Mini Project Report

on

"Emotion Detection using Artificial Intelligence"

Submitted for the partial fulfillment of Bachelor of Engineering

by

1. **Abhinav Kumar (1SI17CS001)**
2. **Athmiya HB (1SI17CS017)**
3. **Ayesha Sulthana (1SI17CS019)**

Under the guidance of

Ms. Bhargavi K BE, MTech
Assistant Professor

Department of Computer Science and Engineering

Siddaganga Institute of Technology, Tumakuru – 572103
(An Autonomous Institution, Affiliated to VTU, Belagavi & Recognized by AICTE, New Delhi)

2019 -2020

SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-3

(An Autonomous Institution, Affiliated to VTU, Belagavi & Recognized by AICTE, New Delhi)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that that the mini project entitled **"Emotion Detection Using Artificial Intelligence"** is a bonafide work carried out by **Abhinav Kumar (1SI17CS001), Athmiya HB (1SI17CS017), Ayesha Sulthana (1SI17CS019)** of VI semester **Computer Science and Engineering**, **SIDDAGANGA INSTITUTE OF TECHNOLOGY** for the partial fulfillment of Bachelor of Engineering during the academic year 2019-2020.

Signature of the Guide
Ms. Bhargavi K
Assistant Professor

Signature of the Convener
Bhaskar G
Assistant Professor

Signature of the HOD
Dr. R Sumathi
Prof. and Head, Dept. of CSE

Name of the Examiners: **Signature with Date**

1.

2.

ACKNOWLEDGEMENT

We consider this a privilege to express a few words of gratitude to all those who guided and inspired us for the successful completion of our project work.

With respectful prayers to His Holiness **Dr. Sree Sree Sree Shivakumara Swamigalu**, *President, Sree Siddaganga Education Society*, we express our gratitude at his lotus feet for being a constant source of inspiration in the course of study.

With gratitude we acknowledge our beloved Director **DR. M N Channabasappa***M.A, Ph.D.,* and *Principal* **Dr.K.P.Shivananda***M.E., Ph.D.*, for fostering an excellent academic environment in this institution and also providing excellent lab facilities, which made our endeavor possible.

We express our deep sense of gratitude to **Dr. R. Sumathi M.E**., *Ph.D.*, *Professor and Head, Department of Computer Science and engineering*, who has always been a remarkable source of inspiration whose unflagging zeal has catalyzed us to strive for greater heights during the span of course in this Institution.

We thank our project guide **Ms. Bhargavi K** *BE, M.Tech,*,*Assistant Professor, Department of Computer Science and Engineering* for providing the critical support on every step of this project development. Her guidance gave us the environment to enhance our knowledge, skills and to reach the pinnacle with sheer determination, dedication and hard work. Without her aspiring support this project would not have been implemented.

We thank our project coordinators for giving us a platform to showcase the different skills that we have in order to complete the mini project.

Finally, we would like to thank all the faculty and non-teaching staff members of *Dept. of Computer Science and Engineering* and our parents and friends, without whose help and constant encouragement, it would have been difficult for us to complete this project.

ABSTRACT

Artificial Intelligence can detect emotion just the way humans do, from multiple channels. This project proposes an implementation of a general convolutional neural network (CNN) building framework for designing CNNs which accomplishes the task of emotion detection. After presenting the details of the training procedure setup it proceeds to evaluate on standard benchmark datasets. Guided backpropagation uncovers the dynamic of the weight changes and evaluates the learning features.

Most of the human communication is done through speech, hand gestures and facial expressions. Interpreting correctly any of these elements using machine learning (ML) techniques has proven to be complicated due to the high variability of the samples within each task. This leads to models with millions of parameters trained on thousands of samples. These tasks require CNN architectures with millions of parameters. We are trying to manually classify the dataset images within the following classes {“angry”, “disgust”, “fear”, “happy”, “sad”, “surprise”, “neutral”}.

The final native web application which will take the input image of the user, will process it and predict the emotion of the person. According the emotion detected it will suggest the activity that need to be performed e.g. if the person is sad it will play some pinky- ponky type of music or can suggest some posts in order to change the mood of the person.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Emotions are slaves to our thoughts and we are slaves to our emotions. Facial expressions convey non-verbal cues, which play an important role in interpersonal relations, automatic recognition of facial expressions can be an important component of natural human-machine interfaces. It may also be used in behavioral science and in clinical practice.

Facial expressions are the visible manifestation of the affective state, activity, personality and intention of a person. Human facial expressions can be easily classified into 7 basic emotions: happy, sad, surprise, fear, anger, disgust, and neutral. Our facial emotions are expressed through activation of specific sets of facial muscles. These sometimes subtle, yet complex, signals in an expression often contain an abundant amount of information about our state of mind.

On a day to day basis human commonly recognize emotions by characteristic features, displayed as a part of a facial expression. For instance happiness is associated with a smile on the face. Similarly, all other emotions are detected and identified through different facial expressions.Automatic recognition of facial expressions can be an important component of natural human machine interfaces; it may also be used in behavioral science and in clinical practice. It have been studied for a long period of time and obtaining the progress recent decades. Though much progress has been made, recognizing facial expression with a high accuracy remains to be difficult due to the complexity and varieties of facial expressions.[1]

But this project is developed and trained in such a way that 97 % of accuracy is obtained. The project is trained by kaggle dataset. The data consists of 48x48 pixcel greyscale images of faces. The faces have been automatically registered in such a way that face is centered and occupies same amount of space in each image. The dataset trains the model in such a way that it categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral)

The model is trained with training set consisting of 28704 examples and the test set consists of 3,589 examples. This dataset was prepared by Pierre-Luc Carrier and Aaron Courville, as

part of an ongoing research project.[2]

In machine learning, a convolutional neural network (CNN) is a type of feedforward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex.
They have wide applications in image and video recognition, recommender systems and natural language processing. The convolutional neural network is also known as shift invariant or space invariant artificial neural network (SIANN), which is named based on its shared weights architecture and translation invariance characteristics.

1.1 Motivation

According to global surveythat design to access the feelings of people around the world, people are feeling more angry, sad and worried than ever before. The poll is called the Global State of Emotions report and it was compiled by analytics firm Gallup. After 1,51,000 interviews with adults in over 140 countries during 2018, the outlook seems pretty bleak. Above research has noted that negative emotions can have a strong link to a person's health.

Links have been found between anger and an elevated risk hearts and stress and chronic worry and sadness can lead to anxiety disorders and depression. Nowadays most of the people are subjected to major depression, or major depressive disorder. It's a diagnosable condition that's classified as a mood disorder and can bring about long-lasting symptoms such as overwhelming sadness, low energy, loss of appetite, and a lack of interest in things that used to bring pleasure. Left untreated, depression can lead to serious health complications, including putting your life at risk.

1.2 Aim of the Application

Nowadays as many individual are facing depression, anxiety, headache, stress and lot of other feelings which are suppressing their life development. Many youngsters are facing this problem. As a result of this they are facing backlogs in their regular academic subjects and they can't concentrate on the personal development. Due to this reason, death rates are increasing which is not a tolerable news. Many a times, the individuals search history is filled only with the topics like related to suicide. To avoid this and in order to guide the individuals in right way and to properly motivate them this project is very helpful.

