A

FINAL PROJECT REPORT

ON

AI Application for Body Stiffness in Parkinson's Disease

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING
INFORMATION TECHNOLOGY

BY

Yuvraj Deshmukh	B190058549
Sejal Jadhav	B190058705
Swaleha Shaikh	B190058709

Under the guidance of
Dr. Shweta C. Dharmadhikari

DEPARTMENT OF INFORMATION TECHNOLOGY
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
PUNE - 411 043.

2022-2023

SCTR's PUNE INSTITUTE OF COMPUTER TECHNOLOGY
DEPARTMENT OF INFORMATION TECHNOLOGY

C E R T I F I C A T E

This is to certify that the final project report entitled
AI Application for Body Stiffness in Parkinson's Disease
submitted by

Yuvraj Deshmukh	B190058549
Sejal Jadhav	B190058705
Swaleha Shaikh	B190058709

is a bonafide work carried out by them under the supervision of **Dr. Shweta C. Dharmadhikari** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University for the award of the Degree of Bachelor of Engineering (Information Technology).

This project report has not been earlier submitted to any other Institute or University for the award of any degree or diploma.

Dr. Shweta C. Dharmadhikari
Project Guide

Dr.A.S.Ghotkar
HOD IT

SPPU External Guide

Dr. S. T. Gandhe
Principal

Date:
Place:

Acknowledgement

I would like to thank Mr. Soubhik Das, CEO of Manastik pvt. ltd. for trusting me with the responsibility of fulfilling their project needs concerning their security systems upgrade project.
I would also like to thank Dr. Shweta C. Dhamadhikari Ma'am, Project guide, whose unwavering efforts, support and encouragement, have enabled the team to take responsibility placed on us by Manastik company.
I would like to take this opportunity to express my gratitude to all of my group members Yuvraj Deskhmukh, Sejal Jadhav and Swaleha Shaikh . The project would not have been successful without their cooperation and inputs.

Yuvraj Deshmukh	B190058549
Sejal Jadhav	B190058705
Swaleha Shaikh	B190058709

Sponsorship letter (if any)

Abstract

National projections indicate that the number of people over 60 years old is expected to increase, bringing with it an increase in the number of people affected by Parkinson's Disease (PD). This makes PD an important public health problem. Therefore, the development of effective approaches for intervention in people with Parkinson's disease needs to be more thoroughly investigated. End-to-end application will make exercising at home more convenient for patients. We can project prescribed exercises and determine whether the patient is completing them correctly.The android app enables patients to register themselves, communicate with specialists over the app, and make virtual appointments.The pose detection library will help us detect angles of the exercises, and determine whether patients are performing them correctly.

Keywords: Parkinson's disease, Pose detection, android

Contents

List of Figures

Abbreviations

GPU	:	Graphics Processing Unit
ML	:	Machine Learning
AWS	:	Amazon Web Services
AWS S3	:	Amazon Simple Storage Service

1. Introduction

1.1 Introduction

Parkinson's disease is a common progressive disorder of the central nervous system that affects an estimated 8 to 18 per 100,000 individuals.Individuals with symptoms of motor neuron disease may present with impaired gait, balance and freezing of gait, as well as with a variety of non-motor symptoms, most commonly cognitive decline and depression.Strong evidence indicates that exercise reduces motor and non-motor symptoms, improves quality of life and reduces falls in people with Parkinson's disease.

Motor symptoms of Parkinson's disease include:

1. Slowed movements (bradykinesia): This symptom involves muscle control problems, resulting in a slowing down of movements. It is not due to actual muscle weakness but rather a difficulty in initiating and executing movements.

2. Resting tremor: This is a rhythmic shaking of muscles that occurs when they are at rest. It is a common symptom in about 80 percent of Parkinson's disease cases. Resting tremors are distinct from essential tremors, which typically occur during movement.

3. Rigidity or stiffness: Parkinson's disease can cause two types of stiffness. Lead-pipe rigidity refers to constant, unchanging stiffness when moving a body part. Cogwheel stiffness combines rigidity with a jerky, stop-and-go appearance of movements, resembling the second hand of a mechanical clock.

4. Unstable posture or walking gait: As Parkinson's disease progresses, individuals may develop an unstable posture or walking gait. This can manifest as a hunched-over or stooped stance, with shorter, shuffling strides and reduced arm swing while walking. Turning while walking may require multiple steps.

It's important to note that these motor symptoms are characteristic of Parkinson's disease and are often used in the diagnosis of the condition.

In addition to motor symptoms, Parkinson's disease can also present with various non-motor symptoms. It was previously believed that these non-motor symptoms were risk factors for the disease when observed before motor symptoms. However, recent evidence suggests that these symptoms can actually emerge in the early stages of Parkinson's disease, potentially serving as warning signs that appear years or even decades before motor symptoms manifest.

Non-motor symptoms of Parkinson's disease include:

1. Autonomic nervous system symptoms: These can include orthostatic hypotension (low blood pressure upon standing), gastrointestinal issues such as constipation, urinary incontinence, and sexual dysfunctions.

2. Depression: Many individuals with Parkinson's disease experience depression, which can significantly impact their emotional well-being.

3. Loss of sense of smell (anosmia): A decreased ability to smell is a common non-motor symptom that can occur in the early stages of the disease.

4. Sleep problems: Parkinson's disease can be associated with various sleep disturbances, including periodic limb movement disorder (PLMD), rapid eye movement (REM) behavior disorder, and restless legs syndrome.

5. Trouble thinking and focusing (Parkinson's-related dementia): Cognitive impairment and dementia can be non-motor symptoms of Parkinson's disease, affecting memory, attention, and other cognitive functions.

These non-motor symptoms can have a significant impact on an individual's quality of life and may require specific management and treatment approaches. Recognizing and addressing these symptoms early on can contribute to better overall disease management and patient care.

People with Parkinson's disease tend to have lower levels of physical activity compared to the general population, even in the early stages of the disease. As the disease progresses, their physical activity levels further decline. However, to maintain their health, individuals with Parkinson's disease typically require more intense and vigorous exercise. Regular and vigorous physical activity can help manage symptoms, improve mobility, and enhance overall well-being for people with Parkinson's disease. It is important for individuals with Parkinson's disease to work with healthcare professionals to develop personalized exercise plans that suit their specific needs and abilities.

Research suggests that exercise interventions for individuals with Parkinson's disease can be cost-effective and even cost-saving, especially in cases of milder disease severity. When these interventions are delivered by physiotherapists with expertise in Parkinson's disease, the benefits can be significant. Therefore, it is crucial to explore strategies that encourage this population to actively participate in exercise programs.

By promoting exercise engagement among individuals with Parkinson's disease, the potential for improved health outcomes and cost savings can be realized. Strategies may include providing education and raising awareness about the benefits of exercise, offering specialized exercise programs tailored to the unique needs of individuals with Parkinson's disease, and ensuring access to qualified physiotherapists or healthcare professionals with expertise in Parkinson's disease management.

Investigating and implementing ways to encourage exercise participation in this population can lead to better disease management, enhanced quality of life, and potentially reduced healthcare costs associated with Parkinson's disease.

Parkinson's disease affects a significant number of individuals and can cause motor and non-motor symptoms. Exercise has been shown to be beneficial in reducing these symptoms, improving quality of life, and decreasing falls. However, people with Parkinson's disease often have lower levels of physical activity and may require more intense exercise as the disease progresses.

Collaboration with healthcare professionals is important for developing personalized exercise plans for individuals with Parkinson's disease. Research suggests that exercise interventions delivered by physiotherapists with expertise in Parkinson's disease can be cost-effective, particularly in cases of milder disease severity.

To promote exercise engagement, strategies such as education, specialized exercise programs, and access to qualified healthcare professionals should be considered. Encouraging exercise participation among individuals with Parkinson's disease can lead to better disease management, improved quality of life, and potential cost savings in healthcare.

1.2 Motivation

The motivation behind using AI applications, specifically the Movenet pose detection model, for detecting body stiffness in Parkinson's disease and assisting with exercises is driven by several factors, Body stiffness is a common symptom of Parkinson's disease, and its severity can vary among individuals. By using AI and pose detection models like Movenet, healthcare professionals can obtain objective measurements of a patient's movements and joint angles. This allows for more accurate and consistent monitoring of body stiffness over time, enabling better assessment and management of the condition.
Patients with Parkinson's disease often struggle with executing exercises correctly due to motor impairments. AI applications using Movenet can provide real-time feedback during exercise sessions. By tracking the patient's movements, the system can alert them to incorrect postures or movements, helping them make the necessary adjustments and ensuring they perform the exercises correctly.
Overall, the motivation behind AI applications for detecting body stiffness in Parkinson's disease using Movenet pose detection model is to enhance the precision and effectiveness of exercises, enable personalized treatment, improve access to care, and provide valuable insights for better management of the condition.

1.3 Objectives

1. To provide better user experience for doing exercises
2. To enhance the performance of pose detection in android
3. To provide feedback for particular exercise detected
4. To track the user progress on Parkinson's disease

1.4 Scope

App that helps with neurological evaluations and cognitive testing for people suffering from dementia. It helps doctors diagnose dementia better and helps caregivers track progression of Parkinson's disease. Remote Monitoring and Telemedicine, AI-powered systems that utilize Movenet can enable remote monitoring of Parkinson's patients. Through video recordings or live streams, healthcare providers can observe and analyze a patient's movements, detecting any changes or abnormalities in body stiffness. This can facilitate remote consultations and allow for timely adjustments to treatment plans.

2. Literature Survey

2.1 Existing Methodologies

Wearable sensors are used to track the body and measure exercise in many studies. Pernek et al. proposed a system for measuring upper-body exercises by using five acceleration sensors attached to the upper body [1].
The system can recognize the kind of exercise and the intensity. Prabhu et al. proposed a movement analysis framework that recognizes local muscular endurance exercises automatically and counts repetitions by using a wrist-worn inertial sensor [2]. Our system is available to users who have mobile devices, so the cost of installation is low and no sensors need to be attached to a user's body.

