B.E. Project Report *on*

DEPRESSION DETECTION USING MACHINE LEARNING

By

Gauri Duddalwar(B9114)

Pallavi Kolhe(B9139)

Balaji Kokate(B9230)

Sanket Yeole(B9234)

Under the guidance of

Prof.S.M.Kamble

Department of Information Technology Smt. Kashibai Navale College of Engineering, Pune-41

Accredited by NBA

SAVITRIBAI PHULE PUNE UNIVERSITY

2019-2020

Sinhgad Technical Education Society,
Department of Information Technology

Smt. Kashibai Navale College of Engineering , Pune 411041

Date:

CERTIFICATE

This is to certify that,

Gauri Duddalwar(B9114)

Pallavi Kolhe(B9139)

Balaji Kokate(B9230)

Sanket Yeole(B9234)

of class B.E IT; have successfully completed their project work on **"DEPRESSION DETECTION USING MACHINE LEARNING"** at Smt. Kashibai Navale College of Engineering, Pune in the partial fulfillment of the Graduate Degree course in B.E at the Department of **Information Technology**, in the academic Year 2019-2020 as prescribed by the Savitribai Phule Pune University, Pune.

Prof.S.M.Kamble — Project Guide

Prof. R.H.Borhade — Head of the Department, Department of Information Technology

(External Examiner) — Savitribai Phule Pune University , Pune

Principal — Smt.Kashibai Navale College of Engineering,Pune

ACKNOWLEDGEMENT

We are very thankful to all the teachers who have provided us valuable guidance towards the completion of this on **"DEPRESSION DETECTION SYSTEM USING MACHINE LEARNING''** as part of the syllabus of bachelor's course. We express our sincere gratitude towards cooperative department who has provided us with valuable assistance and requirements for the system development. We are very grateful and want to express our thanks to **Prof. S.M.Kamble** for guiding us in right manner, correcting our doubts by giving us their time whenever we required, and providing their knowledge and experience in making this project. We are also thankful to the HOD of our Information Technology department **Prof. R. H. Borhade** for his moral support and motivation which has encouraged us in making this project. The acknowledgement will be incomplete if we do not thank our Principal **Prof. Dr.A.V.Deshpande** who gave us his constant support and motivation which has been highly instrumental in making our project.

.

Gauri Duddalwar(B9114)

Pallavi Kolhe(B9139)

Balaji Kokate(B9230)

Sanket Yeole (B9234)

ABSTRACT

Depression is a common mental disorder and one of the main causes of disability worldwide. Lacking objective depressive disorder assessment methods is the key reason that many depressive patients can't be treated properly. Developments in affective sensing technology with a focus on acoustic features will potentially bring a change due to depressed patient's slow, hesitating, monotonous voice as remarkable characteristics. So, our motivation is to find out a speech feature set to detect, evaluate and even predict depression. For examining the correlation between depression and speech, we extract features as many as possible according to previous research to create a large voice feature set.

In the previous system if the person is detected as depressed then that person was treated using medicines but in our system, we are providing some techniques based on which persons level of depression is being determined. Using BDI technique some questions are to be answered. Facial expressions detect the face, expressions of person. In rare cases the nearby doctor/suggestions/ notification are used for depression detection.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1	CNN	Convolutional Neural Network
2	NLP	Natural Language Processing
3	FER	Facial Emotion Recognition
4	WHO	World Health Organisation

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 PROBLEM ON HAND

Every Human being in day to day life is being diagnosed with depression due to affection of different parameters. It disturbed mental state of the human being . So as consider to technology we have one solution to solve this issue in terms of machine learning. Machine learning is a process which learns from past experience and provide the best result when the same issue or event occurs in the future.[2] It considers different parameters like user emotions. Depression is a leading cause of mental ill health. It is a major cause of suicidal ideation and leads to significant impairment in daily life. Machine Learning can help detection and can generate possible solutions to tackle depression .

1.2 BASIC CONCEPT

Depression is a mental illness that is not taken seriously in some countries that can cause us depression.[1] Depression is a psychiatric disorder that needs to be addressed with medication. According to Our World in Data Website, Depressive disoprders occur with varying severity,The WHO's International Classification of Diseases defines this set of disorders ranging from mild to moderate to severe. The Institue of Health Metrics and Evaluation adopt such definitions by disaggregating to mild, persistent depression (dysthymia) and major depressive order (severe).

1.3 PROJECT OBJECTIVE

Using machine learning techniques, our objective from this project is to identify and tackle the problem which is tremendously affecting todays youth's depression. This project aims at properly identifying depression levels by using two approaches such as facial expression based

emotion recognition[8] and calculation of depression level from answers belonging to the questions asked to the user. From questions asked to the answer, we with this software try to recommend movies and tv series with certain genres to target and reduce depression level from users.

1.4 SCOPE OF PROJECT WORK

Though this project can use varieties of techniques such as facial expression detection, social media feeds , questionnaire **,** etc. to target and identify users depression levels. We limit the scope of this project by using only facial expression detection[8] and questionnaire based solution to tackle depression. This scope can be extended by adding extra techniques in emotion detection and using variety of machine learning models as we have looked in the papers we have selected as literature for this project.

1.5 APPLICATIONS

Application of this system is to aim at early detection of level of depression for the real time users.

CHAPTER 2
LITERATURE REVIEW

2.1 RELATED WORK DONE

TABLE 2.1 LITERATURE SURVEY

SR. NO.	Title	Author	Publication Year	Description
1.	Depression Detection Using Emotional Artificial Intelligence	Vignesh Rao, Mandar Deshpande	ICISS 2017, IEEE 2018	This paper aims to apply NLP on twitter feeds for conducting emotion analysis focusing on depression. Individual tweets are classified as neutral or negative based on curated word list to detect depression tendencies.[1]
2.	Emotion Recognition and drowsiness detection using python	Anmol Uppal, S. Tyagi Rishi Kumar	IEEE 2019	This paper uses detection of eye movements such as blinking to avoid any accidents or mishappening like in vehicles or just for security surveillance.[2]
3.	Emotion based mood enhancing music Recommendation	Viral Prasad, Smita Sankhe , Karan Prajapati, Aurobind V. Iyer	IEEE 2017	This paper gives us inspiration for making use of machine learning technologies and making a personal use software out of it.[3]
4.	Facebook social media for depression detection in the thai community	Panida Yomaboot, Kantinee Katchapakirim, Konlakorn Wongpatikaseree, Yongos Kaewpitakkun	JCSSE 2018	This research employs NLP techniques to develop a depression detection algorithm. [4]

5.	Clinical Depression Detection in Adolescent by Face	Prajakta Bhalchandra Kulkarni, Meenakshee M. Patil	IEEE 2017	For implementation of a depression detection method, two algorithms were used named as Fisher vector algorithm and LTrP. Fisher vector is used for representation and description of an image. It uses a Gaussian mixture model. Efficiency of Fisher vector encoding is great for a computation. [5]
6.	Facial Feature Detection using Haar Classifiers	Philip Ian Wilson, Dr. John Fernandez	Journal of computing sciences, 2014	This paper introduced a method to accurately and rapidly detect faces within an image through Haar classifiers.[6]

CHAPTER 3
PROJECT STATEMENT

3.1 PROBLEM DEFINITION

To develop a user centric application program which addresses growing problem of depression in teenagers. Basically to design and develop an application which can be helpful to the normal user , where machine learning is playing a big role to calculate the depression level of the user according to the user Input or face expression detection (parameters like face edges) .

3.2 PROPOSED ALGORITHM/ METHODOLOGIES

3. **Haar cascade Classifier for Face Detection:**

In this system we used Haar classifier algorithm for face detection when one of these features is found, the algorithm allows the face candidate to pass to the next stage of detection. A face candidate is a rectangular section of the original image called a sub-window.[6] Generally these sub-windows have a fixed size (typically 24×24 pixels). This Sub-window is often scaled in order to obtain a variety of different size faces. The algorithm scans the entire image with this window and denotes each respective section a face candidate. The algorithm uses an integral image in order to process Haar features of a face candidate in constant time. It uses a cascade of stages which is used to eliminate non-face candidates quickly. Each stage consists of many different Haar features. Each feature is classified by a Haar feature classifier. The Haar feature classifiers generate an output which can then be provided to the stage comparator. The stage comparator sums the outputs of the Haar feature classifiers and compares this value with a stage threshold to determine if the stage should be passed. If all stages are passed the face candidate is concluded to be a face. [6]

3) **Haar Feature Classifier**

A Haar feature classifier uses the rectangle integral to calculate the value of a feature. The Haar feature classifier multiplies the weight of each rectangle by its area and the results are added together. Several Haar feature classifiers compose a stage. A stage comparator sums all the Haar feature classifier results in a stage and compares this summation with a stage threshold. Each

stage does not have a set number of Haar features. Depending on the parameters of the training data individual stages can have a varying number of Haar features.[6]

b) Haar Features:

Haar features are composed of either two or three rectangles. Face candidates are scanned and searched for Haar features of the current stage. Each Haar feature has a value that is calculated by taking the area of each rectangle, multiplying each by their respective weights, and then summing the results. [6]

2. Goldberg Depression Questionnaire:

Goldberg questionnaire consist of this questionnaire to help determine if you need to see a mental health professional for diagnosis and treatment of depression, or to monitor yourmood. This questionnaire consists of a scale which can be used on a weekly basis to track moods.It might be used to show your doctor how your symptoms have changed from one visit to the next. Changes of five or more points are significant. This sacle is not designed to make a diagnosis of depression or take the place of a professional diagnosis**.**

3.3 MODULES

Login module:

This module is responsible for creating account for the user and storing results and suggestions generated by the system.

