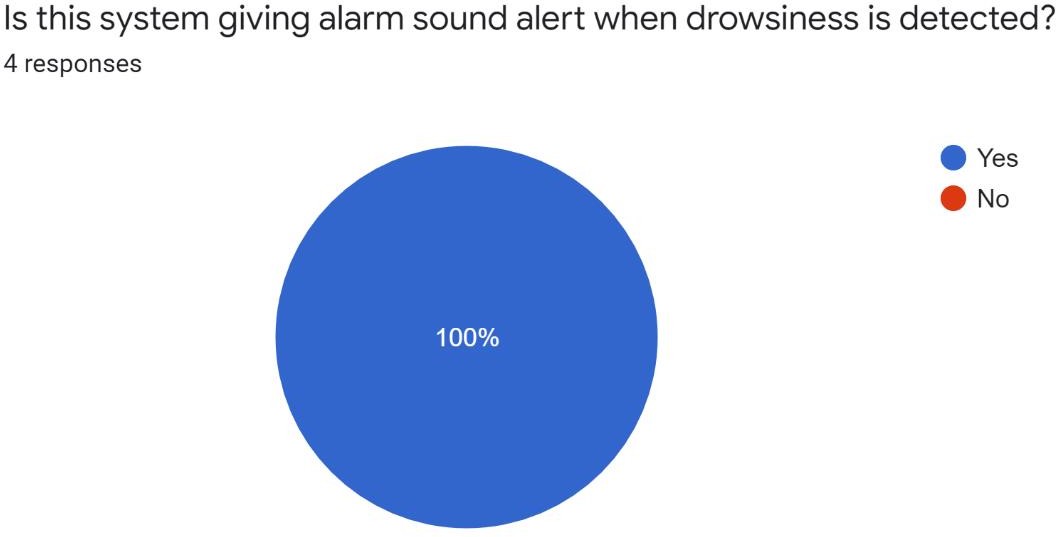
All respondets said that this system is detecting the facial landmarks correctly. All of them responded yes.

## Question 2:



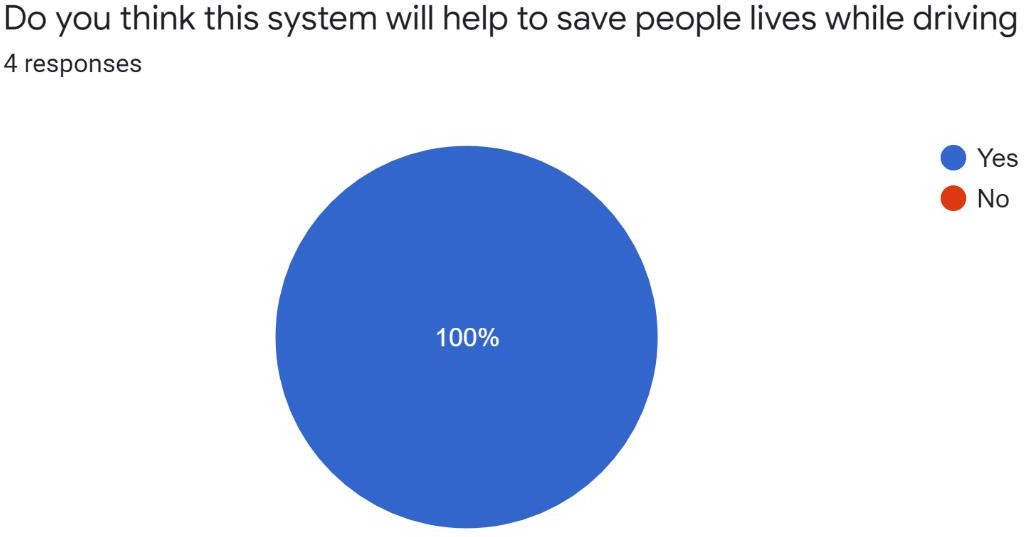
All of the respondents said that this system detecting the drowsiness correctly .All of them said yes.

## Question 3:



All of the respondents said that this system is giving alert when drowsiness is detected .All of them said yes.