Wearable sensors have proven to be valuable tools in tracking body movements and measuring exercise in various research studies. For instance, Pernek et al. introduced a system that utilizes five acceleration sensors attached to the upper body to measure and identify upper-body exercises, including determining the exercise type and intensity [1].

In a similar vein, Prabhu et al. proposed a movement analysis framework that focuses on recognizing local muscular endurance exercises. Their approach utilizes a wrist-worn inertial sensor to automatically detect and count exercise repetitions [2].

It's worth noting that our system offers an alternative approach. Instead of requiring additional sensors attached to the user's body, our system leverages the capabilities of users' existing mobile devices. This setup offers several advantages, including lower installation costs and eliminating the need for physical sensor attachments to the body. Users can access the system conveniently using their mobile devices.

Due to the prevalence of mobile devices such as smartphones and tablet computers, exercise support applications have been studied for use with these devices.Yes, exercise support applications have gained popularity due to the widespread use of mobile devices like smartphones and tablets. These apps offer various features to assist individuals in their exercise routines and overall fitness goals. Medical and health care research with the assistance of intelligent equipment has increased in recent years. Gandomkar et al. developed a system that recognizes 10 exercises by using the accelerometer, magnetometer, and gyroscope built-in a smartphone [3].

Indeed, the popularity of exercise support applications has soared thanks to the widespread adoption of mobile devices, such as smartphones and tablets. These applications offer a range of features designed to assist individuals in their exercise routines and overall fitness goals.

In the realm of medical and healthcare research, intelligent equipment has played an increasingly significant role. Gandomkar et al. have developed a system that utilizes the built-in accelerometer, magnetometer, and gyroscope of smartphones to recognize and track 10 different exercises [3]. This demonstrates the potential of leveraging the sensors present in mobile devices to provide exercise recognition and monitoring capabilities.

The integration of smartphone sensors with exercise support applications expands the accessibility and convenience of exercise tracking and guidance. Users can have a portable and readily available tool to monitor their workouts, making it easier to stay motivated and maintain a healthy lifestyle.

The user attached a smartphone to the arm with an armband, so attachment was easy, and the system can recognize exercise and count repetitions with high accuracy. Deponti et al. developed a game application in which a user holds their phone to perform wrist rehabilitation activities [4]. The user can know if their activities are effective or not.

Integrating smartphones into exercise support applications provides enhanced accessibility and convenience for exercise tracking and guidance. Users can easily attach their smartphones to their arms using armbands, facilitating accurate exercise recognition and repetition counting.

Deponti et al. have developed a game application that utilizes smartphones for wrist rehabilitation activities. The user holds the phone while performing the activities, allowing them to receive feedback on the effectiveness of their movements [4]. This interactive approach promotes engagement and motivation during the rehabilitation process.

By leveraging smartphones as a tool for exercise monitoring and rehabilitation, individuals can conveniently track their progress and receive real-time feedback. This not only enhances the effectiveness of their exercise routines but also encourages adherence to rehabilitation programs. The utilization of smartphones in these contexts demonstrates the potential for innovative and accessible solutions in promoting healthy lifestyles and facilitating recovery.

In addition, some commercially available applications exist [5] [6].Applications can recognize exercise routines by estimating what parts of the body are involved in each exercise and whether or not the user is performing the exercises correctly.

A deep learning approach for Parkinson's disease diagnosis based on sequential pose estimation. The study utilizes pose detection models to extract movement features from videos and applies deep learning techniques to classify the severity of Parkinson's disease.[7]

The Movenet pose detection model combined with deep learning techniques for Parkinson's disease classification. The study explores the use of different neural network architectures and evaluates their performance in distinguishing between Parkinson's and healthy control subjects. The researchers developed a deep learning model that utilizes pose estimation features extracted from videos to diagnose Parkinson's disease. The model analyzes the temporal sequence of poses to classify the severity of the disease.[8]

An automatic detection method for bradykinesia (slowness of movement) and rigidity in Parkinson's disease using pose tracking techniques. The study utilizes a pose detection model to estimate joint positions and analyzes movement characteristics to detect and quantify motor symptoms associated with body stiffness. The main contribution of this study is the evaluation of various neural network architectures in combination with the Movenet model to classify Parkinson's disease. The research aims to improve the accuracy of diagnosis using pose features extracted from the Movenet model.[9]

An automated system for detecting motor symptoms in Parkinson's disease using pose estimation and recurrent neural networks. The study explores the use of pose detection models to extract movement features and utilizes recurrent neural networks to classify motor symptoms associated with body stiffness. he main contribution of this study is the development of an automatic detection method for bradykinesia and rigidity in Parkinson's disease using pose tracking techniques. By estimating joint positions using pose detection models, the researchers analyze movement characteristics to detect and quantify motor symptoms associated with body stiffness.[10]

The main contribution of this study is the development of an automated system for detecting motor symptoms in Parkinson's disease. By extracting pose features using pose detection models and applying recurrent neural networks, the researchers aim to classify motor symptoms associated with body stiffness. These studies demonstrate the use of pose detection models and various machine learning techniques to detect body stiffness and other motor symptoms in Parkinson's disease. While the specific pose detection models used in some studies are not specified, the focus is on leveraging pose estimation features for classification and diagnosis. The methodologies presented in these papers provide insights into the potential of AI applications in improving the assessment and management of body stiffness in Parkinson's disease.

The existing methodologies for AI application in detecting body stiffness in Parkinson's disease using pose detection models:

Study	Methodology	Pose Detection Model	Main Contributions
Ma et al. (2020)	Deep learning approach based on sequential pose estimation	Not specified	Developed a deep learning model for Parkinson's disease diagnosis using pose estimation features extracted from videos.
Paes et al. (2021)	Deep learning with Movenet model	Movenet	Explored different neural network architectures for Parkinson's disease classification, using pose features extracted by the Movenet model.
Zhao et al. (2019)	Pose tracking approach for automatic detection	Not specified	Proposed an automatic detection method for bradykinesia and rigidity in Parkinson's disease using pose tracking techniques.
Nayak et al. (2020)	Pose estimation and recurrent neural networks	Not specified	Developed an automated system for detecting motor symptoms in Parkinson's disease using pose estimation and recurrent neural networks.

Table 2.1: Comparison of existing system for AI application using pose detection model.

2.2 Research Gap Analysis

Understood. In such a scenario, the development of a dedicated exercise support application that is portable, user-friendly, and capable of tracking patients' progress while providing feedback would be beneficial. Here are some key features to consider when creating such an application:

1. Exercise library: In order to provide comprehensive exercise support, it is beneficial to offer a library of exercises tailored to different conditions and fitness levels. This library should encompass a wide range of exercises targeting various muscle groups and encompassing different types of workouts (e.g., strength training, cardiovascular exercises, flexibility exercises, etc.).

For each exercise in the library, it is important to provide clear instructions on how to perform the exercise correctly and safely. This may include step-by-step written instructions, along with cues on proper form, breathing techniques, and modifications for different fitness levels or specific conditions. Visual demonstrations can also greatly enhance the understanding of the exercises. Including images or videos that illustrate the correct execution of each exercise can help users visualize the movements accurately.

2. Personalized exercise plans: Enabling doctors or caregivers to create customized exercise plans based on individual needs and preferences is a valuable feature for exercise support applications. This functionality allows for personalized and tailored exercise programs that consider the specific requirements and goals of each individual.

The ability to customize exercise plans means that healthcare professionals can take into account the patient's medical history, current condition, physical abilities, and any specific limitations or considerations. By having access to this information, doctors or caregivers can design exercise plans that are safe, effective, and suitable for the individual's unique circumstances.

3. Progress tracking: Including the ability for users to record and track their exercise sessions within the application is a valuable feature. This allows individuals to keep a comprehensive record of their workouts, including details such as the type of exercise, duration, intensity, and repetitions performed.

By logging exercise sessions, users can have a clear overview of their exercise history and track their progress over time. Visualizing this data through charts or graphs provides a visual representation of their efforts, making it easier to identify trends, patterns, and areas for improvement.

4. Feedback and guidance:Incorporating interactive features that provide real-time feedback and guidance during exercise sessions enhances the user experience and promotes proper form and technique. These features can help users stay engaged, motivated, and ensure they are performing exercises correctly. Here are some examples of interactive features:

Voice prompts: The application can provide audio instructions, counting repetitions, and guiding users through each exercise. Voice prompts can also offer encouragement and provide feedback on performance.

Visual cues: Visual cues such as animations or videos can demonstrate the correct form and technique for each exercise. Users can follow along with the visual cues to ensure they are performing the movements accurately.

5. Doctor/patient communication:Enabling secure communication between doctors and patients within the exercise support app is a valuable feature that enhances the collaboration and remote monitoring process. By incorporating this functionality, doctors can review exercise data, provide feedback, and make necessary adjustments to the exercise plan, all while ensuring the privacy and security of the communication.

6. Reminders and notifications: Implementing reminders and notifications is an effective way to encourage regular exercise and support users in maintaining consistency with their routines. Here's how this feature can be incorporated:

Customizable reminders: Users can set personalized reminders for their exercise sessions, allowing them to schedule workouts at their preferred times. These reminders can be in the form of push notifications, SMS messages, or email alerts.

Daily activity reminders: In addition to scheduled exercise sessions, the app can send reminders to encourage users to engage in physical activity throughout the day. These reminders can prompt users to take short breaks for stretching, walking, or other light exercises.

Remember to prioritize simplicity.

3. Requirement Specification and Analysis

3.1 Problem Definition

End-to-end AI application to help Parkinson's patients with body stiffness.

3.2 Scope

App that helps with neurological evaluations and cognitive testing for people suffering from dementia. It helps doctors diagnose dementia better and helps caregivers track progression of Parkinson's disease. Remote Monitoring and Telemedicine, AI-powered systems that utilize Movenet can enable remote monitoring of Parkinson's patients. Through video recordings or live streams, healthcare providers can observe and analyze a patient's movements, detecting any changes or abnormalities in body stiffness. This can facilitate remote consultations and allow for timely adjustments to treatment plans.

