



APPENDIX 1
EMOTION DETECTION WEBSITE

A MINI PROJECT REPORT

Submitted by

V. LOKESH	510618104032
V. HRITHICK	510618104022
A. HUSAIN AHMAD	510618104023
M. MOHAMMED UBADA	510618104047

in partial fulfilment for the award of the degree
of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

**C.ABDUL HAKEEM COLLEGE OF ENGINEERING AND
TECHNOLOGY, MELVISHARAM, VELLORE-632509.**

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023



APPENDIX 1
EMOTION DETECTION WEBSITE

A PROJECT REPORT

Submitted by

V. LOKESH	510618104032
V. HRITHICK	510618104022
A. HUSAIN AHMAD	510618104023
M. MOHAMMED UBADA	510618104047

in partial fulfilment for the award of the degree
of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

**C.ABDUL HAKEEM COLLEGE OF ENGINEERING AND
TECHNOLOGY, MELVISHARAM, VELLORE-632509.**

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

ACKNOWLEDGEMENTS

At the outset, we would like to express our gratitude to our beloved and respected Almighty, for his support and blessings to accomplish the project.

We would like to express our thanks to our Honourable Chairman **Dr.S.ZIAUDDIN AHMED SAHEB B.A.**, and to our beloved Correspondent **Dr. V. M. ABDUL LATHEEF SAHEB B.E., FIE.**, for their encouragement and guidance.

We thank our Principal **Dr.M. SADIQUE BASHA Ph.D.**, for creating the wonderful environment for us and enabling us to complete the project. We wish to express our sincere thanks and gratitude to **Dr.K.ABRAR AHMED M.Tech.,Ph.D.**, Head of the Department, Department of Computer Science and Engineering who has been a guiding force and constant source of inspiration to us.

We express our sincere gratitude and thanks to our beloved Project Guide **Dr.K. ABRAR AHMED M.Tech., Ph.D.**, Assistant Professor, Department of Computer Science and Engineering for having extended his fullest co-operation and guidance without which this project would not have been a success.

Our thanks to all faculty and non-teaching staff members of our department their constant support to complete this project.

ABSTRACT

Human emotion and facial expression are one of the most powerful tools of human interaction. Detecting and recognizing human emotion is a big challenge in computer vision and artificial intelligence. Emotions are a big part of human communication since most of the communication takes place through emotion. Aim of our project is to develop a robust system which can detect human emotion. This can be used by people affected by autism to make them understand these emotions better.

It is done using html, CSS and JavaScript with face-api. The scope of the project is:

- The user login into the website.
- The users select the button to perform 3 actions.
 - The user can upload the image and see the emotion of that particular image.
 - The webcam opens and shows the emotion of the user.
 - It says the accuracy of the user in the webcam in terms of fps (frames per second).
- Users can go back to Home and also go to End as they wish.
- If the user goes to the end page, a feedback form is asked to fill.

APPENDIX 2

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report on “**EMOTION DETECTION WEBSITE**” is the bonafide work of “**V. LOKESH (510618104032), V. HRITHICK (510618104022), A. HUSAIN AHMAD (510618104023), M. MOHAMMED UBADA (510618104047)**” who carried out the project under my supervision.

SIGNATURE

**Dr.K. ABRAR AHMED M. Tech.,
PH.D.,**

HEAD OF THE DEPARTMENT

Department of Computer Science and
Engineering

C. Abdul Hakeem College of
Engineering and Technology
Melvisharam-632509.

SIGNATURE

**Dr.K. ABRAR AHMED M.Tech.,
PH.D.,**

SUPERVISOR

ASSISTANT PROFESSOR

Department of Computer Science and
Engineering

C. Abdul Hakeem College of
Engineering and Technology
Melvisharam-632509.

Submitted for the University Viva-Voce Examination held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

APPENDIX

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	
	LIST OF TABLES	
	LIST OF ABBREVIATIONS	
1.	INTRODUCTION	1
	1.1 OBJECTIVE	1
	1.2. MOTIVATION	2
	1.3. BACKGROUND	3
2.	LITERATURE SURVEY	4
3.	PROPOSED SYSTEM	5
4.	REQUIREMENT ANALYSIS	6
5.	SYSTEM REQUIREMENTS	7
6.	SYSTEM ARCHITECTURE AND DESIGN	8
7.	EXPERIMENTATION AND RESULT	12
8.	TEST CASE	19
9.	USEABILITY TESTING	23
10.	HCI PRINCIPLES INCORPORATED	27
11.	SOURCE CODE	29
12.	SUMMARY AND CONCLUTION	45
13.	REFERENCES	46

CHAPTER 1

INTRODUCTION

Emotion recognition is the process of identifying human emotion. Emotion Recognition is an important area of research to enable effective human-computer interaction. Human emotions can be detected using speech signal, facial expressions, body language, and electroencephalography (EEG). People vary widely in their accuracy at recognizing the emotions of others. Use of technology to help people with emotion recognition is a relatively nascent research area. Generally, the technology works best if it uses multiple modalities in context. To date, the most work has been conducted on automating the recognition of facial expressions from video, spoken expressions from audio, written expressions from text, and physiology as measured by wearables.

1.1. Objective:

To program a user-friendly application to detect some emotions which are universal to all human beings like angry, sad, happy, surprise, fear, disgust and neutral, which can be used by users with minimal to no experience about current trends in technologies and with the GUI and HCI principles.

The purpose of this study is to introduce a method based on facial recognition to identify students' understanding of the entire distance learning process. This project demonstrates that emotion recognition based on facial expressions is feasible in distance education, permitting identification of a student's learning status in real time. Therefore, it can help teachers to change teaching strategies in virtual learning environments according to the student's emotions.

1.2. Motivation:

Artificial intelligence using neural networks is used for recognition of faces in photos and videos. Most techniques process visual data and search for general patterns present in human faces. Face recognition can be used for surveillance purposes by law enforcers as well as in crowd management. An even more advanced development in this field is emotion recognition. Implementing emotion detection can raise accuracy in facial recognition which in return can aid to the area of surveillance and behavioural analysis by law enforcement. The website is mainly built for the people with autism.

With the rapid development of technologies, it is required to build an intelligent system that can understand human emotion. Facial emotion recognition is an active area of research with several fields of applications. Some of the significant applications are:

- i) Alert system for driving.
- ii) Social Robot emotion recognition system.
- iii) Medical Practices.
- iv) Feedback system for e-learning.
- v) The interactive TV applications enable the customer to actively give feedback on TV Program.
- vi) Mental state identification.
- vii) Automatic counselling system.
- viii) Face expression synthesis.
- ix) Music as per mood.
- x) In research related to psychology.
- xi) In understanding human behaviour.
- xii) In interview

1.3. Background:

Decades of scientific research have been conducted developing and evaluating methods for automated emotion recognition. There is now extensive literature proposing and evaluating hundreds of different kinds of methods, leveraging techniques from multiple areas, such as signal processing, machine learning, computer vision, and speech processing. The accuracy of emotion recognition is usually improved when it combines the analysis of human expressions from multimodal forms such as texts, physiology, audio, or video. Different emotion types are detected through the integration of information from facial expressions, body movement and gestures, and speech.

CHAPTER 2

LITERATURE SURVEY

An earlier version of this paper appeared as “Face Recognition” Technical Report, Center for Automation Research, University of Maryland, College Park, MD, 2000.

Authors’ addresses: W. Zhao, Vision Technologies Lab, Sarnoff Corporation, Princeton, NJ 08543-5300;

email: wzhao@sarnoff.com;

R. Chellappa and A. Rosenfeld, Center for Automation Research, University of Maryland, College Park, MD 20742-3275; email: rama,ar@cfar.umd.edu;

P. J. Phillips, National Institute of Standards and Technology, Gaithersburg, MD 20899; email: jonathon@nist.gov.

Permission to make digital/hard copy of part or all of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication, and its date appear, and notice is given that copying is by permission of ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

CHAPTER 3

PROPOSED SYSTEM

The goal of this project is to help the people born with a mental disability such as autism and guide them to understand emotions with the project we have created pertaining to a simple interface that can be used by anyone regardless of their expertise to use technology. Suppose we are trying to recognize the emotions of a person. In general, getting to the truth of what emotion is actually present can take some work, can vary depending on the criteria that are selected, and will usually involve maintaining some level of uncertainty. However, that may be the case, this project does

help the users understand what emotion is being portrayed which in return can help them in situations where interactions are require

CHAPTER 4

REQUIREMENT ANALYSIS

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders.

- **Eliciting requirements:** This is the part of the process where we need to gather requirements by communicating with the customers. Our target customers are those challenged with mental disabilities and casual users with little to no knowledge with current trends in technologies. So the interface needs to be simple and contain instructions. The interface also needs to be user friendly following all the HCI principles
- **Requirements modelling:** The requirements are documented in different formats such as use cases, user stories, natural-language documents, or process specification. The following process has been documented below and been used to create the interface model that feels comfortable to the user.
- **Review and retrospective:** Creating a functional prototype of the desired model and testing the created model to reflect on the previous iterations of requirements gathering in a bid to make improvements in the process going forward.

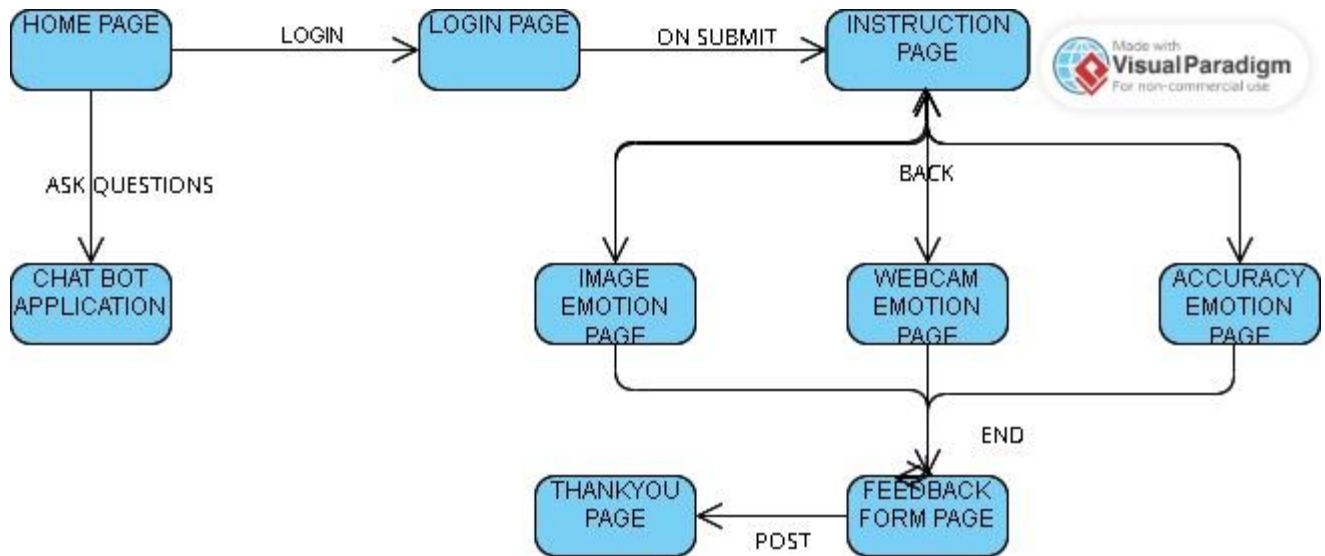
CHAPTER 5

SYSTEM REQUIREMENTS

- Operating System Used – Windows 10 Home 64-bit
- Language Used for Coding – HTML, CSS, JAVASCRIPT
- Tools Used:
 - Visual Studio Code: A source code editor with a primary focus on webdevelopment.
 - Libraries: The **face-api.js** JavaScript module implements convolutional neural networks to solve for face detection and recognition of faces and face landmarks. The face-api.js leverages TensorFlow.js and is optimized for the desktop and mobile web.
- Hardware Requirements:
 - Processor – Intel(R) Core (TM) i7-6500U CPU @ 2.50GHz(\$ CPU),~2.60GHz
 - RAM – 16.0GB
 - ROM - 512GB
 - SSD storage