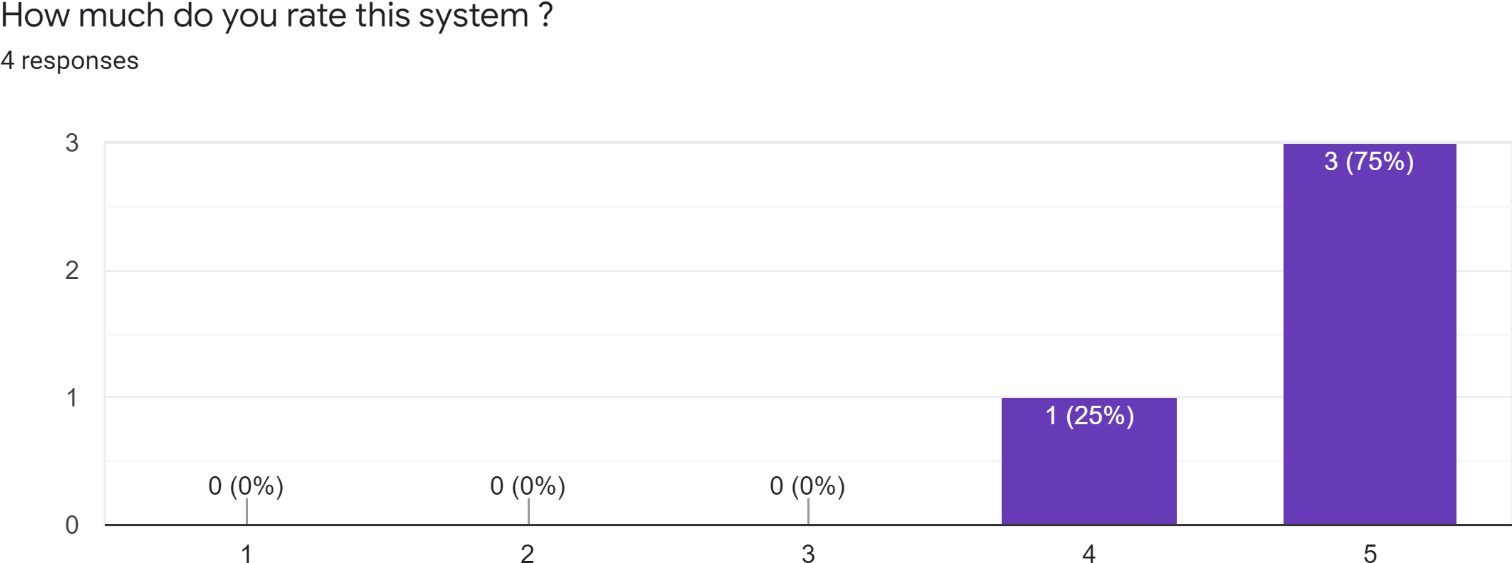
## Question 4 :



All of the respondents said that this system helps people lives from accidents or other unsafe conditions