CHAPTER 2

LITERATURE SURVEY

Web application development is the process and practice of developing web applications. There is a consensus that the process involved are standard software engineering process.[3]

2.1 Development process of Web Application :

The web application development practice is focused on optimum user experience. The development process of web application revolve around the four major phases. The following figure shows the four major phases of application development.

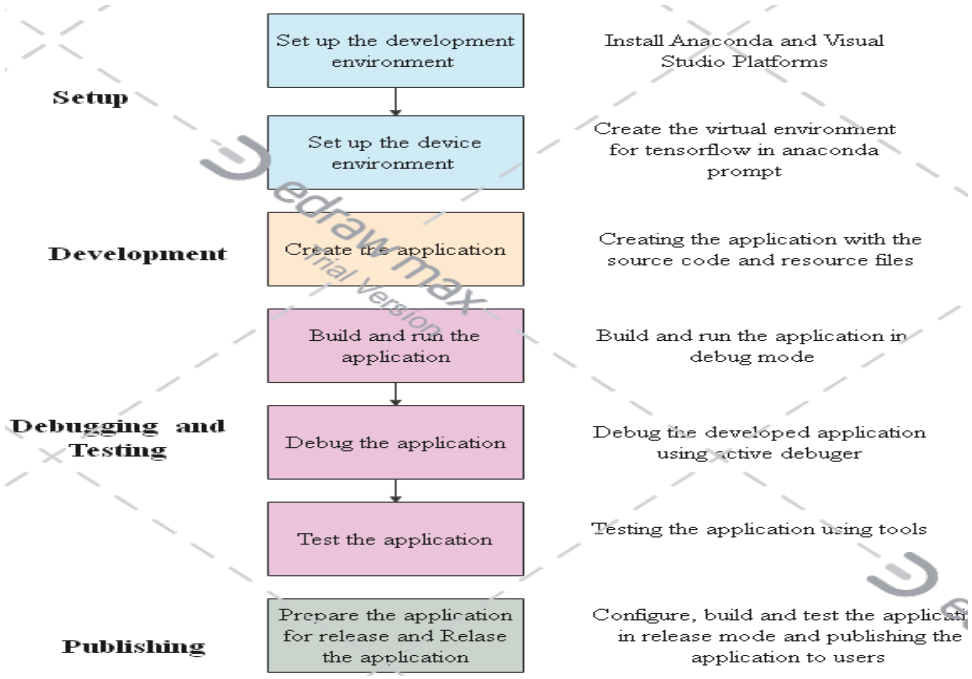


Figure 2.1 Development Process of Web Application

The development phases are as follows:

i. The Development Environment Setup Phase:

In this stage the Anaconda is downloaded, installed and set up for development. The Visual Studio IDE (Integrated Development Environment) is also installed for the development of the application. The virtual environment for tensorflow is created in order to run the developing code in the environment.

ii. Project Setup and Development Phase:

This is the stage where the actual development of the project is done. The application project folder contains all the source code and resource files for the application.

iii. Building, Debugging and Testing:

This is the stage where the developed application code can be built and debug in debug mode.

Then the testing process is used to validate the developed web application so that it can meet specific functionality, security, usability, accessibility, visual and performance criteria.

iv. Publishing:

During this stage, the application is prepared for release by configuring, building and testing it in the release mode. Thereafter, the application is published, sold and distributed to users.

2.2 Background:

There are two different approaches used for facial expression recognition, both of which include two different methodologies. Dividing the face into separate action units or keeping it as a whole for further processing appears to be the first and the primary distinction between the main approaches. In both of these approaches, two different methodologies, namely the 'Geometric based' and the 'Appearance-based' parameterizations, can be used. The image of the face can be taken and processed that can be classified into 7 facial expression prototypes: angry, disgust, fear, happy, sad, neutral . Here, it is assumed that each of the above mentioned emotions have characteristic expressions on face and that's why recognition of them is necessary and sufficient.

In order to obtain the proper output second approach is used. i.e., Appearance-based. The various images of 7 emotions are fed and trained the model. Only face is cropped from the input image. Each pixel of this image is taken and the result is stored in the form of an array. Then by processing the value in the array, the image is classified. The output graph of percentage of emotions is displayed.

Rather than tracking spatial points and using positioning and movement parameters that vary within time, color (pixel) information of related regions of face are processed in Appearance Based Parameterizations. For classification problem, algorithms like Machine learning, Neural Network, Support Vector Machine, Deep learning, Naive Bayes are used.

The emotion detection system is implemented using convolutional neural network. The

following figures[1] shows the block diagram of the system.

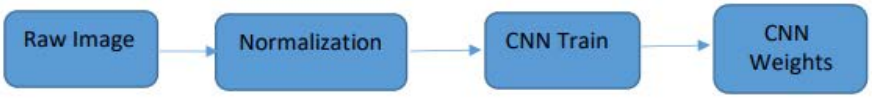


Figure 2.2(a). Training phase

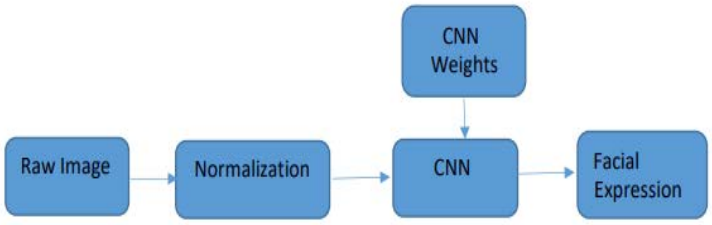


Figure 2.2(b). Testing phase

The model is trained by inputting a training data comprising images of faces with their respective emotion label and the model learns a set of weights for the network. The input for the training phase is the image of face. Normalization is applied on the input image. These normalized images are used to train the network. Validation dataset is used to select the final best set of weights to ensure training performance is not affected. The output of the training is a set of weights that achieve the best result with the training data.

The model is tested by inputting a image of face from test dataset and the output is predicted emotion by using the final network weights.

The number of epochs used to develop the model is 25. An epoch is a hyperparameter which is defined before training a model. One epoch is too big to feed at once. So it is divided into several smaller batches and enhances the performance of the model. By increasing the number of epochs, the model is trained in such a way that application project predicts the output with high accuracy.

CHAPTER 3

REQUIREMENT SPECIFICATION

In Software engineering, application developers need to clearly understand the problems to be solved. It is therefore important for a developer to properly model the scenarios that can influence the solution to the problem by collecting relevant information. This process is called requirement analysis. The requirement analysis focuses on the tasks that determine the needs or conditions to meet the project or product. And provides the opportunity for a developer to get a better understanding of the problem statement [4]. For effective design and development of this project, the following requirements must be met. They can be divided into functional requirements and non-functional requirement.

3.1 Functional Requirement :

Here are the description of the different requirements that accomplished the fulfillment of the 'Emotion Detection using Artificial Intelligence' project. In order to achieve the desired goal of this project, the functional requirement must be met. The following are the major actions performed by the project.