Dashboard module:

Provides the user interface for accessing the depression detection system, which includes feature to capture image using the built-in laptop camera and allows user to select an image used for processing for the other modules. Dashboard module also include questionnaire test which user can give for test analysis.

Face Detection module:

This module is responsible for loading of FER dataset[8] and HAAR feature[6] based cascade classifier. It detects frontal face in an image well. It is real time and faster in comparison to other face detector. We use an implementation from OpenCV.

Expression Detection module:

This module uses an Xception CNN module (Mini_Xception,2017).[8] We will train a classification CNN model architecture which takes bounded face (48*48 pixels) as input and predicts probabilities of 7 emotions in the output layer.

Suggestion module:

Depending on the result of user which is generated from previous module. This module collects movies and teen shows which are similar to the emotions of the current user and also might help to tackle depression related issues and finally we generate and present this list to user.

CHAPTER 4

SOFTWARE REQUIREMENT SPECIFICATION

4.1 HARDWARE REQUIREMENTS

- System : Intel i5 Processor
- Hard Disk : 40 GB
- Speed : 1.1 GHz
- Monitor : 15 VGA colour
- Ram : 4 GB

4.2 SOFTWARE REQUIREMENTS

- Language : Python
- Operating System : Windows 7
- IDE :JetBrains Pycharm

CHAPTER 5
SYSTEM DESIGNS

5.1 SYSTEM ARCHITECTURE

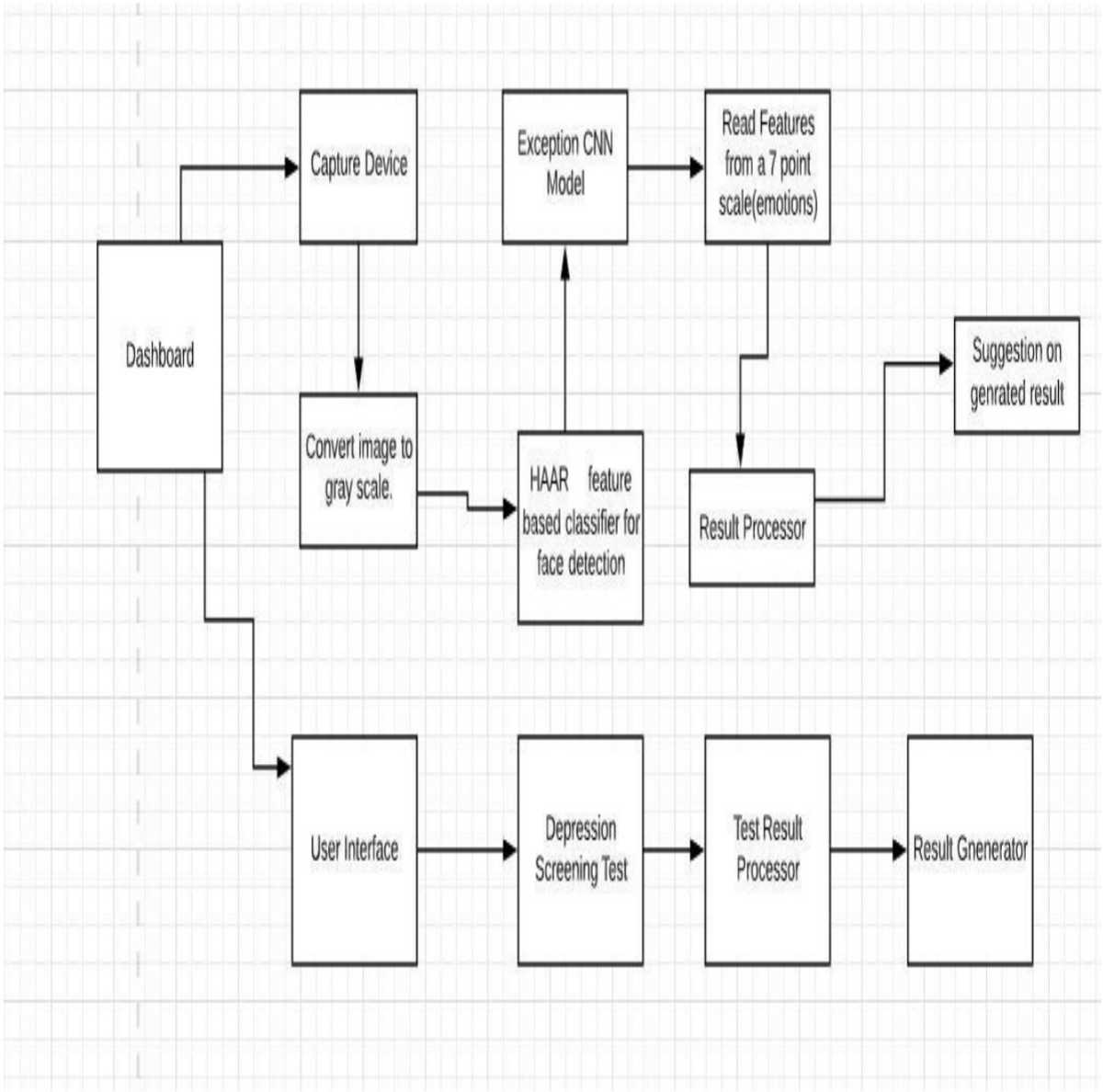


FIG 5.1.SYSTEM ARCHITECTURE

A system architecture is the conceptual model that defines the structure, behaviour and more views of the system. Generally, in the depression detection approaches consists of following

steps. Initially, user will login. Then through dashboard, it provides the user interface for accessing the depression detection system, which includes feature to capture image also include questionnaire test which user can give for test analysis. For the face detection,it is responsible for loading of FER dataset and HAAR feature based cascade classifier. It detects frontal face in an image well. For the detection of expression, it uses an Xception CNN algorithm[9]. And in the end result will be generated and suggestions will be given on the basis of generated result.

