Stress Detection Using Machine And Deep Learning

M.Nasir Abdullah Khan
Usman Baig

Fall 2023

Department of Computer Science
COMSATS University Islamabad
Attock Campus- Pakistan

COMSATS University Islamabad,

Attock Campus, Attock Pakistan

Stress Detection using Machine and Deep Learning

By

M. USMAN BAIG	**CIIT/SP20-BSE-040/ATK**
NASIR ABDULLAH	**CIIT/SP20-BSE-038/ATK**

Supervisor

Mr. KAMRAN

Bachelor of Science in Software Engineering (2020-2024)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

COMSATS University Islamabad,

Attock Campus, Attock Pakistan

Stress Detection using Machine and Deep Learning

A project presented to

COMSATS University, Islamabad (ATTOCK CAMPUS)

In partial fulfillment

of the requirement for the degree of

Bachelors of Science in Software Engineering (2020-2024)

By

M. USMAN BAIG	**CIIT/SP20-BSE-040/ATK**
NASIR ABDULLAH	**CIIT/SP20-BSE-038/ATK**

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other, we will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

M.USMAN BAIG   M.NASIR ABDULLAH KHAN

---------------------------   ---------------------------

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (SE) “STRESS DETECTION USING MACHINE AND DEEP LEARNING” was developed by **M.USMAN BAIG (CIIT/SP20-BSE-040/ATK)** and **NASIR ABDULLAH KHAN (CIIT/SP20-BSE-038/ATK)** under the supervision of “Mr. KAMRAN” and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Software Engineering.

Committee:

1. External Examiner --------------------------------------

Dr. Zeeshan Ali
Assistance Professor
National IT & Automation, Islamabad

2. Supervisor --------------------------------------

Mr. Kamran Ali
Department of Computer Science
COMSATS University Islamabad
Attock Campus

3. Head of the Department --------------------------------------

Dr. Khalid Iqbal
Associate Professor HOD(CS)
COMSATS University Islamabad
Attock Campus

Executive Summary

The Stress Detection Using Machine and Deep Learning project focuses on developing a system to detect stress levels in individuals using facial expression analysis through advanced machine and deep learning techniques. The system is designed to be highly accurate and reliable, with an accuracy rate of nearly 90% or higher. It can be used in various settings, including research projects, hospitals, and offices, to monitor and manage patient stress levels. The system provides real-time feedback to the user on ways to manage their stress effectively. Our system utilizes a large dataset of facial expression images to train the AI model, which can then make predictions about the subject's facial expressions in real-time video streams. The system can be easily integrated into existing software systems and can be tailored to meet the specific needs of different settings and applications. The benefits of this stress detection system are numerous, including improving patient outcomes and enhancing overall quality of life.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Mr. KAMRAN”. Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are grateful to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us with the values of honesty & hard work.

M. USMAN BAIG

NASIR ABDULLAH KHAN

Abbreviations

SRS	Software Require Specification
PC	Personal Computer
CV	Computer Vision
HCI	Human Computer Interaction
CNN	Convolutional Neutral Network
SDD	Software Design Description
BO	Business Objective

Table of Contents

List of Figures

List of Table

Chapter: 1
Introduction

1 Introduction

Stress is a common mental health condition that impacts a lot of people today, especially those who follow strict routines and face daily obstacles that can be very taxing on their minds. Unfortunately, the individual's health cannot always be continuously monitored by anyone, which makes it difficult to detect when they are struggling with Stress. Additionally, treating Stress often requires a long medication process that can be both time-consuming and expensive.

To address this problem, we have created a system that aims to track the current state of the individual's facial expression, evaluate expression levels, predict the percentage of the facial expression and the system also give some questions to the patient which he have to answer, on the bases of facial features and the questionnaire, our system will predict stress level. Our system will predict stress of patient on real time. If the patient have high stress. Our system will recommend medicine to the patient.

The system works by analyzing the user's facial expressions using computer vision and machine learning algorithms. We have use convolutional neural network model for face detection and for extraction of facial features. We are using laptop camera to get image of patient on real time and a questionnaire section thorough which our system will predict stress level.

1.1 Vision Statement

For individuals experiencing high levels of stress and anxiety, who are seeking a convenient and reliable way to monitor and manage their mental well-being, the Stress detection is a smart and user-friendly stress detection system that utilizes machine learning and deep learning algorithms. Unlike traditional self-report methods or other wearable devices that offer limited insights, our product can accurately detect and analyze multiple physiological and behavioral markers of stress in real-time. With its intuitive interface and personalized recommendations, Stress detection empowers users to take proactive steps towards better mental health and overall wellness.

1.2 Related System Analysis

Table 1: Related System Analysis

Projects	**Questionnaire**	**Real-time stress detection**	**Accuracy above 90**	**Gender Classifier**
Blazemeter	✓	✗	✗	✗
Emotient	✓	✗	✗	✗
Affectiva	✓	✗	✗	✗
EmoVu	✓	✗	✗	✗
Nviso	✓	✗	✗	✗
Our project	✓	✓	✓	✓

1.3 Project Deliverables

List down the project deliverables.

- User Interface Design
- Machine learning models for stress detection
- Deep learning models for feature extraction
- Software application for stress monitoring
- Testing and validation framework
- Documentation and user manuals

1.4 System Limitations

- The system will not be able to work in low lightening environment in which the face is not clearly visible and not easy to understand.
- With low RAM and processing power like using old technology CPU's will downgrade theperformance of the system working.

1.5 Tools and Technologies

Table 2: Tool and Technologies

Tools And Technologies	**Tools**	**Version**	**Rationale**
	Pycharm	2022	IDE
	Pyqt5		Front end
	MS Word	2015	Documentation
	MS PowerPoint	2015	Presentation
	Technology	**Version**	**Rationale**
	Python	3.8	Programming language
	Dlib	3.6	Face Detection Module
	Tensor flow, Keras	2.4.3, 2.2.4	Deep Learning Modules

1.6 Relevance to Course Modules

Stress detection using machine learning and Deep Learning is a topic that is relevant to several courses studied during a BSSE (Bachelor of Science in Software Engineering) degree.

Firstly, courses on Artificial Intelligence cover the fundamental concepts and techniques that areused to develop intelligent systems capable of detecting stress.

Secondly, courses on Computer Vision provide a foundation for understanding how facial expressions can be analyzed using computer algorithms to detect stress. These course cover techniques for image segmentation, feature extraction, and pattern recognition that are essential for building a stress detection system that uses facial expression analysis.

Thirdly, courses on Human-Computer Interaction are also relevant, as the stress detection systemmust be designed to be intuitive, user-friendly, and accessible to individuals who may not have atechnical background. These courses teach principles and methods for designing interfaces and interactions that are easy to use and promote engagement.

Finally, courses on Data Science provide a foundation for understanding how data can be collected, analyzed, and interpreted to gain insights into patterns of stress in a population. These course cover techniques for data mining, statistical analysis, and machine learning that are essential for building an effective stress detection system

Chapter: 2
Problem Definition

2 Problem Definition

The precise problem to be solved with a stress detection system is the increasing prevalence of stress-related illnesses and the need for early detection and management of stress levels. Chronicstress has been linked to various physical and mental health conditions, such as heart disease, diabetes, depression, and anxiety. Additionally, stress can have a significant impact on an individual's quality of life, leading to decreased productivity, impaired relationships, and decreased overall well-being.

A stress detection system can help individuals identify when they are experiencing stress and provide them with personalized recommendations for stress reduction techniques. By using advanced machine learning algorithms to analyze facial expressions and other physiological indicators of stress, the system can provide accurate and timely feedback on an individual's stresslevels. This information can help individuals take proactive steps to manage their stress levels, promoting overall wellness and preventing the development of stress-related health conditions.

The outcome of implementing a stress detection system is the reduction of stress-related illnesses and the promotion of overall well-being. By detecting stress early and providing personalized recommendations for stress reduction techniques, individuals can take control of their emotionalwell-being and make positive changes in their lives. Additionally, the system can improve productivity and performance in work and personal life by reducing the negative impact of stresson individuals. Overall, the outcome of a stress detection system is a healthier, happier, and moreproductive population.

2.1 Problem Statement

We must keep track of a person's stress levels to know when to encourage relaxation and rest for the sake of their health. Because not many doctors are available to check on the system continuously for monitoring purposes, our system will monitor facial expressions to identify stress. Stress has been shown to have detrimental effects on worker performance and wellbeing in a large body of literature. These detrimental effects are especially pronounced for service representatives who must constantly tolerate and manage client emotions. This paper presents and describes a deep learning model to predict real-time service agent stress from emotion patterns in voice-to-voice service interactions.

2.2 Objectives of the Proposed System

BO-1: Face Recognition will be in real-time and flexible.

BO-2: This system will be helpful for people with continuous stress peaks in the long term.

BO-3: Doctors use it to monitor patient stress levels.

BO-4: The system will monitor multiple Face at the time.

BO-5: Model accuracy of 90%.

2.3 Scope

This project's focus is on patient stress in research projects, offices, hospitals, and any other setting where people work. It expands on the idea of straightforward face detection and can be very helpful in understanding psychological human nature as well as ensuring the improvement of human communication because it yields very good results with an accuracy of nearly 90% or higher. Stress Detector is a user-interface program that recognizes facial expressions in videos being recorded by cameras.

2.4 Modules

Following are the modules of the System.

- Face Detection System
- Expression Classifier
- Gender Classifier
- Image Drawing
- Recommend Medicine

Module 1: Face Detection System

Here we will detect the faces from the live stream frames and extract the faces from the frames.

Module 2: Expression Classifier

Classify the extracted frame and classify the face with its expression. Here the face is preprocessed and passed into the model and it will predict the Expression as happy, sad,confused, etc.

Module 3: Image Drawing

After the prediction display the label on the opencv drawing function are used to draw boundingboxes on the face image and display the labels on the image.

Module 4: Questionnaire

System will ask some questions to the patient.

Module 5: Check stress

System will check stress on real-time.

Module 6: Recommend medicine

System will check recommend medicine according to the stress level.

CHAPTER: 3
Requirement Analysis

3 Requirement Analysis

Provide an introduction to the chapter in a few lines. Write details in each of the sections provided ahead.

3.1 User classes and characteristics

Different types of user can use the stress detection system. Some possible user classes for a stress Detection system could be:

- **Individuals:** These are the end users who will be using the stress detection system. They could be anyone, from students to employees to athletes, who want to monitor their stress levels.
- **Healthcare professionals:** Healthcare professionals, such as doctors, psychologists, and counselors, could use the stress detection system to monitor the stress levels of their patients and provide appropriate treatment.
- **Employers:** Employers could use the stress detection system to monitor the stress levels of their employees and provide support or interventions when necessary.

The characteristics of a stress detection system will depend on its intended user classes. Some possible characteristics of a stress detection system could include:

- **Real-time:** The system should provide real-time monitoring of the user's stress levels, so that they can take appropriate action if necessary.
- **Accurate:** The system should be accurate in detecting the user's stress levels, and should not provide false alarms or miss instances of high stress.
- **Personalized:** The system should be personalized to the user's individual stress response, and should take into account factors such as age, gender, and lifestyle.
- **User-friendly:** The system should be easy to use and understand, so that users can easily monitor their stress levels and take appropriate action.

3.2 Requirement Identifying Technique

There are several requirement identifying techniques that could be used for a stress detection.

- **Use cases:** Developing use cases that describe the different scenarios in which the stress detection system will be used, and identifying the requirements needed to support these scenarios.
- **Literature review:** Conducting a literature review on stress detection systems to identify best practices, common features, and potential limitations that should be addressed in the design of the system.
- **Benchmarking:** Comparing the stress detection system to similar systems in the market to identify features that are missing or that need to be improved to make the system more competitive.

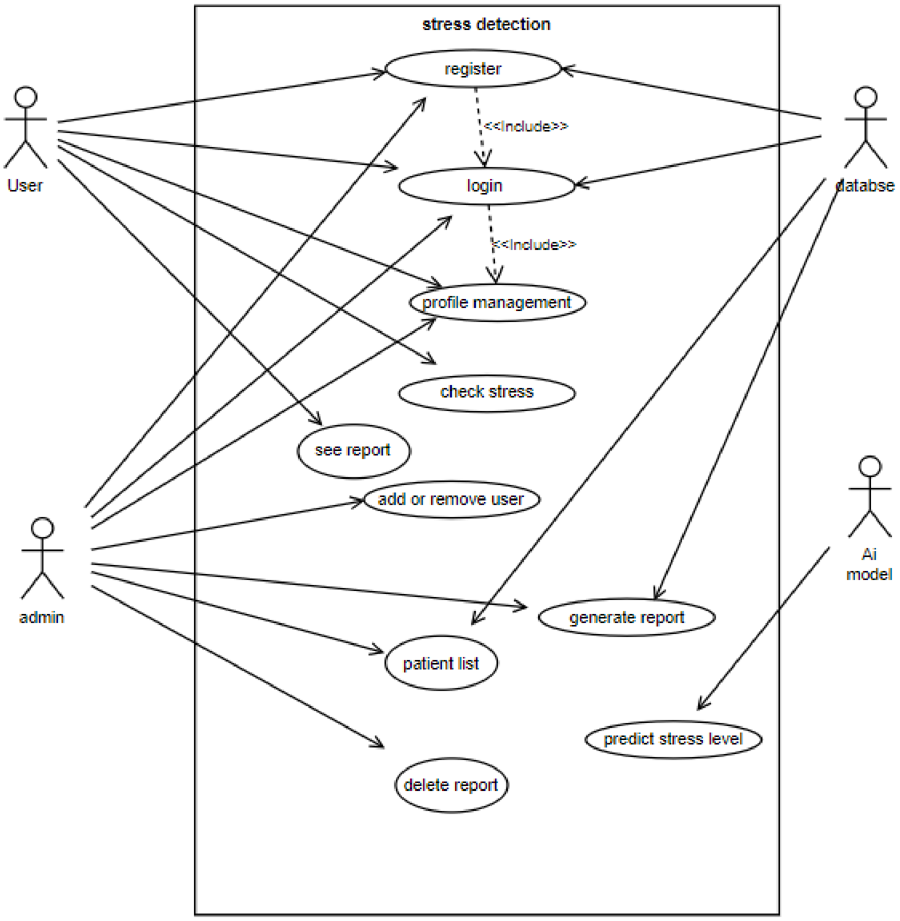


Figure 1: Use case diagram

3.3 Functional Requirements

Functional requirements are the key for a system

- **Process Image**: As the image is in case of both video as well as image as an output.
- **Face Detection:** Detection of face so it will remove the background noise from the image.
- **Expression Detection:** Expression as the model is loaded it will be helpful in detection certain features as the use of CNN layers are concerned.
- **Display Report:** Report is displayed with the level bar.