Input Requirements:

Input requirements are the requirements that a user must fulfill before gaining access to use this web application. First, the users have to register in the website. The registered users can login. The login credentials are saved. Next, the users are allowed to take their picture from the webcam which opens in the website. The users have to save their image. User's account is displayed. There, the user can update their profile picture. The users can update their profile picture by the image which they took from the webcam of the website. Then the graph of the detected emotion is shown. Based on the graph shown, the users are redirected to the recommended page. There the users find some videos and music that can make their mood. The contents are displayed through user interface. After watching the recommended videos user can logout from the website.

3.2 Non-Functional Requirements:

Non-functional requirements are those requirements that do not affect the proper running of this web application. However it is important to consider these requirements for the better analysis and the quality of the software.

Software Requirements:

The project needs well defined specifications to meet the software environment needs to fulfill the aim of the project. The software requirements that are considered for the development of this project are mentioned below:

- This web application is a software program that runs on web server. Users can access this web application through a web browser with an active internet connection.
- Visual Studio, it is the Integrated Development Environment used in developing this project using flask framework and sqLite database.
- Python 3.7, Opencv, Tkinter, PIL, Numpy, Keras, Image classification Technique: Tensorflow.

Hardware Requirements:

The hardware requirements and specifications are considered for the development of this web application for proper functioning. The application is developed to run in web server. With the help of the web browser, this application can utilize the device hardware such as web camera, sensors.

CHAPTER 4

DESIGN AND IMPLEMENTATION

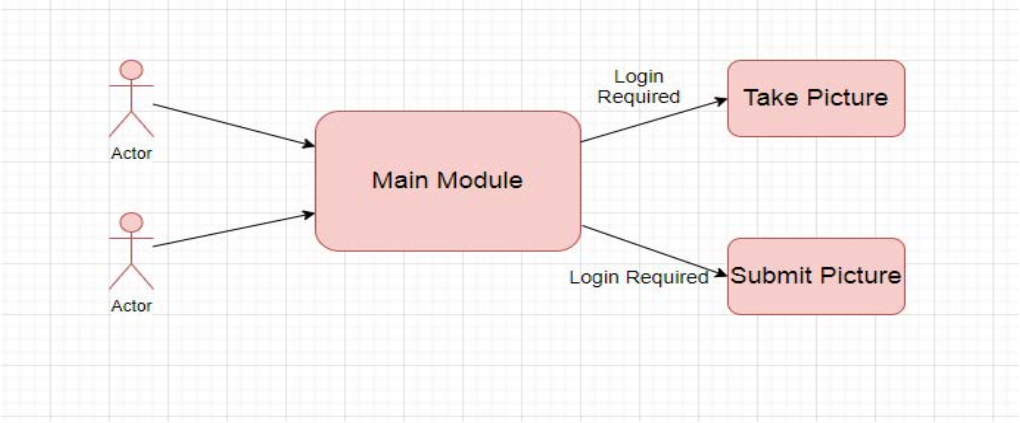
The application follows a modular design. The application is followed in such a way that, it is helpful for the end users. The users need to login with the credentials and all they need to do is to take their picture from the web cam that opens on this web application. The graph of the current emotion is displayed on the website. And recommended videos are displayed for the users on their current emotion.

4.1 Implementation of Modules :

4.1.1Home Page (Main Module)

The screen that the users can see first when they open the website is considered as the main module. This web page acts as a guide to the users. The options such as to take photo from the web cam and to submit the photo is displayed. These web pages can be accessed only by users who are logged in. If new users, first they have to register and then log in to their account.

Use Case:



. Figure.4.1.1(a) Use Case diagram of the main module

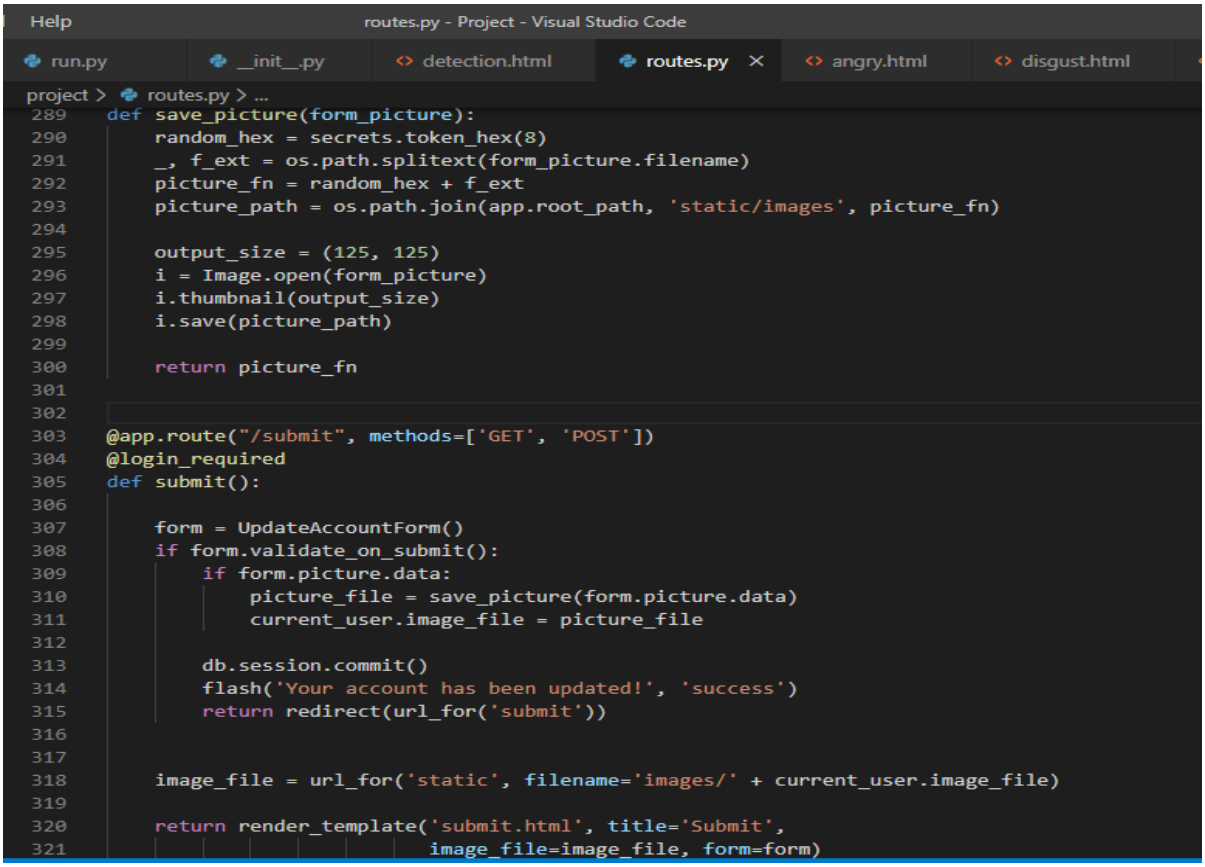


Figure.4.1.1(b) Main Module

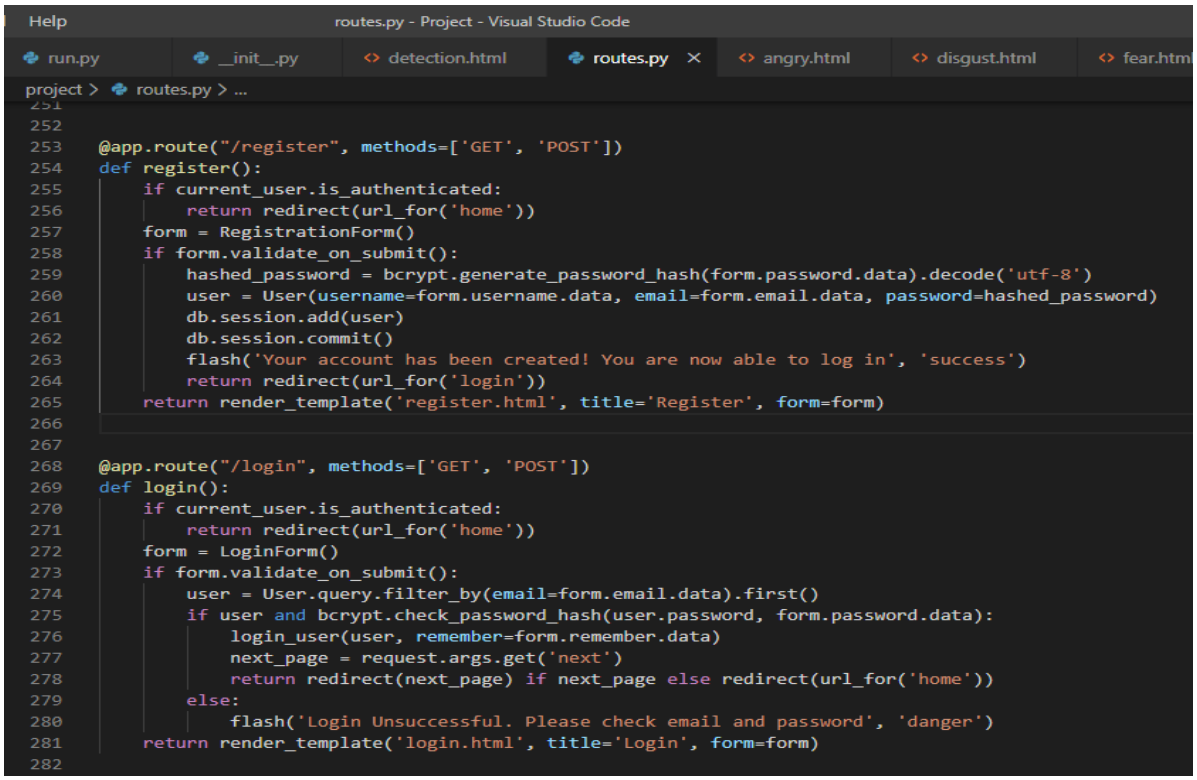


Figure.4.1.1(c) Register and Login

4.1.2 Take Picture

This makes the users to access the web camera from the website. Then the users can take their picture from pressing the button 'Take Picture'. Then the users can save the image which they took now.

Then the users are redirected back to the home page (Main module).

4.1.3 Submit Picture

This makes the users to see the page where their account is displayed. Account is displayed with the users name, email and the default account picture. Users have to update this profile picture by choosing a file. Users have to update it with the same picture which they took on the web cam of the website. Then users account get updated. Then the emotion of the users is detected by cropping only the face of the image. By clicking the 'Detected emotion' route, users can see their emotion detected graph. Then users are suggested with some recommendations and they are redirected to the youtube videos.

Use Case:

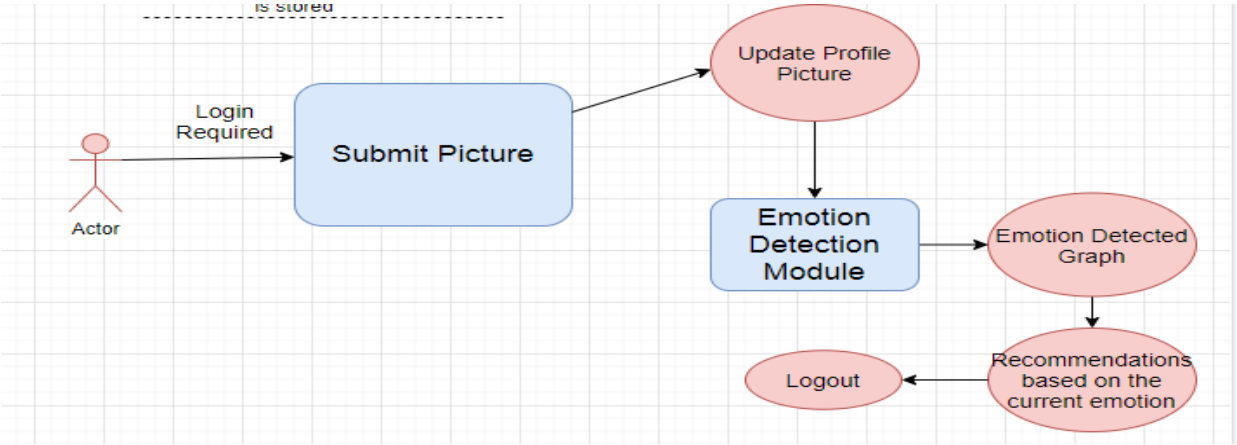


Figure.4.1.3(a) Use Case diagram of the Submit Picture section

Dataflow diagram:

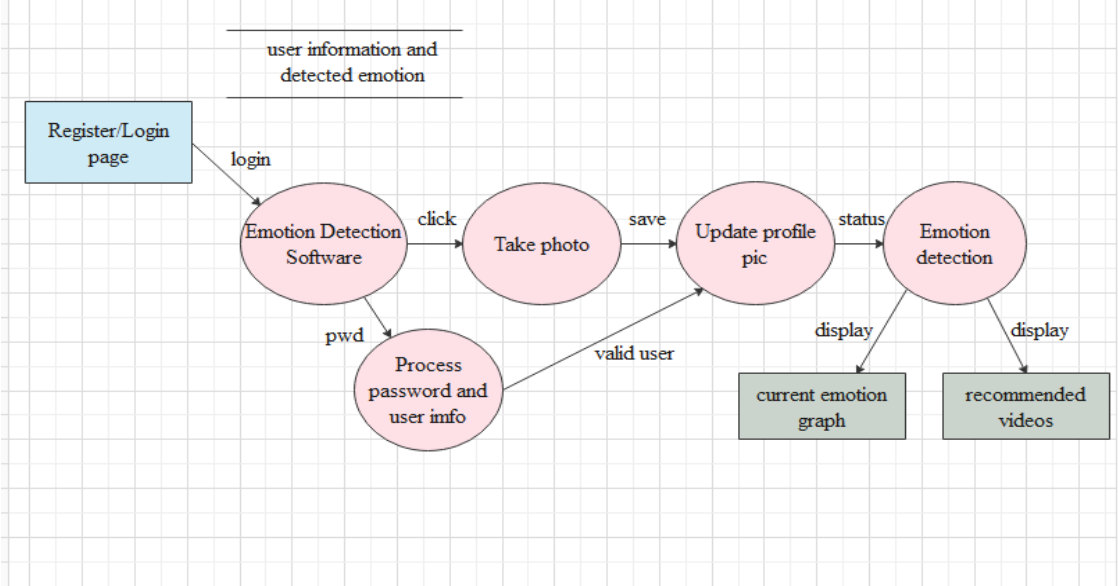


Figure 4.1.3(b) Dataflow Diagram of the project

```
@app.route("/detection", methods=['GET', 'POST'])
@login_required
def detection():

    from keras.preprocessing import image
    from keras.preprocessing.image import ImageDataGenerator

    import numpy as np
    import matplotlib.pyplot as plt



    image_file = url_for('static', filename='images/' + current_user.image_file)


    file = 'project/' + image_file
    true_image = image.load_img(file)
    img = image.load_img(file, grayscale=True, target_size=(48, 48))

    x = image.img_to_array(img)
    x = np.expand_dims(x, axis = 0)

    x /= 255

    custom = model.predict(x)
    emotion_analysis(custom[0])

    x = np.array(x, 'float32')
    x = x.reshape([48, 48]);

    plt.gray()
```

Figure.4.1.3(c) Emotion Detection code

```
357     plt.imshow(true_image)
358     plt.show()
359
360     u = current_user.username
361     f_ext =  'png'
362     data_file = url_for('static', filename='data/' + (u+'.'+f_ext))
363     emotion = Emotion(title=data_file, visitors=current_user)
364     db.session.add(emotion)
365     db.session.commit()
366
367
368     return render_template('detection.html', title='Detection',
369                            image_file=image_file)
370
371
```

Figure.4.1.3(d) Emotion Detection Code

Help routes.py - Project - Visual Studio Code

routes.py | angry.html | disgust.html | fear.html | happy.html | neutral.html

project > routes.py > ...

```
73  #initialize trainset and test set
74  '''
75  x_train, y_train, x_test, y_test = [], [], [], []
76
77  #------------------------------
78  #transfer train and test set data
79  for i in range(1,num_of_instances):
80      try:
81          emotion, img, usage = lines[i].split(",")
82
83          val = img.split(" ")
84
85          pixels = np.array(val, 'float32')
86
87          emotion = keras.utils.to_categorical(emotion, num_classes)
88
89          if 'Training' in usage:
90              y_train.append(emotion)
91              x_train.append(pixels)
92          elif 'PublicTest' in usage:
93              y_test.append(emotion)
94              x_test.append(pixels)
95      except:
96          print("", end="")
97
98  #------------------------------
99  #data transformation for train and test sets
100 x_train = np.array(x_train, 'float32')
101 y_train = np.array(y_train, 'float32')
102 x_test = np.array(x_test, 'float32')
103 y_test = np.array(y_test, 'float32')
104
```

Figure.4.1.3(e) Training the model

(Code is commented because once the model is trained, need not run this again)

project > routes.py > ...

```
133   model.add(Flatten())
134
135   #fully connected neural networks
136   model.add(Dense(1024, activation='relu'))
137   model.add(Dropout(0.2))
138   model.add(Dense(1024, activation='relu'))
139   model.add(Dropout(0.2))
140
141   model.add(Dense(num_classes, activation='softmax'))
142   #------------------------------
143   #batch process
144   gen = ImageDataGenerator()
145   train_generator = gen.flow(x_train, y_train, batch_size=batch_size)
146
147   #------------------------------
148
149   model.compile(loss='categorical_crossentropy'
150       , optimizer=tf.keras.optimizers.Adam()
151       , metrics=['accuracy']
152   )
153
154   #------------------------------
155
156   fit = True
157
158   if fit == True:
159       #model.fit_generator(x_train, y_train, epochs=epochs) #train for all trainset
160       model.fit_generator(train_generator, steps_per_epoch=batch_size, epochs=epochs) #train for randomly sel
161   else:
162       model.load_weights('/data/facial_expression_model_weights.h5') #load weights
163
164   #------------------------------
```

Figure.4.1.3(f) Training the model

(Code is commented because once the model is trained, need not run this again)

Help — models.py - Project - Visual Studio Code

<> disgust.html | <> fear.html | <> happy.html | <> neutral.html | <> surprise.html | <> sad.html

project > models.py > User

```
1   from project import db, login_manager
2   from flask_login import UserMixin
3
4
5   @login_manager.user_loader
6   def load_user(user_id):
7       return User.query.get(int(user_id))
8
9
10  class User(db.Model, UserMixin):
11      id = db.Column(db.Integer, primary_key=True)
12      username = db.Column(db.String(20), unique=True, nullable=False)
13      email = db.Column(db.String(120), unique=True, nullable=False)
14      image_file = db.Column(db.String(20), nullable=False, default='default.png')
15      password = db.Column(db.String(60), nullable=False)
16      emo = db.relationship('Emotion', backref='visitors', lazy=True)
17
18      def __repr__(self):
19          return f"User('{self.username}', '{self.email}', '{self.image_file}')"
20
21  class Emotion(db.Model):
22      id = db.Column(db.Integer, primary_key=True)
23      title = db.Column(db.String(100), nullable=False)
24      user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)
25
26      def __repr__(self):
27          return f"Emotion('{self.title}')"
28
```