5.2 USE CASE DIAGRAM

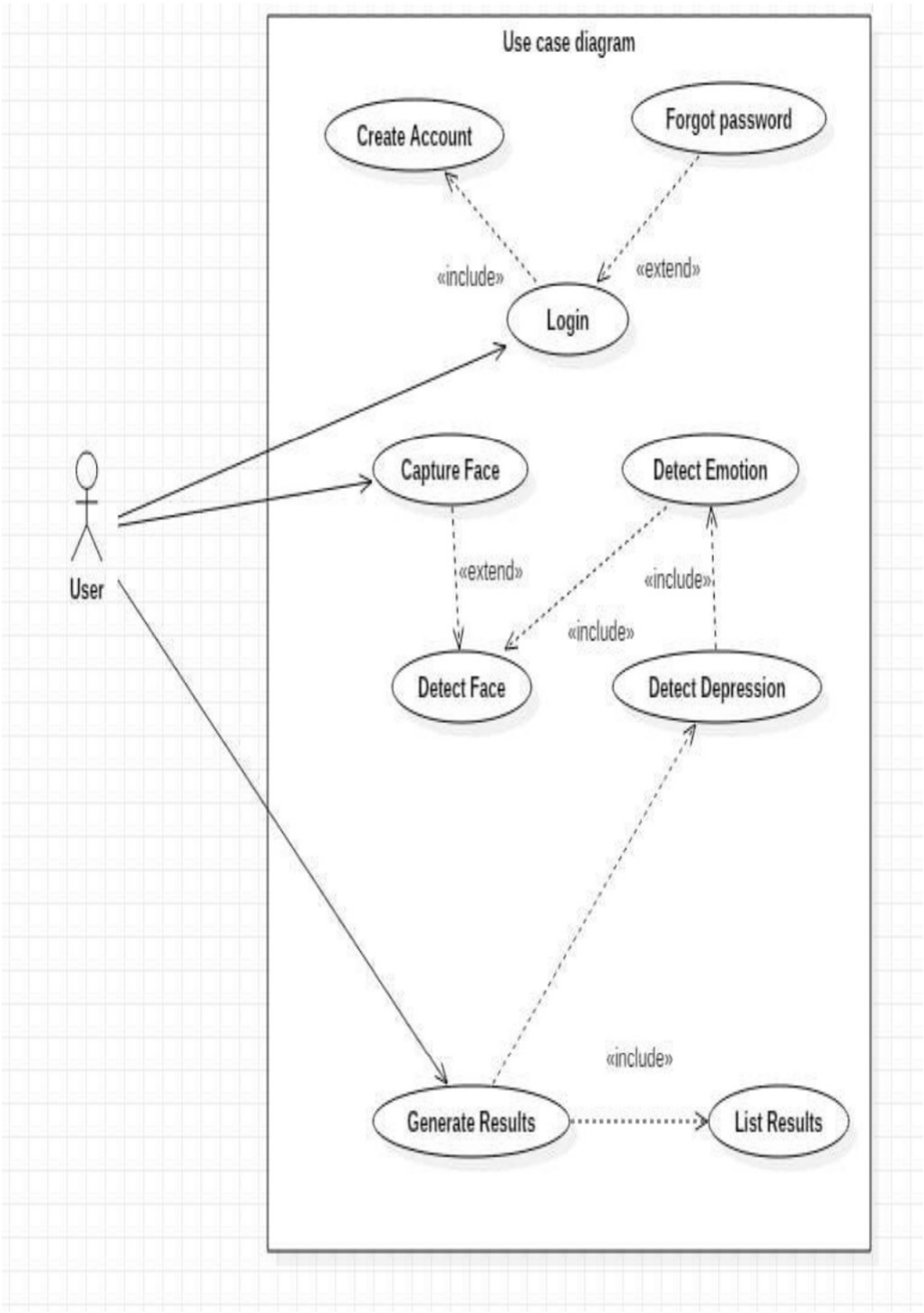


FIG 5.2 USE CASE DIAGRAM

A **use case diagram** at its simplest is a representation of a user's interaction with the system shows the relationship between the user and the different use cases in which the user is

involved .A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

5.3 ACTIVITY DIAGRAM

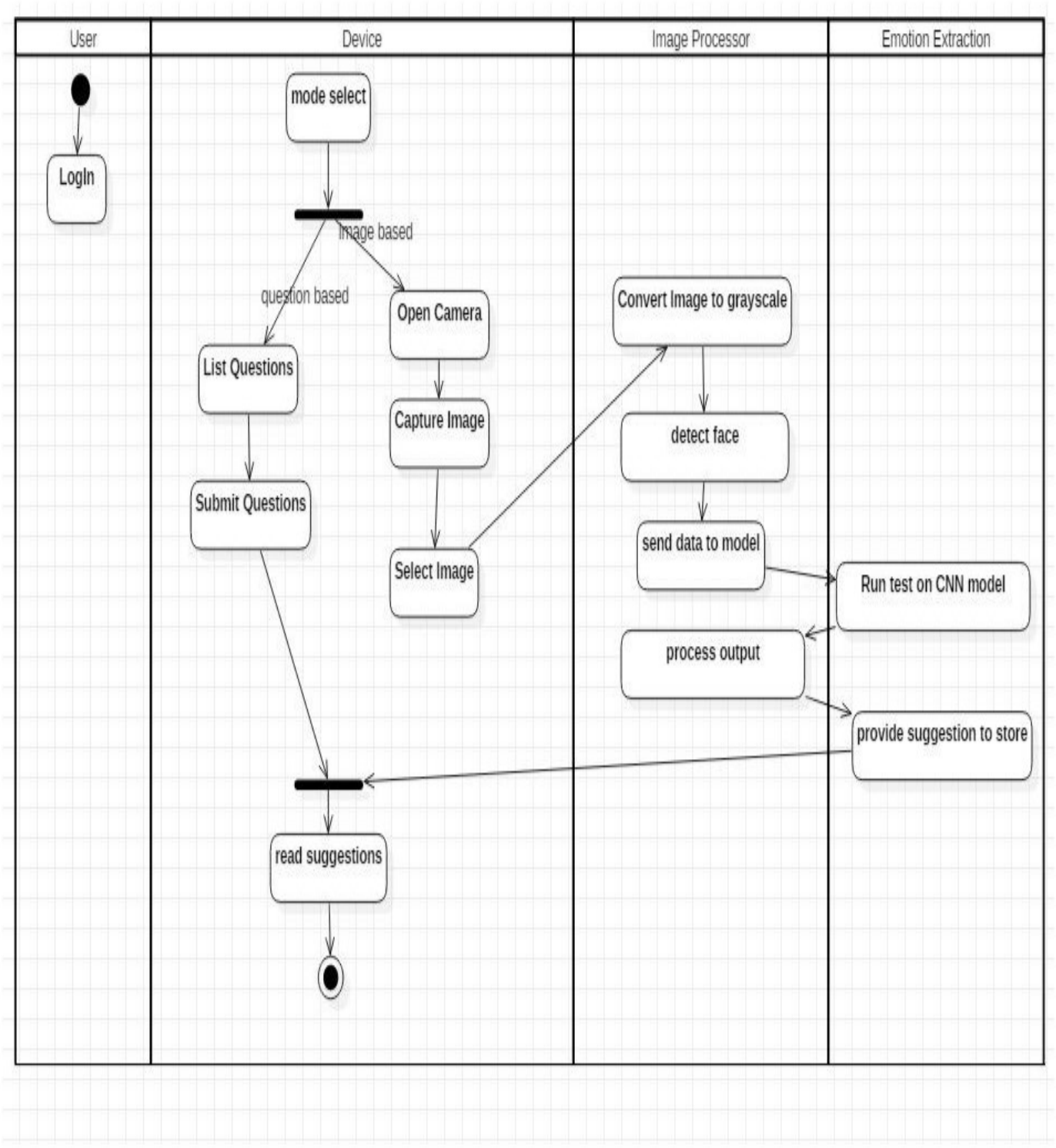


FIG 5.3 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities.

5.4 SEQUENCE DIAGRAM

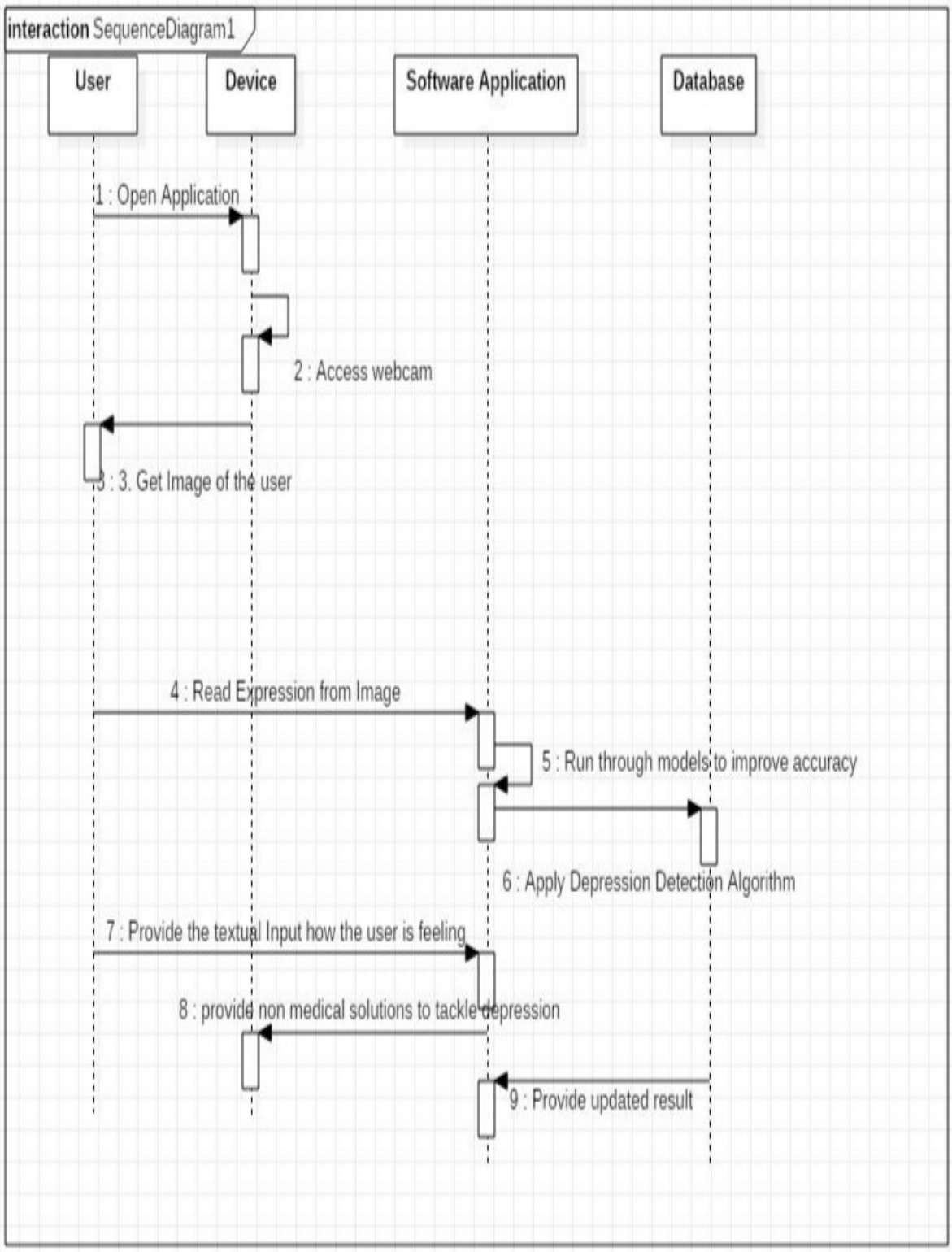


FIG 5.4 SEQUENCE DIAGRAM

A **sequence diagram** shows object interaction arranged in time sequence. It depicts the objects and classes involved in the scenario in the sequence of message exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are ty;pically associated with use case realizations in the logical view of system under development. Sequence diagrams are sometimes called event diagram or event scenarios.