CHAPTER 6

SYSTEM ARCHITECTURE AND DESIGN



Architecture Diagram: Fig 1

Analysis

- A. **Goal:** Infer and project the emotion that is given
- B. **Plans:** Using faceapi.js, a JavaScript library, initialize the program and **Task** classify an output with the projected image
- C. **Object:** JavaScript libraries(faceapi.js), Camera module, CPU System, VSCode editor, live server, browser
- D. **Objectives:** Write code to input facial image, image pre-processing with faceapi.js, and detection of the emotion presented with the 3 modules
- E. **Procedure:** Write code for each of the modules present, and corroborate with each module present to validate the workings of the code
- F. **Contingencies:** Refer to the research paper and online sources if tackled with any errors, keep track of progress made with the code and resolve the obstacles faced along the way

Task Analysis

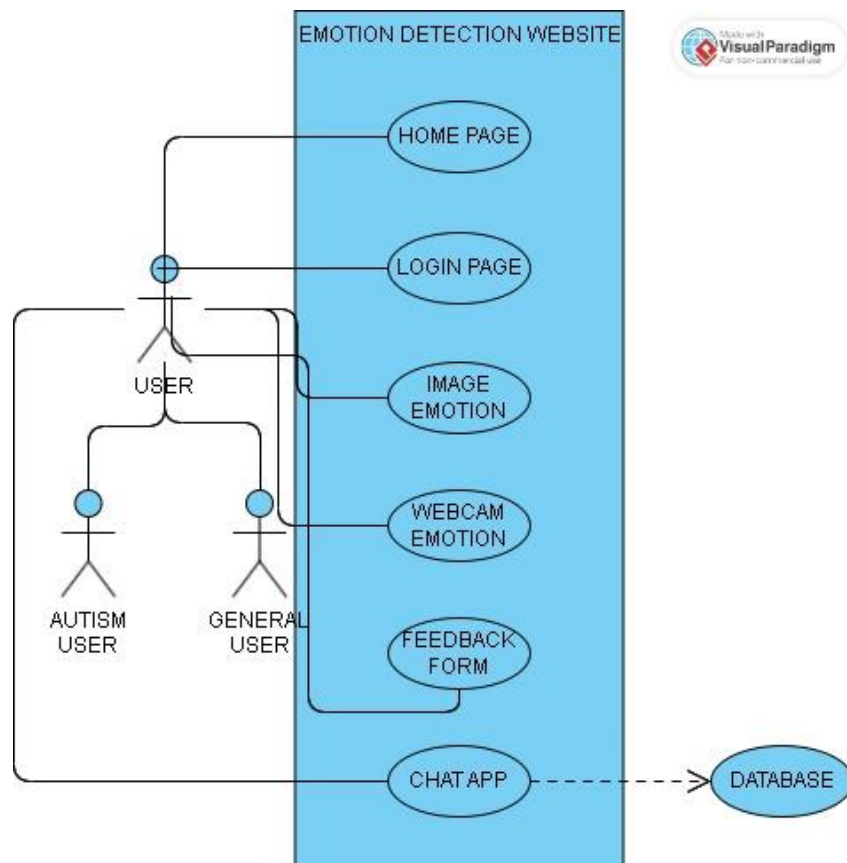
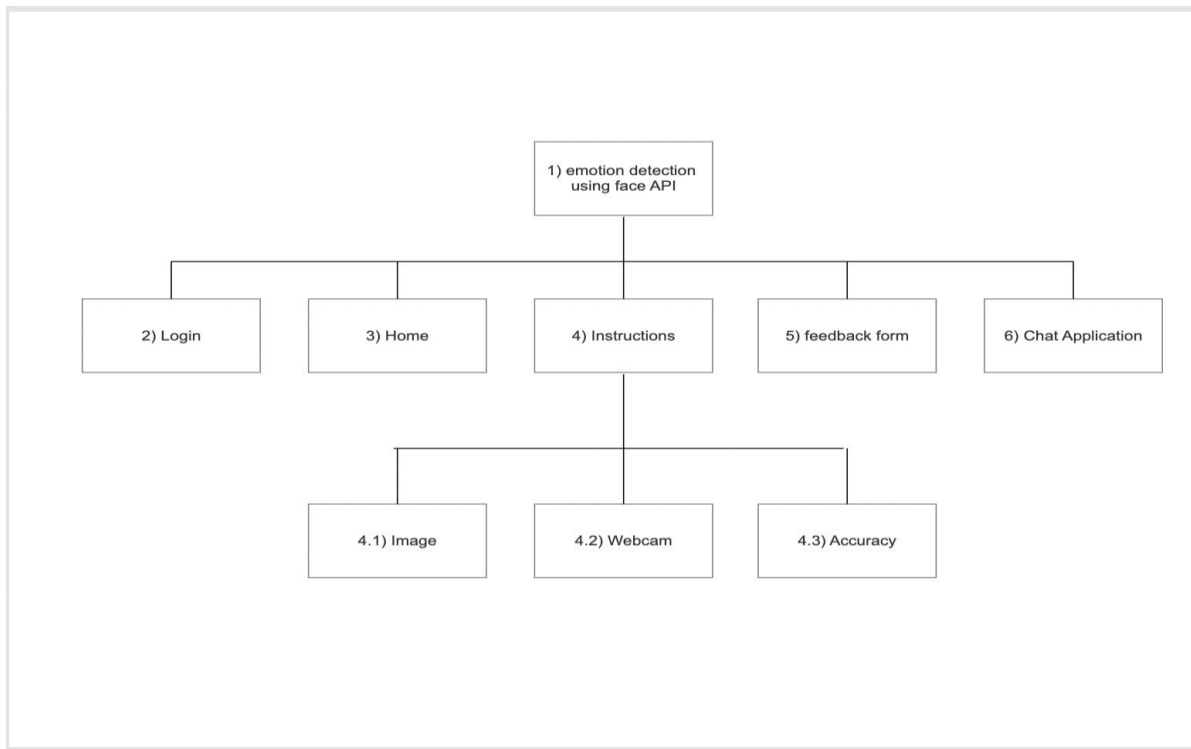


Fig 2: Use Case Diagram



Fig 3: Storyboard I depict how important emotions are.



Fig 4: Storyboard II depicts the emotion of each person

Work Approach

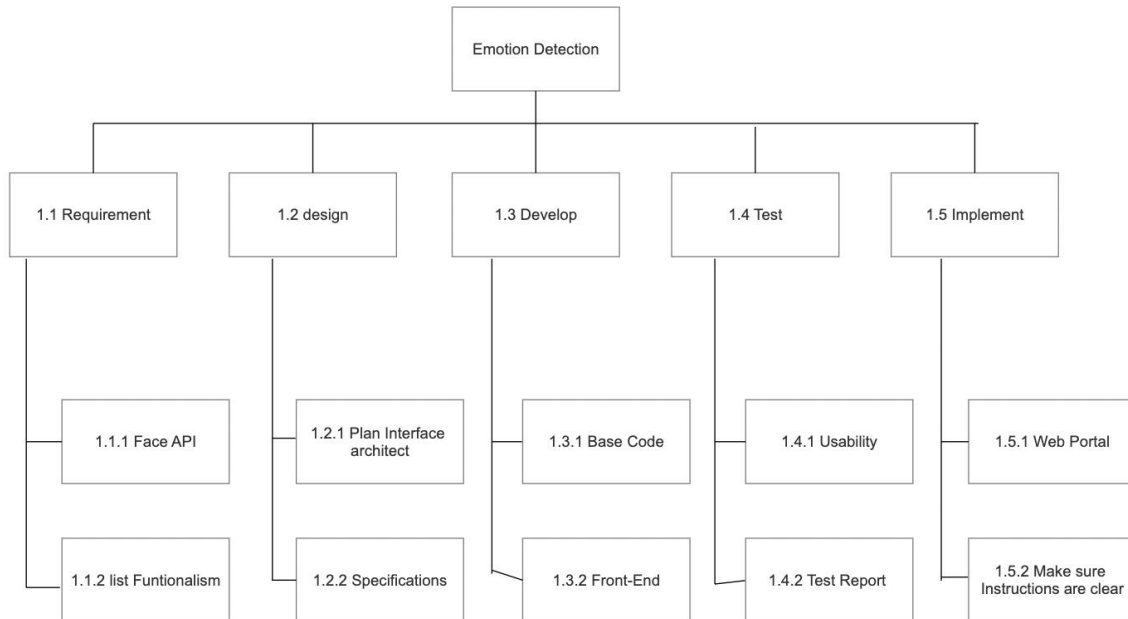
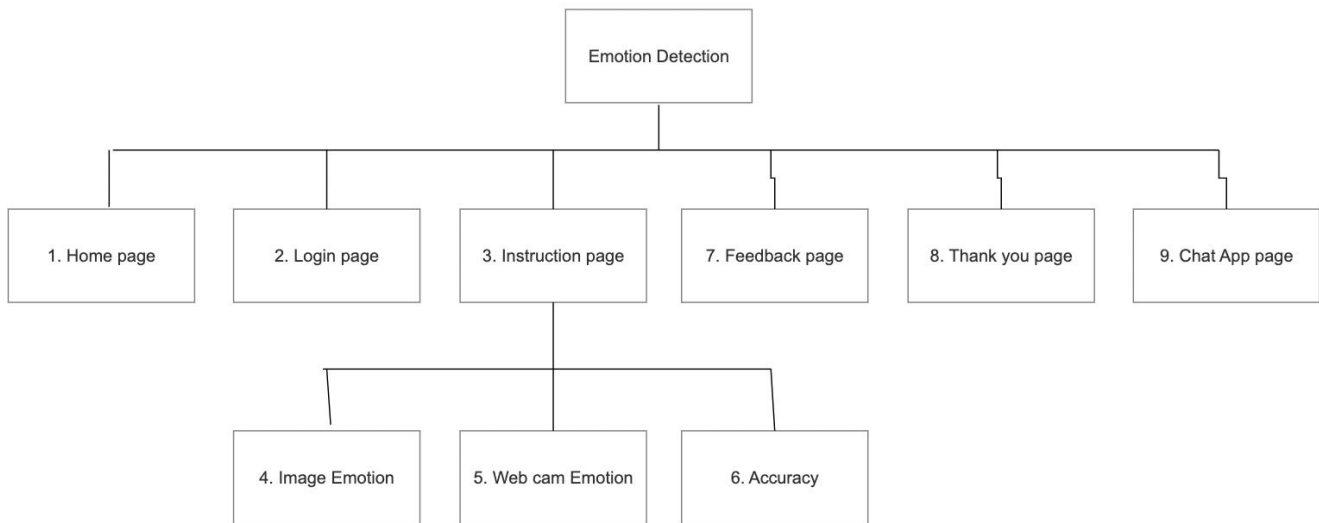