Figure4.2.3(g). Database creation in the project

CHAPTER 5

RESULT OF THE WEB APPLICATION

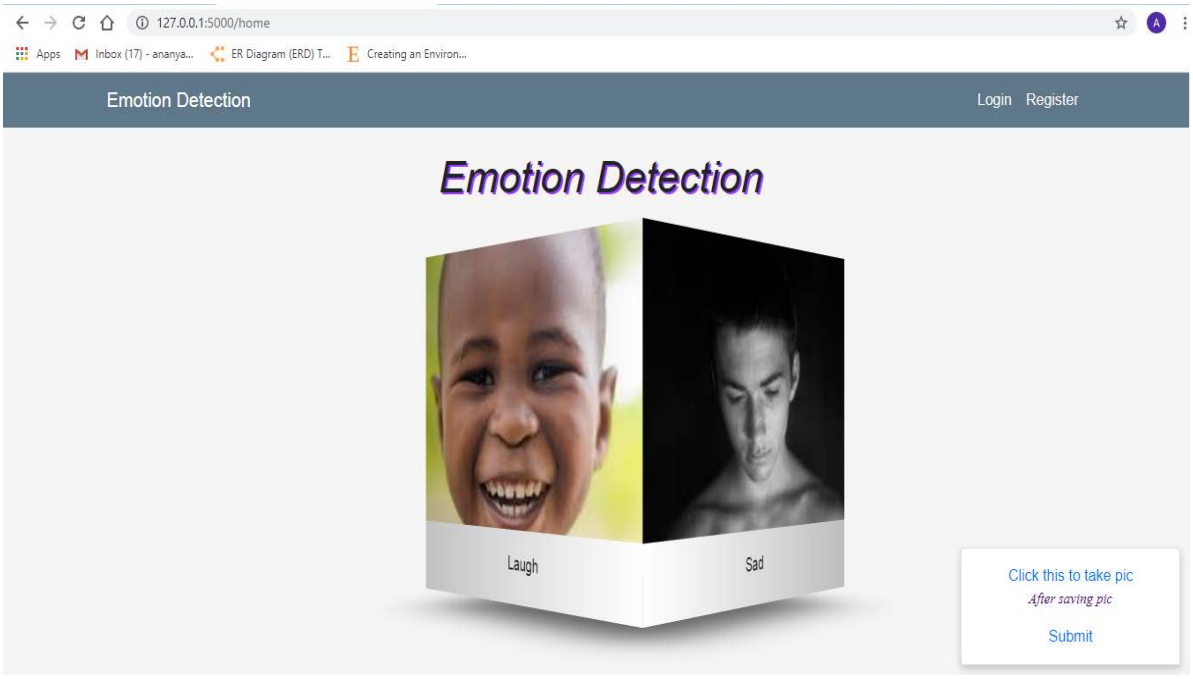


Figure.5.1 Home Page

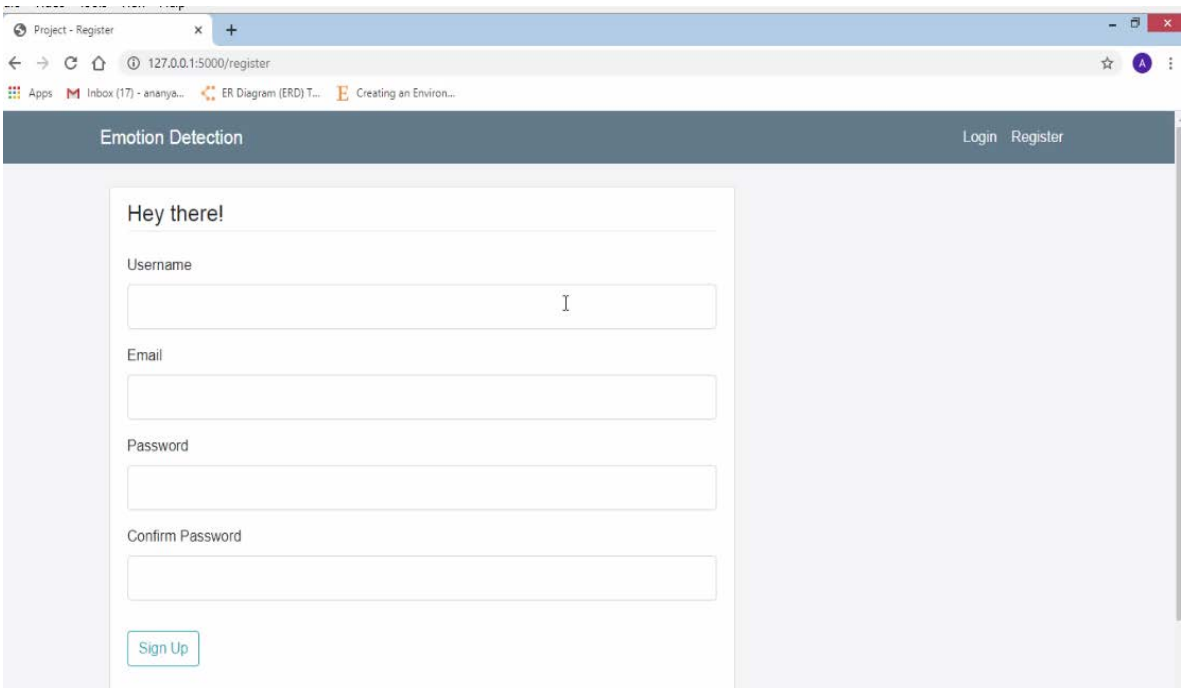


Figure.5.2 Registration Page

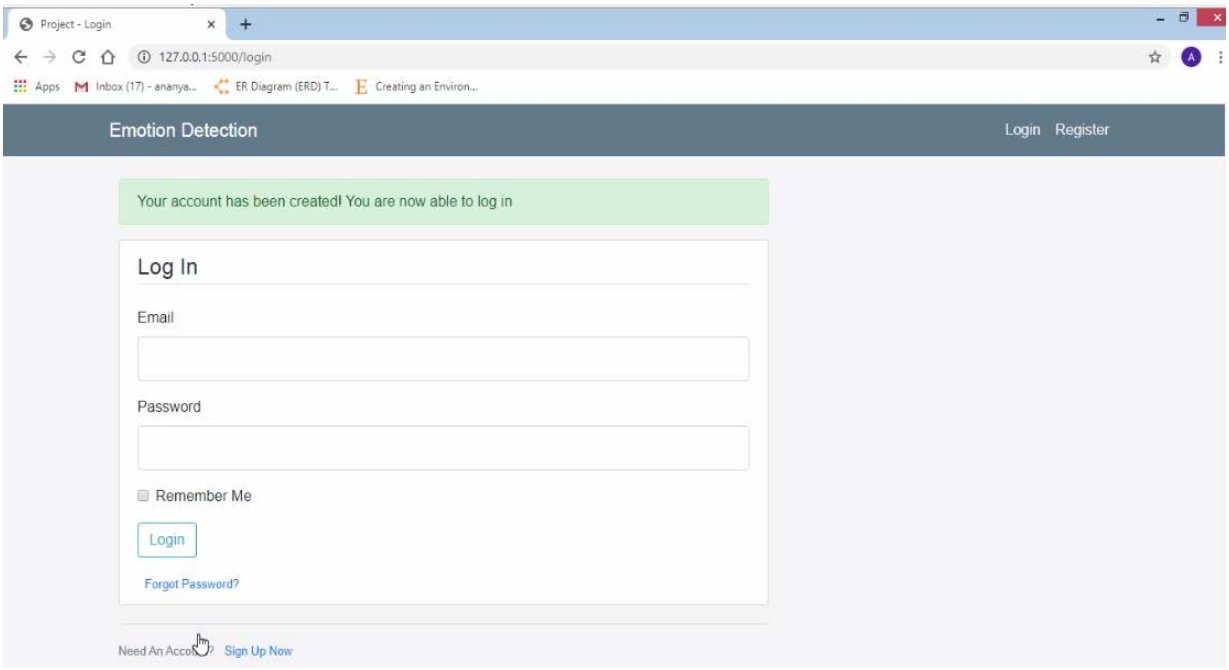


Figure.5.3 Login Page

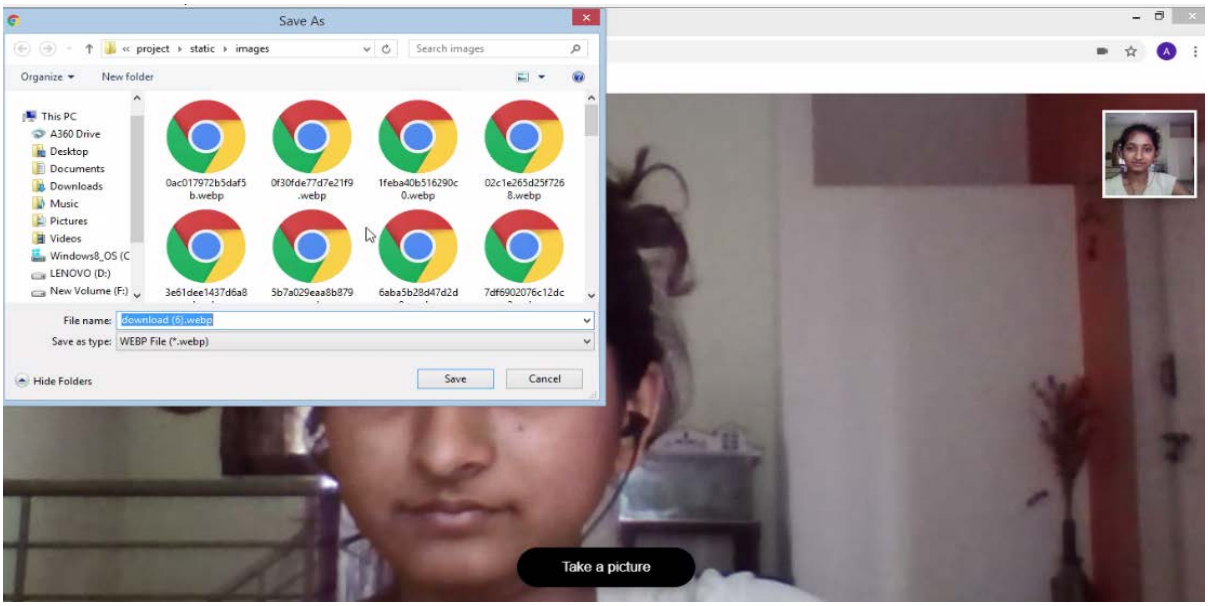


Figure.5.4 Saving the image which is clicked from the web cam of the website

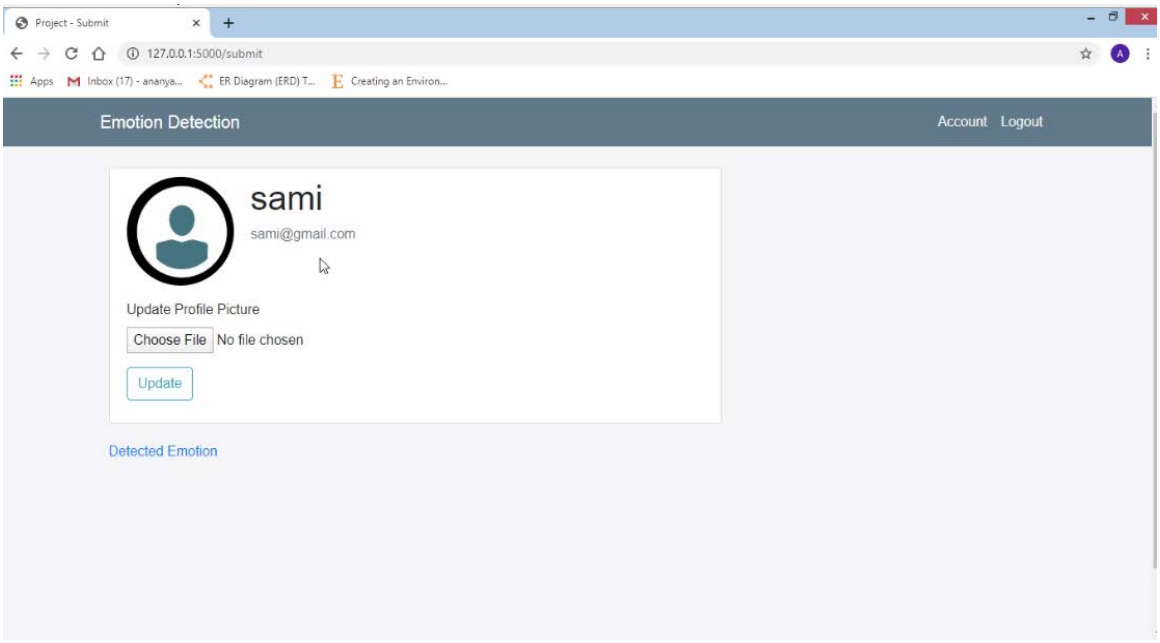


Figure.5.5 User Account

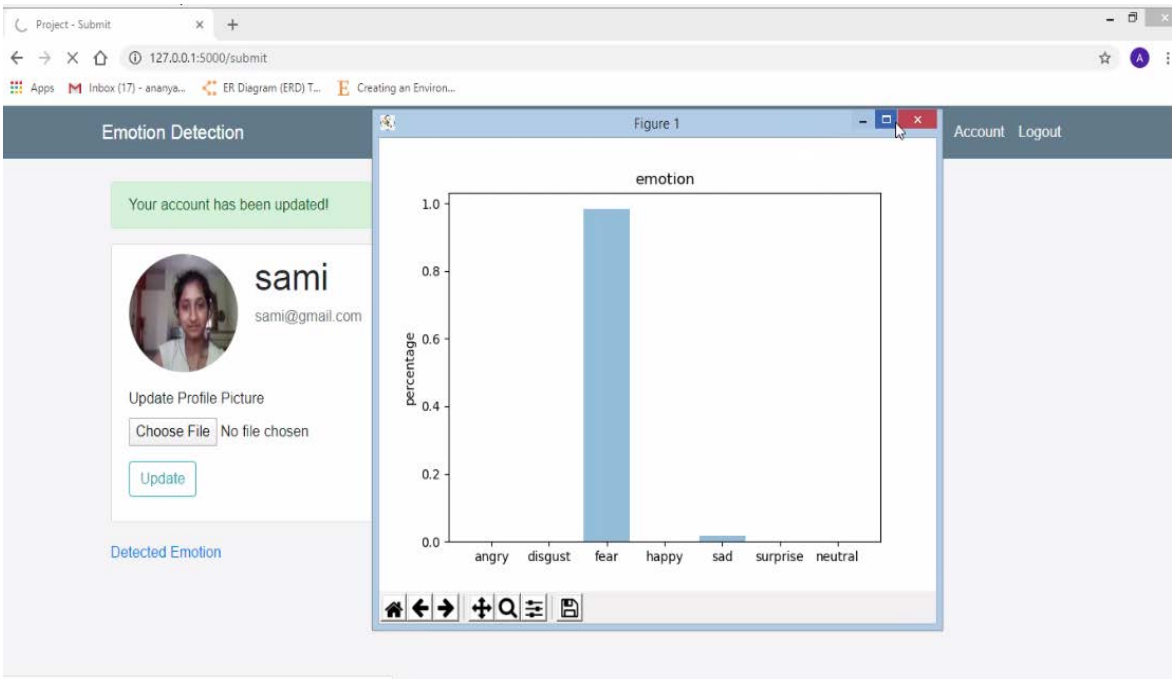