3.3.1 Login

Table 3: Description for login

Identifier	FR-1
Title	Login
Requirement	Use should have login id and password. If it do not have id password. Then he have to first register itself.
Source	Main step of stress detection
Rationale	User enter its id and password.
Business Rule (if required)	User will first login itself into the app.
Dependencies	
Priority	High

3.3.2 Check Stress

Table 4: Description for check Stress

Identifier	FR-2
Title	Check stress
Requirement	User has to place its face in front of camera.
Source	Main step of stress detection
Rationale	User has to place its face in front of camera.
Business Rule (if required)	User must be in light. So that camera can easily get its picture and our app can easily tell its stress.
Dependencies	FR-1
Priority	High

3.3.3 Detect and measure stress

Table 5: Description for Detect and measure stress

Identifier	FR-1
Title	Detect and measure stress
Requirement	System must be able to detect and measure the stress using psychological behavior and use eyes movement.
Source	Main step of stress detection
Rationale	User places it face in front of camera and through using camera it will detect.
Business Rule (if required)	Detection is main step of our system if the system do not detect stress correctly than it will not be able to produce accurate result.
Dependencies	FR-2
Priority	High

3.3.4 See report

Table 6: Description for See report

Identifier	FR-3
Title	See report
Requirement	After generating report. User can see its report on app.
Source	Main step of stress detection
Rationale	User can see its stress report.
Business Rule (if required)	Through the Ai model system will generate the report.
Dependencies	FR-1 and FR-2
Priority	High

3.3.5 Stress level

Table 7: Description for stress level

Identifier	FR-3
Title	Stress level
Requirement	Through the user answer and psychological behavior. System will be in condition to predict stress level.
Source	Main step of stress detection
Rationale	User will get its stress level.
Business Rule (if required)	Through the user answer and psychological behavior. System will be in condition to predict stress level.
Dependencies	FR-1 and FR-2
Priority	High

3.4 Non-Functional Requirements

- The system must be user-friendly and intuitive to use.
- The system must be accurate and reliable in detecting stress levels.
- The system must be accessible to users with disabilities.
- The system must be responsive and perform well under varying conditions.
- The system must comply with privacy and security regulations and best practices.
- The system must be scalable and able to handle increasing user loads.
- The system must be compatible with multiple devices and platforms.

3.4.1 Usability

Our System will be user-friendly, so that the user will not face any difficulty while using our application.

3.4.2 Performance

The performance of our system will be that it can handle a certain number of user within a specified time frame.

3.5 External Interface Requirements

3.5.1 User Interfaces Requirements

- **User-Friendly:** User interface should be easy to use, with clear navigation, understandable instruction and clear button
- **Consistent design:** The app should have consistent design across all pages and screen to provide better user experience

- **Responsive design:** The app should be design to work well on different devices such as smart phone, tablet
- **Performance:** The user interface must be responsive and provide fast feedback to user
- **Security:** The user interface must provide appropriate security measure to protect user data and prevent unauthorized access

3.5.2 Software Interface

- **Database:** we are using excel sheet as our database in the project.

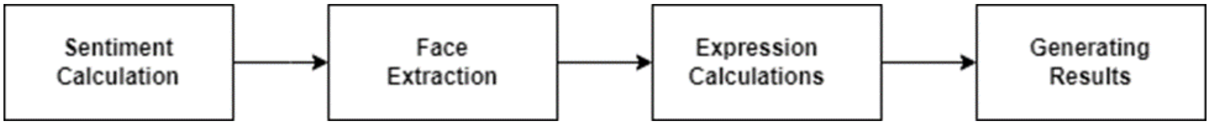
CHAPTER: 4
Design and Architecture

4 Design and Architecture

4.1 Architectural Design

4.1.1 System Architecture

Following is the simple line-box diagram consisting major features of the model architect:



The simple Line-Box-Diagram showing features calculated in this model are sentiment calculation, generating results according to the requirements, face extraction, and expressions calculations.

4.1.2 Data flow diagram:

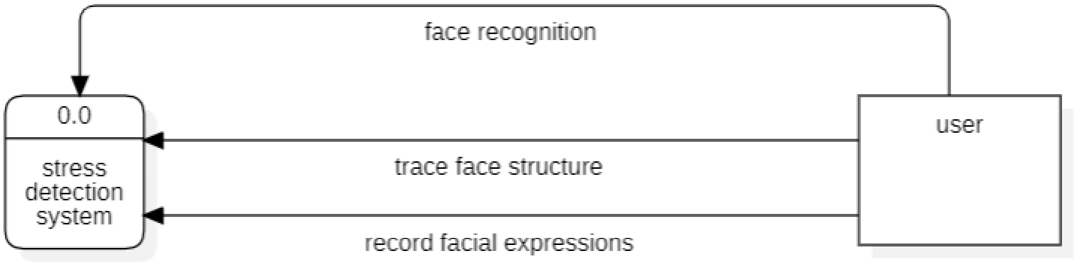


Figure 2: Data flow diagram level 0

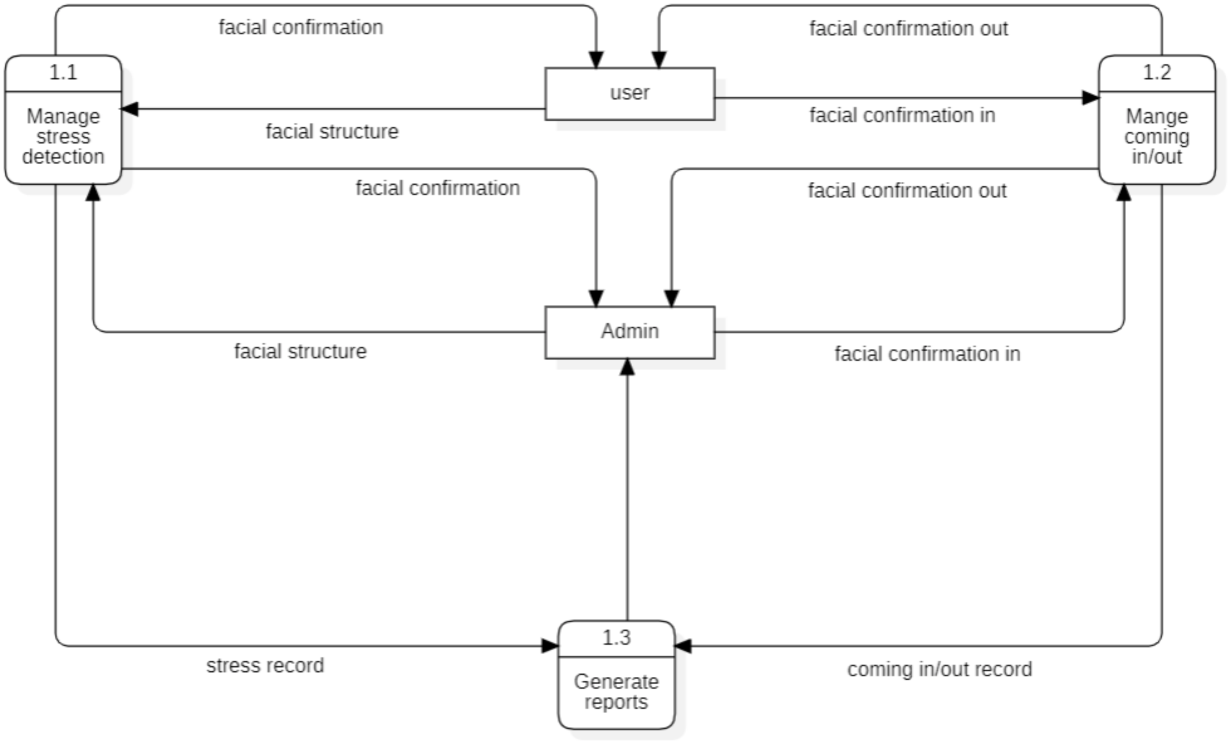


Figure 3: Data flow diagram level 1

A Level 1 Data Flow Diagram (DFD) provides a detailed view of a system and breaks down the main process into sub-processes from the Level 0 DFD. It describes the flow of data between processes, external entities, and data storage with labeled arrows. This diagram includes rectangles for processes, open rectangles for data storage, rounded rectangles for terminals, and notes for clarity.

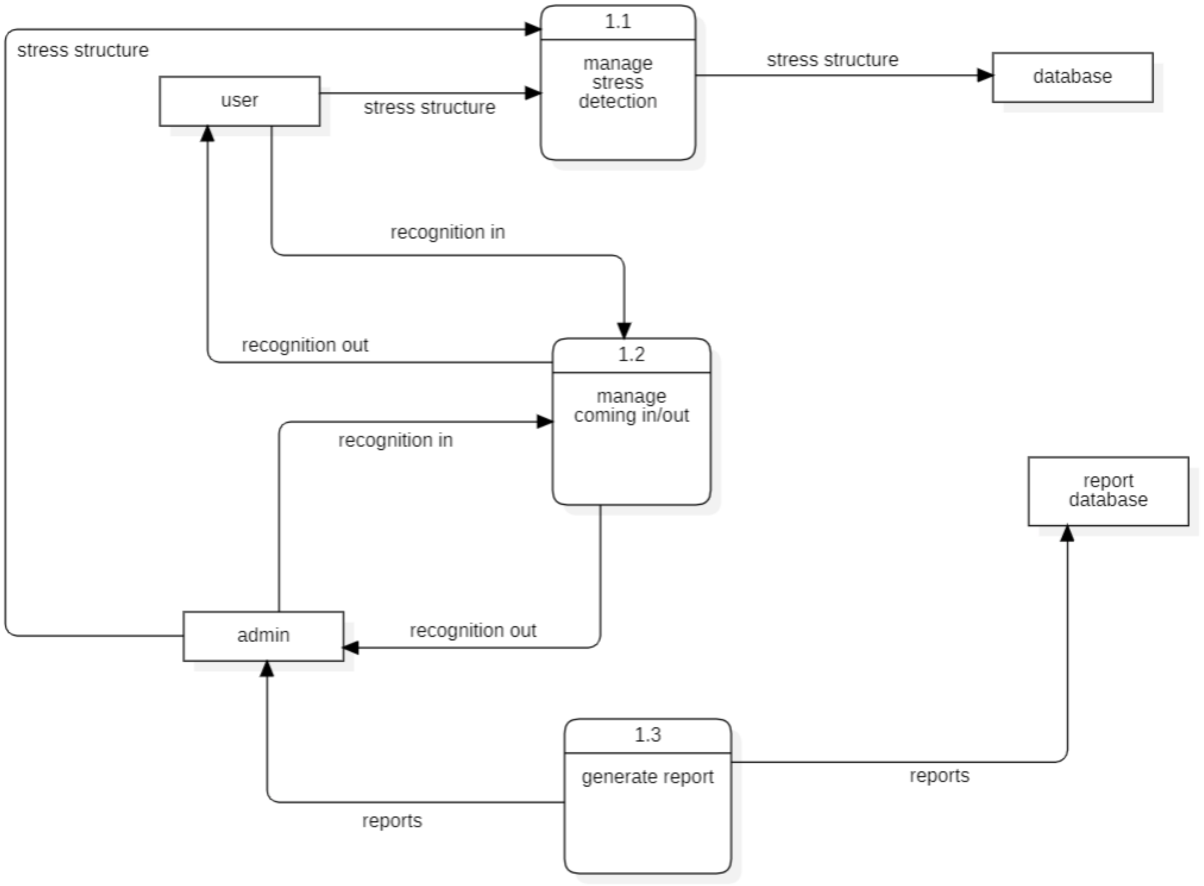


Figure 4: Data flow diagram level 2

Data flow diagrams provide a visual representation of systems and processes that are challenging to explain verbally. These diagrams can be used to plan out a new system for implementation or to map out an existing system and improve it. This is a general data flow diagram, it consists user and user can login, check the stress and see the generate report. The admin will login, and see the user list and delete and update the data.