Fig 5: Work Structure

CHAPTER 7

EXPERIMENTATION AND RESULT

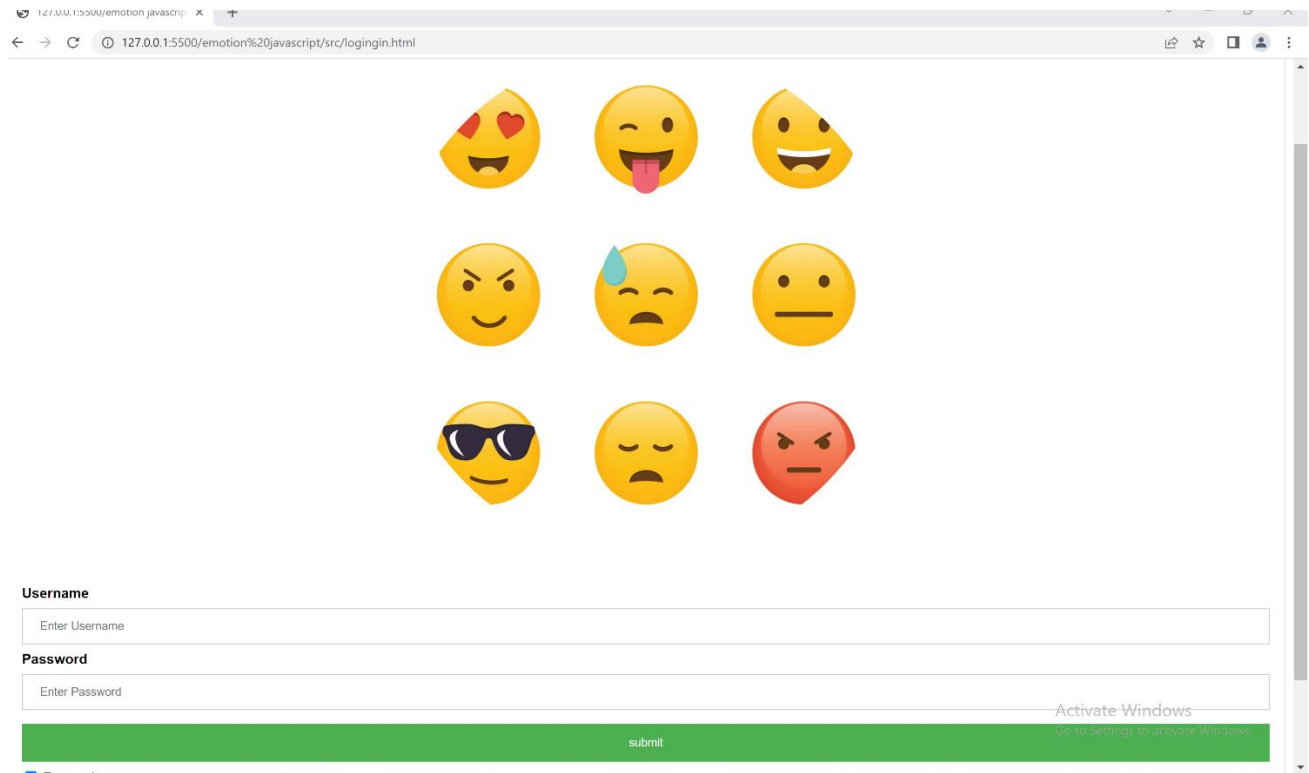


Fig 6: Login Page

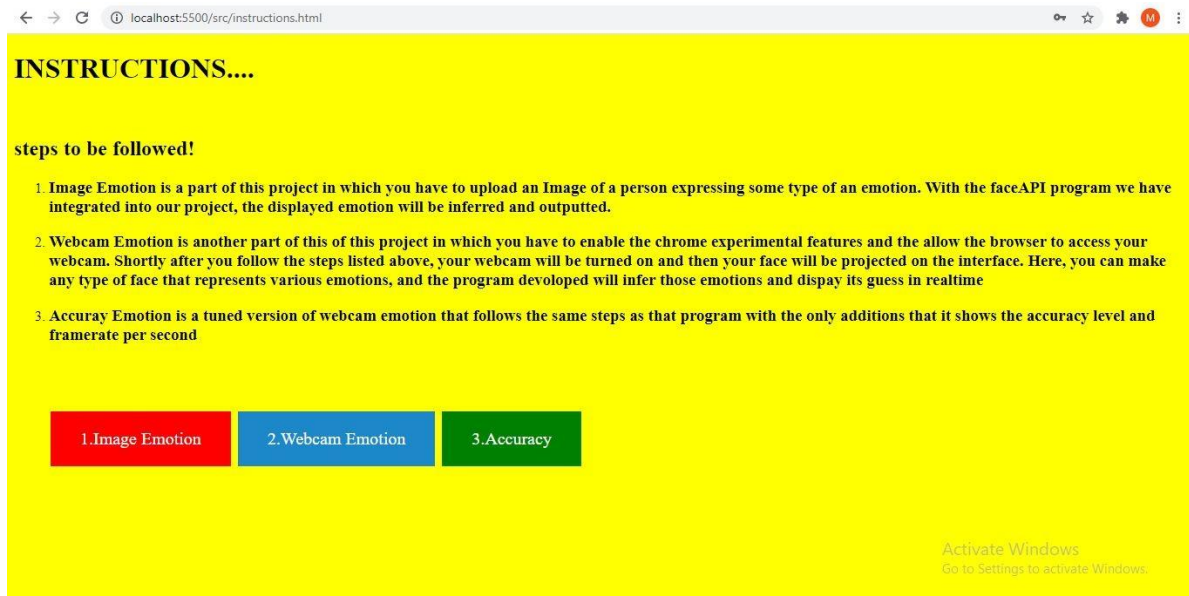


Fig 7: Instruction Page

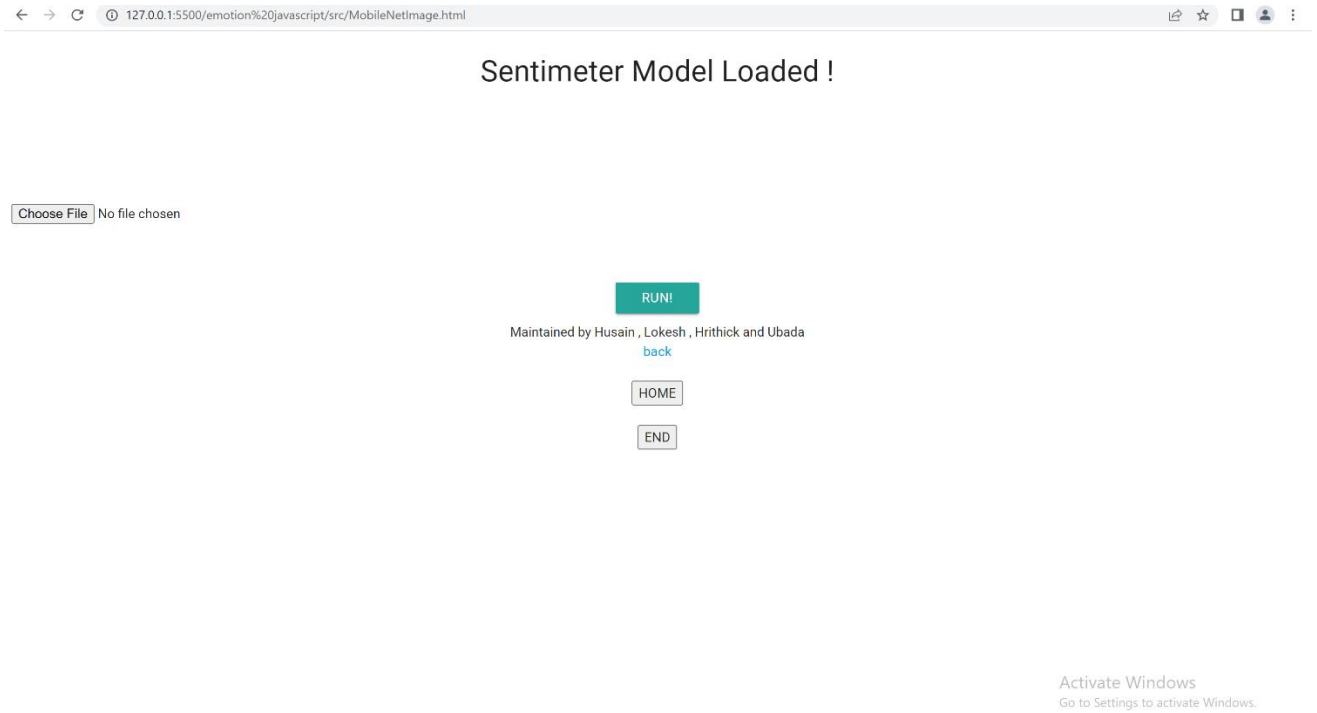


Fig 8: Image Emotion I

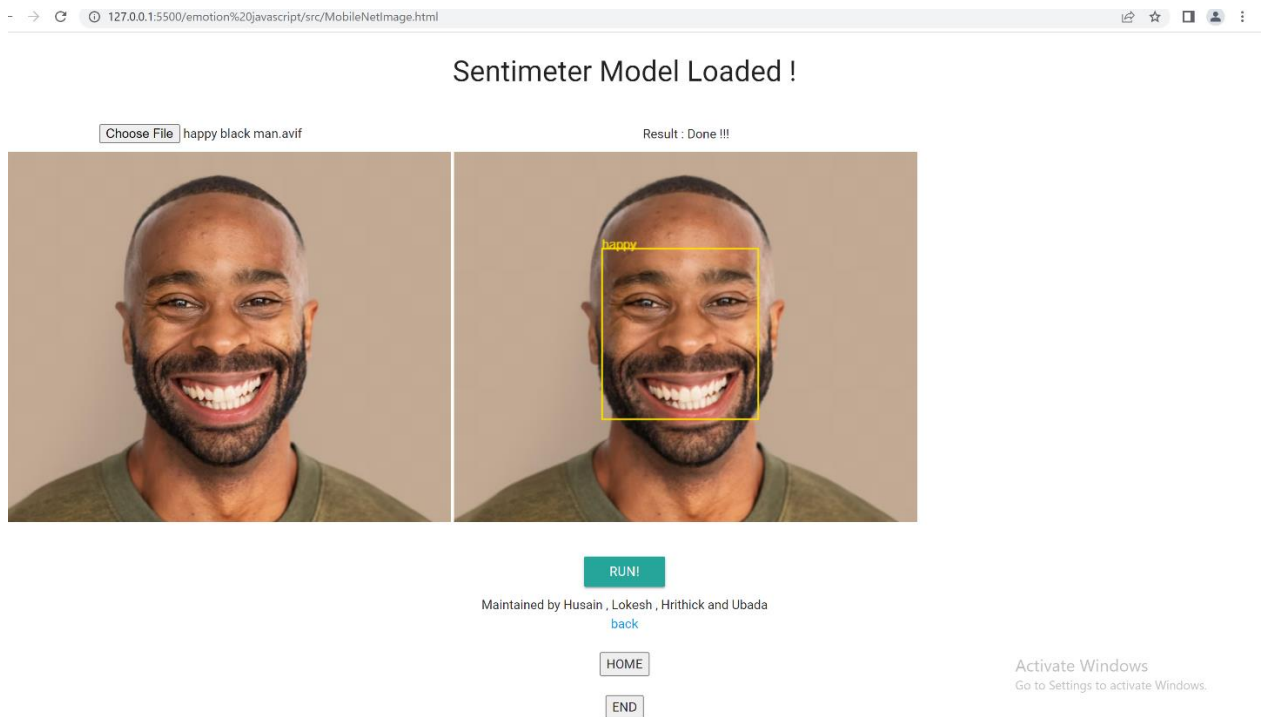


Fig 9: Image Emotion II

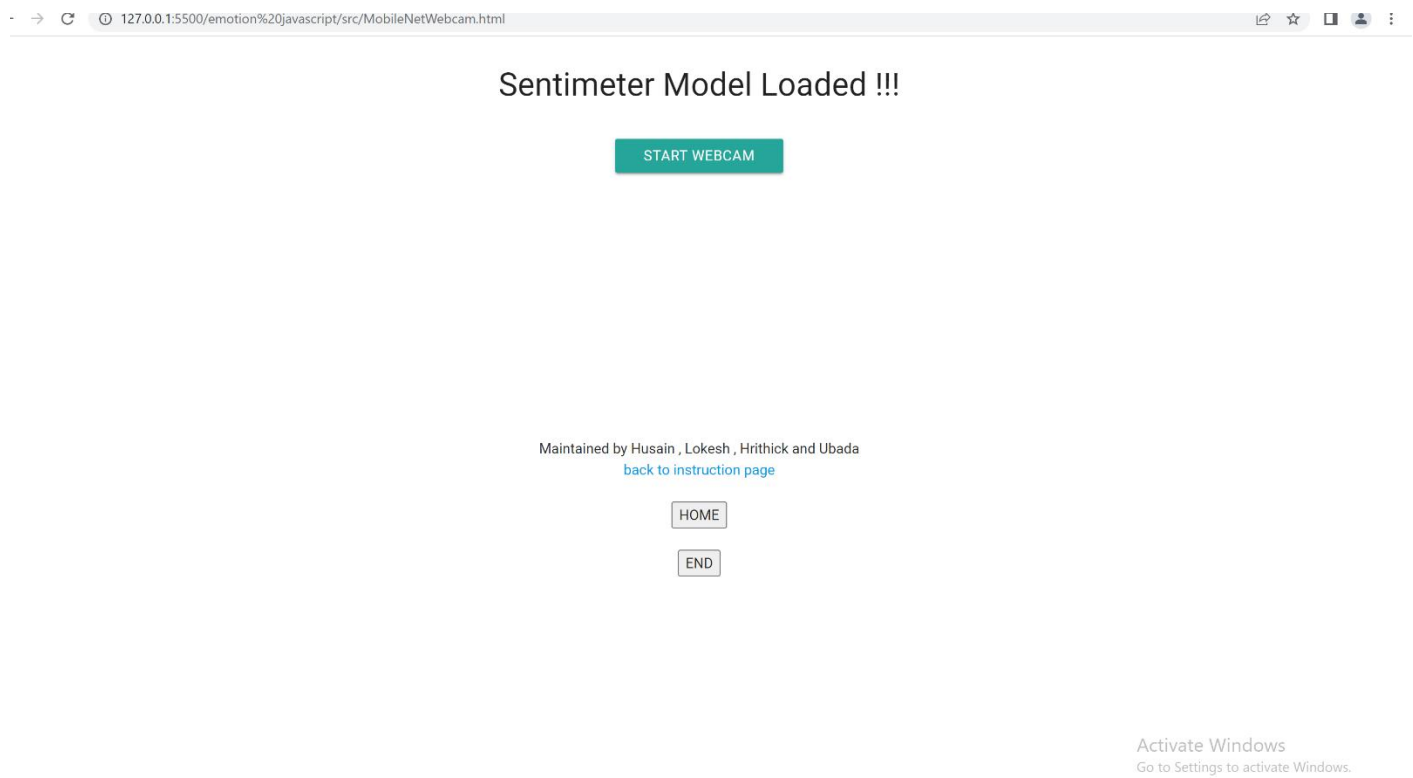


Fig 10: Webcam Emotion I