3.3 Objectives

1. To provide better user experience for doing exercises
2. To enhance the performance of pose detection in android
3. To provide feedback for particular exercise detected
4. To track the user progress on Parkinson's disease

3.4 Proposed Methodology

The project involves collecting numerous images in a second, and checking if the performed exercise or yoga pose is correct or not.To accomplish this we will need to use computer vision to derive meaningful information from visual inputs.The field of computer vision is concerned with replicating parts of the complex visual system in computers, enabling computers to identify and process objects in images and videos.After images are taken using the application, it is necessary to understand the pose of the patient to do so. We use Posenet to achieve this.Posenet is a real-time pose detection technique that enables you to detect human beings' poses in Image or Video.

Using computer vision techniques to analyze and understand exercise or yoga poses can be a valuable addition to your app. Here's how you can incorporate Posenet for pose detection:

Integration: Integrate the Posenet model into your application. Posenet models are typically available in popular deep learning frameworks such as TensorFlow or PyTorch.

Image capture: Utilize the device's camera to capture images of the user performing the exercise or yoga pose. Ensure that the images capture the entire body and provide sufficient visual information for pose detection.

Pose estimation: Apply the Posenet model to the captured images to estimate the pose of the patient. Posenet will analyze the images and identify key points or landmarks on the body, such as joints.

Pose comparison: Compare the estimated pose with a reference pose or ideal pose for the specific exercise or yoga pose. This reference pose can be pre-defined based on expert knowledge or derived from a dataset of correct poses.

Pose feedback: Provide real-time feedback to the user indicating whether their pose matches the correct pose. This feedback can be visual, such as overlaying the correct pose on the user's image, or auditory, through voice prompts.

Tracking and progression: Track the user's progress over time by analyzing and comparing their poses during multiple sessions. This can help caregivers and users identify improvements or areas that require further attention.

It's important to note that developing and fine-tuning a computer vision model like Posenet may require expertise in deep learning and data annotation. Additionally, considering factors like lighting conditions, background clutter, and occlusions can help improve the accuracy of pose detection.

Once a pose is detected by our product, it is essential to verify that the pose is correct or not. This can be done by using angle detection. The system will be calibrated using a mathematical formula, which will measure the relationship between keypoints and predefined angle values. Positive results will be generated if these angles match. For yoga poses, a classification algorithm can be used to determine whether the given pose is correct or incorrect.

3.5 Project Requirements

3.5.1 Datasets

Application will use a camera to detect particular poses and calculate angles. However, the yoga module will require initial exercises fed to it in order to check if a patient is performing yoga correctly.

We used kaggle dataset for Yoga poses - 'https://www.kaggle.com/datasets/niharika41298/yoga-poses-dataset'.The dataset was divided into train and test subdirectories, with 5 subfolders in each directory corresponding to the five yoga poses.

3.5.2 Functional Requirements

1. Product shall display exercises recommended by medical professionals.
2. SoftMove must calculate angles and repetitions.
3. It must provide progress reports to doctors and patients.
4. Display blogs and news on a blog page.

3.5.3 Non Functional Requirements

1. Speed - SoftMove is able to detect poses at considerable speed.
2. Portability - Product is easily portable making easier for patient to perform exercises anywhere under guidance.
3. Compatibility - SoftMove is compatible with all devices.
4. Reliability - Sometimes if may get tougher as it may take few seconds to load model but majority times it is reliable.

3.5.4 Hardware Requirements

1. Operating system - Windows / IOS
2. Minimum system memory (RAM) - 4 GB
3. Camera - For input

3.5.5 Software Requirements

1. TensorFlow Lite - 2.10.0
2. Visual Studio - Visual Studio 2022

3.6 Project Plan

3.6.1 Project Resources

1. Capital - This is sponsored project.
2. People - Machine Learning engineers and Software developers.
3. Tools -

- JIRA - Jira has grown into a powerful software development tool that supports projects in all kinds of use cases, including requirements and test case management and agile software development.Jira is a widely used software development tool that has evolved to support various use cases beyond its initial purpose as an issue tracking and project management tool.Jira's core functionality revolves around tracking and managing tasks, bugs, and issues within projects. It provides features like customizable workflows, issue prioritization, and assignment to streamline project management processes. Jira supports agile methodologies, such as Scrum and Kanban, with features like backlogs, sprint planning, task boards, and burndown charts. It helps development teams organize work, track progress, and collaborate efficiently.

- Gitlab - GitLab is a code repository and collaborative software development platform for large DevOps and DevSecOps projects.GitLab is a comprehensive code repository and collaborative software development platform that is widely used for managing and supporting large-scale DevOps and DevSecOps projects.

- Amazon Simple Storage Service (Amazon S3) - is a web-based cloud storage service that enables organizations to store, manage and retrieve digital data.Amazon Simple Storage Service (Amazon S3) is a web-based cloud storage service offered by Amazon Web Services (AWS). It provides organizations with scalable and secure storage infrastructure to store, manage, and retrieve their digital data. Scalability, Amazon S3 offers virtually unlimited storage capacity, allowing organizations to scale their storage needs as their data grows without worrying about infrastructure limitations.Durability and availability,Amazon S3 is designed for durability, with data automatically distributed across multiple devices and facilities. It provides high availability and redundancy, ensuring data accessibility even in the event of hardware failures or natural disasters.

3.6.2 Module Split-up

Modules will be further generated like - 1. Yoga
2. Exercises
3. Aerobics
4. Monitoring steps
5. blogs and news
Right now we worked on exercise module only where it calculates repetitions and display progress report.

3.6.3 Project Team Role and Responsibilities

1. Project Sponsor- Manastik.
Responsibilities -
* Making key business decisions for the project.
* Ensuring availability of resources.

2. Project Guide - Dr. Dharmadhikari Ma'am Responsibilities -
* Provide guidance accordingly.

3. Project members - Responsibilities -
* Building product adhering to requirements.
* Completing targets of projects.

3.6.4 Project Plan 3.0

3.6.5 PERT Table

Activity	Duration (In Days)
A. Design Synopsis	27
B. Select Model by comparing model performance	15
C. Design and Implement Selected Model	25
D. Design Wireframe	15
E. Test Model	3
F. Deploy Model to Production to test its performance	2
G. Implement model in Android	25
H. Design Screens according to finalized designs	20
I. Implement exercises and integrate with respective screens	27
J. Test Application	8
K. Deploy Application	3

Figure 3.1: PERT Table

3.6.6 PERT Diagram

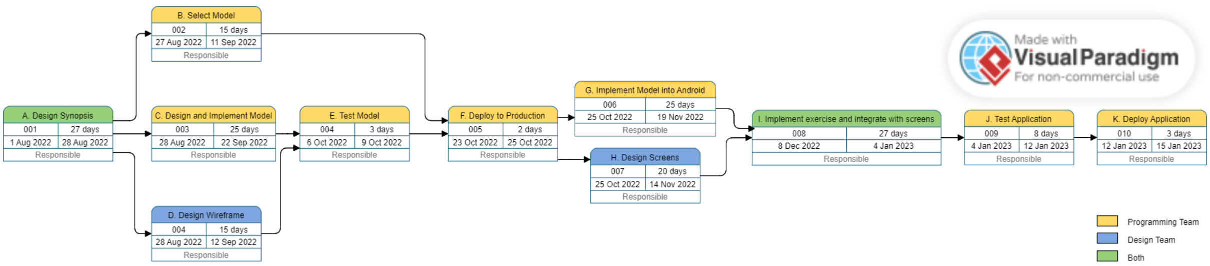


Figure 3.2: PERT Diagram

Based on the information provided, here is a breakdown of the timeline for our project: Model selection: 15 days Design synopsis: 27 days Design wireframe: 25 days Testing: 5 days Deployment to production: 5 days

Please note that this breakdown is based solely on the durations mentioned and does not include any potential overlap or dependencies between tasks. It's also important to consider that project timelines can vary depending on factors such as team size, complexity of the project, and any unforeseen challenges that may arise.