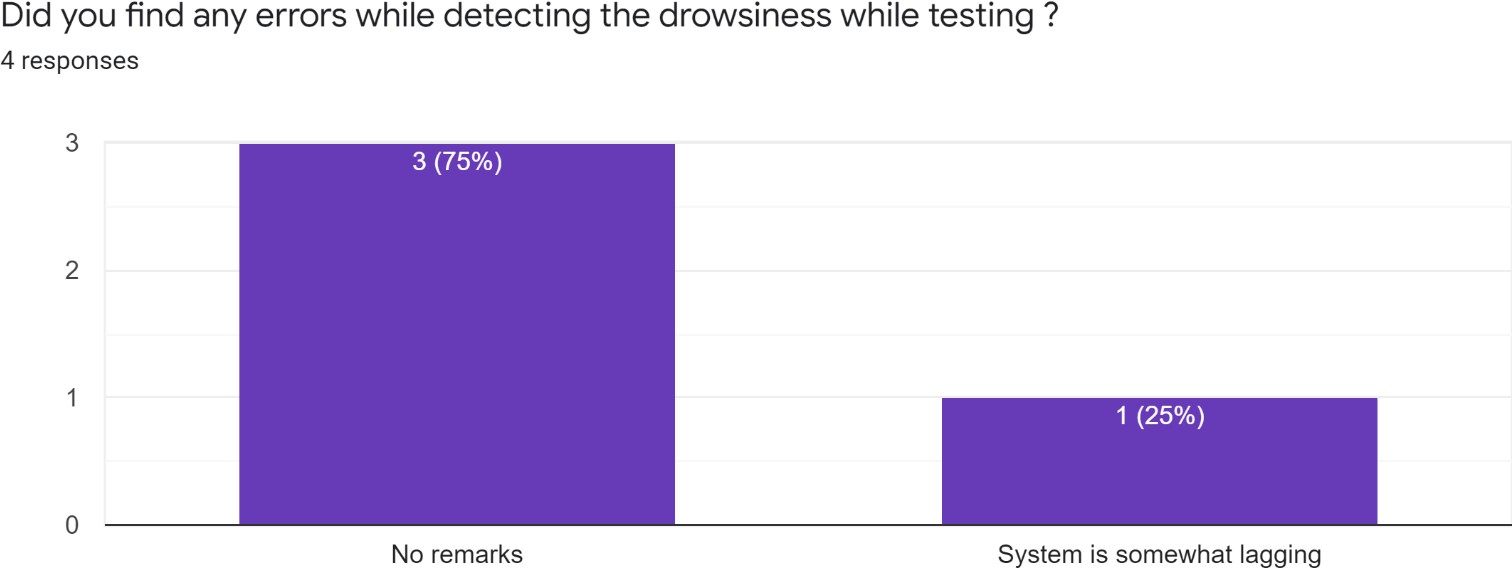
.All of them said yes.

## Question 5:



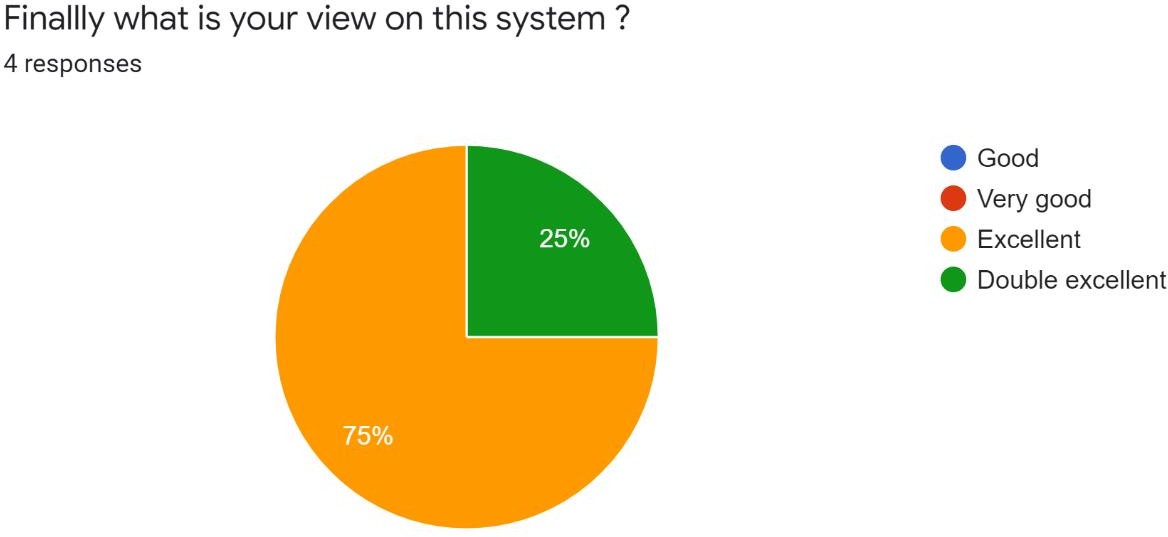
Most of the respondents rated 4 and 5 to this system because they tested this system and they belieave this system will work correctly detecting the drowsiness .