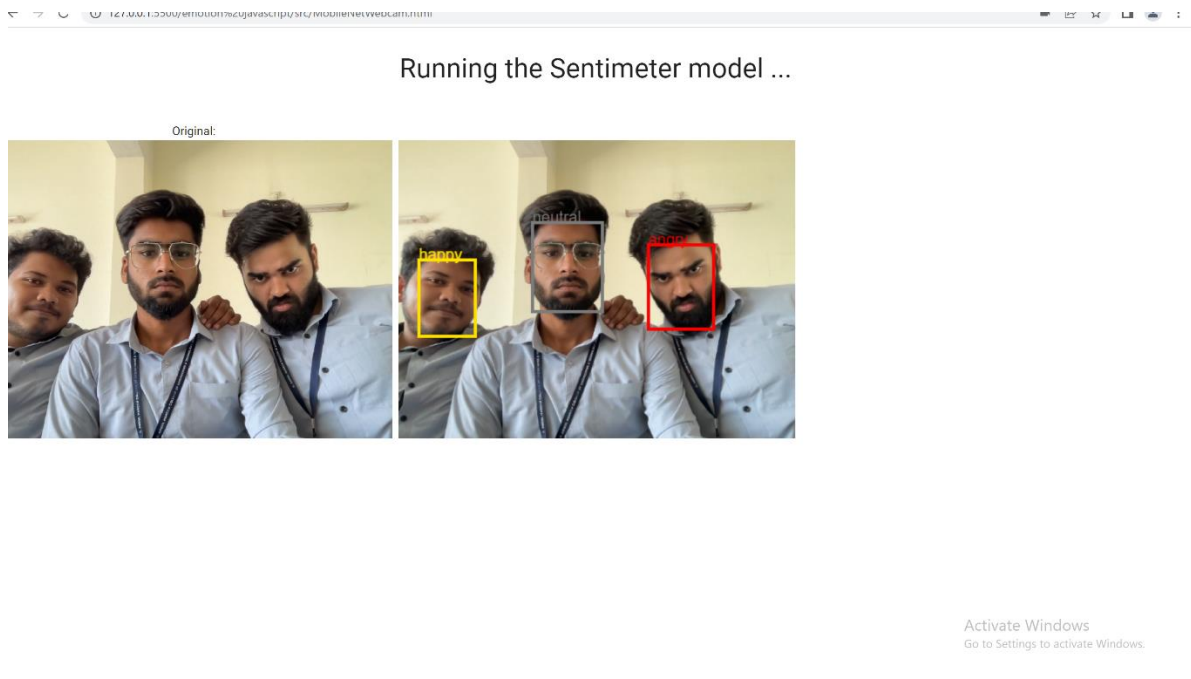


Fig 11: Webcam Emotion II

Running the Sentimeter model ...

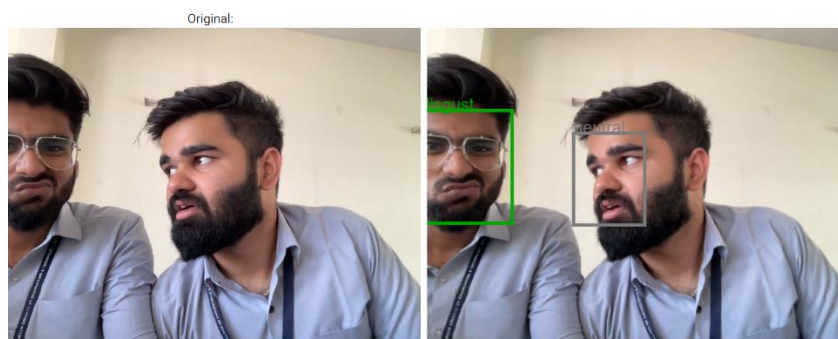


Activate Windows
Go to Settings to activate Windows.

Fig 12: Webcam Emotion III

127.0.0.1:5500/emotion%20javascript/src/MobileNetWebcam.html

Running the Sentimeter model ...



Activate Windows
Go to Settings to activate Windows.