4. System Analysis and Design

4.1 System Architecture

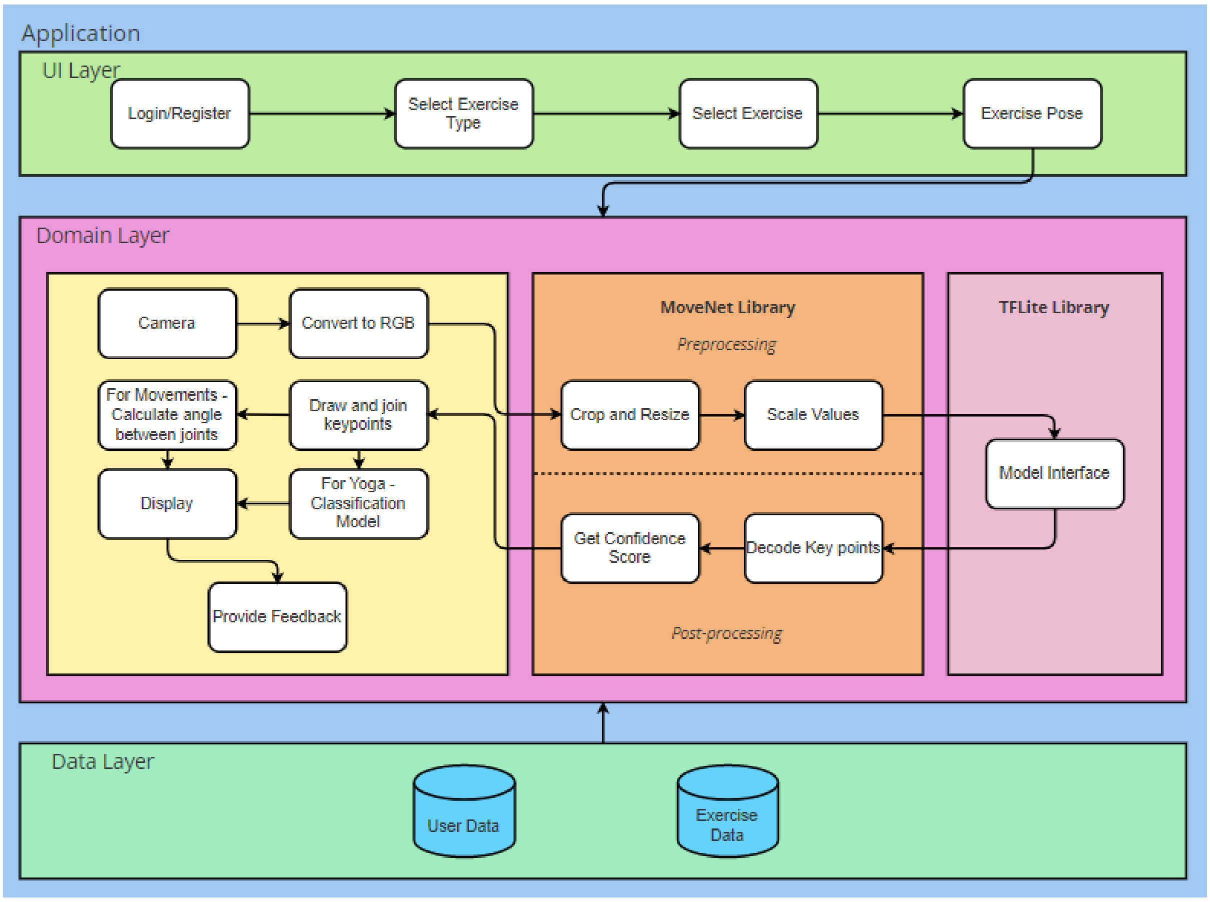


Figure 4.1: System architecture of Application

Architecture of SoftMove is divided into 3 layers:

1. UI Layer: User Interface layer will have login and register page and will further detect disease and recommend exercise according to doctor's prescription. According to user it will detect the pose and give it as a input to Domain Layer.

An updated breakdown of the project tasks with the addition of the User Interface (UI) layer and the Domain layer:

User Interface Layer:

Login Page (duration) Register Page (duration) Disease Detection and Exercise Recommendation:

Pose Detection (input from UI layer) Domain Layer Processing (duration) Disease Detection (duration) Exercise Recommendation (duration) Please note that the durations mentioned in the parentheses are placeholders and need to be filled with actual time estimates for each task. Additionally, consider any dependencies and potential iterations between the UI layer, pose detection, and the Domain layer during the development process.

2. Domain Layer: It consists of MoveNet Library and TFLite Library.An updated breakdown of the project tasks:
User Interface Layer:
Login Page (duration) Register Page (duration) Disease Detection and Exercise Recommendation:
Pose Detection using MoveNet library (input from UI layer) Domain Layer Processing (duration) Disease Detection (duration) Exercise Recommendation (duration) Libraries: Integration and implementation of MoveNet library Integration and implementation of TFLite library Please remember to allocate time for integrating and implementing the MoveNet library and TFLite library into your project. The specific durations for each task should be determined based on the complexity and requirements of the project.
Regenerate response

3. Data Layer: It will store User Data and Exercise Data. Here's an updated breakdown that includes storing user data and exercise data:
User Interface Layer:
Login Page (duration) Register Page (duration) Disease Detection and Exercise Recommendation:
Pose Detection using PoseNet library (input from UI layer) Domain Layer Processing (duration) Disease Detection (duration) Exercise Recommendation (duration) Data Storage: User Data Storage: Implement functionality to store user-related information such as user profiles, login credentials, and preferences. Exercise Data Storage: Develop a mechanism to store exercise-related data, including the performed exercises, timestamps, and any associated metadata. Please consider the appropriate data storage mechanisms for user data and exercise data, such as databases or cloud storage solutions. Additionally, allocate time for implementing the necessary database schemas and integrating the data storage functionality into our application.