5.5 CLASS DIAGRAM

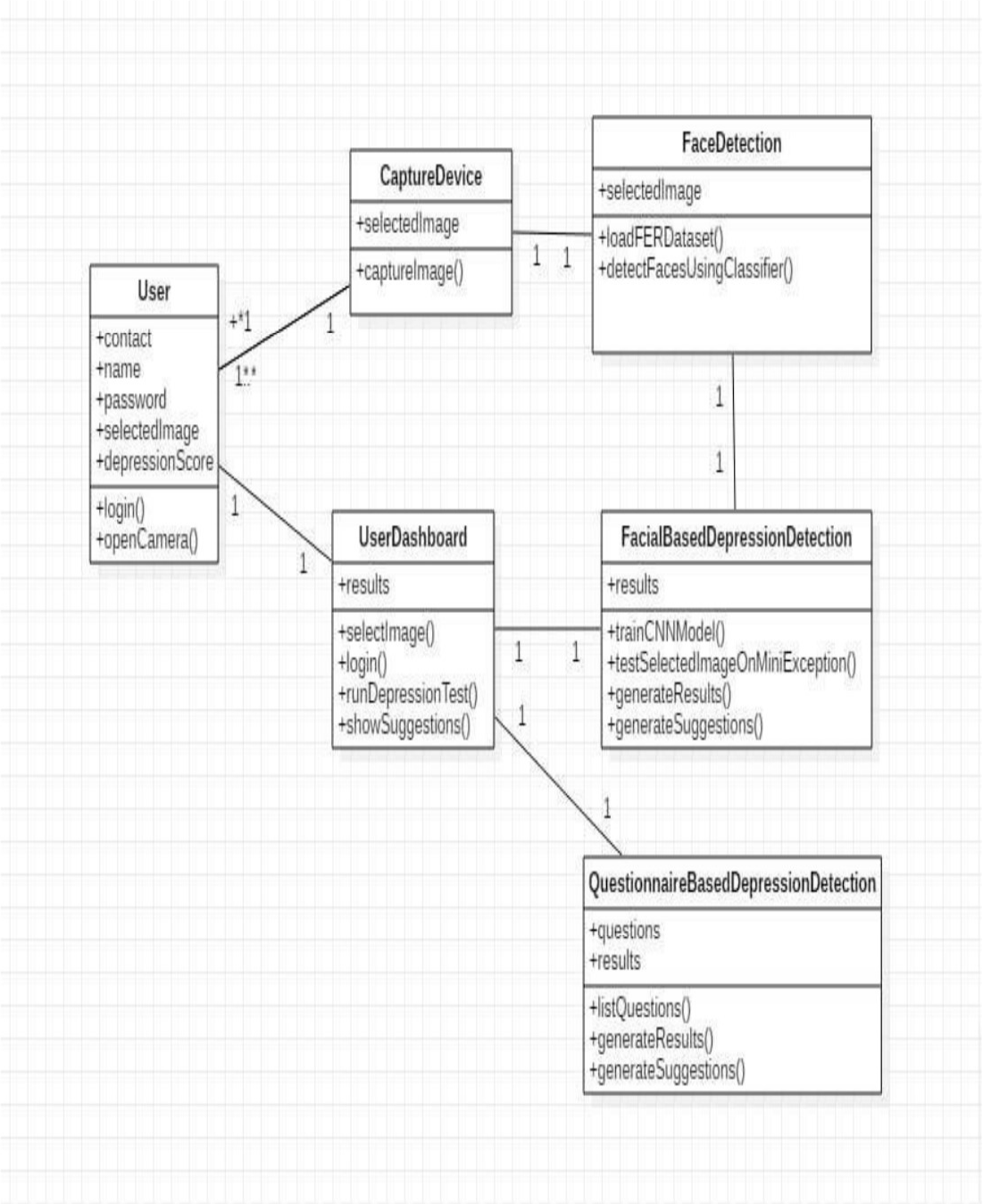
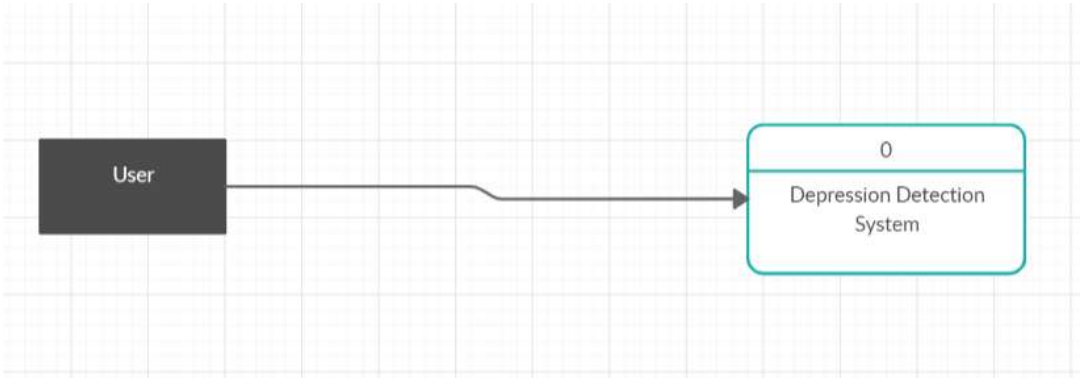


FIG 5.5 CLASS DIAGRAM

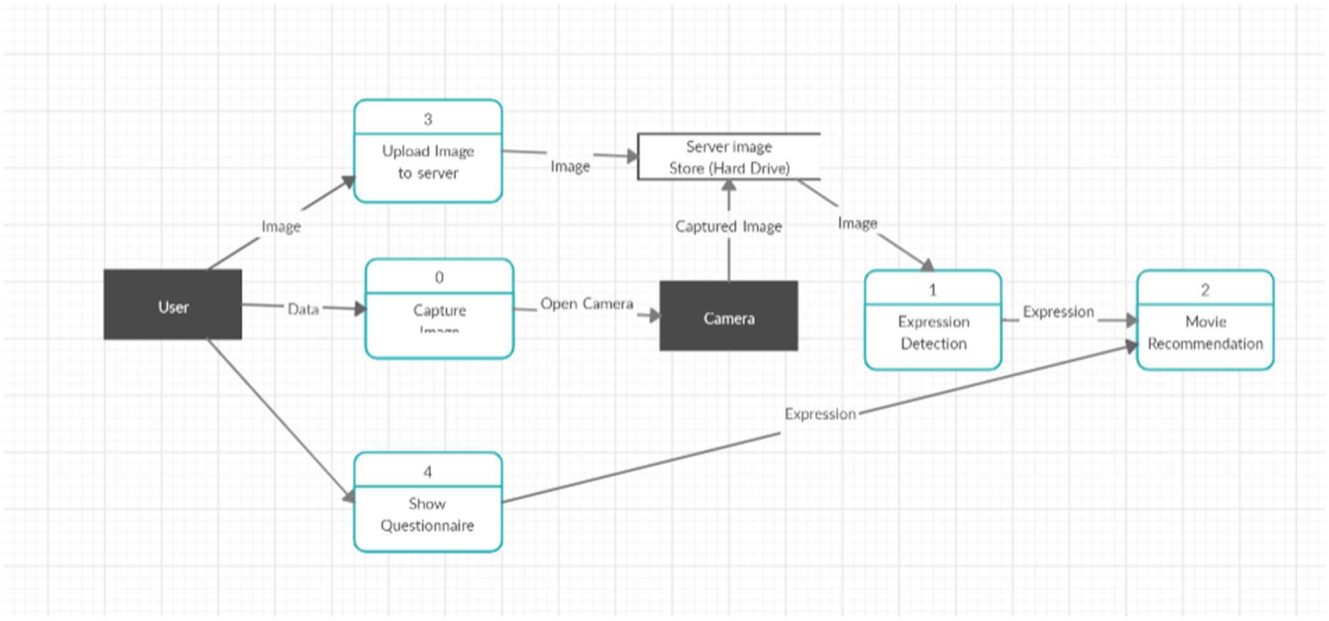
In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modelling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