Figure.5.6 Emotion Detected Graph of the user for the updated profile picture

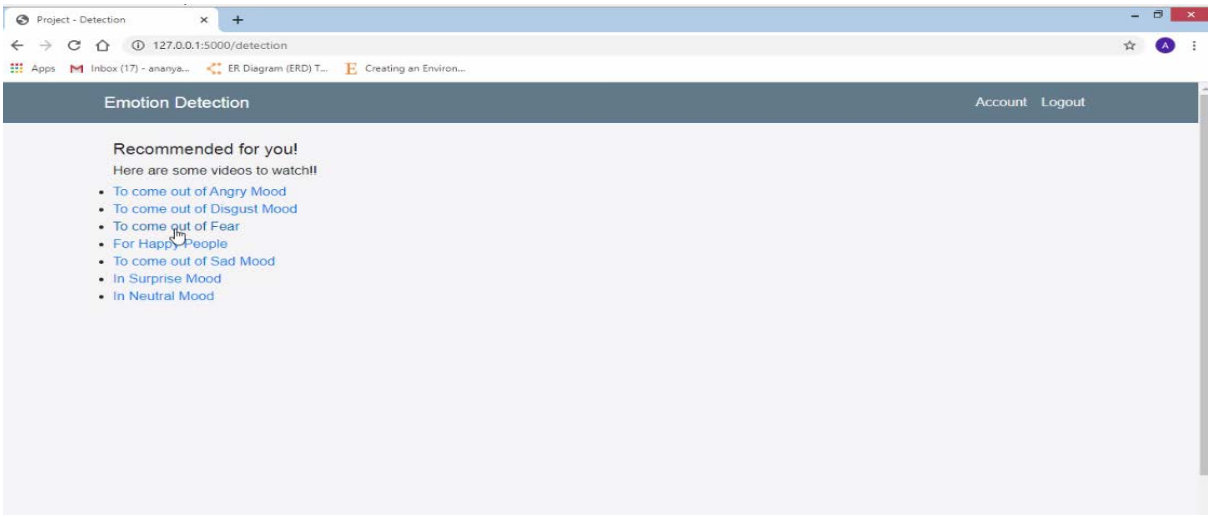


Figure.5.7 Recommended

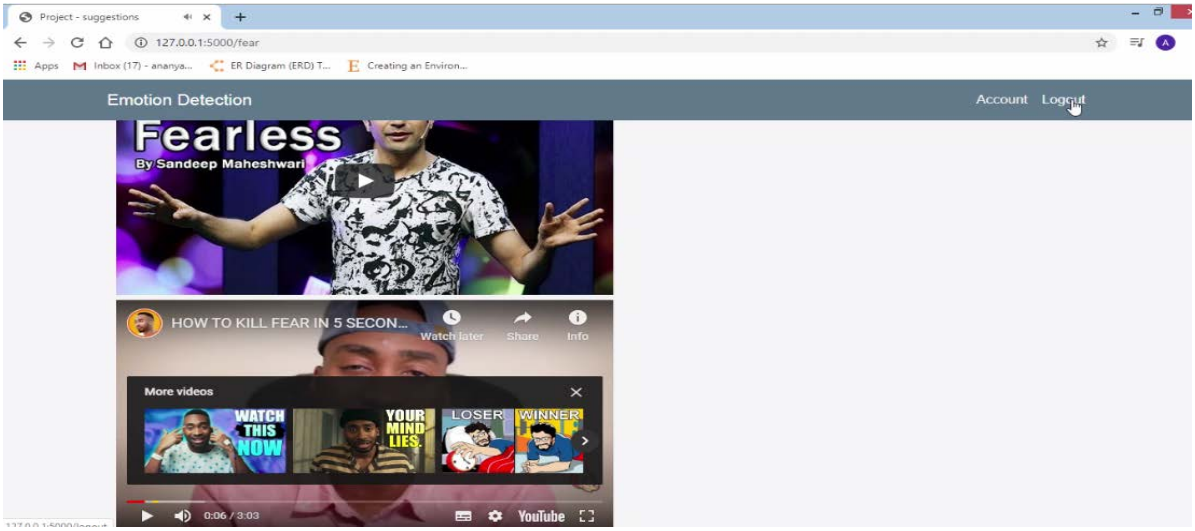


Figure.5.8 Displayed videos

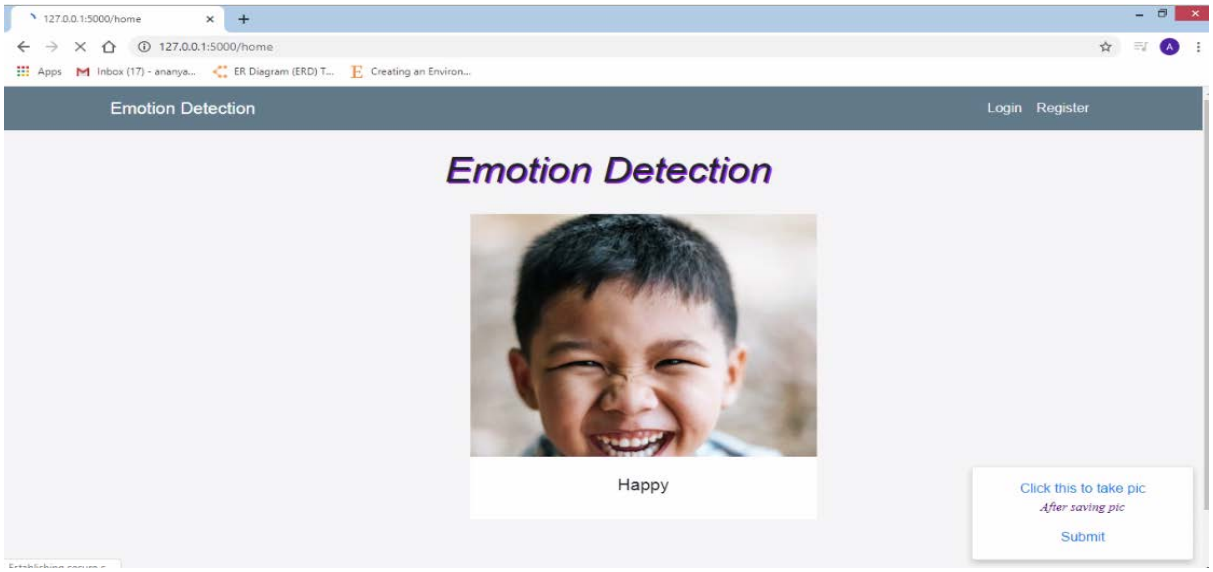


Figure.5.9 Redirecting the users back to home page after logout

CHAPTER 6

CONCLUSION

Thus, the aim of the web application is to detect the emotion of the user and suggest the activity that needs to be performed for the user. The aim of the project is to motivate and encourage the users in the right way. This helps people to come out of depression, stress, anxiety and other such things. In the year 2017, the emotion detected model was developed with 56.77 % only for black and white images. But this web application has come up with 90 % accuracy for colored images.

CHAPTER 7

REFERENCES

1. https://www.researchgate.net/publication/322103128_Facial_Expression_Recognition System_using_Convolutional_Neural_Network
2. https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge
3. https://en.wikipedia.org/wiki/Web_application
4. https://en.wikipedia.org/wiki/Requirements_analysis