4.1.3 Process Flow:

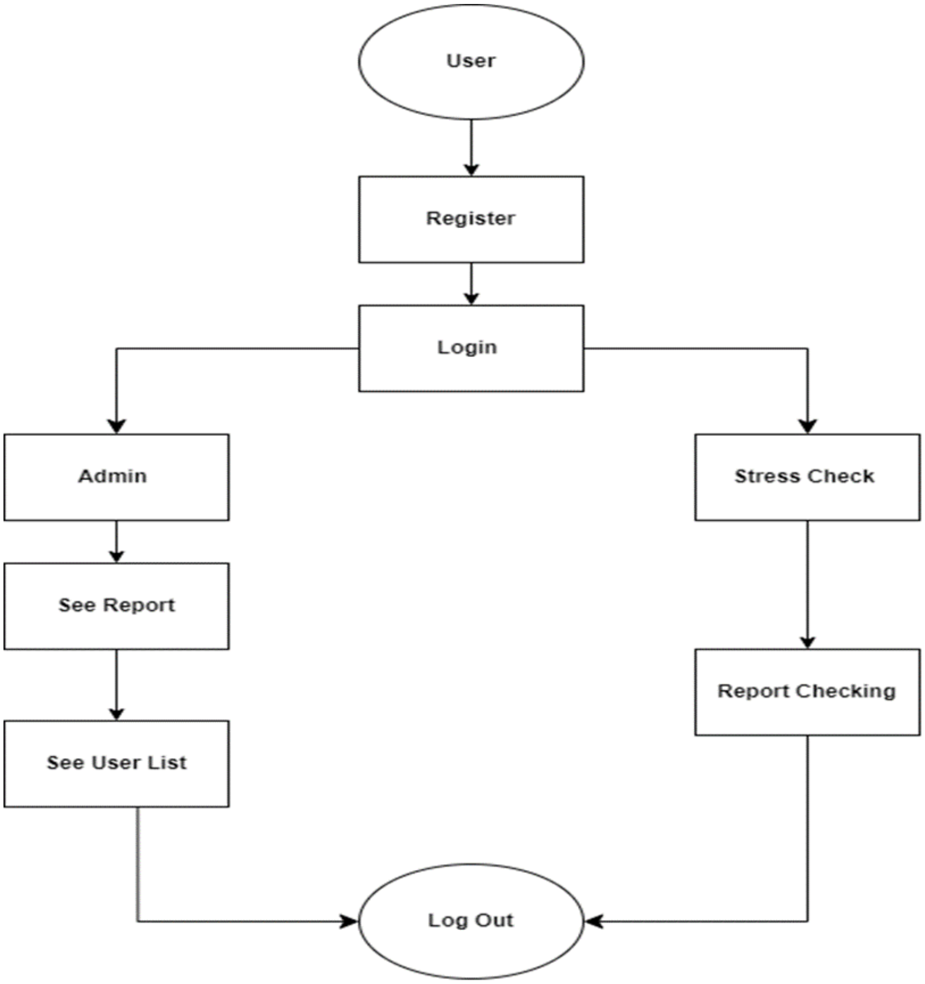


Figure 5: Process Flow

Process Flow or representation of general flow is quite useful while understanding the main working of the application. Hence general flow is useful because it ensure that the system will perform as intended, user will be able to understand and recognize the linkages between complicated systems, it will be used to verify that the design requirements are still met in various operating scenarios, user will be able to work along the system dynamics of surge-related and non-surge-related transient operations are evaluated, the planned future system changes and expansions will be simple, clients will get the tools they need to understand their systems better, troubleshoot efficiently, and assess the effects of changes