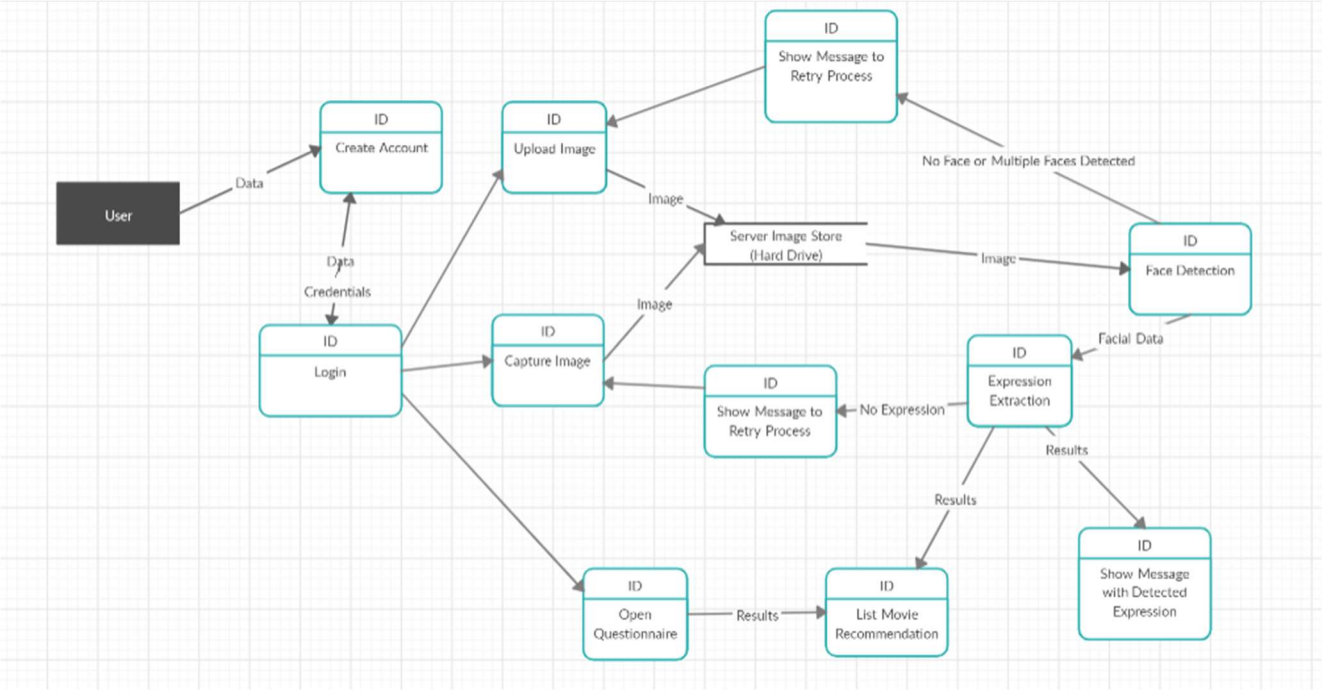
5.6 DATA FLOW DIAGRAM



5.6.1 Level 0 Data Flow Diagram



5.6.2 Level 1 Data Flow Diagram



5.6.3 Level 2 Data Flow Diagram

A **data-flow diagram** (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.[1]

The data-flow diagram is part of the structured-analysis modelling tools. When using UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.

5.7 DEPLOYMENT DIAGRAM

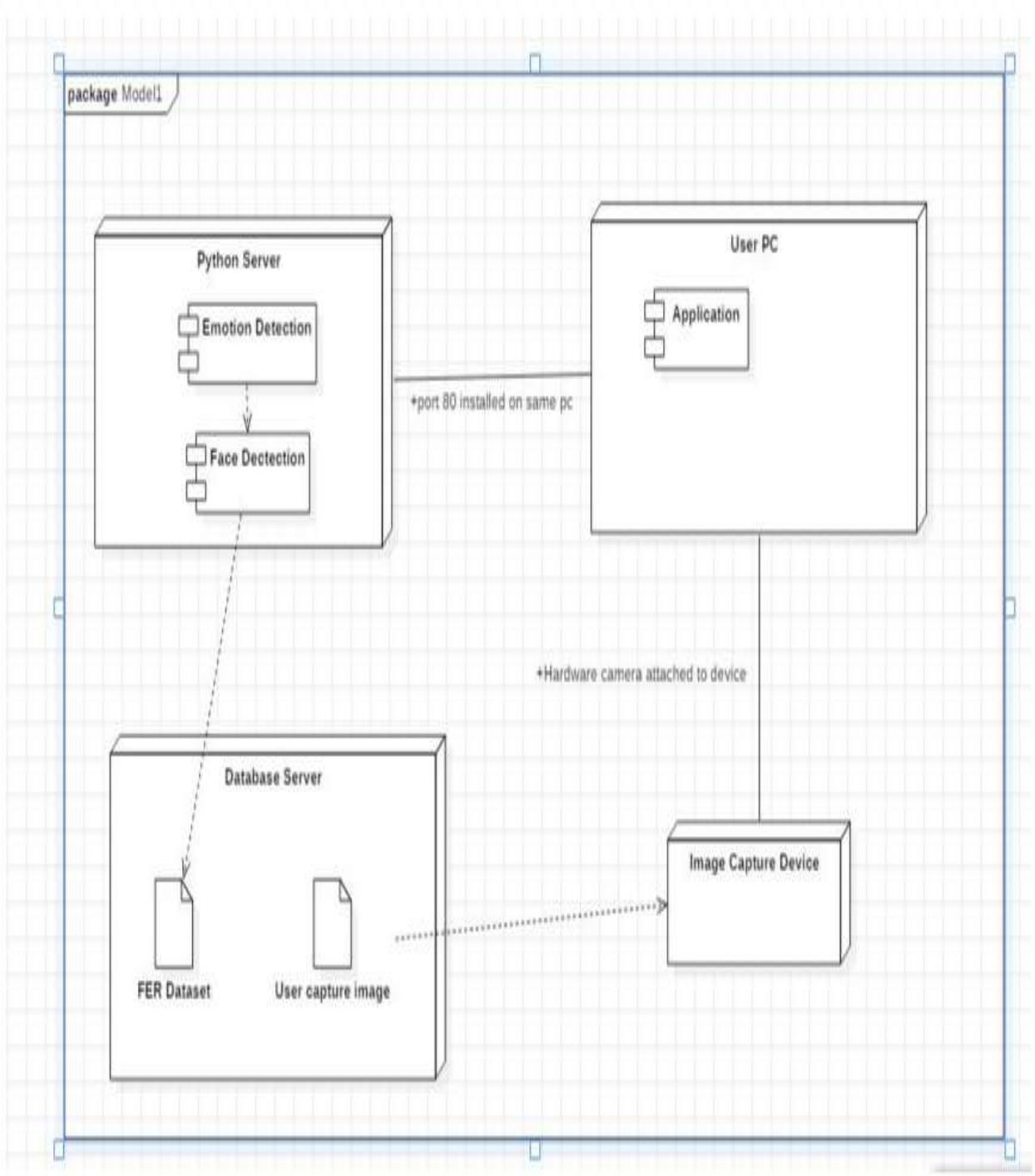


FIG 5.7 DEPLOYMENT DIAGRAM

A **deployment diagram** in the Unified Modelling Language models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what