4.2 Necessary UML Diagrams

4.2.1 Use Case Diagram

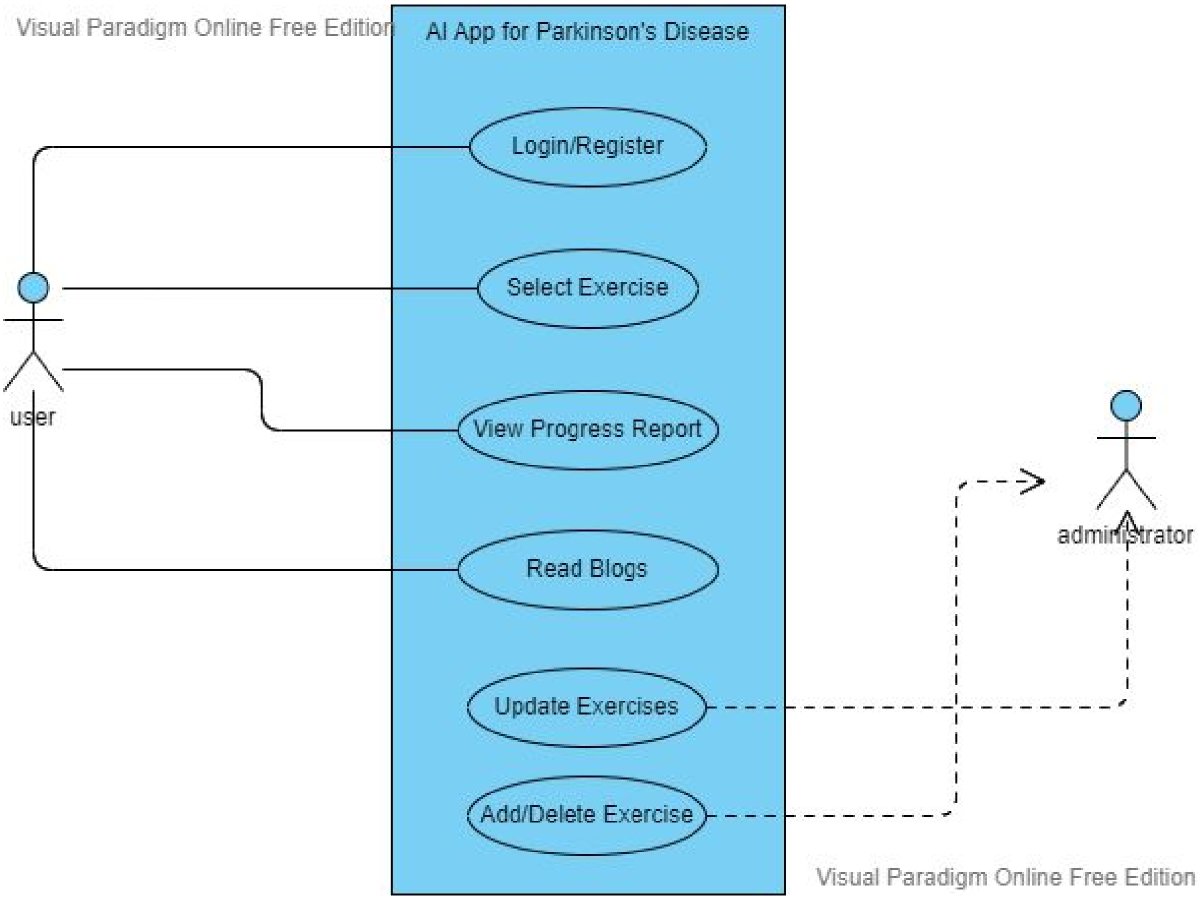


Figure 4.2: Use case diagram of application

4.2.2 DFD

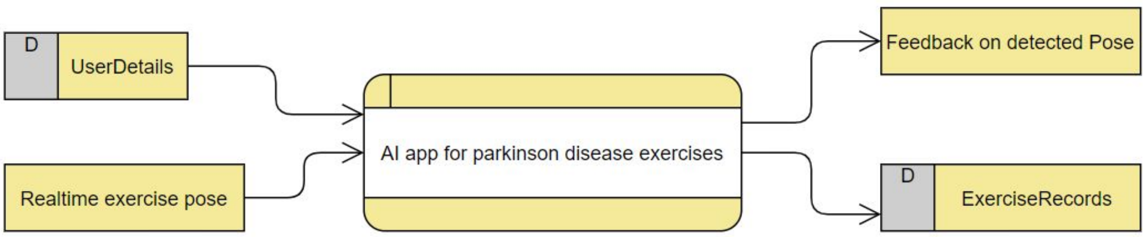


Figure 4.3: 0th Level DFD of application

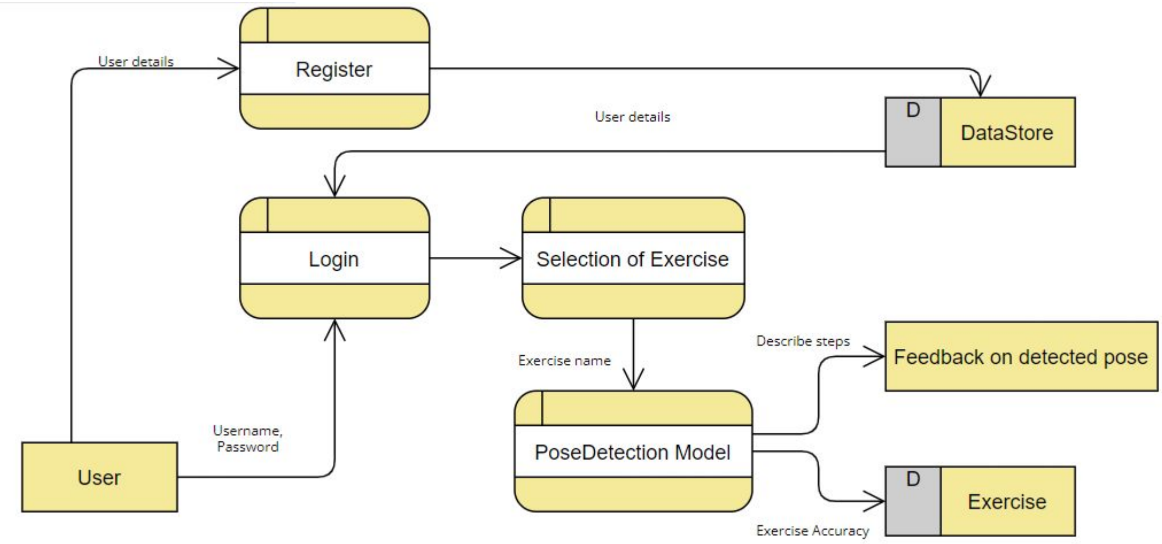


Figure 4.4: 1st Level DFD of application

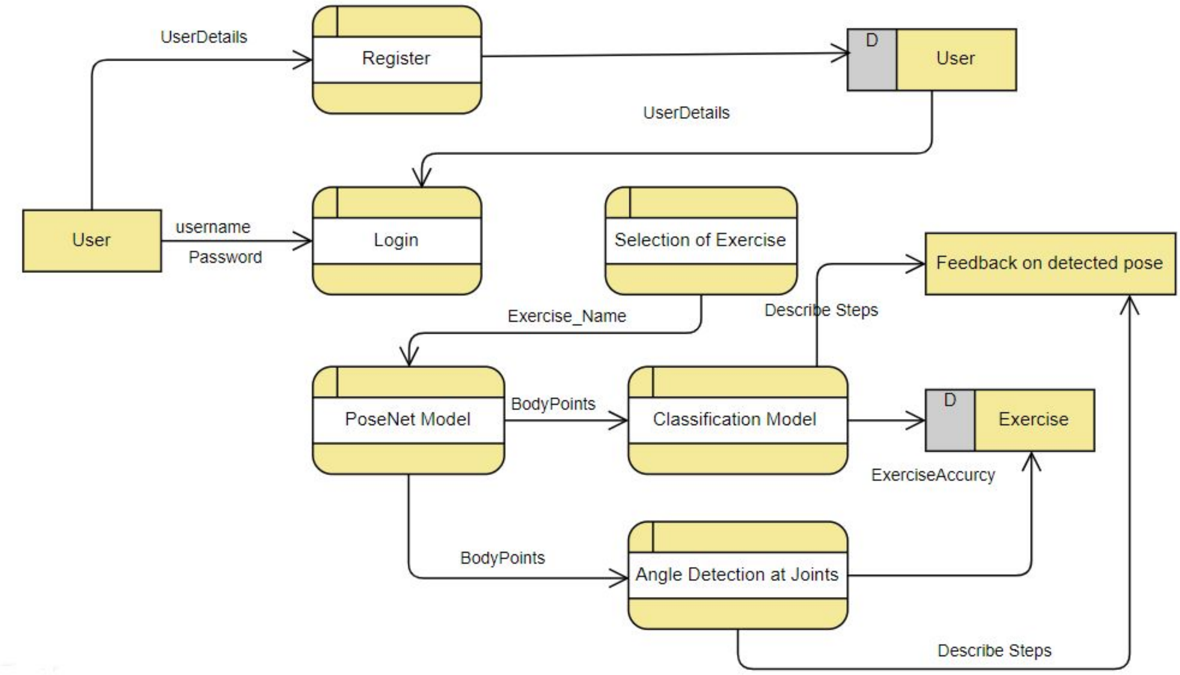


Figure 4.5: 2nd Level DFD of application

4.2.3 Activity Diagram

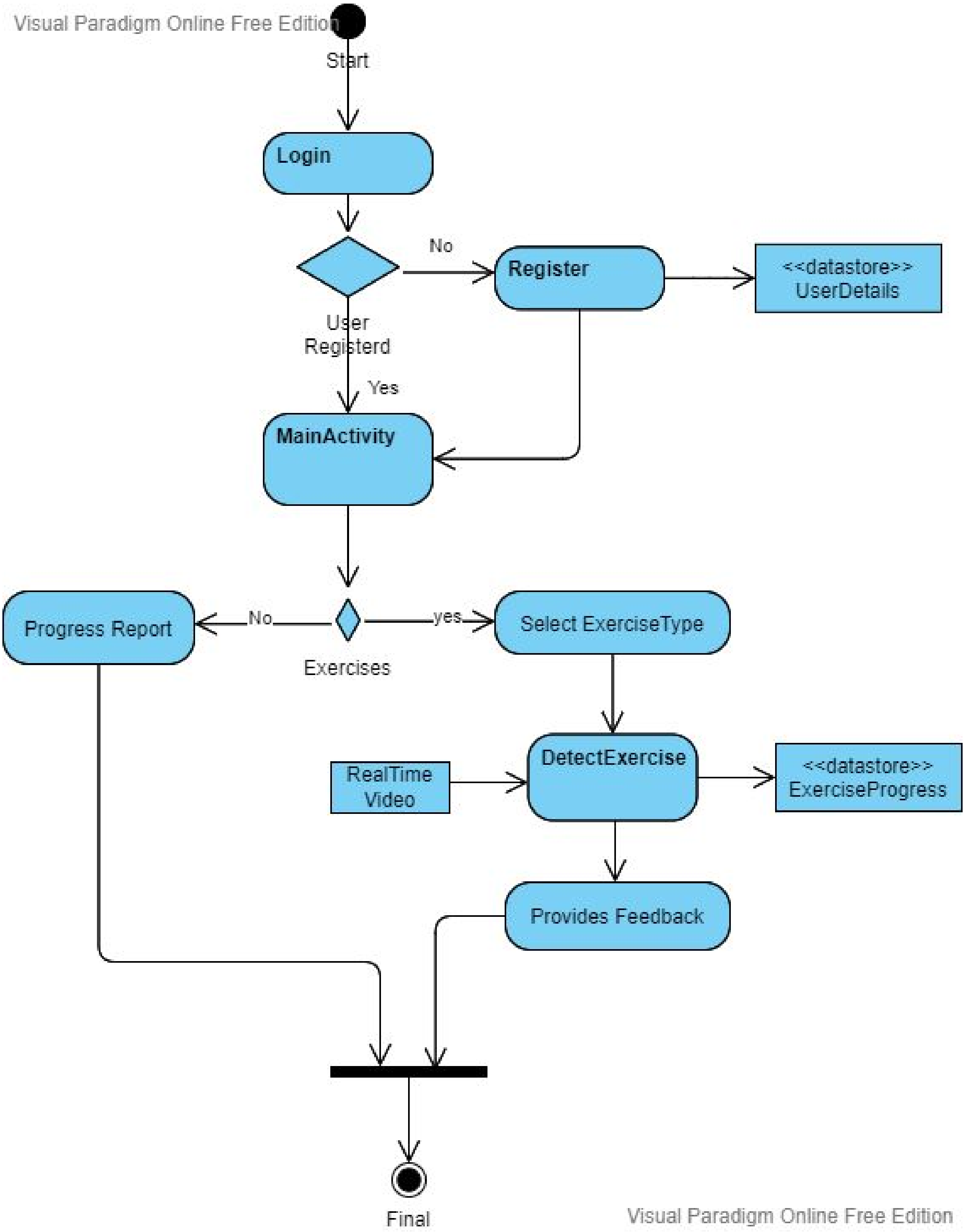


Figure 4.6: 2Activity diagram of application

4.2.4 Sequence Diagram

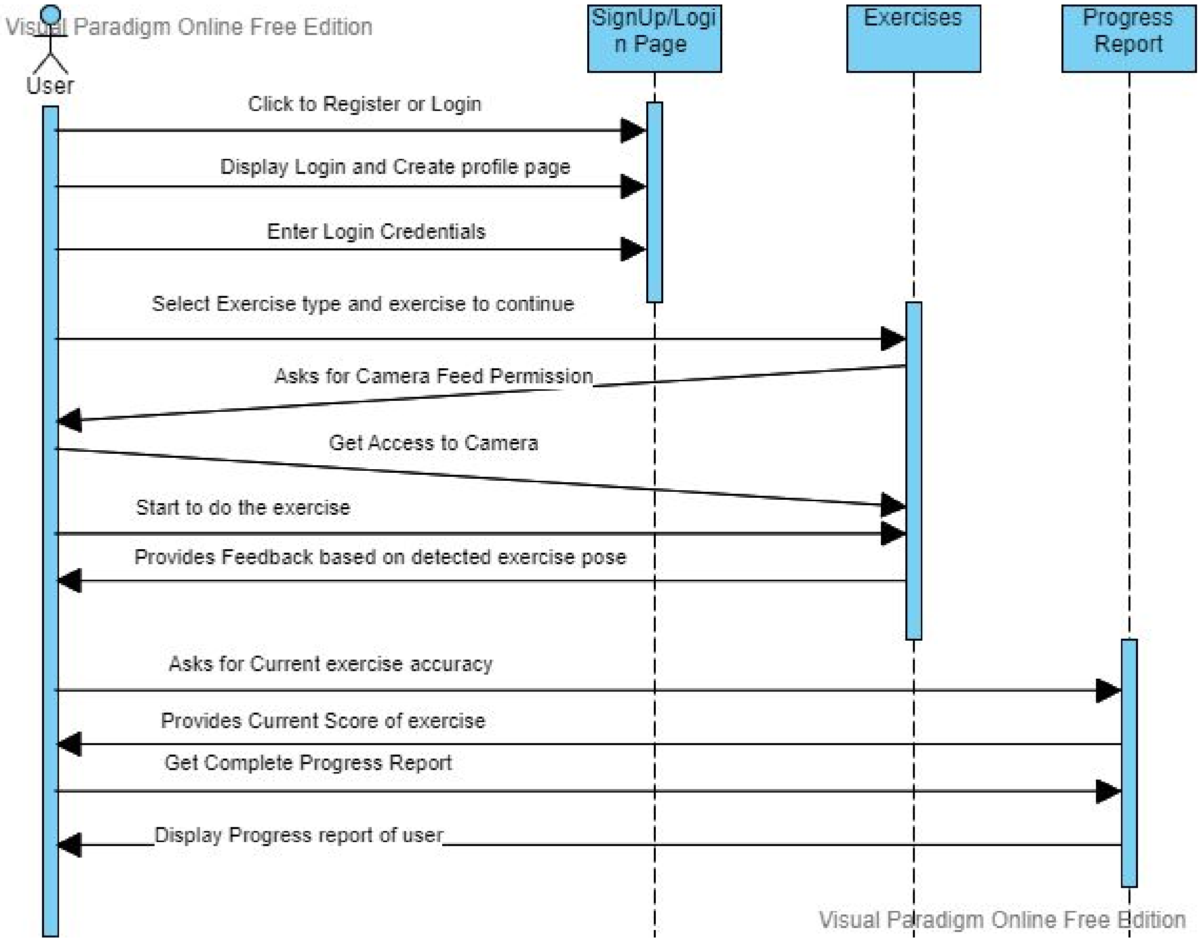


Figure 4.7: Sequence diagram of application

4.3 Algorithm and Methodologies

The app will make sure you reach your desired goal through the prescribed treatment. Both yoga and exercise are available in this app. It is recommended to follow the routine of exercises already chosen by the user. It utilises posedetection and angle detection for checking if the user is exercising correctly. The app can create a progress report showing how many repetitions and exercises have been performed throughout the day and show it in a dashboard to share with others. Yoga is a popular exercise, which combines physical and mental discipline with meditative and emotional intelligence. If yoga is selected then again with help of posenet and classification model it is further classified if performed pose is correct or not.

Posesnet is a stand-alone detection module that uses advanced convolutional neural networks to detect human poses in video and images. It is well-suited for capturing the current state of the art pose recognition technology, while being engineered with the flexibility to innovate further on future use cases.Posenet is a deep learning tensorflow model that predicts human pose by estimating the parts of the body designated as key points(which are 17 in total for this model) namely being nose,right elbow,right wrist etc. The pose estimation algorithm happens when the input RGB image is processed through a Convolutional neural network on which either multiple poses or single pose algorithms are applied by outputting the pose in the image, its pose confidence scores,keypoints positions and its key points confidence scores.

Angle Detection X and Y axis are used to describe position of a body part, but angles are used to describe the movement of different body parts. To convert x y points to lines Example - Left armpit angle — using the left shoulder, left elbow, and left hip Right armpit angle — using the right shoulder, right elbow, and right hip Left shoulder angle — using the left shoulder, right shoulder, and left hip Right shoulder angle — using the right shoulder, left shoulder, and right hip