4.2 Design Models for Object Oriented Development Approach

4.2.1 Activity Diagram

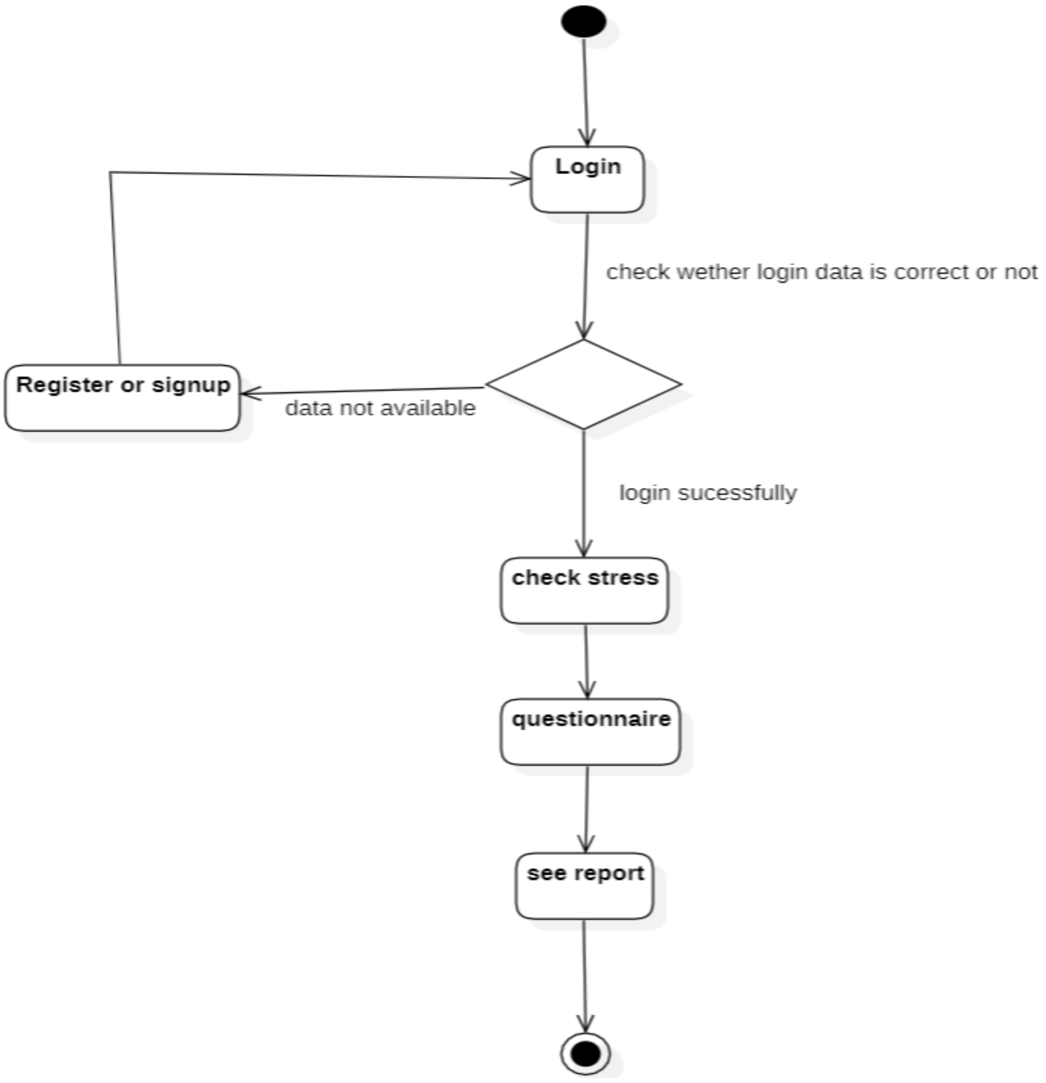


Figure 6: Activity Diagram

4.2.2 Class Diagram

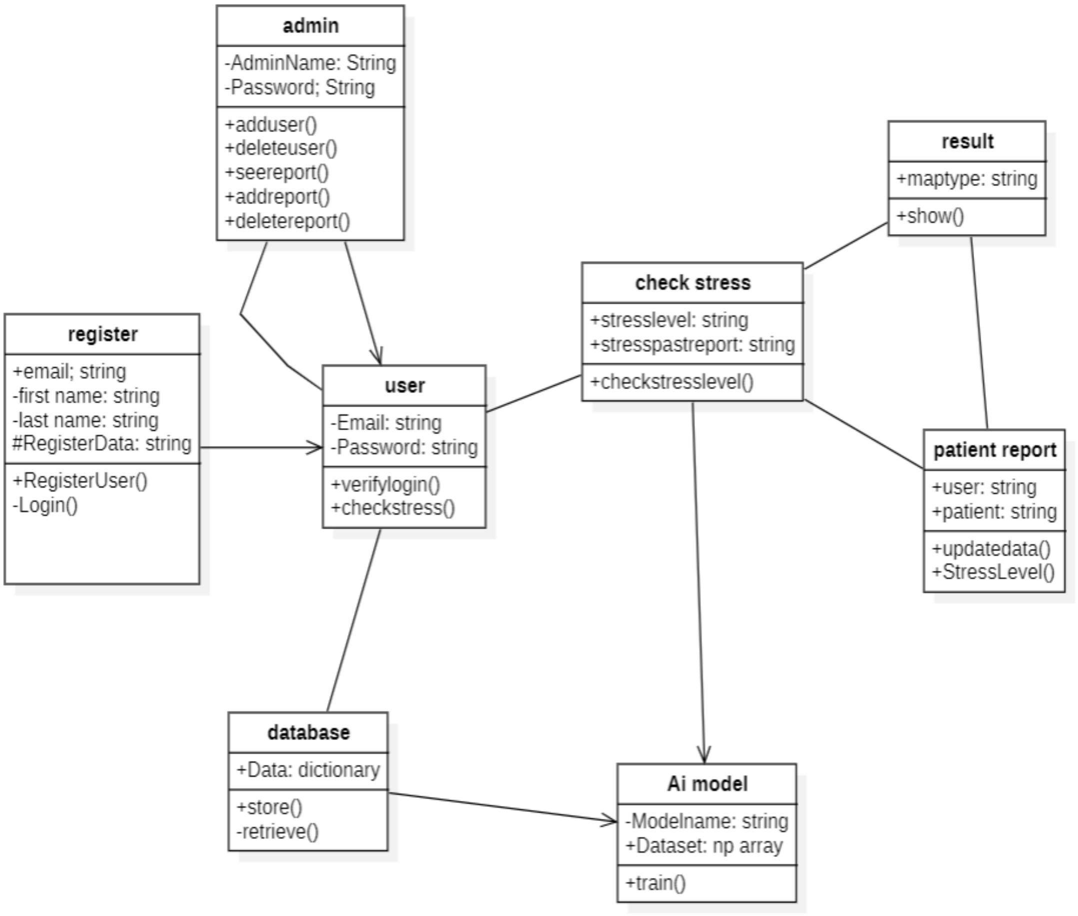


Figure 7: Class Diagram

4.2.3 Sequence Diagram

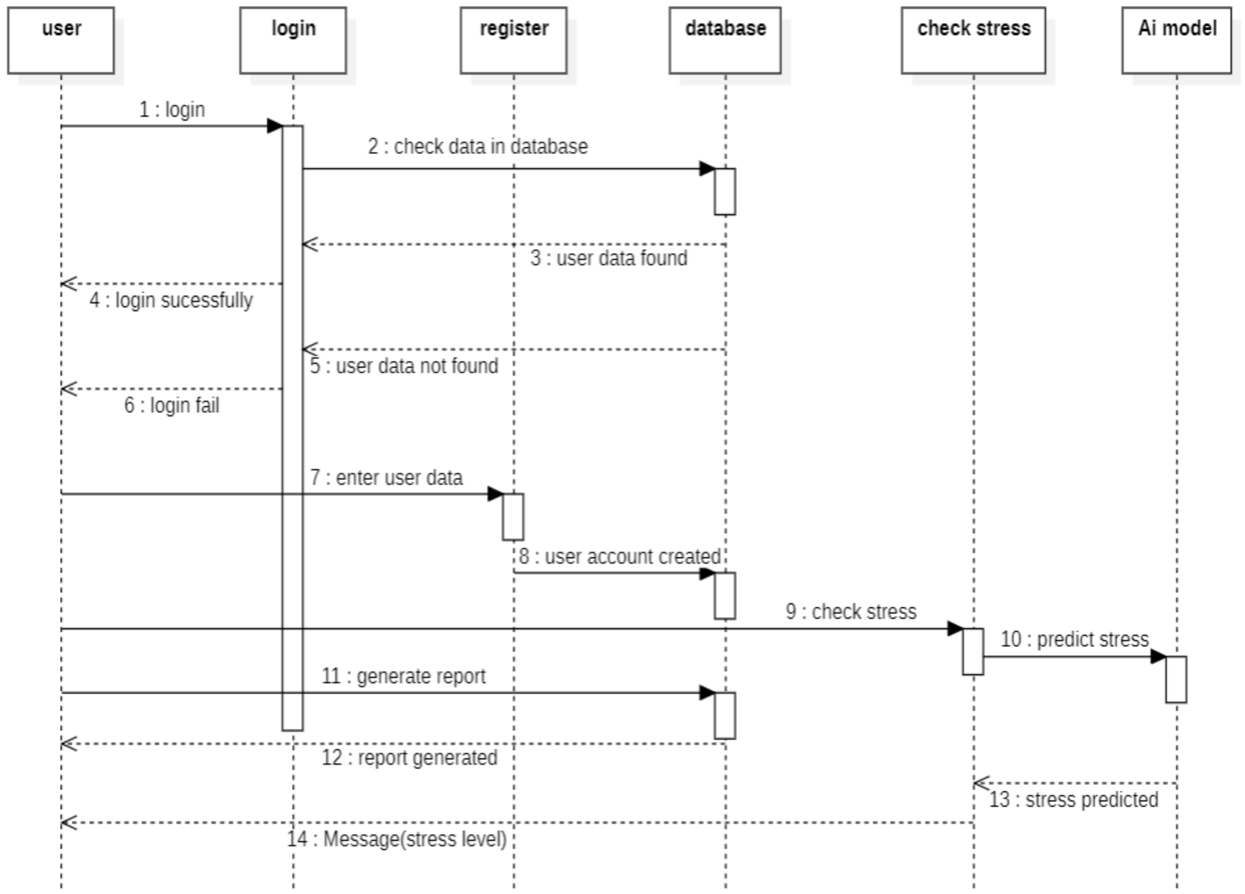


Figure 8: Sequence Diagram

4.2.4 State Transition Diagram

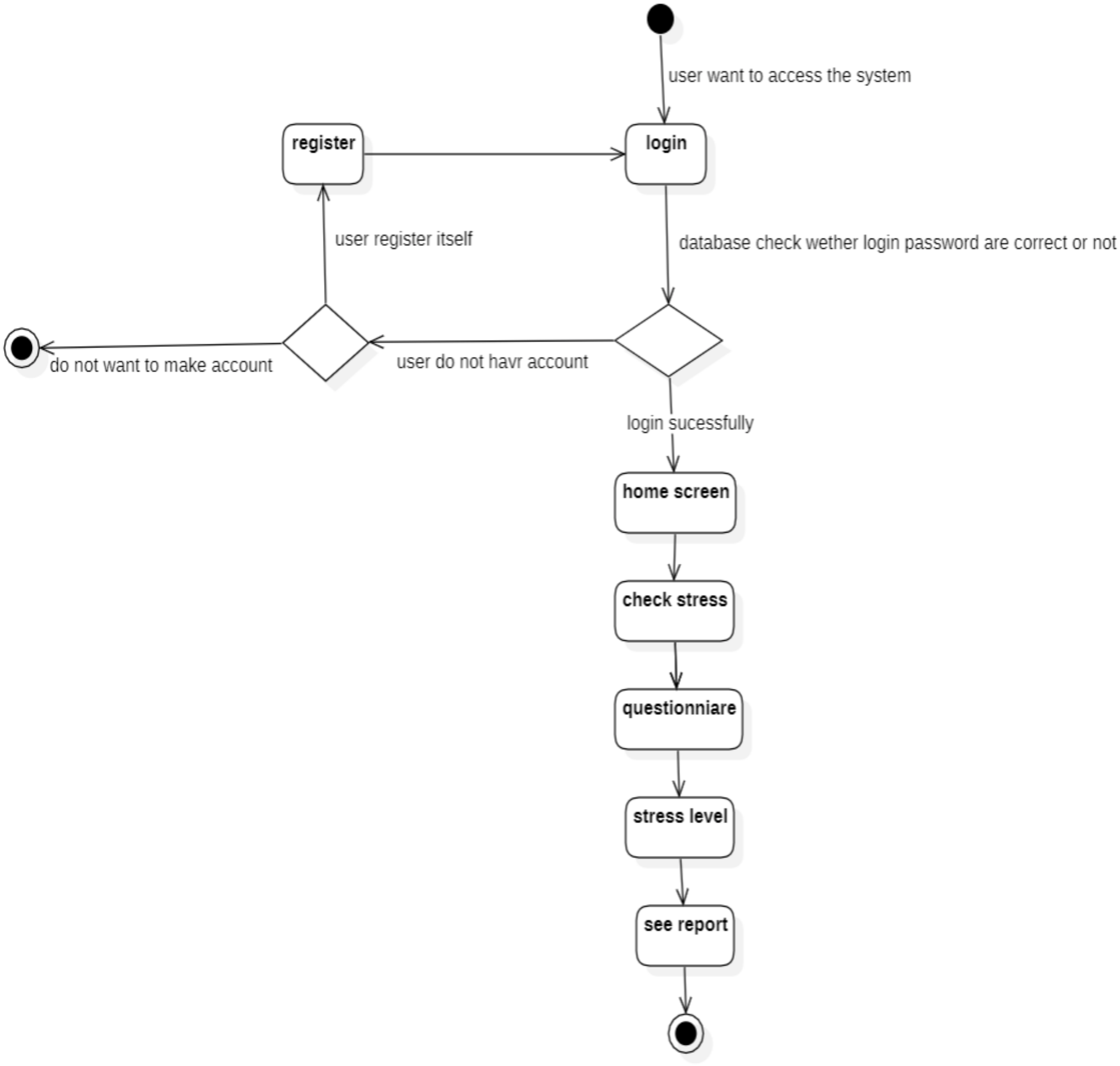


Figure 9: State Transition Diagram

4.3 Data Design

4.3.1 Data Dictionary

Login

In login we are going to use variable

1. **Email:** email used to store user email using use state
2. **Password:** password used to store user password using use state
3. **Click me:** function to authenticate user
4. **Button:** To authenticate and navigate user

Register

In Sign up we are going to use variable

1. **Email:** user enter its email.
2. **Password:** user enter its password.
3. **Register:** then user account will created.
4. **Button:** for navigation

Menu

In menu, we have options of patient list, check stress, patient report and log out button

1. **Patient list:** Admin can check patient list.
2. **Check stress:** User can check stress.
3. **Patient report:** patient can see its report.
4. **Log out:** user can easily logout.

Check Stress

In this user can easily check its stress.

1. **Open camera:** user can open camera and check
2. **Stress level**: it display stress level.

4.4 Human Interface Design

Login screen

On this Screen user will write email and password and login himself. The login page is same for both patients and admin. If the email or password is not correct then system will give notification of email or password not correct. New user can register himself by going on register page through register button.

.

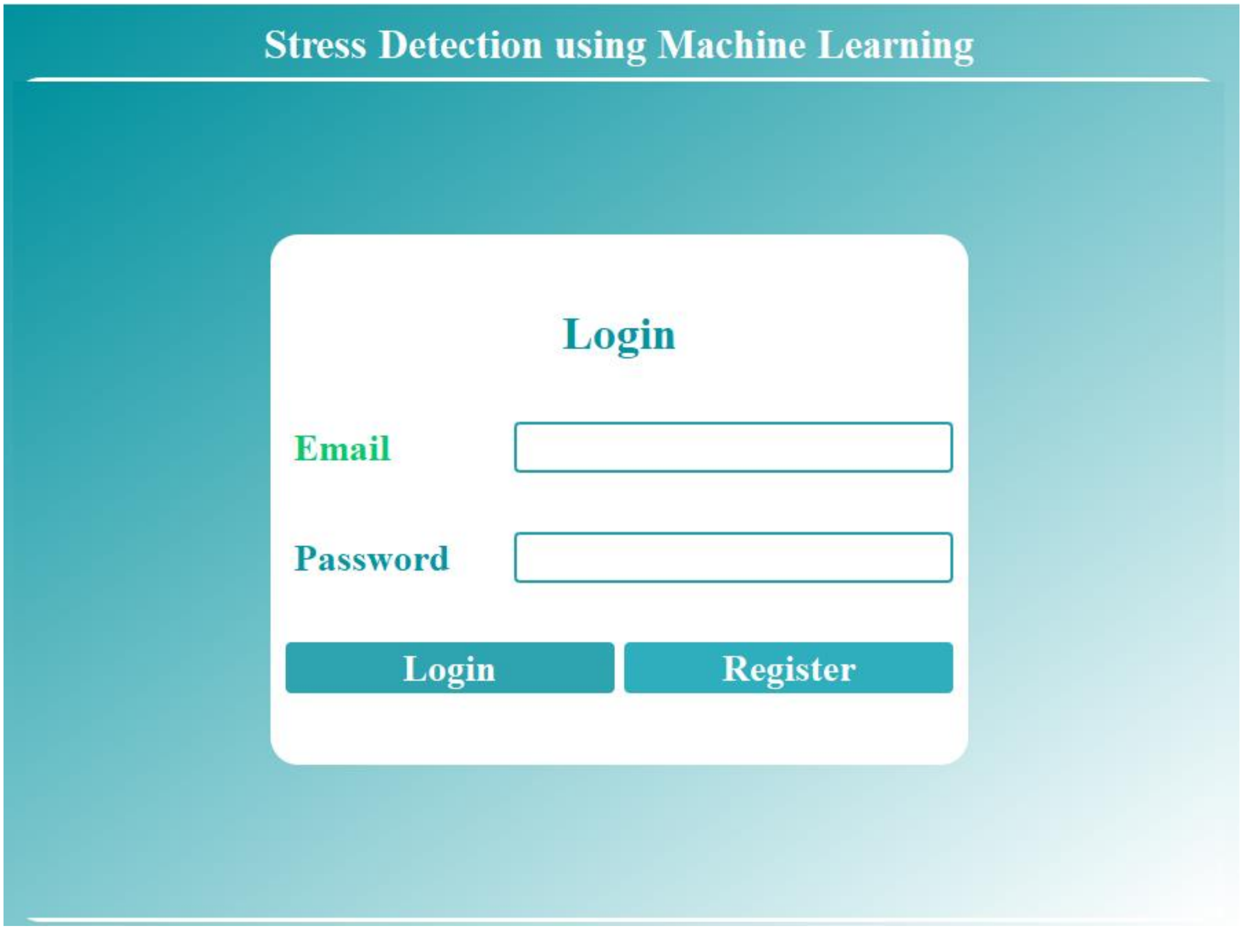
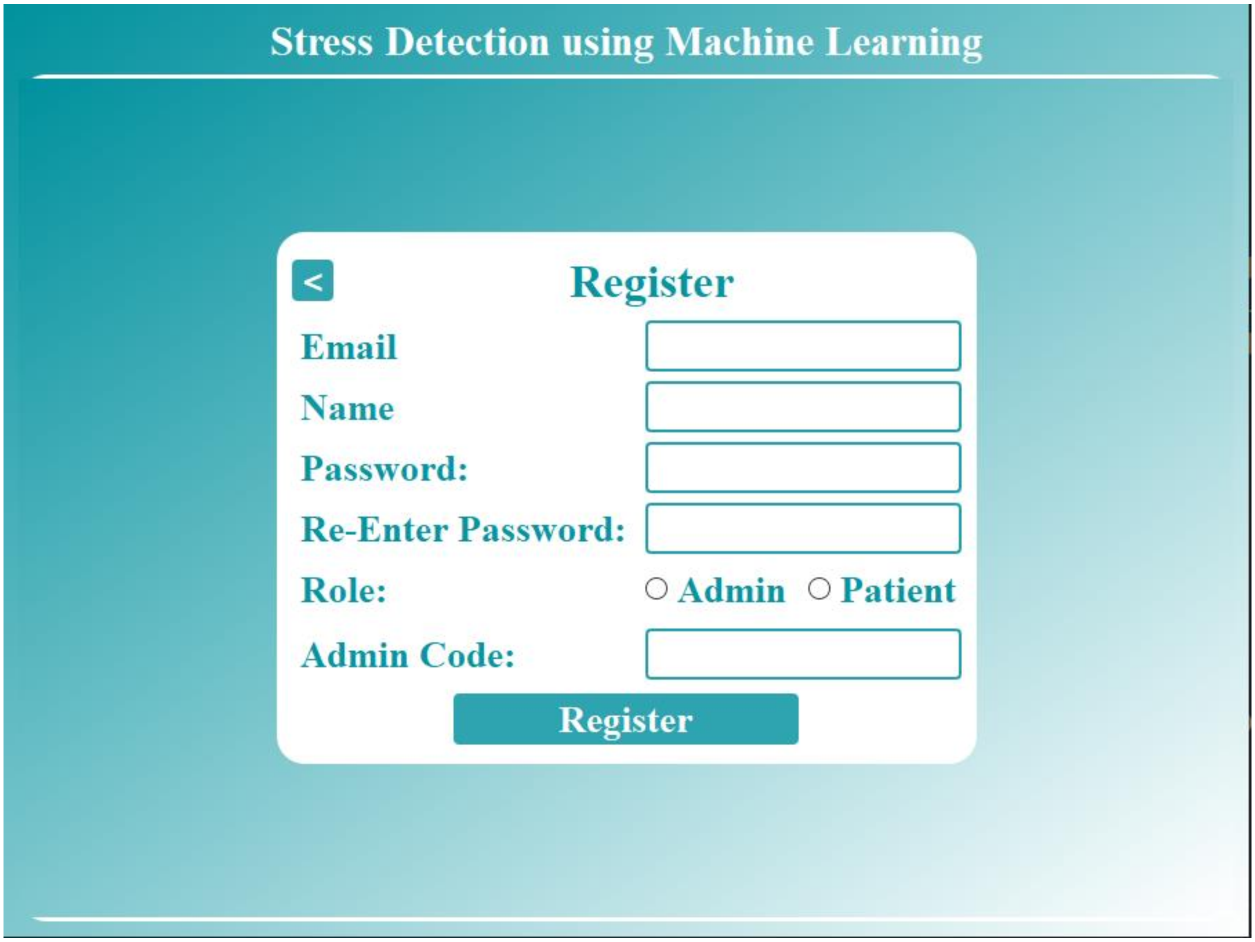


Figure 10: Login Screen

Register Screen

On this Screen user can sign up or register himself. We have created a code for admin. If the admin want to register himself then he have to write admin code. Patients can simply register themselves



Stress Detection using Machine Learning

< Register

Email

Name

Password:

Re-Enter Password:

Role: ○ Admin ○ Patient

Admin Code:

Register

Figure 11: Sign up Screen

Menu screen

On the Home Screen user can see different options. User can perform different operations. Patient list is available to only admins. Patients cannot get this option. Patient can check its stress by clicking check stress button and see its report by clicking patient report button. User can logout himself by clicking logout button.

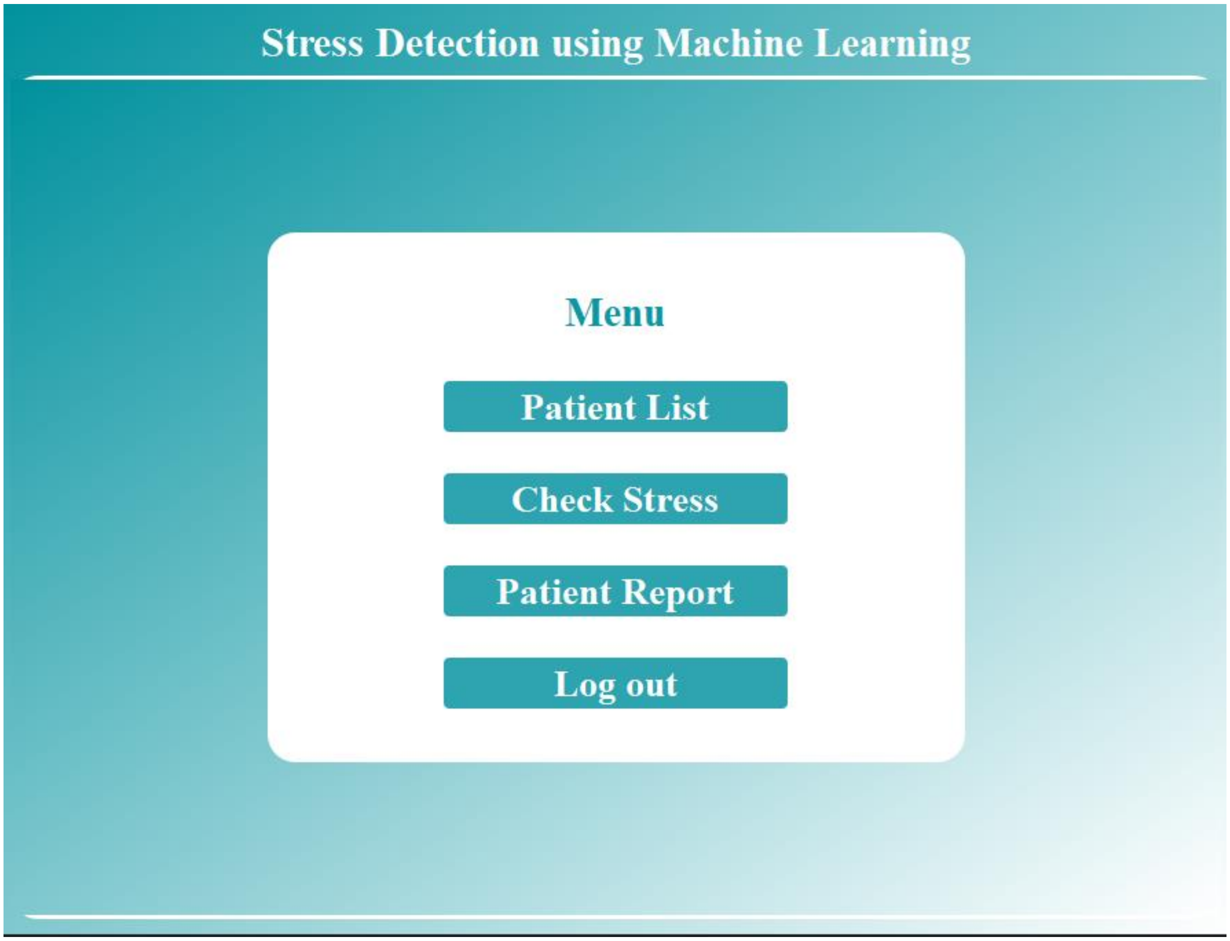


Figure 12: Home screen

Check Stress

On this screen user will come in front of camera so application can check stress. User have to click start button to start camera. Patient have to be in front of camera so that system can get information about his condition. Stress level monitor will show stress level.