hardware components (“nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

5.8 COMPONENT DIAGRAM

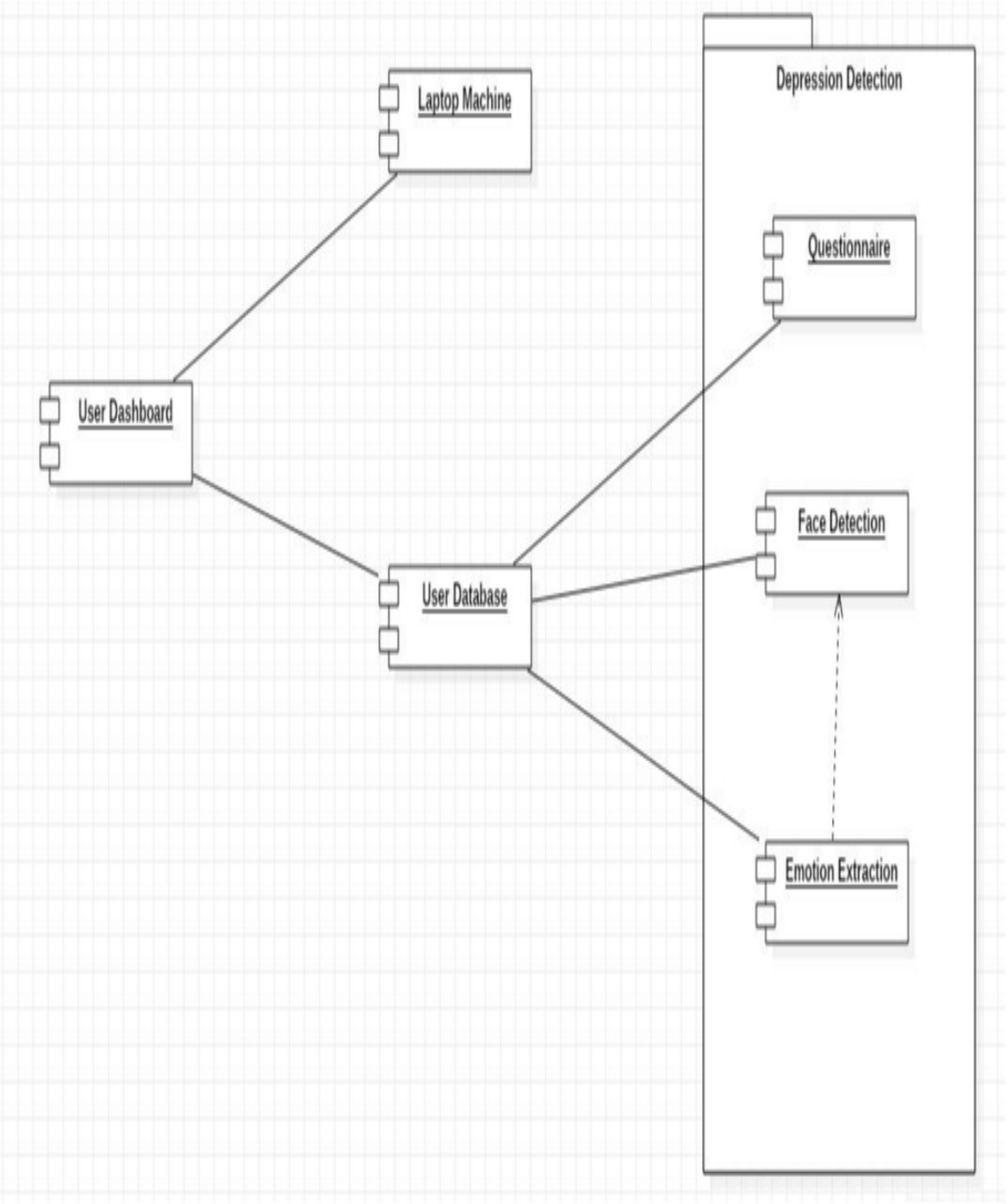


FIG 5.8 COMPONENT DIAGRAM

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to

help model implementation details and double-check that every aspect of the system's required functions is covered by planned development. In the first version of UML, components included in these diagrams were physical: documents, database table, files, and executables, all physical elements with a location. In the world of UML 2, these components are less physical and more conceptual stand-alone design elements such as a business process that provides or requires interfaces to interact with other constructs in the system.

.

5.9 COLLABORATION DIAGRAM

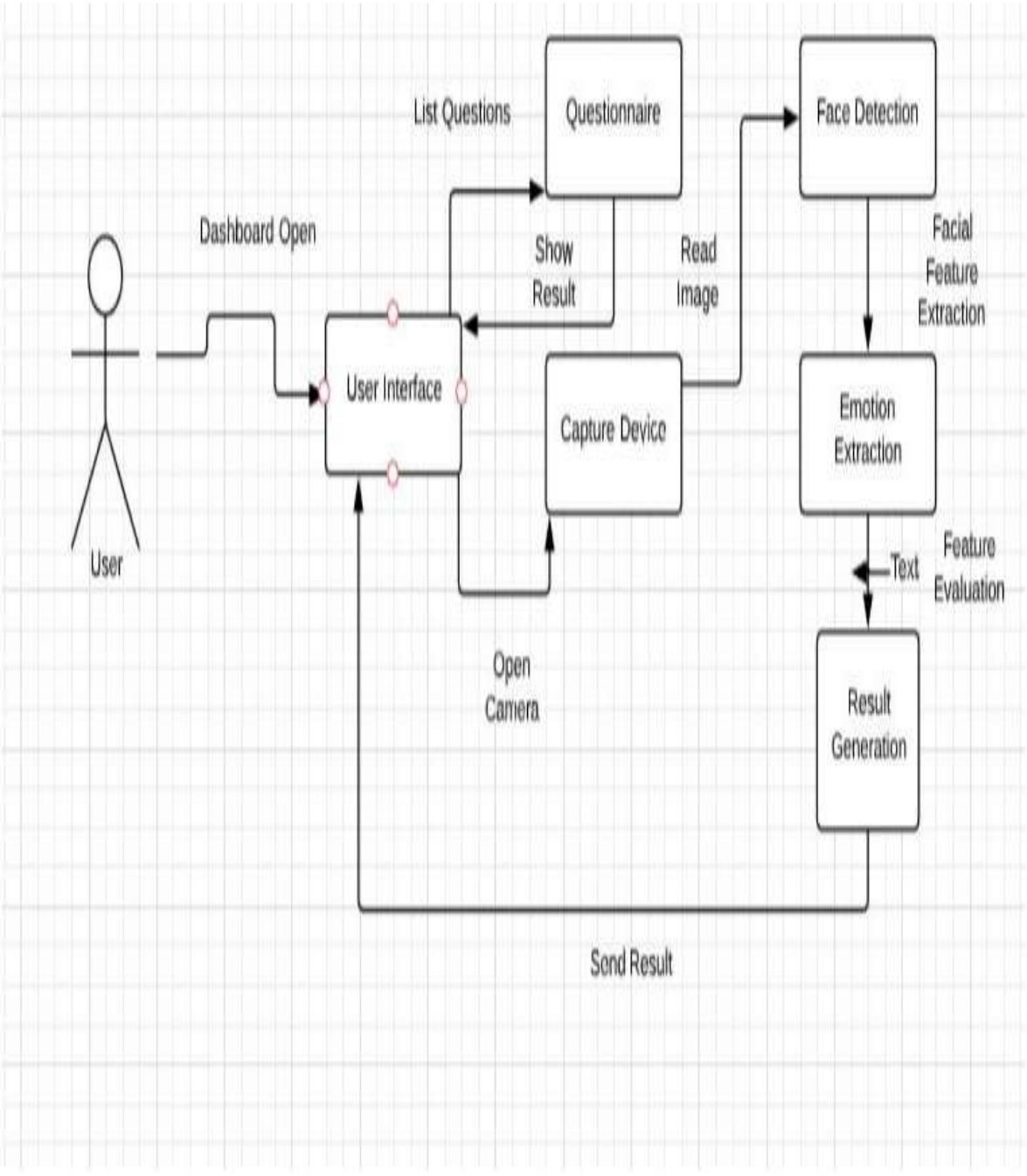


FIG 5.9 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case

and define the role of each object. Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams

CHAPTER 6
TESTING

6.1 Types of Testing

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, in-dependent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

It describes which testing types we are going to follow in our testing life cycle. Here we are using:

- Black Box Testing
- Functional Testing
- UI Testing
- Integration Testing

6.1.1 Module Level Testing

We can design the various modules for any software and each module contains the small part of the code. If the error will be found at any module, then it encourages the programmer to search and rectify that error without disturbing the other modules.

6.1.2 Function Testing

The prime objective of Functional testing is checking the functionalities of the software system. It mainly concentrates on Mainline functions: Testing the main functions of an application.

Basic Usability: It involves basic usability testing of the system. It checks whether an user can freely navigate through the screens without any difficulties. Accessibility: Checks the accessibility of the system for the user

Error Conditions: Usage of testing techniques to check for error conditions.

It checks whether suitable error messages are displayed.

6.1.3 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and ows, emphasizing pre-driven process links and integration points.

6.1.4 Black Box Testing

Black box testing methods focus on the functional requirements in the software. That is, black box testing enables us to derive sets of input conditions that will fully exercise. All functional requirements of the program Black box testing attempts to find errors in the following categories:

- Incorrect or missing function
- Interface Errors
- Errors in data structure or external job access
- Performance errors
- Initialization and termination errors.

In the proposed application with the help of this technique, we do not use the code to determine a test suite; rather, knowing the problem that we're trying to solve, we come up with four types of test data:

- Easy-to-compute data
- Typical data
- Boundary / extreme data

But in our application we utilize external dataset as input.