Classification model A classification is a technique where we categorize data into a given number of classes. The classification is based on similarity or distance. The main goal of classification problems is to identify the category/class to which a new data will fall under. ml5.js is an open source machine learning library for the web. As a result, the folks building TensorFlow.js figured out that it is possible to use the web browser's built in graphics processing unit (GPU) to do calculations that would otherwise run very slowly using a central processing unit (CPU). At first, TensorFlow.js relied on GPU support in Webkit.

5. Implementation

5.1 Stages of Implementation

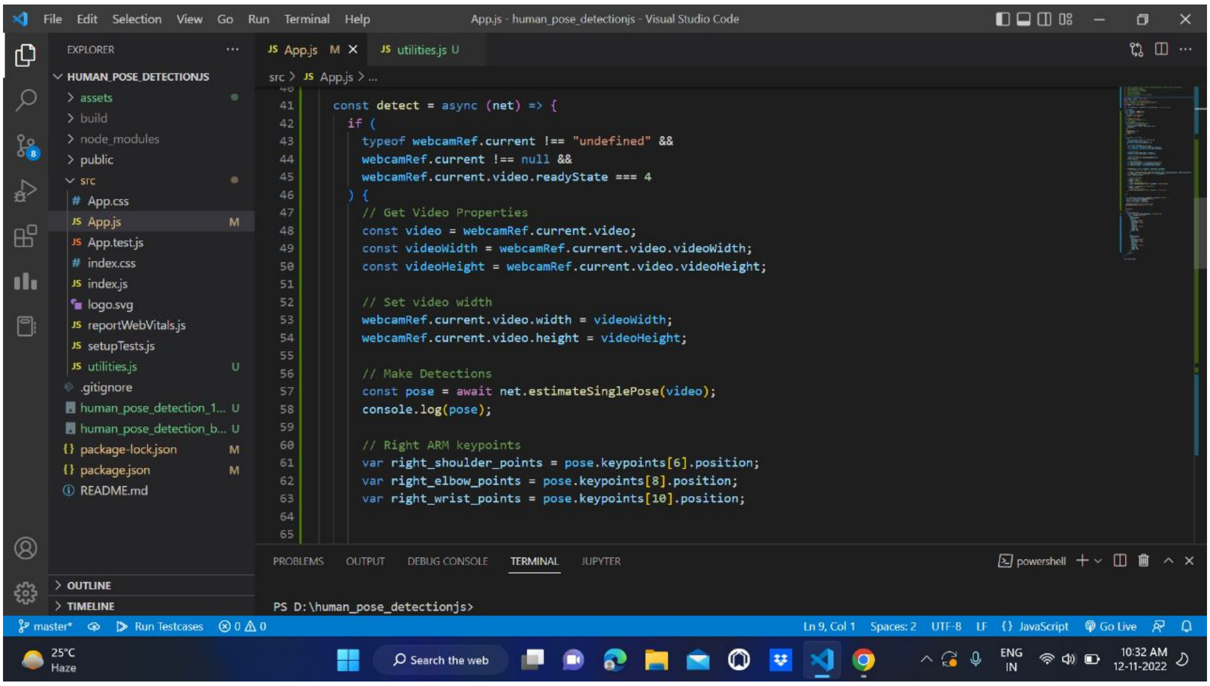


Figure 5.1: Code of detection of pose

This code helps us to detect patient poses during exercises in order to perform them. It detects the poses through video as input and processes it.

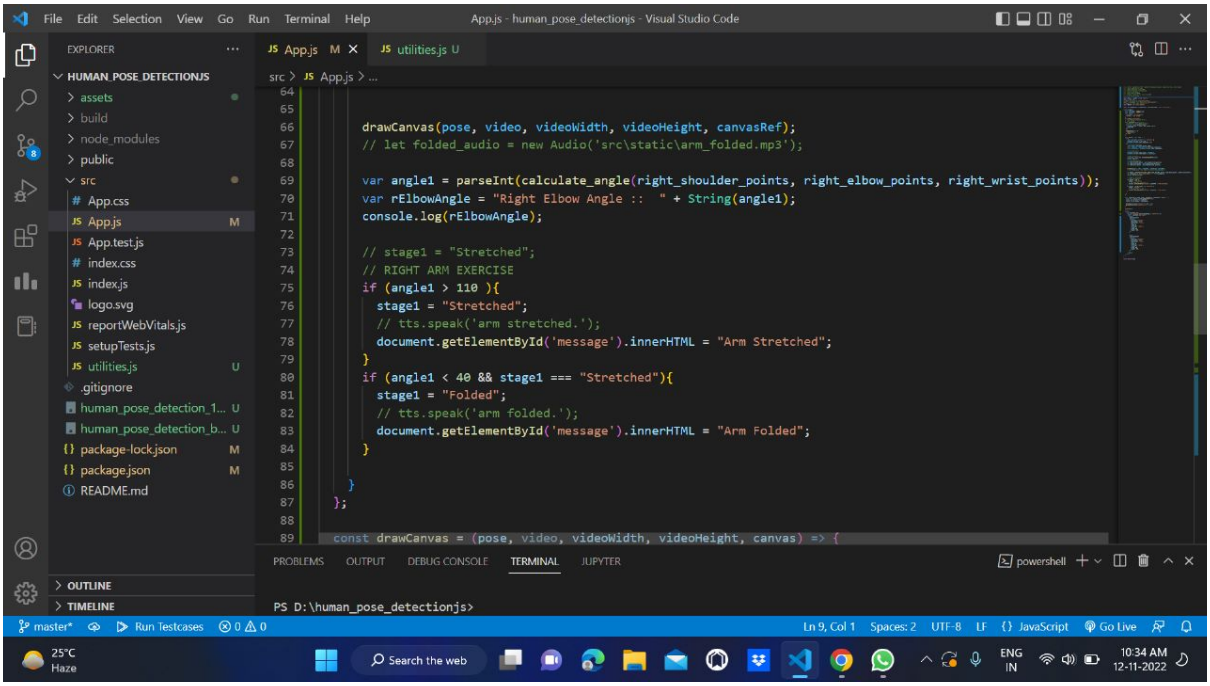


Figure 5.2: Canvas code to display message

This block of code generates a canvas to display whether the patient's arm is folded or stretched. If the angle posed by the patient is greater than 100 degrees, then the arm is said to be in an stretched position; otherwise, it is considered to be in a folded position. After finalizing the model by testing it on web platform, we have decided to integrate the Movenet model with the Android and followed the below steps for integration and testing:

1. To use the Movenet model in an Android app using Kotlin, we start by integrating the model into our project. We can download the pre-trained Movenet model, available in TensorFlow Lite format, from the official TensorFlow Lite model repository.

2. TensorFlow Lite is a lightweight version of the TensorFlow framework specifically designed for mobile and embedded devices. It enables efficient deployment of machine learning models on Android devices. In our Android project, we include the TensorFlow Lite library and dependencies to work with the Movenet model.

3. Once the model is integrated, we can use the TensorFlow Lite API to load the Movenet model and perform inference. In Kotlin, we typically use the TensorFlow Lite Interpreter class to load the model and run inference on input data.