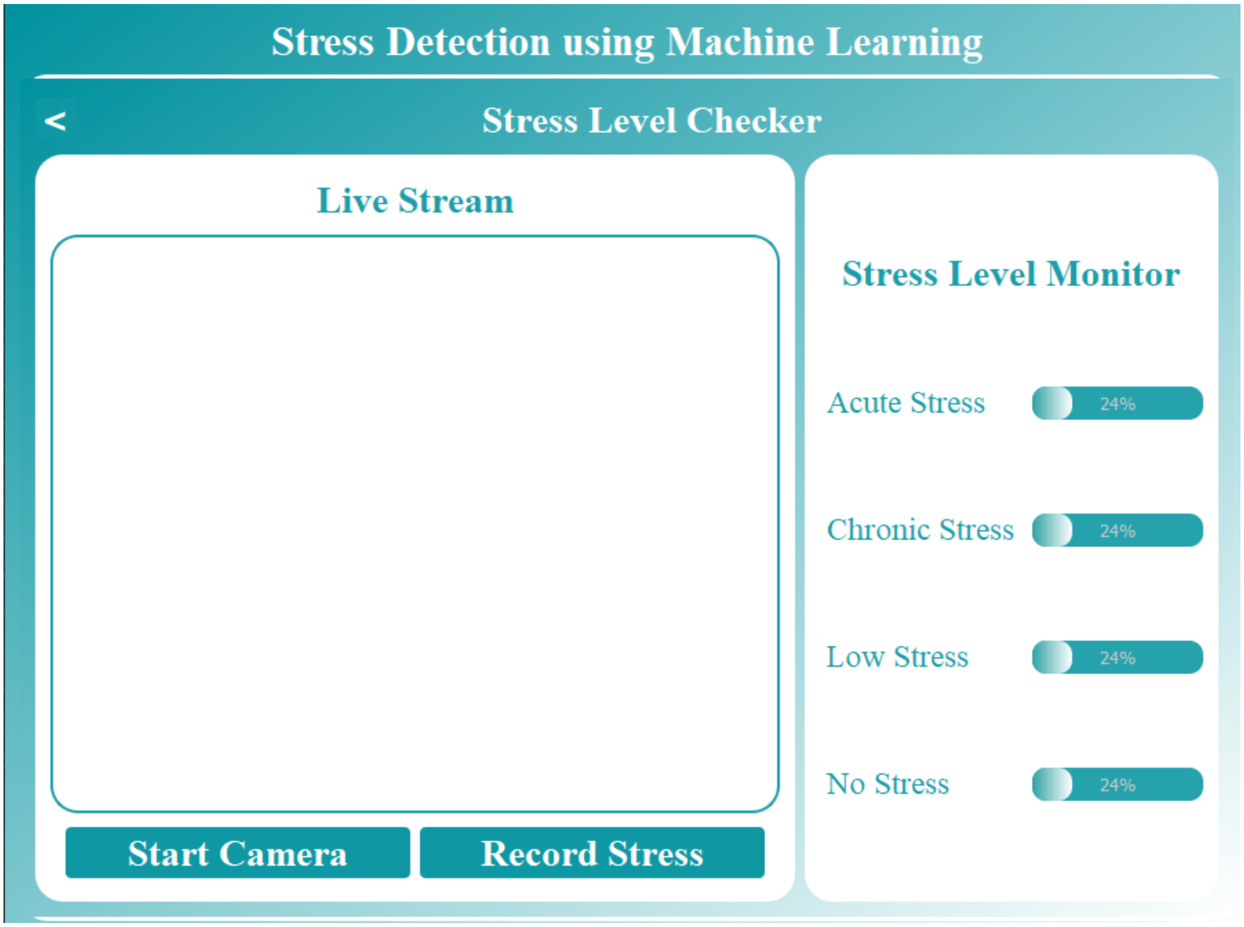


Figure 13: Check Stress

4.4.1 Screen Objects and Actions

Screen object

- ✓ **Login:** User can login into system.
- ✓ **Register:** new user can register itself.
- ✓ **Menu:** Menu screen display different functions.
- ✓ **Check stress:** User can check stress on run time.
- ✓ **See report:** User can see report.

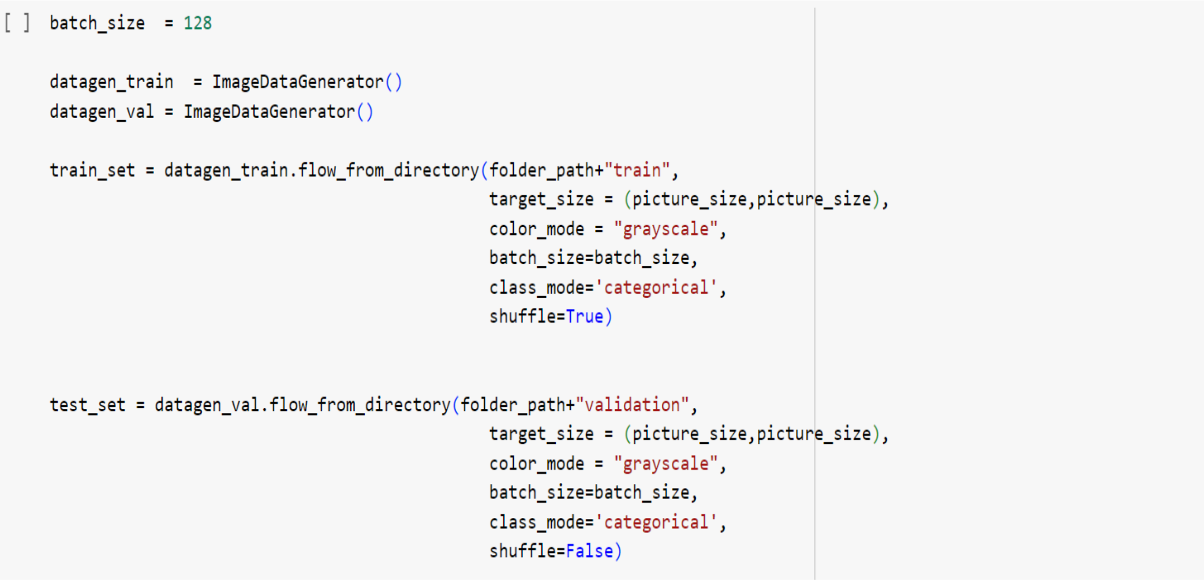
Action

- ✓ **Stress detection:** user can check its stress at run time.

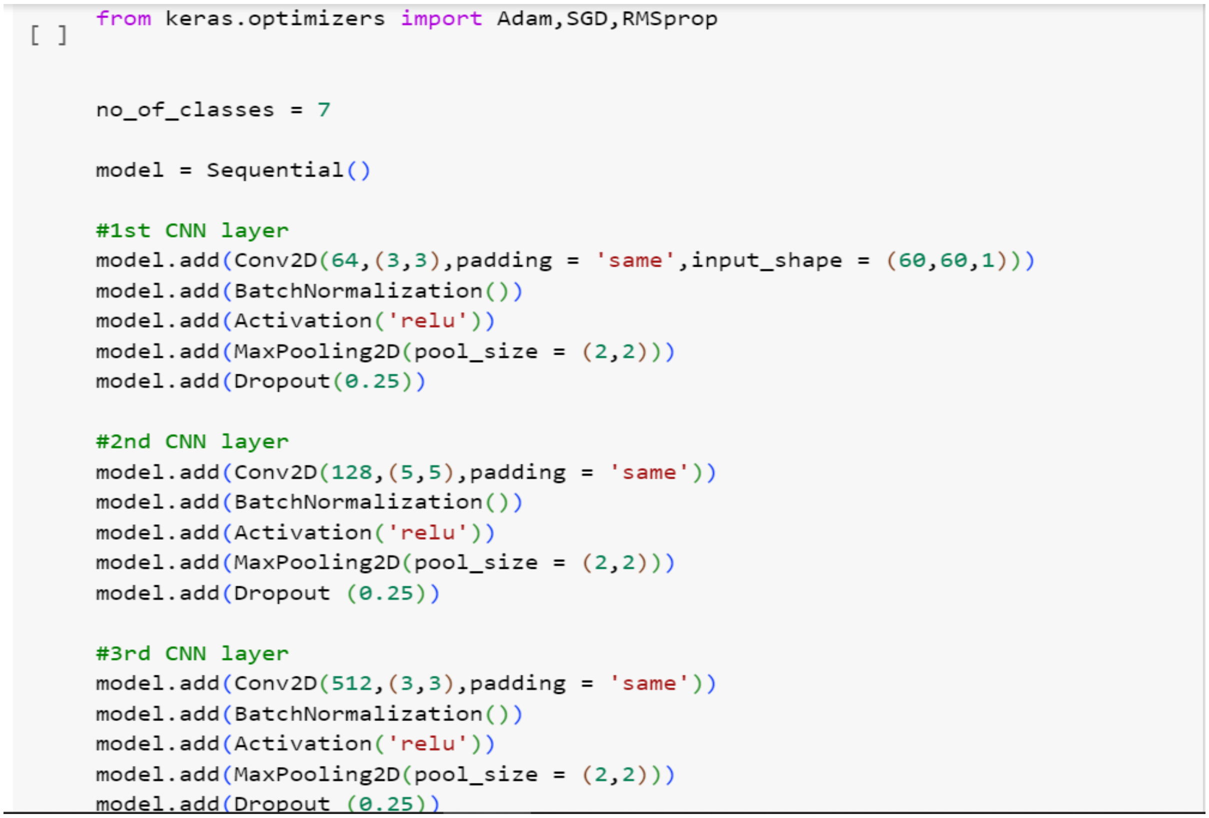
CHAPTER: 5
Implementation

5 Implementation

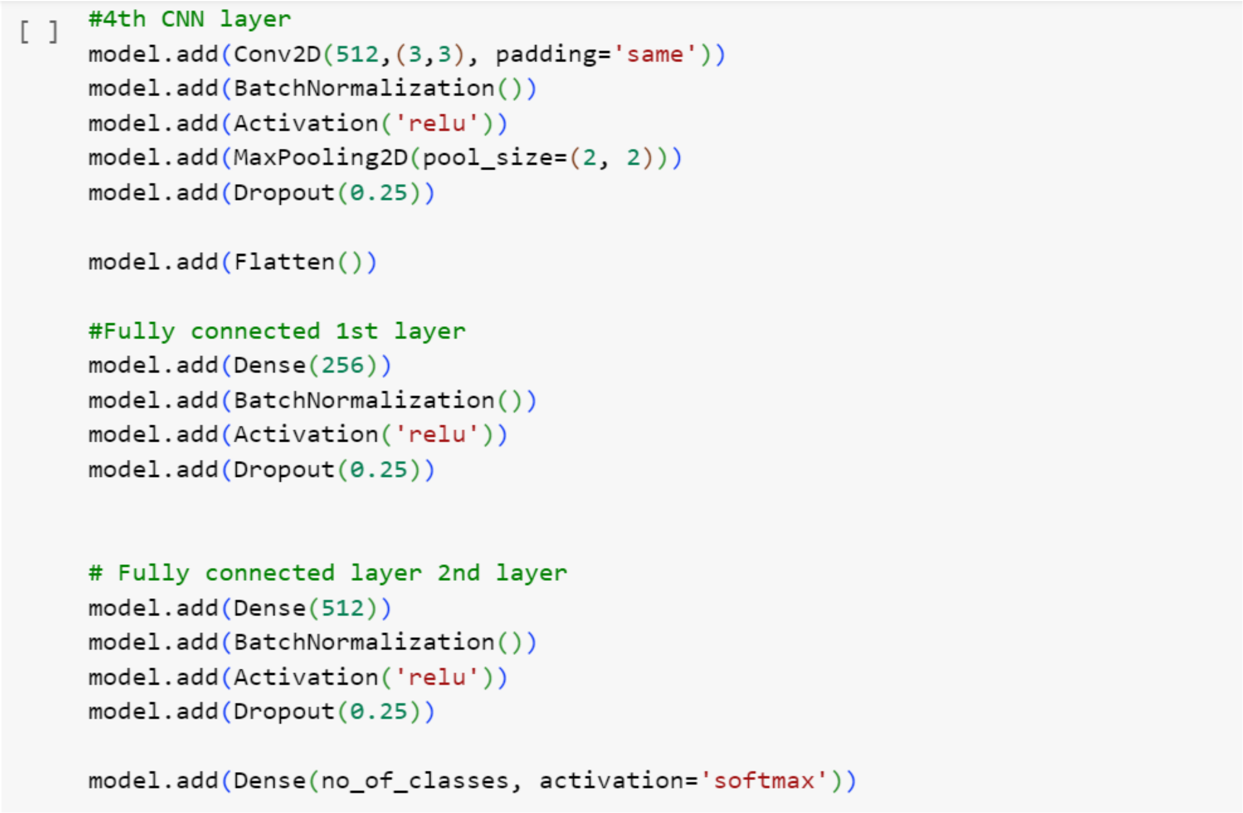
5.1 Algorithm



```
[ ] batch_size  = 128

    datagen_train  = ImageDataGenerator()
    datagen_val = ImageDataGenerator()

    train_set = datagen_train.flow_from_directory(folder_path+"train",
                                                  target_size = (picture_size,picture_size),
                                                  color_mode = "grayscale",
                                                  batch_size=batch_size,
                                                  class_mode='categorical',
                                                  shuffle=True)


    test_set = datagen_val.flow_from_directory(folder_path+"validation",
                                                  target_size = (picture_size,picture_size),
                                                  color_mode = "grayscale",
                                                  batch_size=batch_size,
                                                  class_mode='categorical',
                                                  shuffle=False)
```



```
     from keras.optimizers import Adam,SGD,RMSprop
[ ]

     no_of_classes = 7

     model = Sequential()

     #1st CNN layer
     model.add(Conv2D(64,(3,3),padding = 'same',input_shape = (60,60,1)))
     model.add(BatchNormalization())
     model.add(Activation('relu'))
     model.add(MaxPooling2D(pool_size = (2,2)))
     model.add(Dropout(0.25))

     #2nd CNN layer
     model.add(Conv2D(128,(5,5),padding = 'same'))
     model.add(BatchNormalization())
     model.add(Activation('relu'))
     model.add(MaxPooling2D(pool_size = (2,2)))
     model.add(Dropout (0.25))

     #3rd CNN layer
     model.add(Conv2D(512,(3,3),padding = 'same'))
     model.add(BatchNormalization())
     model.add(Activation('relu'))
     model.add(MaxPooling2D(pool_size = (2,2)))
     model.add(Dropout (0.25))
```



```
#4th CNN layer
model.add(Conv2D(512,(3,3), padding='same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Flatten())

#Fully connected 1st layer
model.add(Dense(256))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))


# Fully connected layer 2nd layer
model.add(Dense(512))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))

model.add(Dense(no_of_classes, activation='softmax'))
```