## Question 6



Most of the respondents said No remarks for the system because they tested the system. only 1 member said this system is lagging. But that was not the developed system issue. We can consider It maybe the laptop network issue.

## Question 7:



All of the respondents said that this system is “Excellent” and “Double excellent”.

* + - So With this user acceptance test we can conclude that this system is working perfectly and it will safe lives from accidents or in unsafe conditions. So users accepted our system to release.

Chapter 6 : Conclusion

This study proposes a less-cost, real-time Vehicle driver sleepiness monitoring system based on visual behaviour and machine learning techniques. Visual behaviour features such as eyeaspect ratio, mouth openness ratio, and length of nose ratio are estimated from Video streaming taken by a camera in this case. To identify driver tiredness in real time, an adaptive thresholding technique has been used. The designed system performs flawlessly with the provided synthetic data. Following that, the feature values are saved, and machine learning methods are employed for classifying.

**References :**

1. Horng W. B., C. Y. Chen, Y. Chang, C. H. Fan, “Driver Fatigue Detection Based on Eye Tracking and Dynamic Template Matching”, IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, March 21-23, 2004.
2. Abas A., J. Mellor, and X. Chen, “Non-intrusive drowsiness detection by employing Support Vector Machine,” 2014 20th International Conference on Automation and Computing (ICAC), Be.dfordshire, UK, 2014, pp. 188- 193.
3. Shih, M. H., W. L. Ou, C. W. Chang, X. H. Yu, C. P. Fan, "Intelligent Video-Based Drowsy Driver Detection System under Various Illuminations and Embedded Software Implementation", 2015international Conf. on Consumer Electronics - Taiwan, 2015.
4. Mandalapu Sarada Devi and Dr. Preeti R. Bajaj, “Driver Fatigue Detection B.ased on Eye Tracking”, ICETET, July-2008.
5. Raees Ahmad Mr. and Prof. J.N. Borole, “Drowsy Driver Identification Using E.ye Blink detection” International Journal of Computer Science and Information Technologies, Vol. 6 (1), 2015, 270-274.
6. Matthew Sacco and Reuben A. Farrugia, “Driver Fatigue Monitoring System Using Support Vector Machines”, 5th International Symposium on Communications, Control and Signal Processing, May 2012.
7. Karchani M., A. Mazloumi, G. N. Saraji, A. Nahvi, K. S. Haghighi, B. M. Abadi, A. R. Foroshani, A. Niknezhad, “The Steps of Proposed Drowsiness Detection System Design based on Image Processing in Simulator Driving”, International Research Journal of Applied and Basic Sciences, vol. 9(6), pp 878-887, 2015
8. W. L. Ou, M. H. Shih, C. W. Chang, X. H. Yu, C. P. Fan, "Intelligent Video-Based

Drowsy D.river Detection System under Various Illuminations and Embedded Software Implementation", 2015 international Conf. on Consumer Electronics - Taiwan, 2015.

16

1. W. B. Horng, C. Y. Chen, Y. Chang, C. H. Fan, “Driver Fatigue Detection based on Ey.e Tracking and Dynamic Template Matching”, IEEE International Conference on Networking,, Sensing and Control, Taipei, Taiwan, March 21-23, 2004.
2. S. Singh, N. P. papanikolopoulos, “Monitoring Driver Fatigue usi.ng Facial Analysis T.echniques”, IEEE Conference on Intelligent Transportation System, pp 314-318.
3. B. Alshaqaqi, A. S. Baquhaizel, M. E. A. Ouis, M. Bouumehed, A. Ouamri, M. Keche, “Driver Drowsiness Detection System”, IEEE International Workshop on Systems, Signal Processing and their Applications, 2013.
4. R. Ahmad, and J. N. Borole, “Drowsy Driver Identification Using Eye Blink D.etection,” IJISET - International Journal of Computer Science and Information Technologies, vol. 6, no. 1, pp. 270-274, Jan. 2015. [14] A. Abas, J. Mellor, and X. Chen, “Non-intrusive drowsiness detection by employing S.upport Vector Machine,” 2014 20th International Conference on Automation and Computing (ICAC), Bedfordshire, UK, 2014, pp. 188- 193.
5. A. Sengupta, A. Dasgupta, A. Chaudhuri, A. George, A. Routray, R. Guha; "A M.ultimodal System for Assessing Alertness Levels Due to Cognitive Loading", IEEE Trans. on Neural Systems and Rehabilitation Engg., vol. 25 (7), pp 1037-1046, 2017.
6. K. T. Chui, K. F. Tsang, H. R. Chi, B. W. K. Ling, and C. K. Wu, “An accurate ECG b.ased transportation safety drowsiness detection scheme,” 343 IEEE Transactions on Industrial Informatics, vol. 12, no. 4, pp. 1438- 1452, Aug. 2016.
7. https://pyimagesearch.com/wp-