4. Movenet requires image data as input to perform pose detection. We need to capture images from the device's camera or obtain images from other sources, such as image files or video frames. The input images should be appropriately preprocessed, such as resizing and normalization, to match the requirements of the Movenet model.

5. Using the TensorFlow Lite Interpreter, we feed the preprocessed image data as input to the Movenet model and invoke the interpreter's run() method to perform inference. The model will process the input image and output pose estimation results.

6. After running inference, we can extract the pose estimation results from the model's output. Movenet provides information about key body joints and their respective positions in the input image. We can access and utilize this data to detect body stiffness, track movements, or provide real-time feedback during exercises.

7. Finally, we can visualize the pose estimation results by overlaying them on the input image or displaying them in a user interface. This allows users to see the detected body poses and provides a more interactive experience within our application.

5.1.1 Implementation of Modules

Implementation Issues -

1. It lags in mobile devices sometimes hence it was challenging to develop a code which reduces the lag created.

2. We were trying to generate audio output as well but text to speech implementation was quite difficult.

3. For angle detection we had to select trial and error method for selection of modules which was time consuming.

Techniques -

1. Pose Detection

2. Angle Detection

3. Classification model - ml5.js

5.2 Experimentation Setup

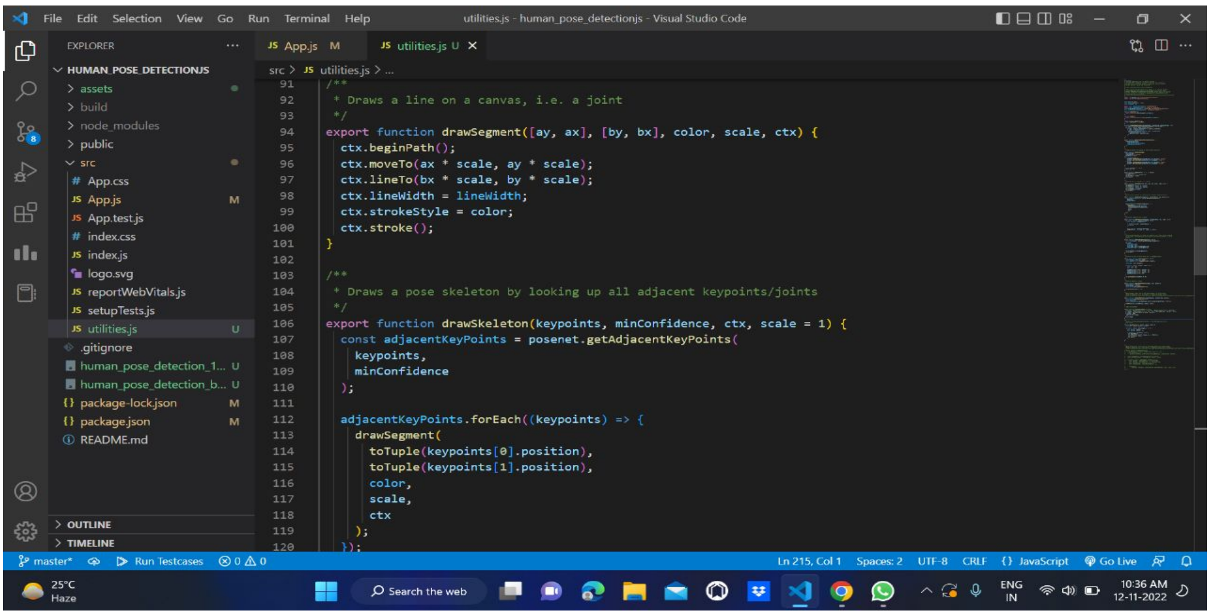


Figure 5.3: B

lock of code to draw segments and skeletons

This code helps us to generate code to draw skeleton layout of humans for angle detection

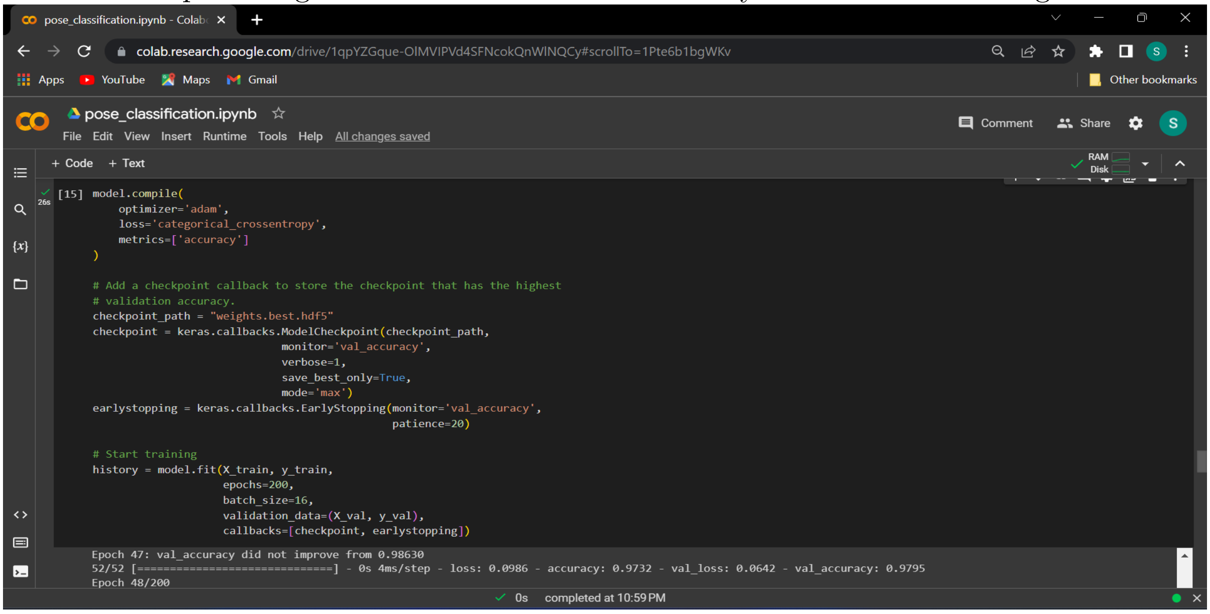


Figure 5.4: Block of code to define the Movenet model

This code helps us to execute the movenet pose detection algorithm.

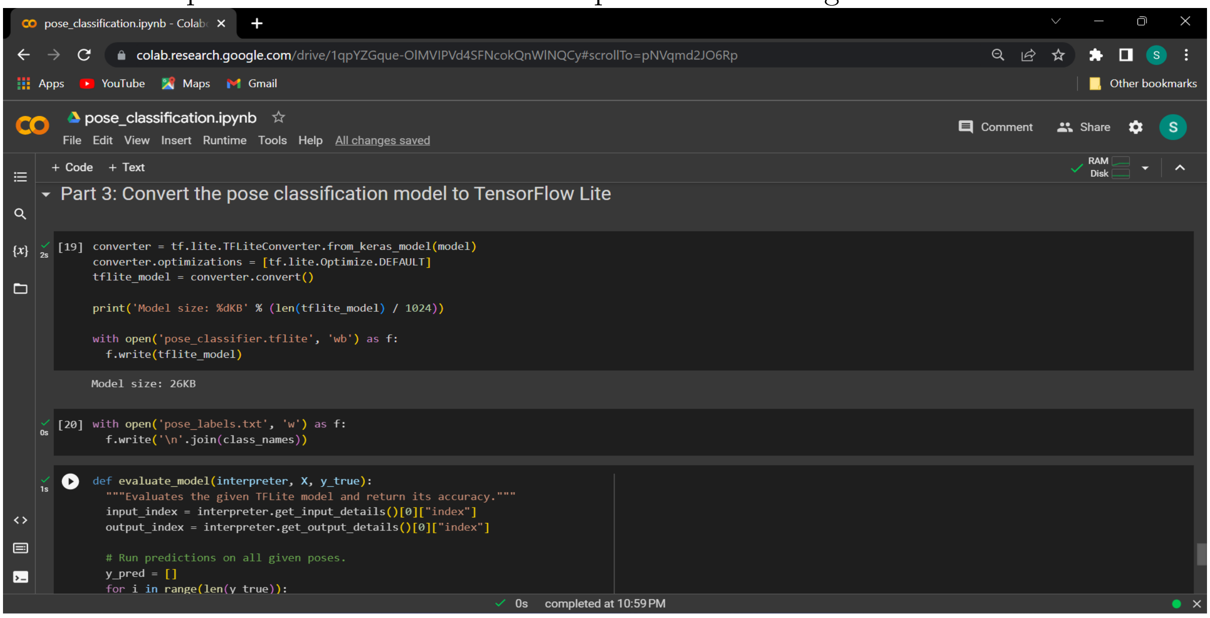
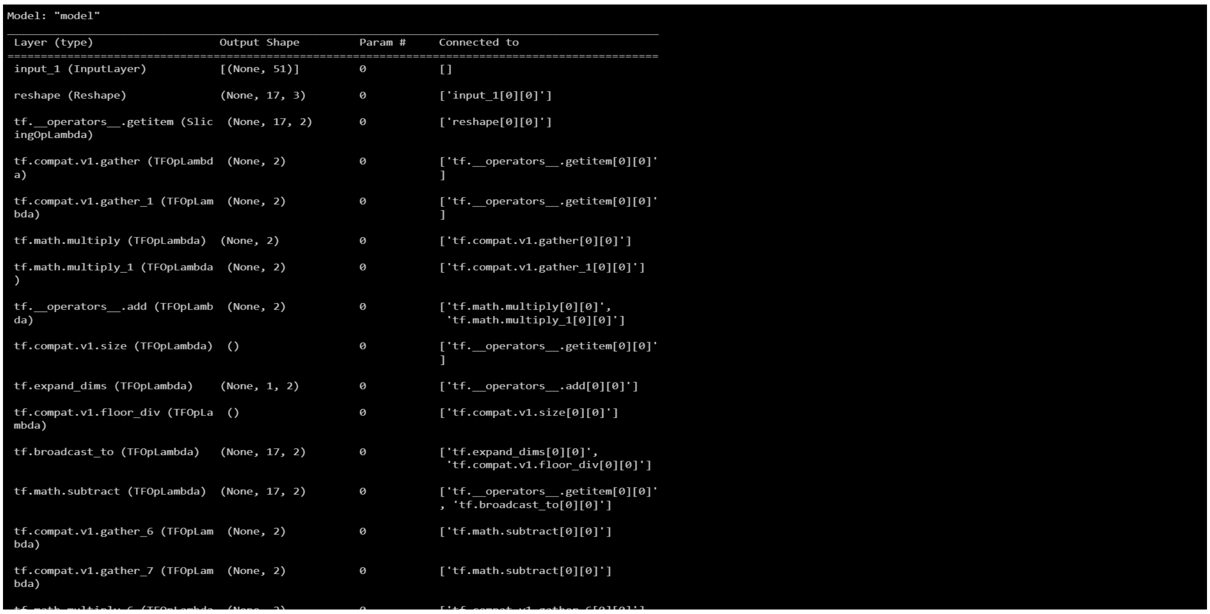