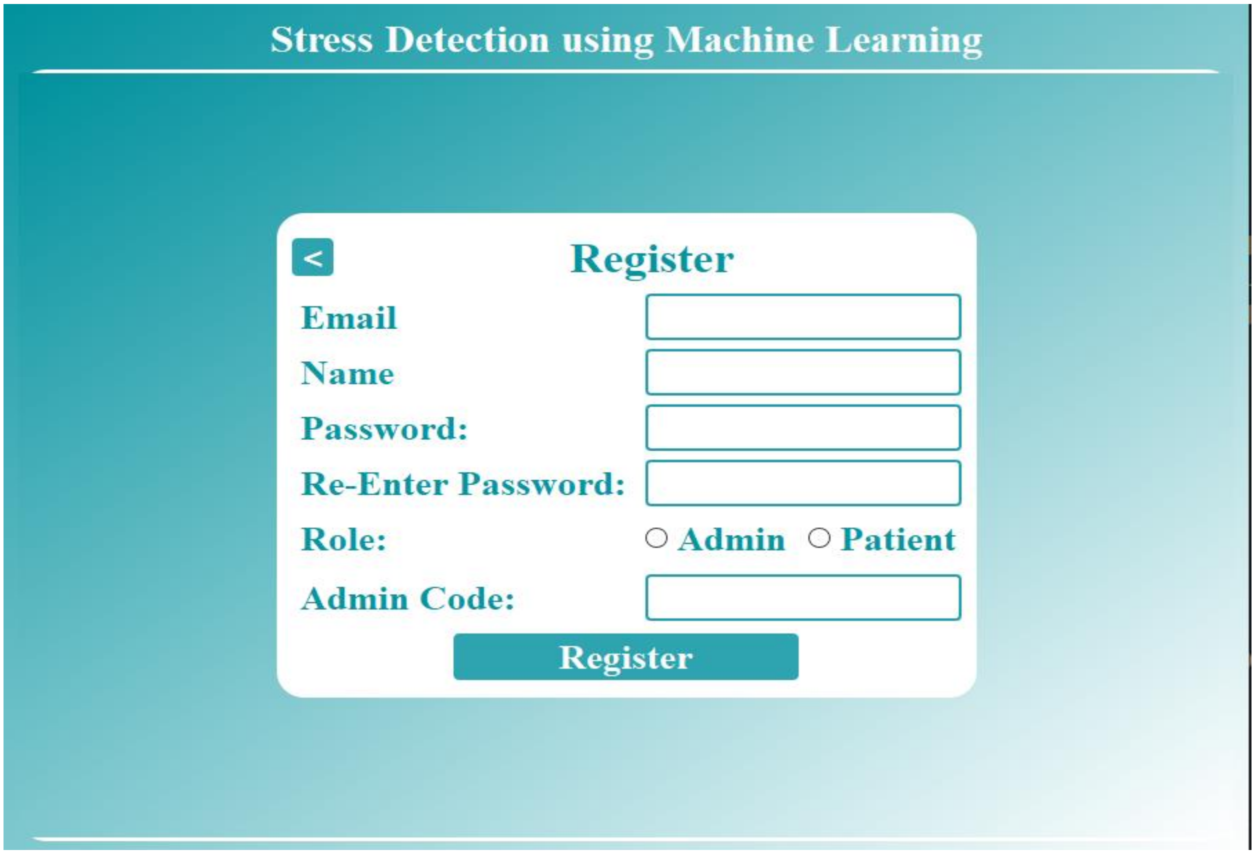
5.2 User Interface

The implementation phase involves the implementation of the project plan. Implementing the system requires back-end coding, front-end interfaces, and their connections in order for the system to run properly. This section describes all the backend coding needed to build this project. It also contains the system's graphical user interface (GUI). In this phase, all functions are integrated to form a meaningful diagram. In User Interface, there are different activities for users which were given below.

As, there will be login required for the users to utilize this application as first they register in the app and then login and use this app. So, users just launch the application and then login in app and goes to the home screen related to it.

5.2.1 Registration Screen

On this Screen user can sign up or register himself. We have created a code for admin. If the admin want to register himself then he have to write admin code. Patients can simply register themselves.



Stress Detection using Machine Learning

Register

Email

Name

Password:

Re-Enter Password:

Role: ○ Admin ○ Patient

Admin Code:

Figure 14: Registration screen

5.2.2 Login Screen

On this Screen user will write email and password and login himself. The login page is same for both patients and admin. If the email or password is not correct then system will give notification of email or password not correct. New user can register himself by going on register page through register button.

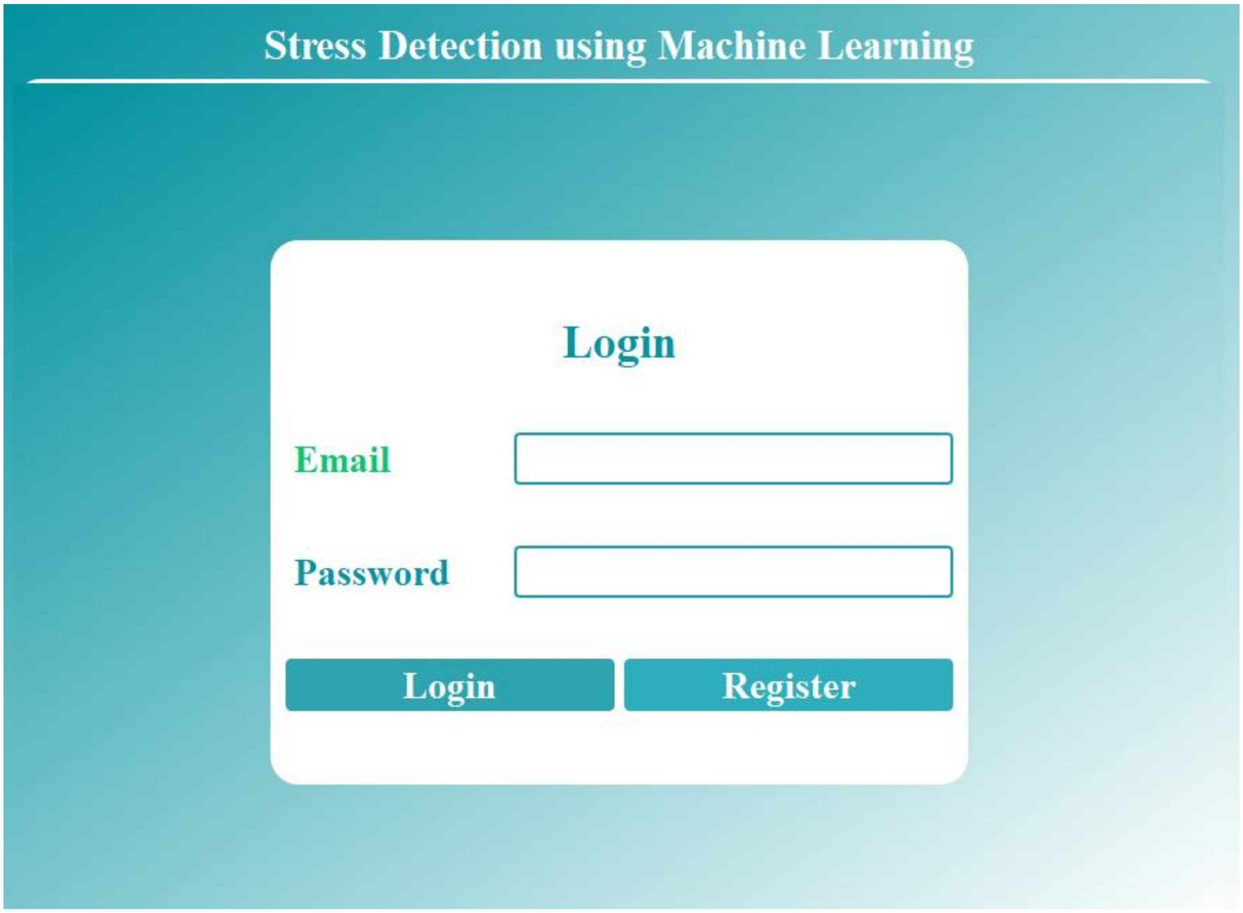


Figure 15: Login Screen

5.2.3 Home Screen

On the Home Screen user can see different options. User can perform different operations. Patient list is available to only admins. Patients cannot get this option. Patient can check its stress by clicking check stress button and see its report by clicking patient report button. User can logout himself by clicking logout button.