Fig 13: Webcam Emotion IV

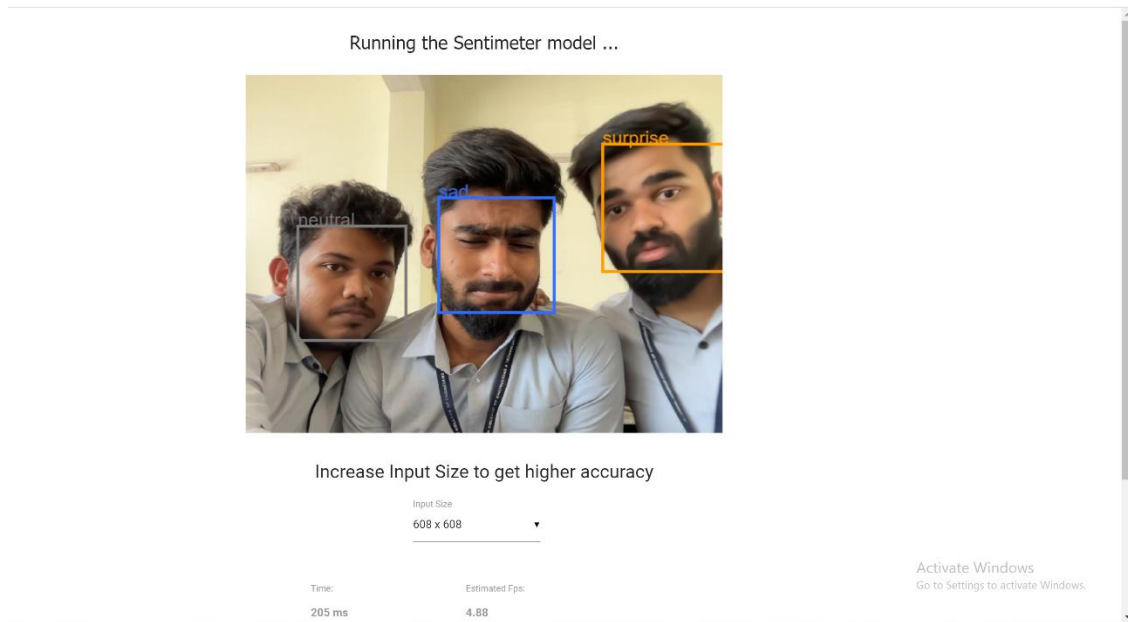


Fig 14: Accuracy Emotion I

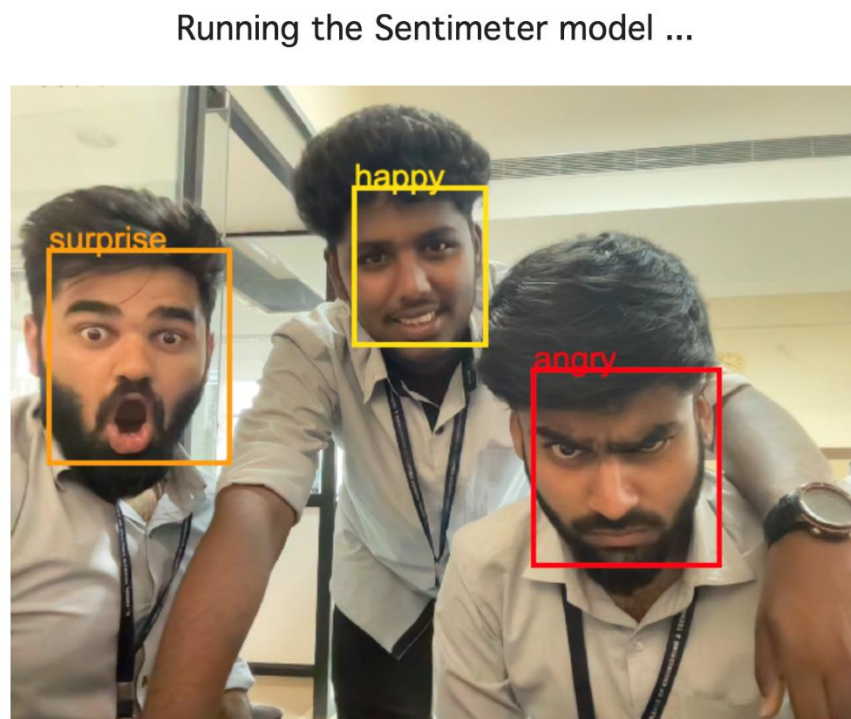


Fig 15: Accuracy Emotion II

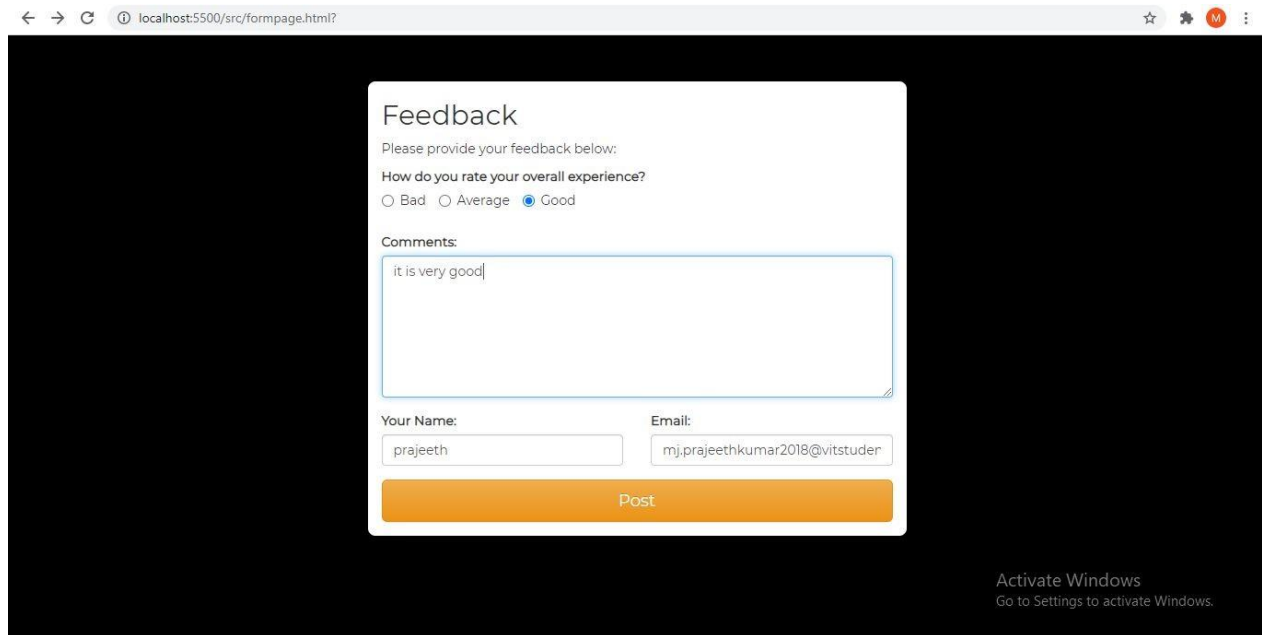


Fig 16: Feedback Page

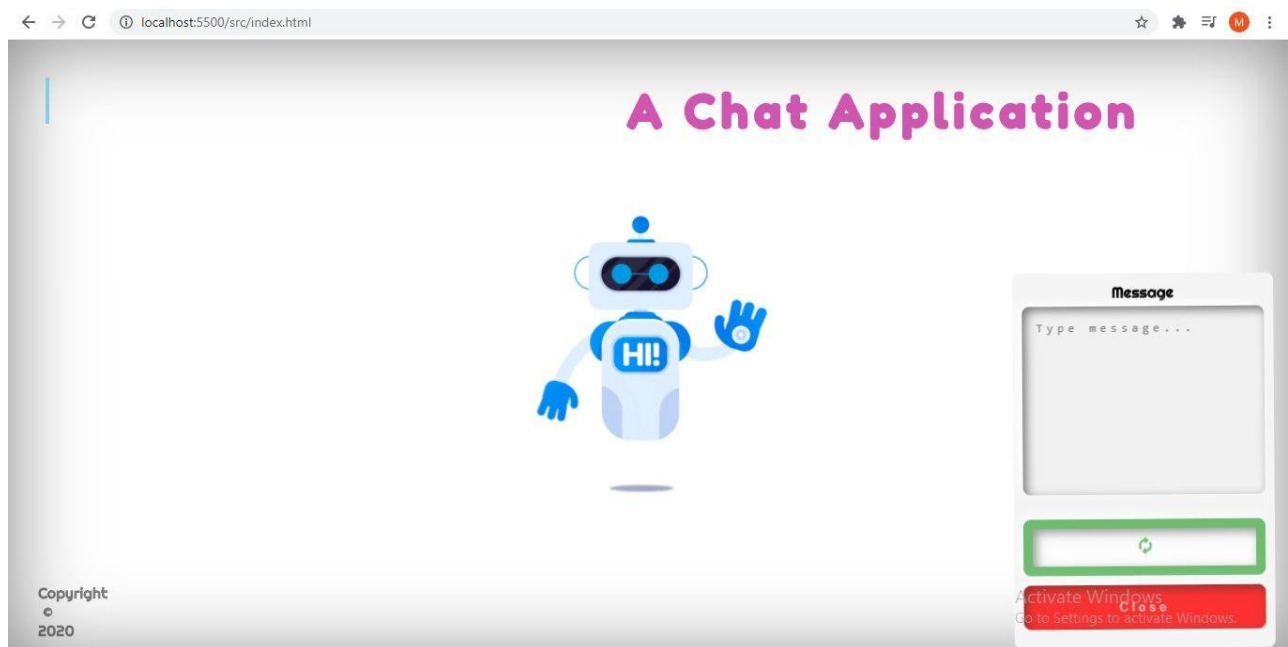


Fig 17: Chat Application Page

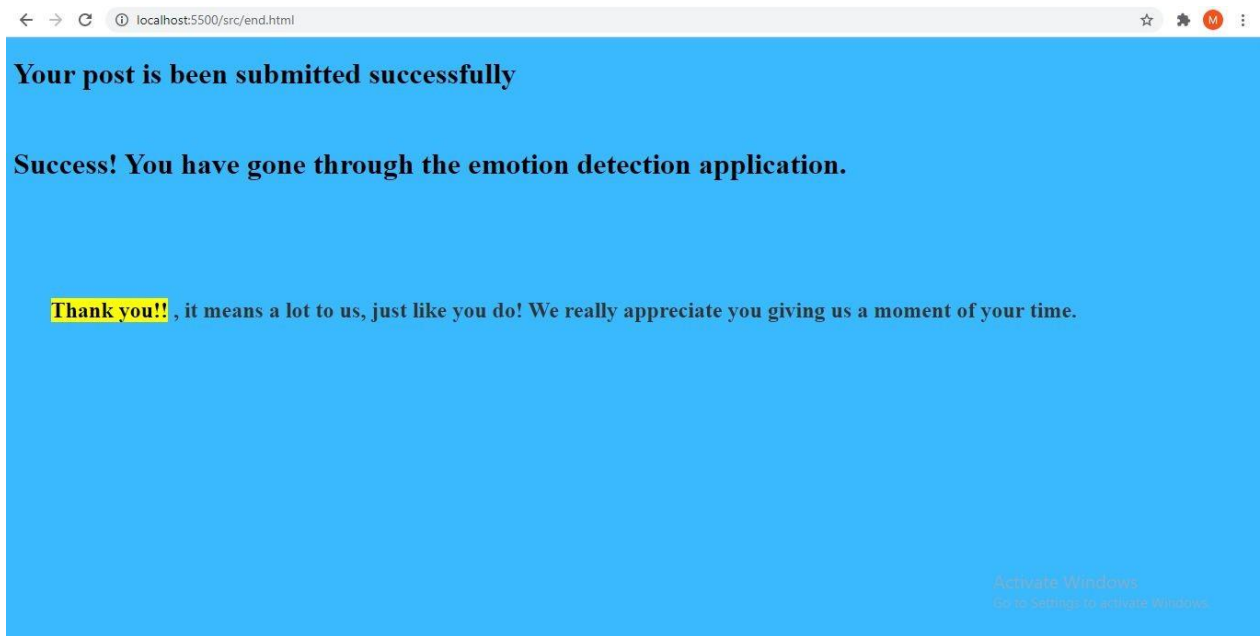


Fig 18: End Page

CHAPTER 8

TEST CASE

Prototype testing

Username and password should be made mandatory.

Error messages should come if the face is not shown in webcam.

Test report (validate the proposed work)

The validation test is a complete test before giving it to the client. Validating the login page