content/uploads/2017/05/drowsiness\_detection\_alert.jpg

1. https://graspcoding.com/wp-content/uploads/2020/09/Driver-DrowsinessDetection-System-

AI-Project-1-800x445.png

[https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcRuX2BSSzoJuQTNrQcxJW9W](https://encryptedtbn0.gstatic.com/images?q=tbn%3AANd9GcRuX2BSSzoJuQTNrQcxJW9WFpLDJebUSwd2VA&amp;usqp=CAU) [FpLDJebUSwd2VA&usqp=CAU](https://encryptedtbn0.gstatic.com/images?q=tbn%3AANd9GcRuX2BSSzoJuQTNrQcxJW9WFpLDJebUSwd2VA&amp;usqp=CAU)

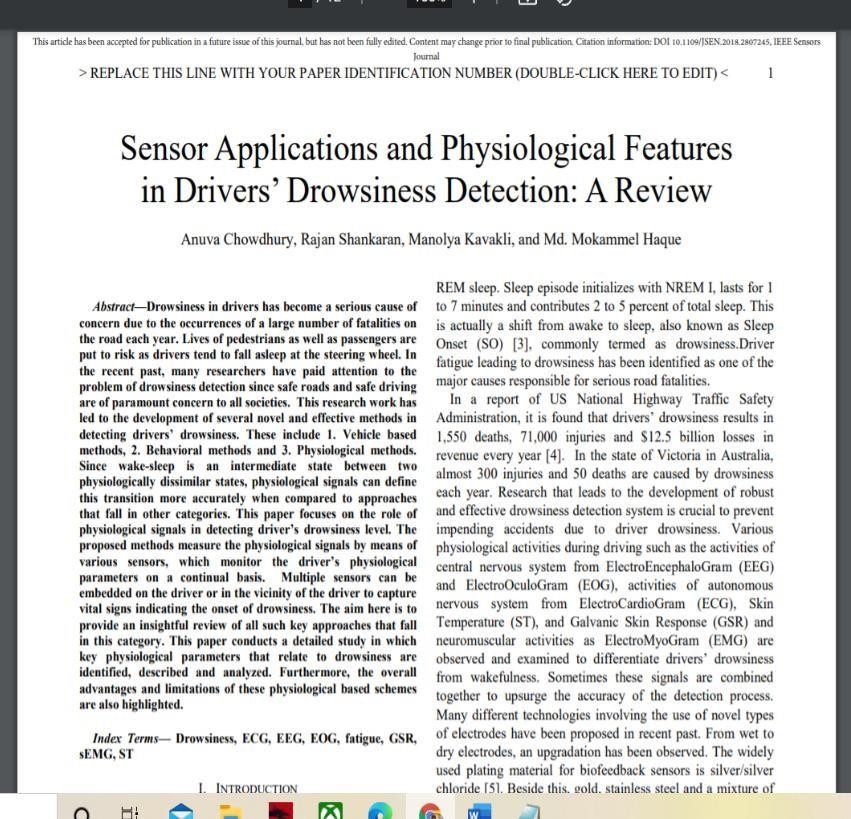
**Appendix**



**Screenshot of journal :**



**Screenshot of journal :**



17

**The End**