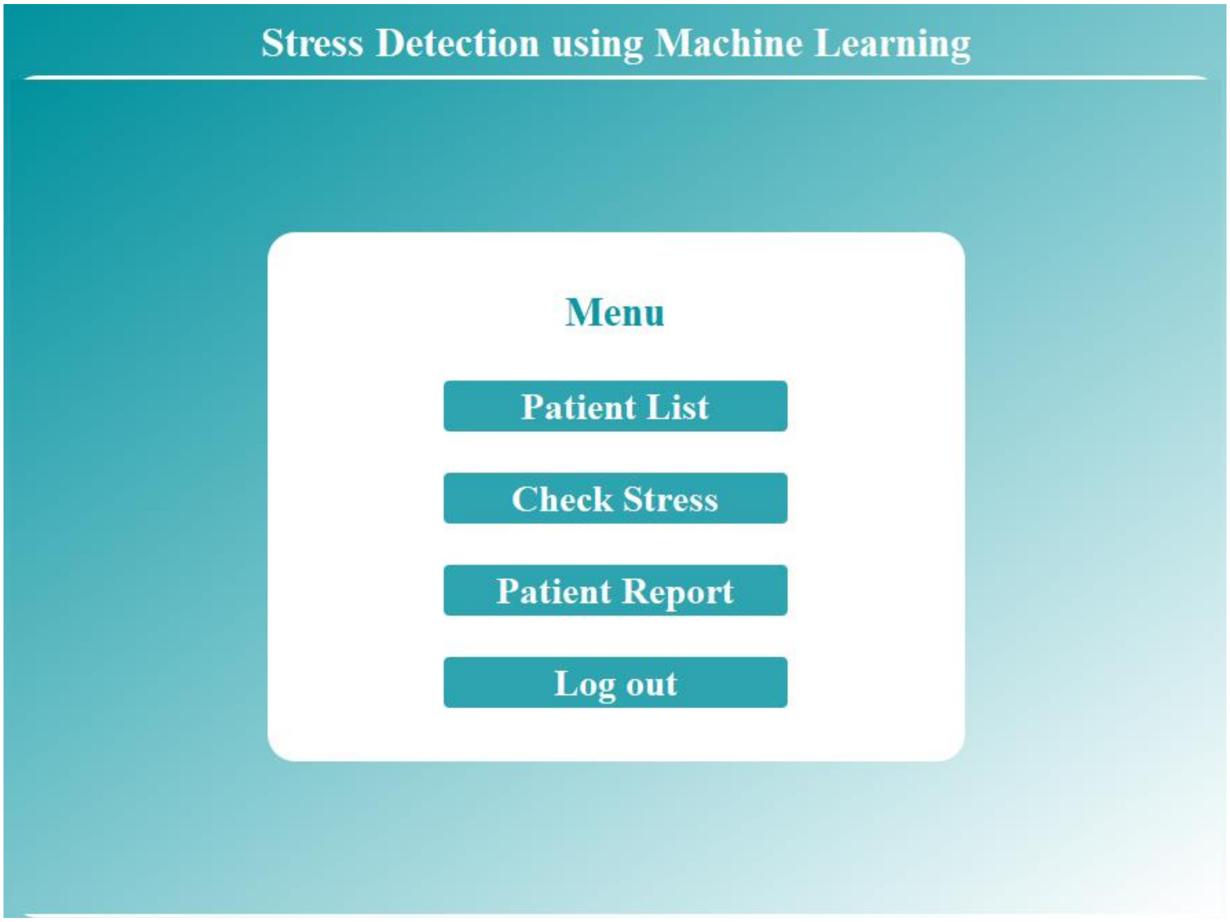
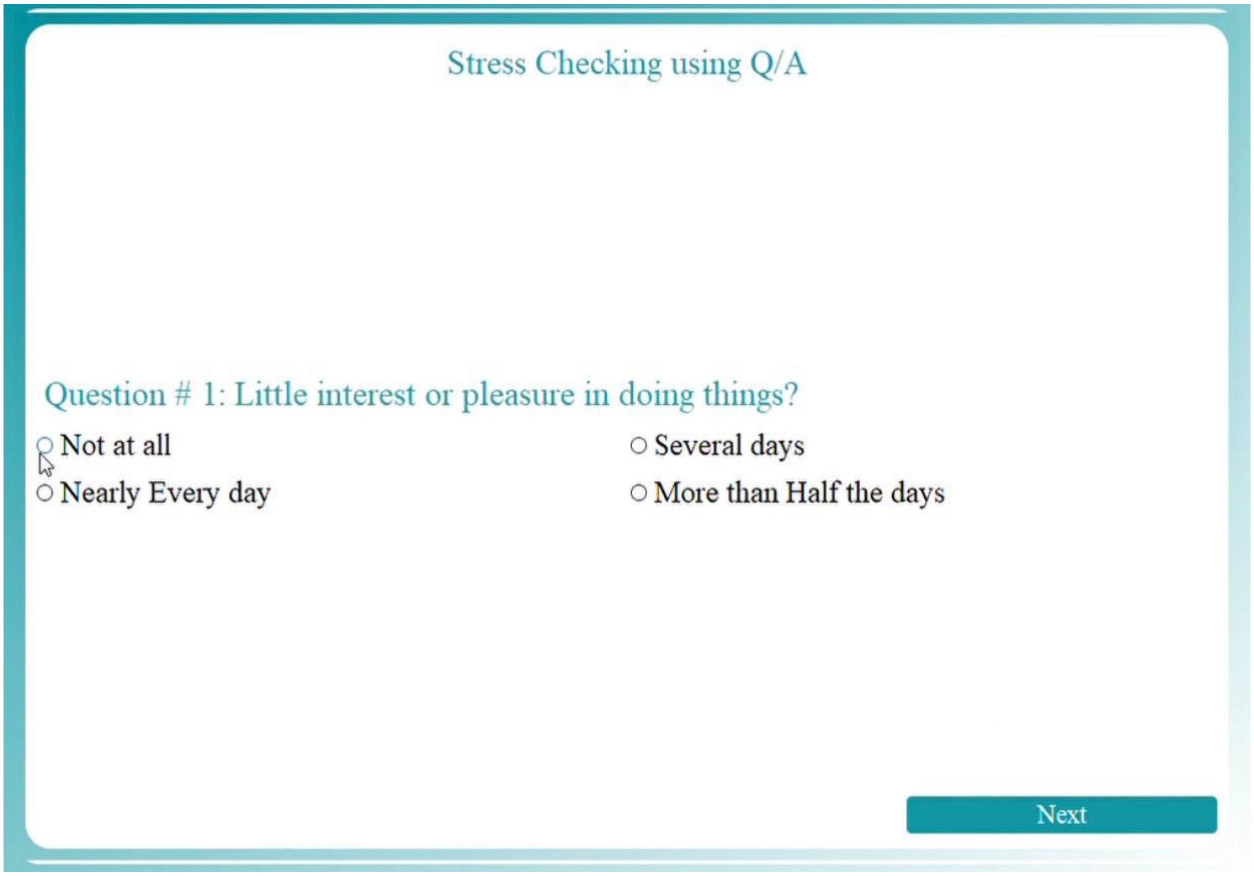


Figure 16: Home Screen

5.2.4 Questionnaire

On this screen patient have to answer the question so that system can guess its stress according to it. System will give 17 questions to the patient which he have to answer. On the bases of these answers and patient face, system will predict patient stress.



Stress Checking using Q/A

Question # 1: Little interest or pleasure in doing things?

○ Not at all ○ Several days

○ Nearly Every day ○ More than Half the days

Figure 17: Questionare

5.2.5 Stress level checker

On this screen user will come in front of camera so application can check stress. User have to click start button to start camera. Patient have to be in front of camera so that system can get information about his condition. Stress level monitor will show stress level.

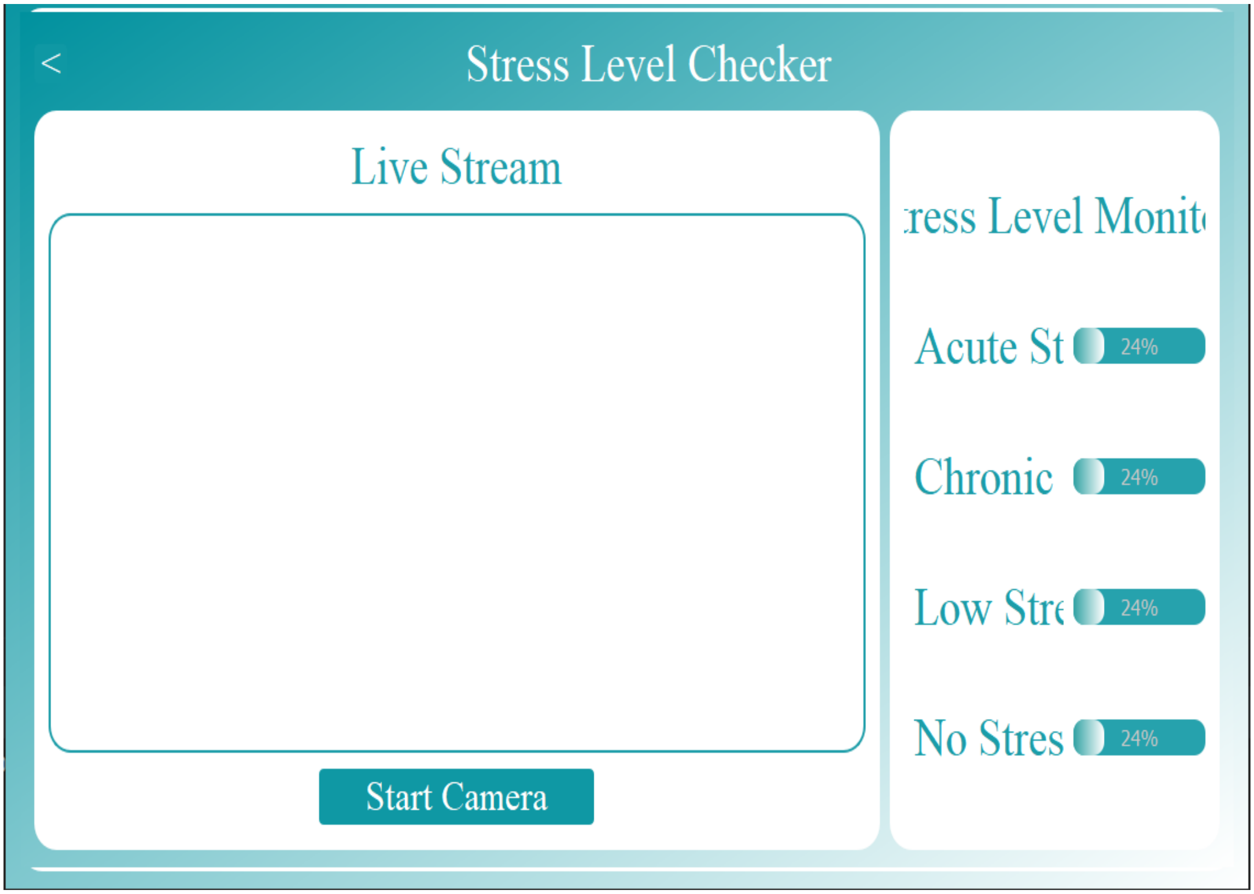
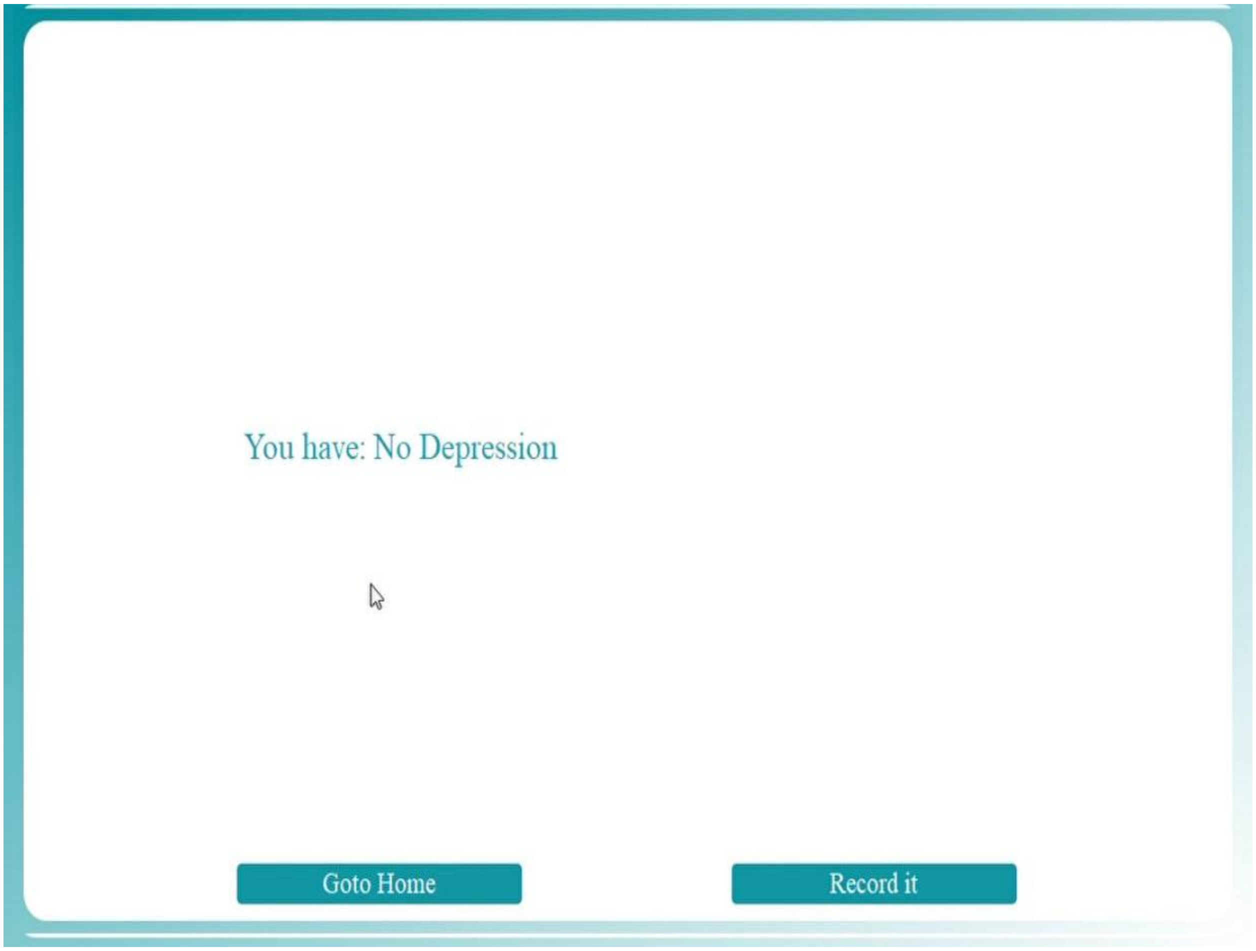


Figure 18: Stress level checker

5.2.6 Patient stress report

On this screen user will see its stress report. After getting information of face features of patient and questionnaire. System will tell patient stress level which will show on this page. Patient can record its report by clicking on record it button and can go to menu screen by clicking on go to home button.