1. Login page

1.1. Giving username and password

1.1.1. Test case 1:

Purpose: to verify that username and password entered or not

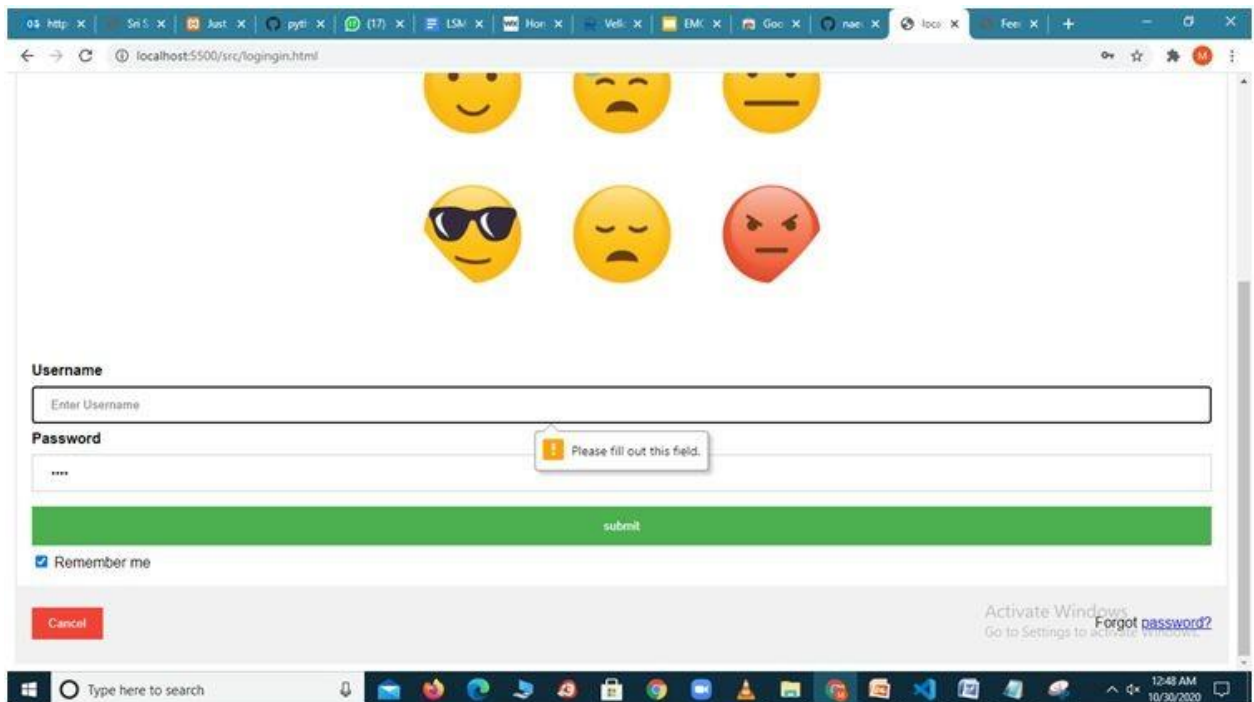


Fig 19: Validation Testing I

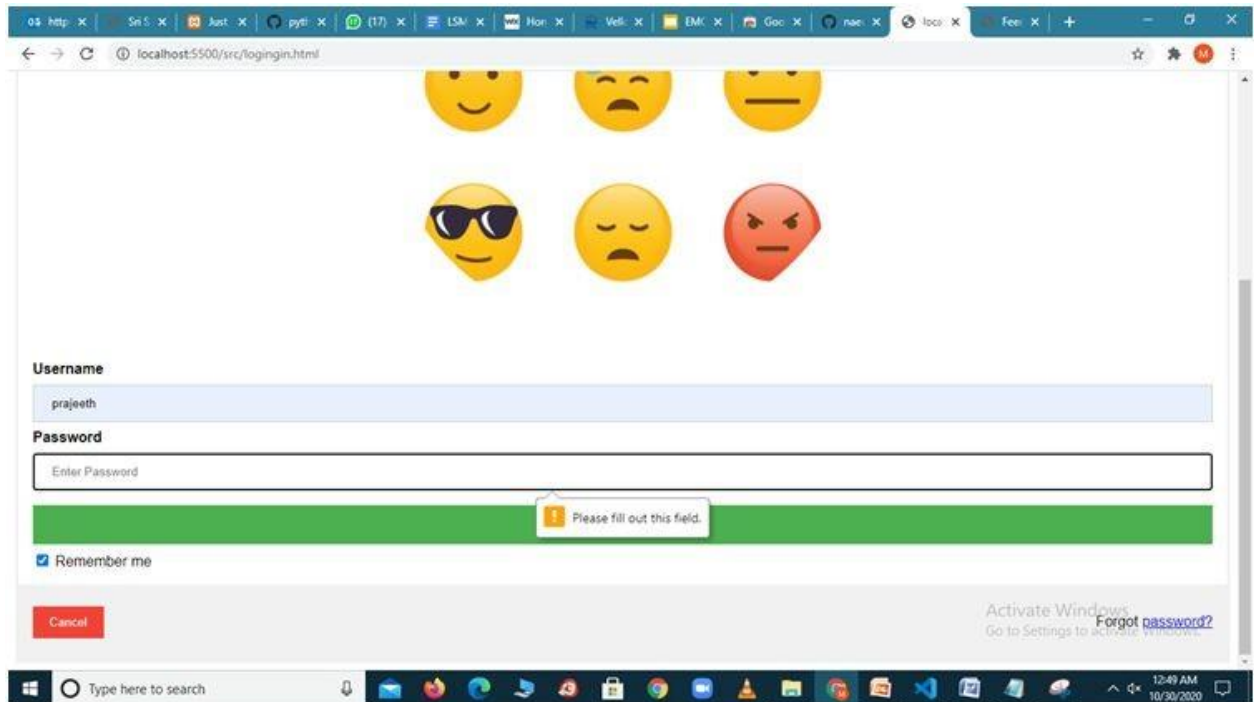


Fig 20: Validation Testing II

Validating webcam

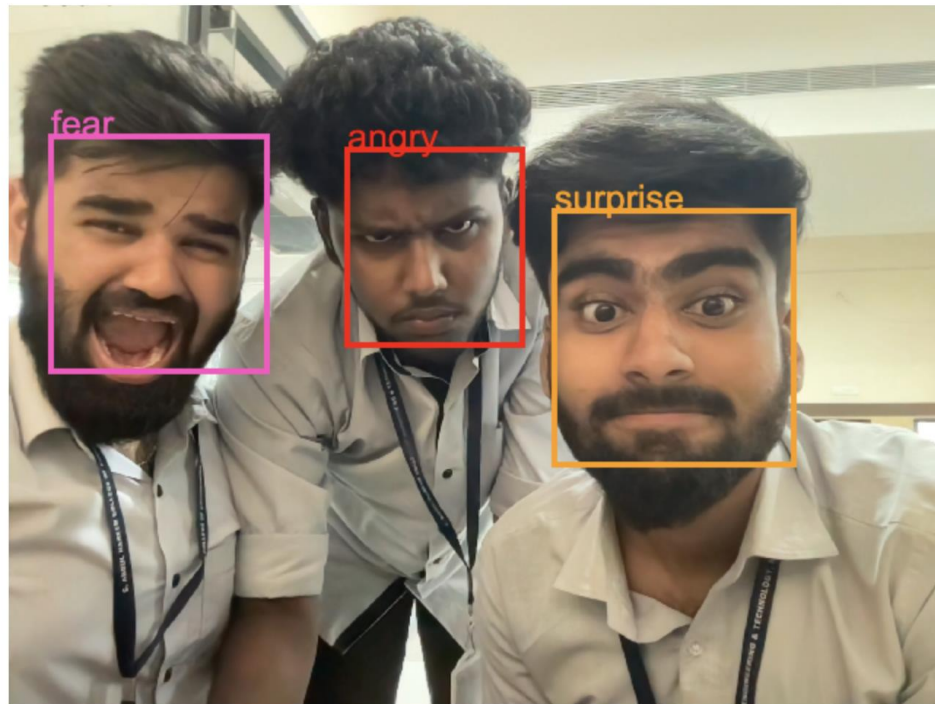
2. webcam page

2.1. Face detected or not

2.1.1. Test case 2

Purpose: to verify that whether the face is detected or not in webcam Itdetects only the faces and not any other things.

Running the Sentimeter model ...



Increase Input Size to get higher accuracy

Fig 21: Validation Testing III

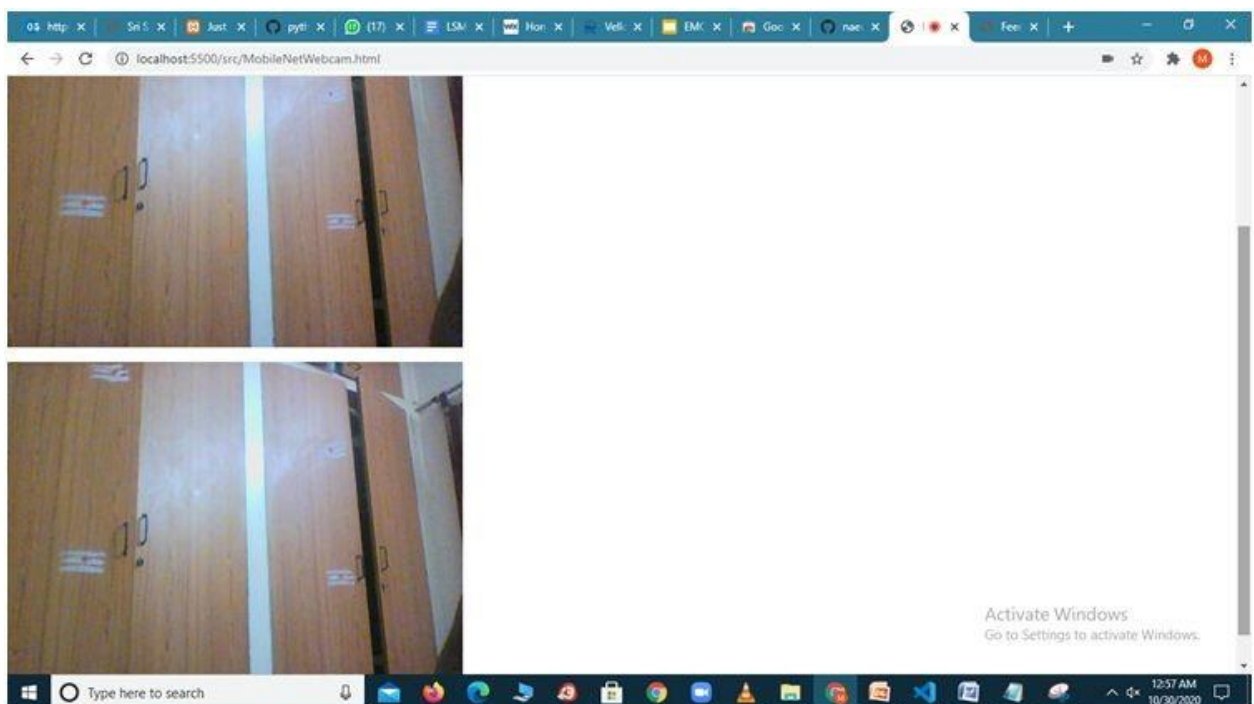


Fig 22: Validation Testing IV

Validating image emotion

Only the image with face reactions only the emotion is detected. The emotion isn't detected for any other images. It shows no face in the result if face is not in the image.

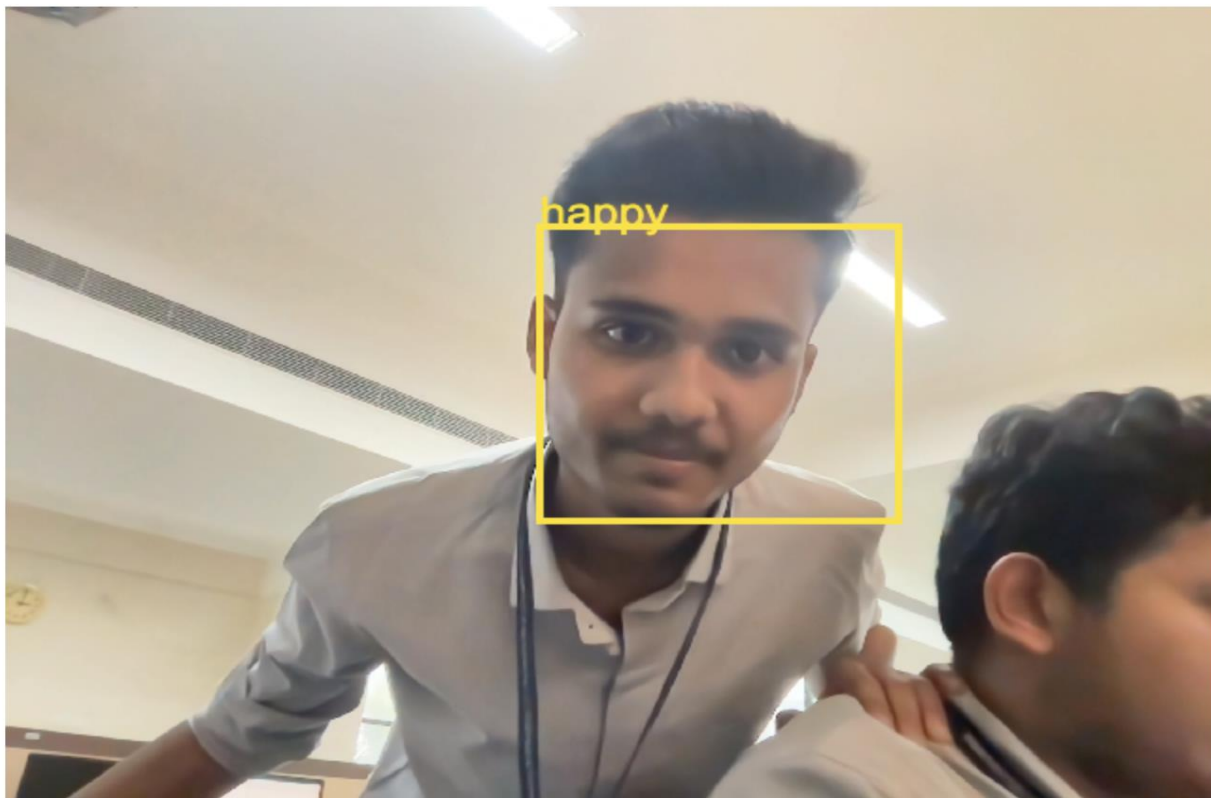
1. Webcam page

3.1. Face detected or not in image

3.1.1. Test case 3

Purpose: to verify that whether the face is detected or not in image emotion
It detects only the faces and not any other images

Running the Sentimeter model ...



Increase Input Size to get higher accuracy

Fig 23: Validation Testing V

CHAPTER 9

USEABILITY TESTING

We collected the feedback of our website from users through google forms.

Questions:

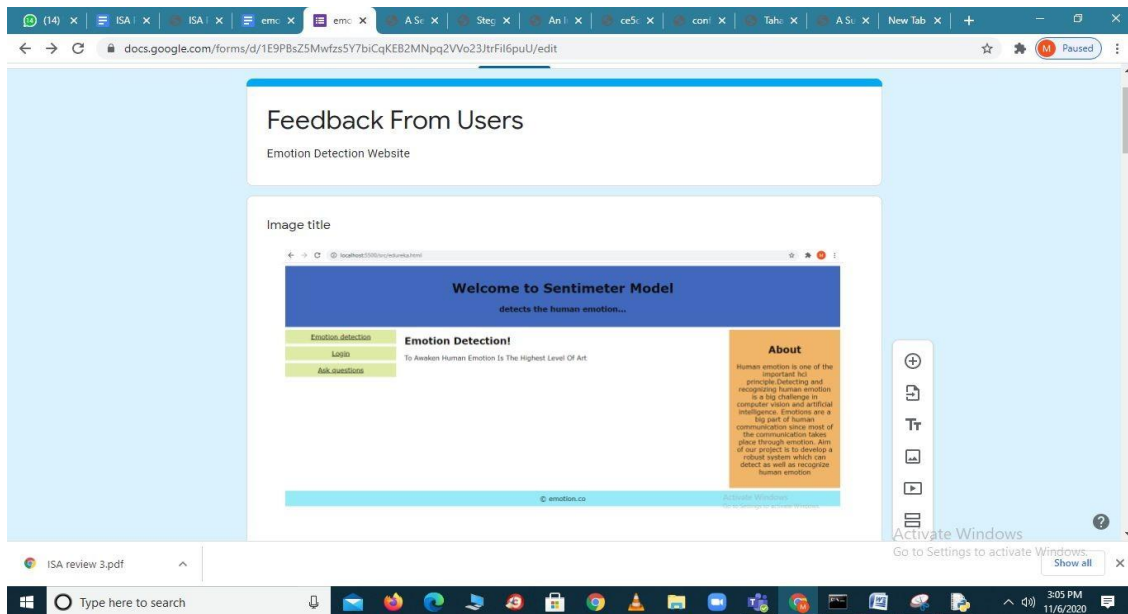


fig 24: Feedback Form I

Fig 25: Feedback Form II

The screenshot shows a Google Forms editor interface for a form titled "emotion detection". The form is currently in "Questions" mode, and there are 3 responses recorded. The questions are as follows:

- Question 1: "Is there any feature that needs to be added to the site?*" (Long answer text)
- Question 2: "What did you dislike about the site?*" (Long answer text)
- Question 3: "How would rate the website (from 1 to 5)?" (Long answer text)

The interface includes a top navigation bar with various icons and a "Send" button. The bottom of the screen shows a Windows taskbar with the date 11/6/2020 and time 3:05 PM.