6.1.5 White Box Testing

White box testing is a set case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, we can derive test cases that

Guarantee that all independent paths within a module have been exercised at least once

Exercise all logical decisions on their true and false sides

Execute all loops at their boundaries and within their operational bounds

Exercise internal data structures to ensure their validity In the proposed application the white box testing is done by the developer implemented the code, the implements code is studied by the coder, determines all legal (valid and invalid) AND illegal inputs and verifies the outputs against the expected outcomes, which is also determined by studying the implementation code.

6.2 Test Cases and Test Results

Table 6.1: Test Case 1

Test case id	1
Test case name	View Home page
Test Case Description	After running the application should open the home page.
Test Steps	1.Open Command Prompt 2.Run Project 3.Home page opened
Expected Result	Application should provide the home page.
Actual Result	Application has valid home pages
Status	PASS

Table 6.2: Test Case 2

Test case id	2
Test case name	Registration
Test Case Description	After opening the home page the user should register with valid username , mobile number, address and password into the system.
Test Steps	1.Click on Register link 2.Enter all details 3.Click on register button
Expected Result	User registration should be done
Actual Result	Registration success message.
Status	PASS

Table 6.3: Test Case 3

Test case id	3
Test case name	Login
Test Case Description	After Successfully registration user should login to the system
Test Steps	1.Enter Valid Username and Password 2.Click on Login Button
Expected Result	User Home Page should be displayed
Actual Result	Login success message
Status	PASS

Table 6.4: Test Case 4

Test case id	4
Test case name	Capture Image
Test Case Description	Selecting image for emotion detection
Test Steps	1.Click on capture image button 2. Camera window opens 3. Click capture image now button
Expected Result	Image Captured Successfully
Actual Result	Captured Image success message.
Status	PASS

Table 6.5: Test Case 5

Test case id	5
Test case name	Select Image
Test Case Description	Select Image for detection of emotion
Test Steps	1. Click on Upload Image Button
Expected Result	Captured Emotion Shown on screen
Actual Result	Emotion from image shown on screen
Status	PASS

Table 6.6: Test Case 6

Test case id	6
Test case name	Questionnaire Test
Test Case Description	Select YES or NO in questions
Test Steps	Provide answers to questions
Expected Result	Generation of Results
Actual Result	Display of Emotion based on questions
Status	PASS