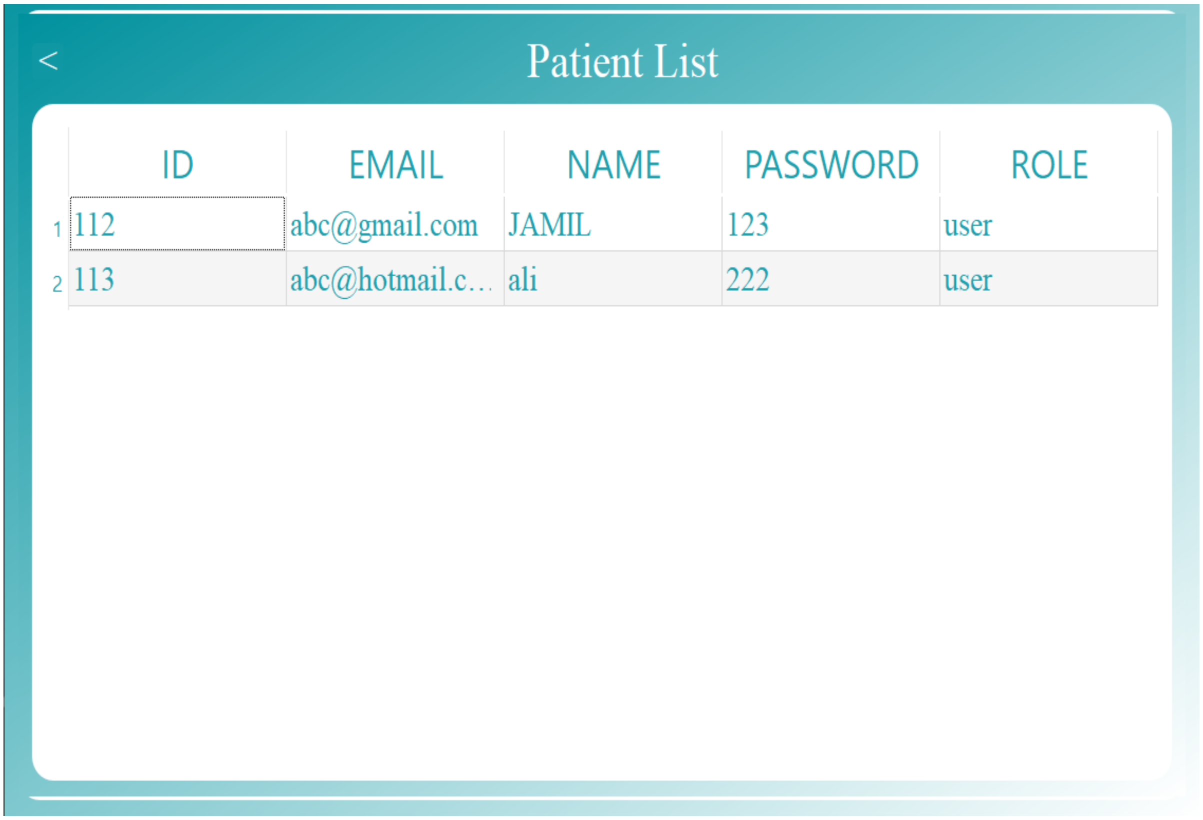


You have: No Depression

Figure 19: Patient stress report

5.2.7 Patient list

On this screen admin can see patient list. Admin can see there emails, name, and password. If he want to go back he has to simply click on back button in top left corner

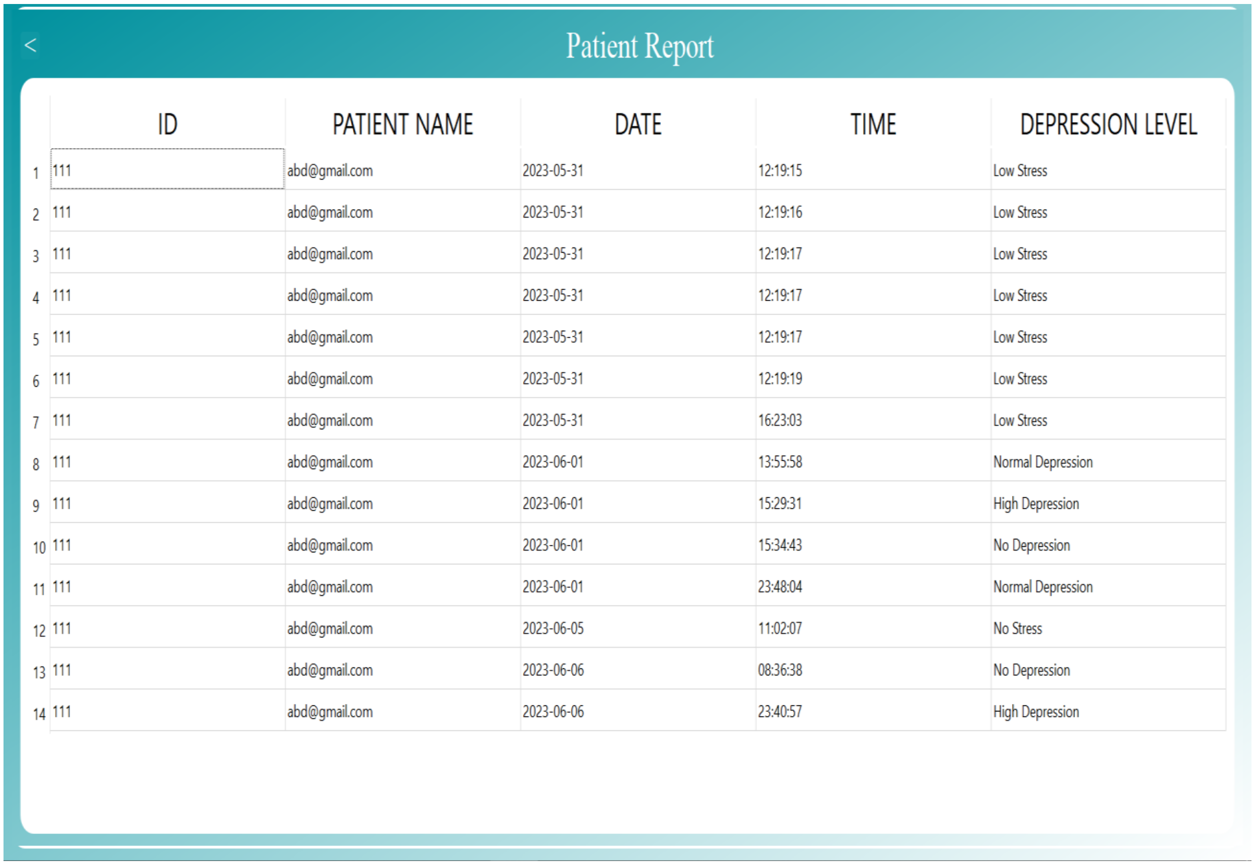


	ID	EMAIL	NAME	PASSWORD	ROLE
1	112	abc@gmail.com	JAMIL	123	user
2	113	abc@hotmail.c…	ali	222	user

Figure 20: Patient list

5.2.8 Patient report

On this screen user will see its stress reports. All his reports will be save there. He can check its reports anytime.



	ID	PATIENT NAME	DATE	TIME	DEPRESSION LEVEL
1	111	abd@gmail.com	2023-05-31	12:19:15	Low Stress
2	111	abd@gmail.com	2023-05-31	12:19:16	Low Stress
3	111	abd@gmail.com	2023-05-31	12:19:17	Low Stress
4	111	abd@gmail.com	2023-05-31	12:19:17	Low Stress
5	111	abd@gmail.com	2023-05-31	12:19:17	Low Stress
6	111	abd@gmail.com	2023-05-31	12:19:19	Low Stress
7	111	abd@gmail.com	2023-05-31	16:23:03	Low Stress
8	111	abd@gmail.com	2023-06-01	13:55:58	Normal Depression
9	111	abd@gmail.com	2023-06-01	15:29:31	High Depression
10	111	abd@gmail.com	2023-06-01	15:34:43	No Depression
11	111	abd@gmail.com	2023-06-01	23:48:04	Normal Depression
12	111	abd@gmail.com	2023-06-05	11:02:07	No Stress
13	111	abd@gmail.com	2023-06-06	08:36:38	No Depression
14	111	abd@gmail.com	2023-06-06	23:40:57	High Depression

Figure 21: Patient report

CHAPTER: 6
Testing and Evaluation

6 Testing and Evaluation:

The purpose of testing is to discover errors. Testing is the process of trying to discover faults or defects in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user exceptions and does not fail in an unacceptable manner. A good test case is the one that has a high probability of finding a yet undiscovered error. A successful test is the one that uncovers an undiscovered error. Testing may be carried out during the implementation phase to verify that the software behaves as intended by its designers and after implementation is complete.

There are few types of testing which includes the unit testing, functional testing and integration testing.

6.1 Unit Testing:

Unit Testing 1: Login as Patient with valid and invalid credentials

Testing Objective: To ensure the login form is working correctly with valid and invalid credentials/inputs.

Table 8: Login as Patient with valid and invalid credentials

No.	Test case	Attribute and value	Expected result	Result
1	Check the email field of login to validate that it takes proper email	Email: abc@gmail.com	Validates email address and moves cursor to next textbox	Pass
2	Check the email field of login to validate that it displays error message.	Email: abc.gmail.com	Highlights field and displays error message	Pass
3	When Patient enter incorrect password	Password: 12345	Invalid password	Pass
4	Patient enter the correct Password	Password : 13456	Patient move to the next screen	Pass

Unit Testing 2: Login as Admin with valid and invalid credentials

Testing Objective: To ensure the login form is working correctly with valid and invalid credentials/inputs.

Table 9: Login as Admin with valid and invalid credentials

No.	Test case	Attribute and value	Expected result	Result
1	When Admin enter incorrect password	Password: 12345	Invalid password	Pass
2	Check the email field of login to validate that it displays error message.	Email: abc.gmail.com	Highlights field and displays error message	Pass
3	When Admin enter correct password	Password: 14563	Admin move to the next screen	Pass
4	Check the email field of login to validate that it takes proper email	Email: abc@gmail.com	Validates email address and moves cursor to next textbox	Pass

6.2 Functional Testing:

Functional Testing 1: To ensure Register/sign up is working correctly

Description: New users can register themselves into this application by using sign up form.

Table 10: To ensure Register/sign up is working correctly

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
01	Enter Name in the field	Nasir khan	It should register the name	Registered successfully	Pass
02	Enter Password in the field	12345	It should register the Password	Registered successfully	Pass
03	Enter already existing email in the field	Na786@gmail.com	It should not register the Email	User already exists	Pass

Functional Testing 2: Login with different roles (Patient, Doctor)

Objective: To ensure that the correct page with the correct navigation bar is loaded.

Table 11: Login with different roles

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Login as a Patient.	Username: (correct username M003) Password: (correct password 1234)	Main page for the Management is loaded with the patient navigation bar.	Logged in and redirected to management main page.	Pass
2.	Login as a 'Doctor'.	Username: D003 Password: 1234	Main page for the Doctor is loaded with the doctor navigation bar.	Login failed – invalid credentials error	Fail

Functional Testing 3: Stress detection (To ensure that it detect stress correctly).

Objective: Only registered users in this application can access this module

Table 12: Stress detection

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
01	Click on Stress detection button	Web cam must be opened	When someone is in front of camera, it will detect their emotions	Detect emotion and show stress	Pass

Functional Testing 3: patient List

Objective: it will show the all the patient.

Table 13: Patient List

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
01	Click on Patient list button	List of patient must be shown	It will show all the names of patient which are registered	It generate the report of patients.	Pass

6.3 Business Rules Testing:

Business rules testing for stress detection involves assessing the effectiveness of rules and logic designed to identify and manage stress-related issues within a system or process. Stress in this context could refer to various types of stress, such as system overload, performance bottlenecks, or high user demand. Here are steps and considerations for testing business rules related to stress detection:

1. **Understand Business Rules:**
    - Clearly understand the business rules and logic implemented to detect and handle stress situations.
    - Document the expected behavior under stress conditions.
2. **Execution of Stress Tests:**
    - Execute stress tests by applying the predefined stress scenarios to the system.
    - Monitor system behavior, performance, and resource utilization during the stress conditions.
3. **Failover and Recovery Testing:**
    - Test the failover mechanisms and recovery processes in place when stress is detected.

4. **Performance Monitoring:**
    - Implement performance monitoring tools to continuously track system performance during stress testing.
5. **User Experience Testing:**
    - Consider the impact of stress on the user experience.
    - Test how the system communicates stress-related issues to users and whether it provides clear instructions or guidance.
6. **Documentation and Reporting:**
    - Document the test scenarios, results, and any issues encountered during stress testing.
    - Provide a comprehensive report on the system's ability to detect and handle stress conditions.

6.4 Integration Testing:

Table 14: Integration testing

No.	Test cases/Test scripts	Attributes and values	Expected result	Result
1.	The users start the camera to detect face.	Live face using camera	Face is detected live using camera.	Pass
2.	Loading of live face on CNN.		Live face gets loaded on CNN algorithm for processing.	Pass
3.	Processing of Live face	CNN algorithm	Decision is being made.	Pass
4	Results		Results are shown on the App And recorded in Patient report.	Pass

CHAPTER: 7
Conclusion and Future Work

7 Conclusion and Future Work

7.1 Conclusion of thesis

In conclusion, our project aimed to develop a stress and depression detection system using machine learning techniques. We utilized a combination of patient reports, a convolutional neural network (CNN) algorithm for stress detection using live face detection, and a questionnaire to predict Depression of patient. Through our comprehensive approach, we successfully created a system that can provide valuable insights into an individual's mental well-being.

The inclusion of a questionnaire in our system helped augment the predictive power of our model. This integration of subjective and objective measures improved the accuracy and reliability of our stress prediction model.

In summary, our project successfully developed a stress and depression detection system that leverages machine learning techniques. By incorporating patient reports, a CNN algorithm, live face detection, and a questionnaire, we created a comprehensive approach to predict stress levels. This system has the potential to be applied in various domains, such as healthcare, mental wellness, and workplace environments, to support early identification and intervention for individuals experiencing stress and depression.

7.2 Future Work

- **Enhanced Feature Extraction**: Improve the accuracy of stress and depression detection by incorporating advanced feature extraction techniques. Explore different signal processing methods or extract more comprehensive features from patient reports, such as sentiment analysis, linguistic patterns, or contextual information. This can help capture a more nuanced understanding of the patient's mental state and improve the overall performance of the model.

- **Multimodal Approach**: Incorporate multiple modalities to gather a more holistic understanding of the patient's stress and depression levels. In addition to patient reports and questionnaires, consider integrating data from other sources such as social media posts, wearable devices, or physiological signals (e.g., heart rate variability, skin conductance). Combining data from various modalities can provide a richer and more accurate representation of the patient's mental health status.

- **Detect stress of static image:** Extent image to include the feature which detect the stress level of static image. User will be able to upload image of patient or himself and system will detect its stress level.

- **Time based Analysis**: Extend the project to include longitudinal analysis of stress and depression. Instead of focusing solely on the current state of the patient, track changes over time to identify patterns and trends. By analyzing data collected at multiple time points, you can develop predictive models that anticipate potential stressors or detect early signs of worsening mental health. Longitudinal analysis can provide valuable insights for personalized interventions and early intervention strategies.

CHAPTER: 8
References

8 References

1) https://www.bemindfulonline.com/test-your-stress

2) https://www.oprah.com/omagazine/the-stress-detector-test_1

3) https://www.stress.org.uk/individual-stress-test/

4) https://www.sciencedirect.com/science/article/pii/S187705091731904X

5) https://psychcentral.com/quizzes/stress-test#take-the-quiz