Fig 26: Feedback Form III

Responses or Feedback:

The screenshot shows the "Responses" tab for the "emotion detection" form. It displays 9 responses to the question "What was your first impression of the site?". The responses are listed in a table:

Response
Excellent
Wowwww
Nice
I found this to be extremely innovative
It was user friendly and easy to use.
User friendly
Very good site with accurate detection of emotions
Good
It's cool and innovative

The interface includes a top navigation bar with various icons and a "Send" button. The bottom of the screen shows a Windows taskbar with the date 11/6/2020 and time 3:05 PM.

Fig 27: Feedback Form IV

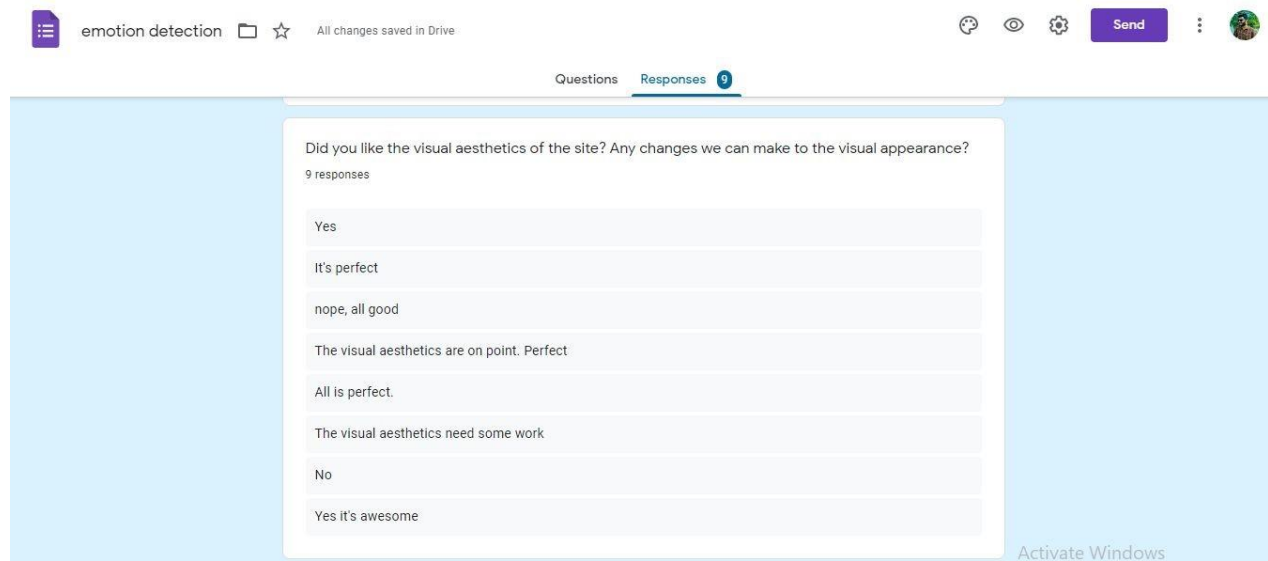


Fig 28: Feedback Form V

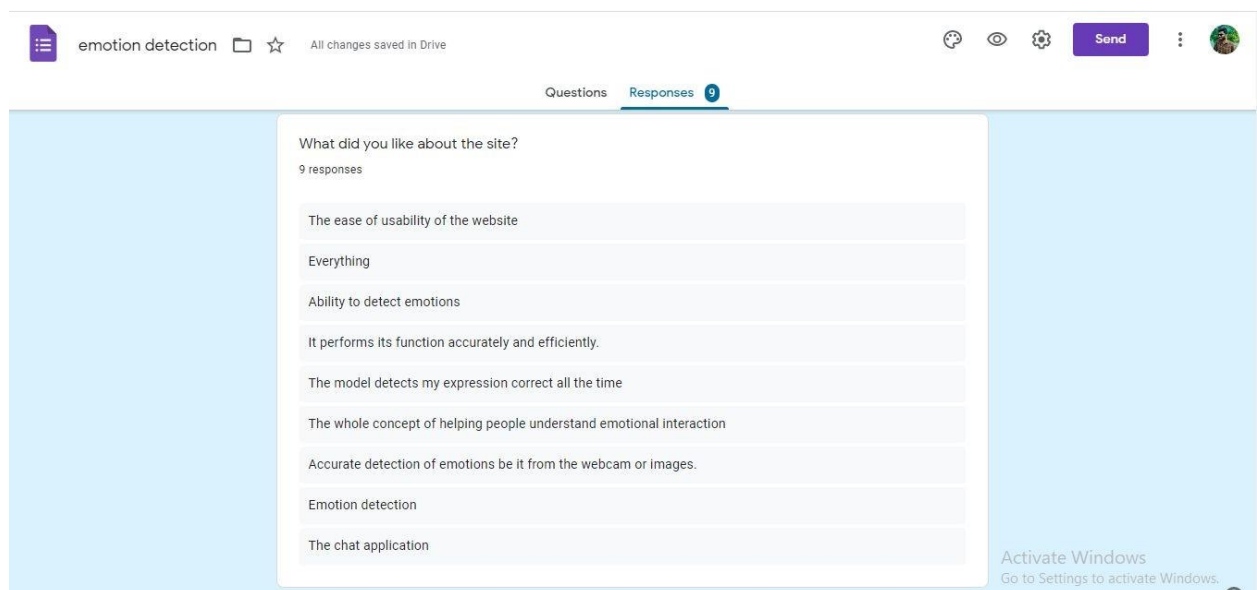


Fig 29: Feedback Form VI

emotion detection All changes saved in Drive

Questions Responses 9

Is there any feature that needs to be added to the site?

9 responses

- No
- More pictures
- Nothing to be added
- nah
- Not necessarily
- This can be integrated with social media apps.
- An improved interface
- Nothing

Activate Windows
Go to Settings to activate Windows.

Fig 30: Feedback Form VII

emotion detection All changes saved in Drive

Questions Responses 9

What did you dislike about the site?

9 responses

- Nothing
- N/A
- Stop joking
- nothing
- Nothing.
- Color scheme
- Nothing

How would rate the website (from 1 to 5)

9 responses

- 5
- 4
- 4.5

Activate Windows
Go to Settings to activate Windows.

Fig 31: Feedback Form VIII

CHAPTER 10

HCI PRINCIPLES INCORPORATED

- **Visibility of system status:** The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
 - A notice be provided by the browser that the camera is to be turned on and the application is going to be launched promptly.
- **Match between system and the real world:** The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.
 - Since the project contains some complexity, and new users might be overwhelmed by its working, we are providing an instruction screen for the users of what the experience might be like.
- **User control and freedom:** Users often choose system functions by mistake and will need a clearly marked exit to leave the unwanted state without having to go through an extended dialogue.
 - If the user wants to leave the application, we have planned to provide them a button that will close the web interface the program is launched in.
- **Consistency and standards:** Users should not have to wonder whether different words, situations, or actions mean the same thing.
 - All platform conventions are going to be followed that support recall and recognition actions for a smoother interaction
- **Error prevention:** Even better than good error messages is a careful design which prevents a problem from occurring in the first place.
 - Dialogue box is going to be set for the user to confirm if they want to terminate the application

- **Recognition rather than recall:** Minimize the user's memory load by making objects, actions, and options visible.

- Instructions will be placed in the screen of the user, so user can comprehend what they need to if they get stuck anywhere

- **Flexibility and efficiency of use:** Cater to both inexperienced and experienced users, and allow users to tailor frequent actions.

- Since the project isn't large scale, accelerators aren't required for efficiency. We cater to the users flexibility with ease in operating the application

- **Aesthetic and minimalist design:** Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

- The UI/UX of the application is very minimalistic, containing only the information and components for a smooth interaction

- **Help users recognize, diagnose, and recover from errors:** Error messages should be expressed in plain language, precisely indicate the problem, and constructively suggest a solution.

- Accessing the webcam from the user might be an issue that we face, and we plan to leave a solution to the users on how to fix this if they encounter these errors along with others

- **Help and documentation:** It may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large

- We plan to have a txt file with proper instructions if the users decide to run the application in their computer system.

CHAPTER 11

SOURCE CODE

```
#REGISTER PAGE
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>
<style>
Body {
    font-family: Calibri, Helvetica, sans-serif;
    background-color: pink;
}
button {
    background-color: #4CAF50;
    width: 100%;
    color: orange;
    padding: 15px;
    margin: 10px 0px;
    border: none;
    cursor: pointer;
}
form {
    border: 3px solid #f1f1f1;
}
input[type=text], input[type=password] {
    width: 100%;
    margin: 8px 0;
    padding: 12px 20px;
    display: inline-block;
    border: 2px solid green;
    box-sizing: border-box;
}
button:hover {
    opacity: 0.7;
}
.cancelbtn {
```

```

width: auto;
padding: 10px 18px;
margin: 10px 5px;
}

.container {
padding: 25px;
background-color: lightblue;
}
</style>
</head>
<body>
<center> <h1> Student Login Form </h1> </center>
<form>
<div class="container">
<label>Username : </label>
<input type="text" placeholder="Enter Username" name="username" required>
<label>Password : </label>
<input type="password" placeholder="Enter Password" name="password" required>
<button type="submit">Login</button>
<input type="checkbox" checked="checked"> Remember me
<button type="button" class="cancelbtn"> Cancel</button>
Forgot <a href="#"> password? </a>
</div>
</form>
</body>
</html>

#LOGIN PAGE
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial, Helvetica, sans-serif;}
form {border: 3px solid #f1f1f1;}

```

```

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #4CAF50;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  cursor: pointer;
  width: 100%;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 40%;
  border-radius: 50%;
}

```

```

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
    display: block;
    float: none;
  }
  .cancelbtn {
    width: 100%;
  }
}
</style>
</head>
<body>

<h2>Login Form</h2>

<form action="instructions.html" method="post">
  <div class="imgcontainer">
    
  </div>

  <div class="container">
    <label for="uname"><b>Username</b></label>
    <input type="text" placeholder="Enter Username" name="uname" required>

    <label for="psw"><b>Password</b></label>
    <input type="password" placeholder="Enter Password" name="psw" required>

```

```

    <button type="submit">submit</button>
    <label>
        <input type="checkbox" checked="checked" name="remember"> Remember me
    </label>
</div>

<div class="container" style="background-color:#f1f1f1">
    <button type="button" class="cancelbtn">Cancel</button>
    <span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>

</body>
</html>

#WEBCAM
<!DOCTYPE html>
<html>

<head>

    <script type="text/javascript" src="https://code.jquery.com/jquery-
2.1.1.min.js"></script>
    <script
src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@0.13.0"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"></sc
ript>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/0.100.2/css/materiali
ze.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></scrip
t>
    <style>
        #wrapper {
            text-align: center;
            margin: 10px auto;

```

```

    }

    #overlay {
        width: 550px;
    }

    #output_image {
        width: 550px;
    }

    .inline {
        display: inline-block;
    }
</style>
</head>

<body>
    <center style="padding:10px; margin:5px">
        <h4>
            <span id="status">Sentimeter Loading ...</span>
        </h4>
    </center>
    <center style="padding:10px">
        <button type="button" id='start' class="btn btn-outline-primary"
onclick='startVideo()'>Start webcam</button>
    </center>
    <div class="inline">
        <center><span id="original_video"></span></center>
        <video id='video' width="500px"></video>
    </div>

    <div class="inline">
        <!--      <center><span id="emotion_video"></span></center>-->
        <canvas id="canvas" src="" style=" margin-top:10px;" />
    </div>
    <div id = "thank">
        <center style="padding:10px; margin:5px">
            <div>Maintained by HLHU</div>
            <a href="instructions.html">back to instruction page</a>
        </center>
    </div>