CHAPTER 7

RESULTS

7.1 Screenshots

FIGURE 7.1 SIGN IN SCREEN

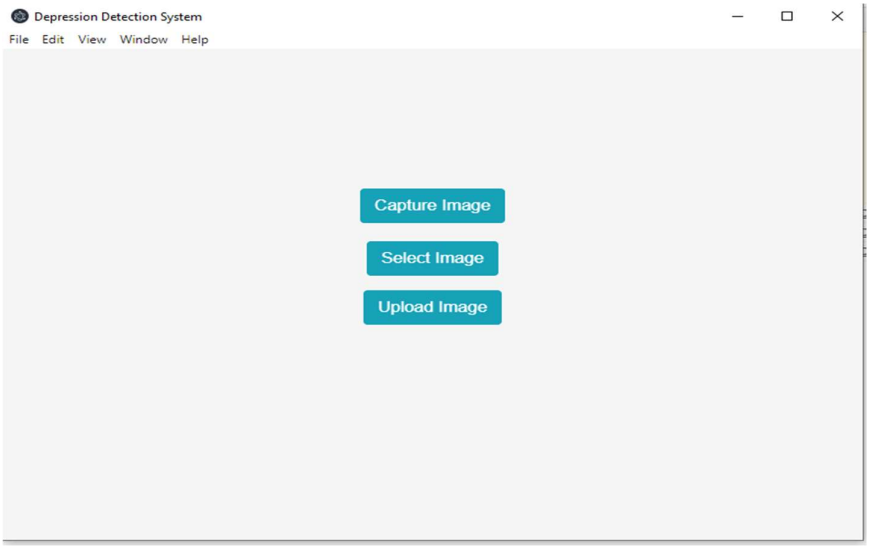


FIGURE 7.2 HOME SCREEN

FIGURE 7.3 NEUTRAL FACE DETECTION

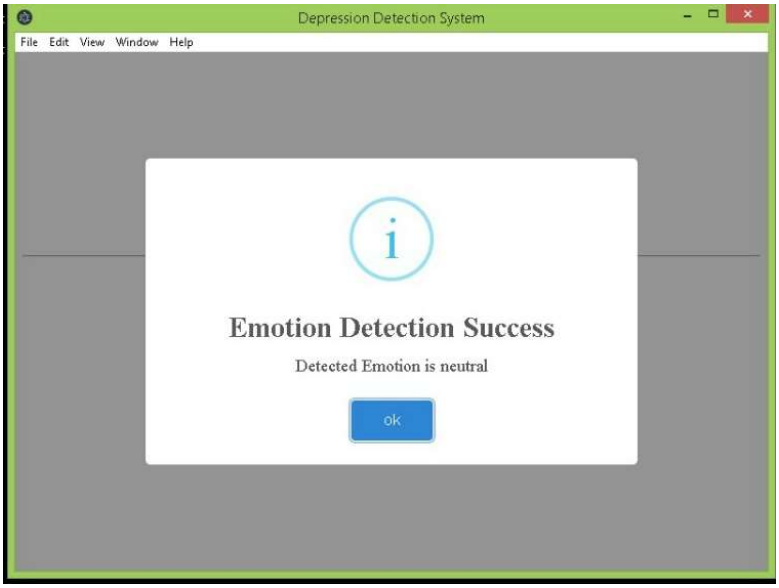


FIGURE 7.4 NEUTRAL FACE DETECTION RESULT

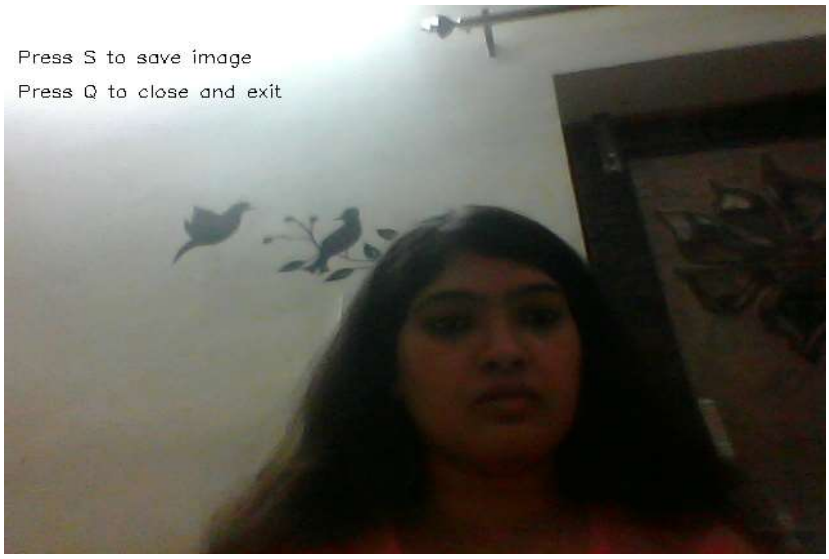


FIGURE 7.5 SAD FACE DETECTION

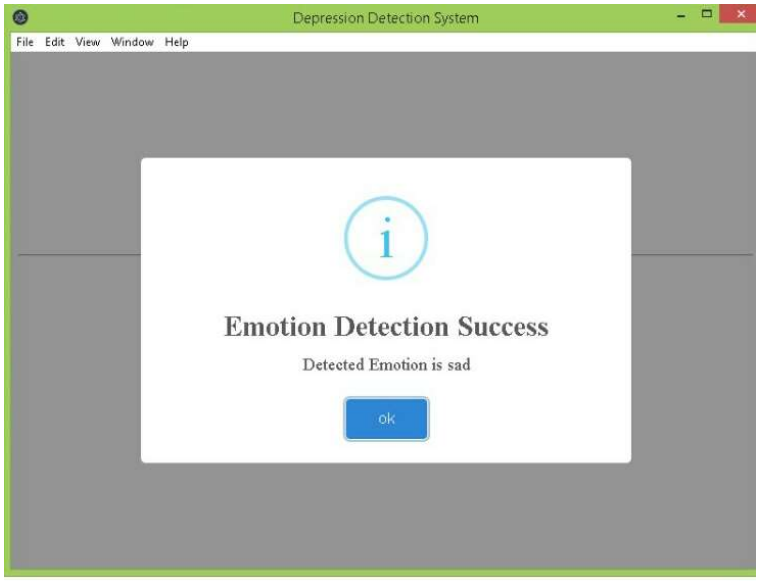


FIGURE 7.6 SAD FACE DETECTION RESULT

FIGURE 7.7 HAPPY FACE

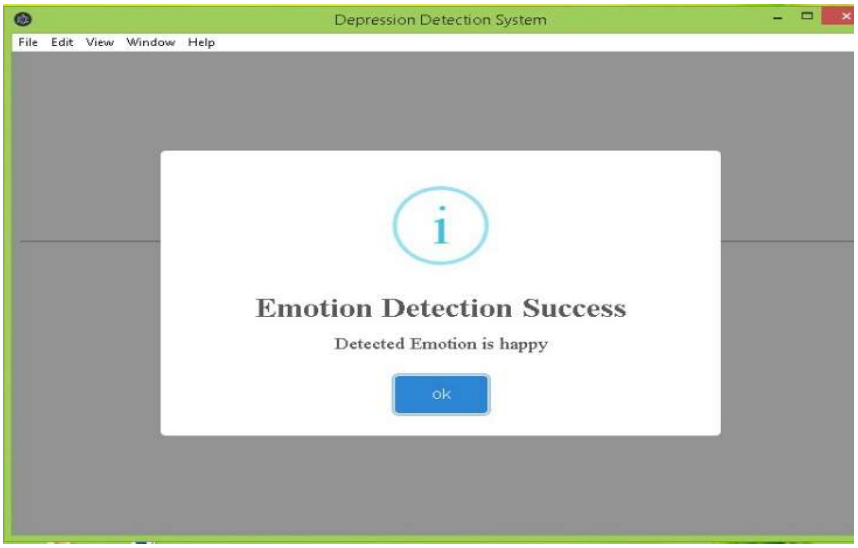


FIGURE 7.8 HAPPY FACE DETECTION RESULT

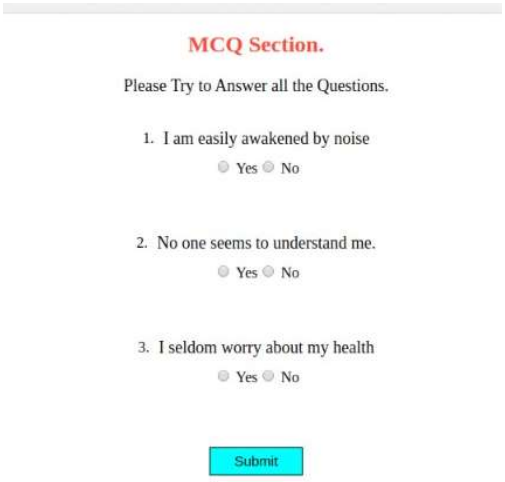


FIGURE 7.9 MCQ QUESTIONS LIST

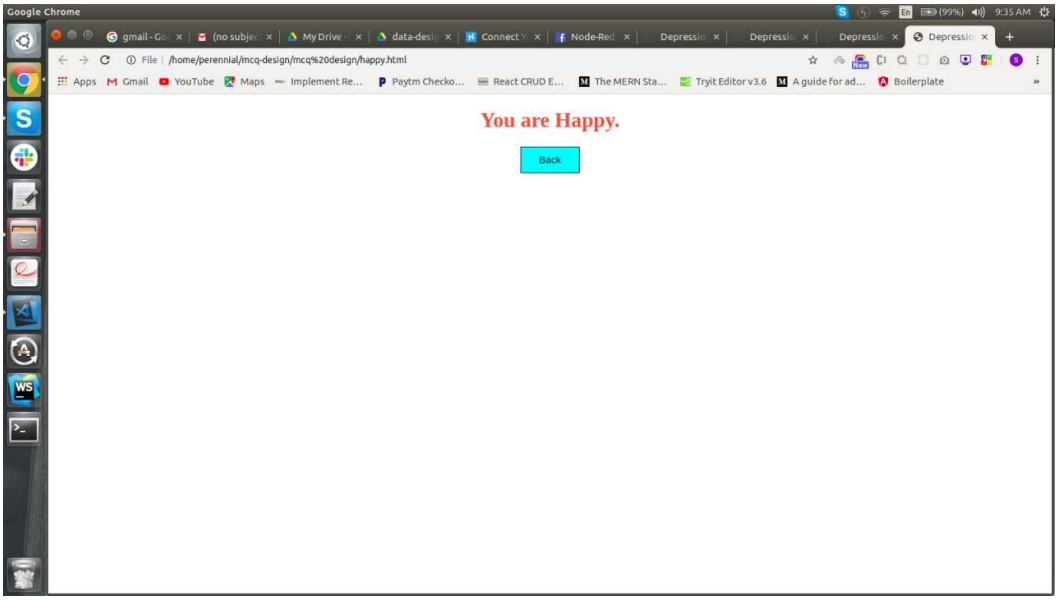


FIGURE 7.10 MCQ QUESTIONS RESULT

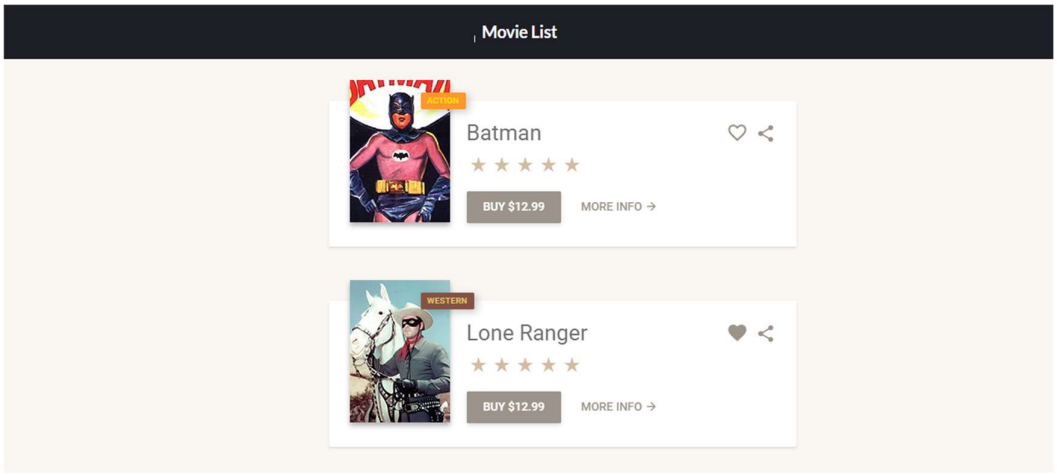


FIGURE 7.11 MOVIES SUGGESTION LIST

CHAPTER 8
CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

Usually in the system if the person faces depression then that person will be treated using medicines but in our system, we are providing some techniques based on which persons level of depression is being determined. Using questionnaire some questions are to be answered and facial expression which detect the face, expressions of person. In rare cases, the nearby doctors or suggestions are used for detecting level of depression.

8.2 Future Scope

We want to scale our existing system to various other platform like mobile app's where user can use our system with the no cost. We are thinking to connect an admin panel to system. Where admin can easily add the questions and it can dynamically reflect to our user.

REFERENCES

[1] Vignesh Rao, Mandar Deshpande ,"Depression Detection using Emotional Artificial Intelligence", IEEE, pp. 858-862, 2018

[2] Anmol Uppal, S. Tyagi, Rishi Kumar, " Emotion Recognition and Drowsiness Detection using Python" ,IEEE, pp. 464-469,2019

[3] Viral Prasad, Smita R. Sankhe, Karan Prajapati, Aurobind V. Iyer, " Emotion based mood enhancing music recommendation" , IEEE, pp. 1573-1577, 2017

[4] Panida Yomaboot, Kantinee Katchapakirim, Konlakorn Wongpatikaseree, Yongos Kaewpitakkun " Facebook Social Media for Depression Detection in the Thai Community", JCSSE, pp. 1-6,2018

[5] Prajakta Balchandra Kulkarni, Meenakshee M. Patil, " Clinical Depression Detection in Adolescent by Face", IEEE, pp. 1-4,2017

[6] Philip Ian Wilson, Dr. John Fernandez, "Facial Feature Detection using Haar Classifiers", journal of computing sciences, pp 127-133, 2014

[7] Selvarajah Thuseethan , Sutharshan Rajasegarar, John Yearwood, " Emotion Intensity Estimation from Video Frames uing Deep Hybrid CNN", Neural Networks(IJCNN) 2019 International Joint Conference , pp 1-10, 2019

[8] Ankit S. Vyas, Harshadkumar B. Prajapati, Vipul K. Dabhi, "Survey on Face Expression Recognition using CNN" , Advanced Computing and Communication System(ICACCS) 2019 5th International Conference , pp 102-106,2019

[9] Octavio Arriaga, Matias Valdenegro-Toro, Paul Ploger, "Real time CNN for Emotion and Gender Classification", ICRA, 2017

[10] Kumar G A Rajesh, Ravi Kant Kumar, Gautam Sanyal, "Facial Emotion Analysis using Deep CNN", International Conference on Signal Processing and Communication (ICSPC'17) , 28th & 29th July 2017