```



```

<br>
<br>
<form>
    <button type="submit" formaction="edureka.html">HOME</button>

    <br>
    <br>

    <button type="submit" formaction="formpage.html">END</button>

</center>
</div>
</body>
<script>
    // check using phone or not
    if (/Android|webOS|iPhone|iPod|BlackBerry/i.test(navigator.userAgent)) {
        alert('Sorry, this website not support mobile devices currently.');
```

history.back();

```

    }

    // check using chrome
    if (!window.chrome) {
        if (confirm('This website needs Chrome browser!!!')) {
            closewin();
        } else {
            history.back();
        }
    }

    var constraints = {
        video: true
    };

    var $body = document.querySelector('body');
    // var status = document.getElementById('status');
    var emotion_labels = ["angry", "disgust", "fear", "happy", "sad",
    "surprise", "neutral"];

```

```

var emotion_colors = ["#ff0000", "#00a800", "#ff4fc1", "#ffe100",
"#306eff", "#ff9d00", "#7c7c7c"];
var offset_x = 15;
var offset_y = 40;

loadModel('../models/mobilenetv1_models/model.json')
// create model
async function createModel(path) {
    let model_tmp = await tf.loadModel(path)
    return model_tmp
}

// load models
async function loadModel(path) {
    var status = document.getElementById('status');
    // status.innerText = "Model Loading ..."
    model = await createModel(path)

    status.innerText = " Sentimeter Model Loaded !!!"
}

function createVideoElement() {
    let $video = document.createElement('video')
    $video.style.maxWidth = '100vw'
    $video.style.width = '100vw'
    $video.style.maxHeight = '0vh'
    $body.appendChild($video)
    return $video
}

function handleError(error) {
    if (error.name === 'DevicesNotFoundError') {
        alert('No camera detected. <br> Do you have any camera
connected?');
    } else if (error.name === 'NotAllowedError') {
        alert('You have to allow camera access in order to run this
experiment.');
```

```

    } else if (navigator.userAgent.indexOf('Chrome') > -1) {
        alert('Error. <br> Enable experimental features on
chrome://flags/#enable-experimental-web-platform-features');
    } else {
        alert('Error. <br> Does your browser supports FaceDetector API?');
    }
    console.error(error)
}

function createCanvas(video) {

    const canvas = document.getElementById('canvas')
    const videoCompStyle = window.getComputedStyle(video)
    canvas.width = videoCompStyle.width.replace('px', '')
    canvas.height = videoCompStyle.height.replace('px', '')
    //      canvas.style.display = 'none'
    document.querySelector('body').appendChild(canvas)

    return canvas
}

function createDrawFunction() {

    const faceDetector = new window.FaceDetector({
        maxDetectedFaces: 3
    })
    let isDetectingFaces = false
    let faces = []
    let hideTimeout

    return async function draw(canvas, video) {
        window.requestAnimationFrame(() => draw(canvas, video))
        const context = canvas.getContext('2d')
        const videoCompStyle = window.getComputedStyle(video)
        const videoWidth = videoCompStyle.width.replace('px', '')
        const videoHeight = videoCompStyle.height.replace('px', '')
        context.drawImage(video, 0, 0, videoWidth, videoHeight)
    }
}

```

```

        //          context.clearRect(0, 0, canvas.width,
canvas.height);
        //          clearTimeout(hideTimeout)
        if (faces.length) {

            //          let canvas =
document.getElementById('canvas')
            let ctx = context;
            //          let scale = 1;

            ctx.lineWidth = 4;
            ctx.font = "20px Arial"
            ctx.fillText('Result', 0, 0);
            for (var i = 0; i < faces.length; i++) {
                ctx.beginPath();
                var item = faces[i].boundingBox;
                //          console.log(item)
                let s_x = Math.floor(item.x - offset_x / 2);
                let s_y = Math.floor(item.y - offset_y / 2);
                let s_w = Math.floor(item.width + offset_x);
                let s_h = Math.floor(item.height + offset_y);

                let cT = ctx.getImageData(s_x, s_y, s_w, s_h);
                cT = preprocess(cT);
                z = model.predict(cT)

                let index = z.argmax(1).dataSync()[0]
                let label = emotion_labels[index];

                ctx.strokeStyle = emotion_colors[index];
                ctx.rect(s_x, s_y, s_w, s_h);
                ctx.stroke();
                ctx.fillStyle = emotion_colors[index];
                ctx.fillText(label, s_x, s_y);
                ctx.closePath();
            }
        } else {
            console.log('NO FACE')

```

```

        //                status.innerHTML = "NO FACE";
    }

    if (isDetectingFaces) {
        return
    }

    isDetectingFaces = true
    faces = await faceDetector.detect(canvas)
    isDetectingFaces = false
    var status = document.getElementById('status');
    status.innerHTML = "Running the Sentimeter model ... ";
}
}

function preprocess(imgData) {
    return tf.tidy(() => {
        let tensor = tf.fromPixels(imgData).toFloat();

        tensor = tensor.resizeBilinear([100, 100])

        tensor = tf.cast(tensor, 'float32')
        const offset = tf.scalar(255.0);
        // Normalize the image
        const normalized = tensor.div(offset);
        //We add a dimension to get a batch shape
        const batched = normalized.expandDims(0)
        return batched
    })
}

function playCameraOnVideo(video) {
    return navigator.mediaDevices.getUserMedia({
        video: {
            facingMode: 'user',
            frameRate: 60
        },
    },

```

```

        audio: false
    })
    .then(srcObject => video.srcObject = srcObject)
    .then(() => video.play())
}

async function main(video) {
    const video_canvas = createCanvas(video)
    const draw = createDrawFunction()
    draw(video_canvas, video)
}

function startVideo() {
    var x = document.getElementById("thank");
    x.style.display = "none";
    let elem = document.getElementById('start');
    elem.parentNode.removeChild(elem);
    var status = document.getElementById('status');
    status.innerHTML = "Initializing the camera ... ";

    var ori = document.getElementById("original_video");
    //      var emo = document.getElementById("emotion_video");
    ori.innerHTML = "Original: "
    //      emo.innerHTML = "Result : "
    playCameraOnVideo(video)
        .then(() => main(video))
        .catch(handleError)

}
</script>

</html>

#APP.JS
// ===== G O O D   L U C K ===== //

// Open or Close the chat button...

```

```

function openForm() {
    document.getElementById("myForm").style.display = "block";
}

function closeForm() {
    document.getElementById("myForm").style.display = "none";
}

// Chat with Bot.....

know = {
    Hello: "Hi, Sir. Good to see you.",
    What is your name?: "MOGA",
    Who are you?: "<b>I am a Chat Bot who created by Jayed Ahsan Saad.</b>",
    Damm: "you bullshit.",
    Tell me something: "I can't do it.",
    Lol: "Don't use this one. I'll kick your butt.",
    Hi: "Hello World! Nice to meet you.",
    XD: "You think yourself smarter than me.",
    I am a human:
        "Human have emotions. It is hard to understand. But I have dream to buy a
bike.",
    Thank you: "No need.",
    This is bs: "I will make you red.",
};

function talk() {
    var user = document.getElementById("userBox").value;
    document.getElementById("userBox").value = "";
    document.getElementById("chatLog").innerHTML = user + "<br>";

    if (user in know) {
        document.getElementById("chatLog").innerHTML = know[user] + "<br>";
    } else {
        document.getElementById("chatLog").innerHTML =
            "I don't understand everything because I am a robot.<br><br><b>Use
capital letter at first.<b><br> It will help me to understad you."; // default
    }
}

```

```

}
// Random Password.....

function getPassword() {
    var chars = "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZKLOPWXYZ";
    var passwordLength = 14;
    var password = "";

    for (var i = 0; i < passwordLength; i++) {
        var randomNumber = Math.floor(Math.random() * chars.length);
        password += chars.substring(randomNumber, randomNumber + 1);
    }
    document.getElementById("password").value = password;
}

// Preloader....

var myVar;

function myLoader() {
    myVar = setTimeout(showPage, 3000);
}

function showPage() {
    document.getElementById("loader").style.display = "none";
    document.getElementById("myDiv").style.display = "block";
    document.getElementById("f").style.display = "block";
    var music = new Audio();
    music.src = "sound/theme.mp3";
    music.play(); // Music
}

// AUTO TYPE.....

var TxtType = function (el, toRotate, period) {
    this.toRotate = toRotate;
    this.el = el;
    this.loopNum = 0;

```



```

    this.period = parseInt(period, 10) || 2000;
    this.txt = "";
    this.tick();
    this.isDeleting = false;
};

TxtType.prototype.tick = function () {
    var i = this.loopNum % this.toRotate.length;
    var fullTxt = this.toRotate[i];

    if (this.isDeleting) {
        this.txt = fullTxt.substring(0, this.txt.length - 1);
    } else {
        this.txt = fullTxt.substring(0, this.txt.length + 1);
    }

    this.el.innerHTML = '<span class="wrap">' + this.txt + "</span>";

    var that = this;
    var delta = 200 - Math.random() * 100;

    if (this.isDeleting) {
        delta /= 2;
    }

    if (!this.isDeleting && this.txt === fullTxt) {
        delta = this.period;
        this.isDeleting = true;
    } else if (this.isDeleting && this.txt === "") {
        this.isDeleting = false;
        this.loopNum++;
        delta = 500;
    }

    setTimeout(function () {
        that.tick();
    }, delta);
};

```

```
window.onpageshow = function () {  
    var elements = document.getElementsByClassName("typewrite");  
    for (var i = 0; i < elements.length; i++) {  
        var toRotate = elements[i].getAttribute("data-type");  
        var period = elements[i].getAttribute("data-period");  
        if (toRotate) {  
            new TxtType(elements[i], JSON.parse(toRotate), period);  
        }  
    }  
    // INJECT CSS  
    var css = document.createElement("style");  
    css.type = "text/css";  
    css.innerHTML = ".typewrite > .wrap { border-right: 0.10em solid skyblue}";  
    document.body.appendChild(css);  
};
```

CHAPTER 12

SUMMARY AND CONCLUSION

We were able to create Sentimeter, a user-friendly interface where people with minimal to no understanding can view and review our application. This has primarily been made to target a mentally challenged audience and assist them in their interaction skills that they are bound to face. In this project report, we went over the 3 modules we developed, image emotion, webcam emotion, and webcam emotion to find accuracy.

- We would like to clean up the interface more and make it look more radiant and professional.
- We would like to add emojis for the respective emotions detected.
- We would also like to create a mobile application for our project since it's more widespread and our application can be used by many more.

CHAPTER 13

REFERENCE

1. Vamshi Krishna Gudipati, Oindrila Ray Barman, Mofica Gafford, Harshavardhan, Abdelshakour Albuneid, Efficient Facial Expression Recognition Using Adaboost and Haar Cascade Classifiers, 2016 IEEE.
2. Namrata Mahajan, Harshad Mahajan, Emotion Detection Algorithm, International Journal of Electrical and Electronics Research, Vol. 2, Issue 2, pp: (56–60), Month: April - June 2014
3. Emotion Recognition from Facial Expressions using Multilevel HMM Ira Cohen, Ashutosh Garg, Thomas S. Huang, Beckman Institute for Advanced Science and Technology.
4. Emotion recognition using facial expressions with active appearance models Matthew S. Ratliff, Eric Paterson
5. Ahmet zdil Metin Mete zbilen A Survey on Comparison of Face Recognition Algorithms IEEE 2015
6. G. Ramkumar, M. Manikandan - Face Recognition Survey International Journal of Advances in Science and Technology (IJAST) 2014 ISSN 2348-5426
7. Sharks, M. Abou Elene, Eigenfaces vs. Fisher faces vs. ICA for Face Recognition; A Comparative Study, 9th International Conference on Signal Processing, 2008, ICSP 2008., 2008, pp. 914919
8. K. Kim, Intelligent Immigration Control System by Using Passport Recognition and Face Verification, in International Symposium on Neural Networks Chongqing, China, 2005, pp.147-156.
9. J. N. K. Liu, M. Wang, and B. Feng, iBotGuard: an Internet based intelligent robot security system using invariant face recognition against intruder, IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews, Vol.35, pp.97-105, 2005.

10. H. Moon, Biometrics Person Authentication Using Projection Based Face Recognition System in Verification Scenario, in International Conference on Bioinformatics and its Applications. Hong Kong, China, 2004, pp.207-213.
11. D. McCullagh, Call It Super Bowl Face Scan 1, in Wired Magazine, 2001.