Figure 5.5: Block of code to convert the Movenet model into tensorflow lite

This code helps us to convert the movenet pose detection model into Tensorflow lite so that it can be easily integrated with android and execute the model smoothly.



```
Model: "model"
_______________________________________________________________________________________
 Layer (type)                Output Shape         Param #     Connected to
=======================================================================================
 input_1 (InputLayer)        [(None, 51)]         0           []

 reshape (Reshape)           (None, 17, 3)        0           ['input_1[0][0]']

 tf.__operators__.getitem (Slic  (None, 17, 2)    0           ['reshape[0][0]']
 ingOpLambda)

 tf.compat.v1.gather (TFOpLambd  (None, 2)        0           ['tf.__operators__.getitem[0][0]'
 a)                                                           ]

 tf.compat.v1.gather_1 (TFOpLam  (None, 2)        0           ['tf.__operators__.getitem[0][0]'
 bda)                                                         ]

 tf.math.multiply (TFOpLambda)  (None, 2)         0           ['tf.compat.v1.gather[0][0]']

 tf.math.multiply_1 (TFOpLambda  (None, 2)        0           ['tf.compat.v1.gather_1[0][0]']
 )

 tf.__operators__.add (TFOpLamb  (None, 2)        0           ['tf.math.multiply[0][0]',
 da)                                                           'tf.math.multiply_1[0][0]']

 tf.compat.v1.size (TFOpLambda)  ()               0           ['tf.__operators__.getitem[0][0]'
                                                              ]

 tf.expand_dims (TFOpLambda)    (None, 1, 2)      0           ['tf.__operators__.add[0][0]']

 tf.compat.v1.floor_div (TFOpLa  ()               0           ['tf.compat.v1.size[0][0]']
 mbda)

 tf.broadcast_to (TFOpLambda)   (None, 17, 2)     0           ['tf.expand_dims[0][0]',
                                                               'tf.compat.v1.floor_div[0][0]']

 tf.math.subtract (TFOpLambda)  (None, 17, 2)     0           ['tf.__operators__.getitem[0][0]'
                                                              , 'tf.broadcast_to[0][0]']

 tf.compat.v1.gather_6 (TFOpLam  (None, 2)        0           ['tf.math.subtract[0][0]']
 bda)

 tf.compat.v1.gather_7 (TFOpLam  (None, 2)        0           ['tf.math.subtract[0][0]']
 bda)
```

Figure 5.6: Movenet Model Layers

This represents the Movenet model different layers which shows the convolutional neural network where the image processing is done.

6. Results

6.1 Results of Experiments

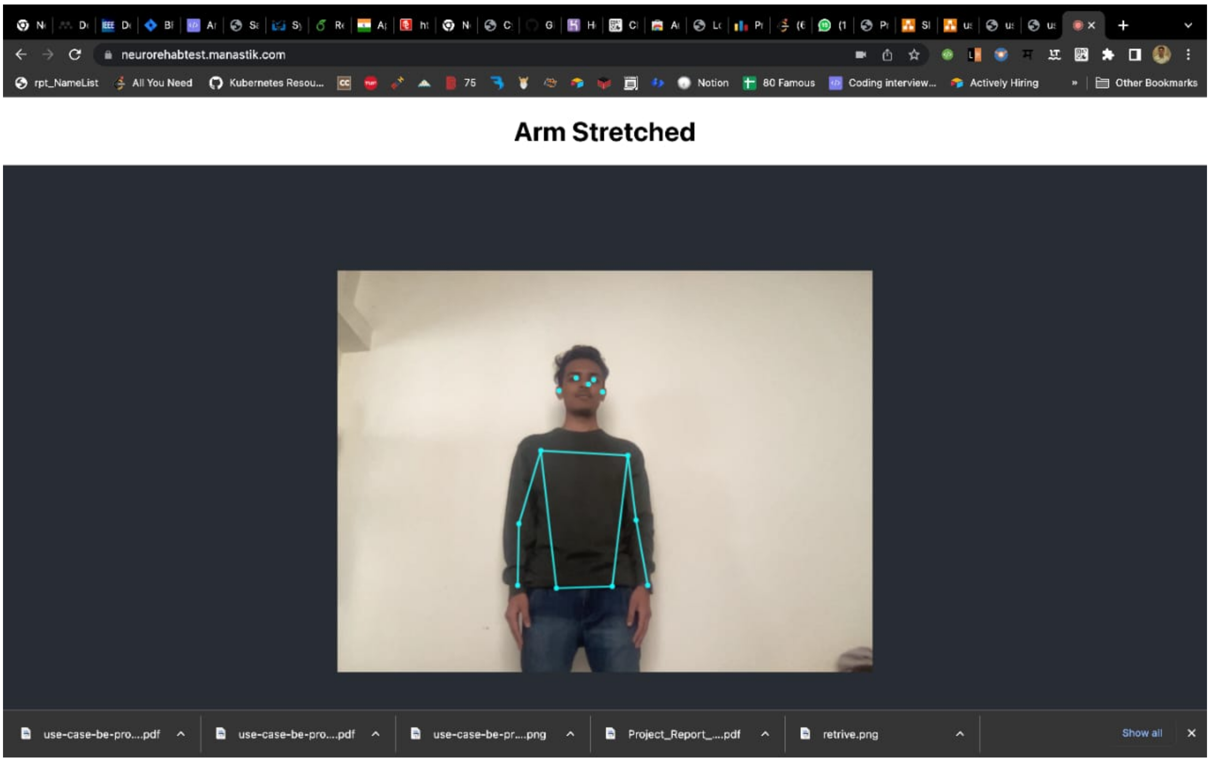


Figure 6.1: Stretching exercise for arm stretched

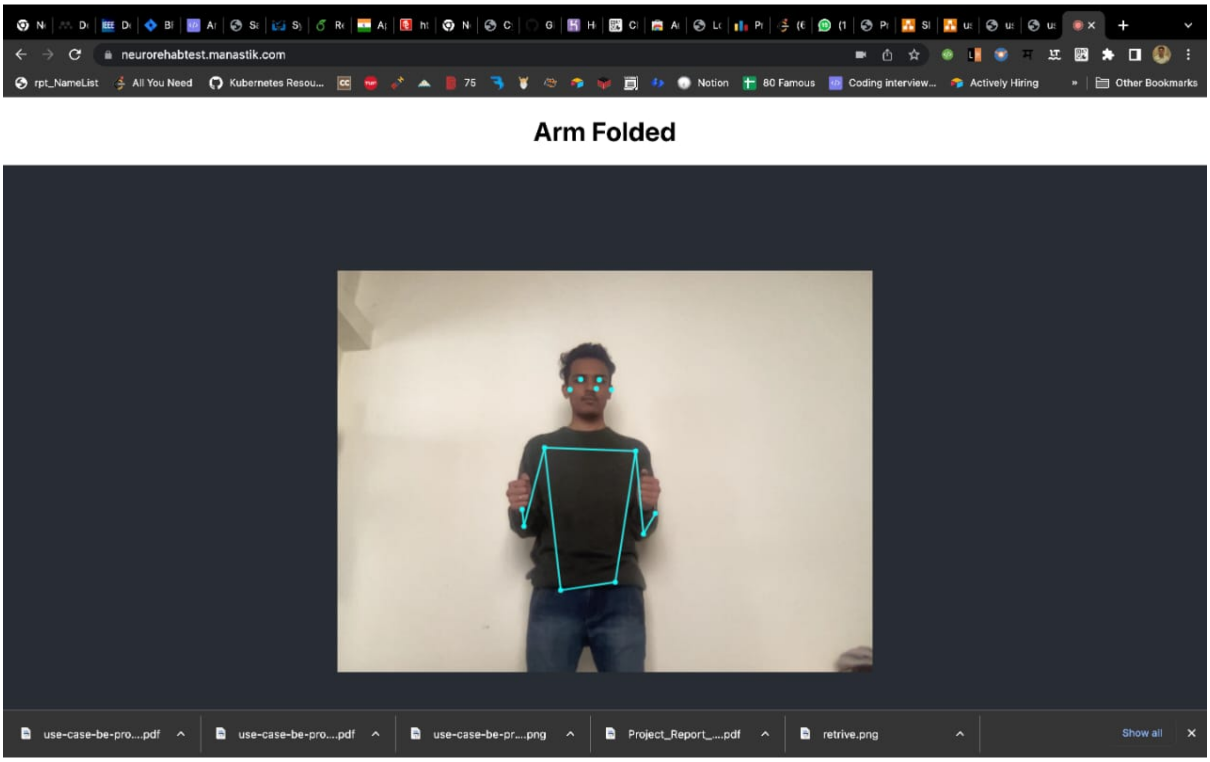


Figure 6.2: Stretching exercise for arm folded

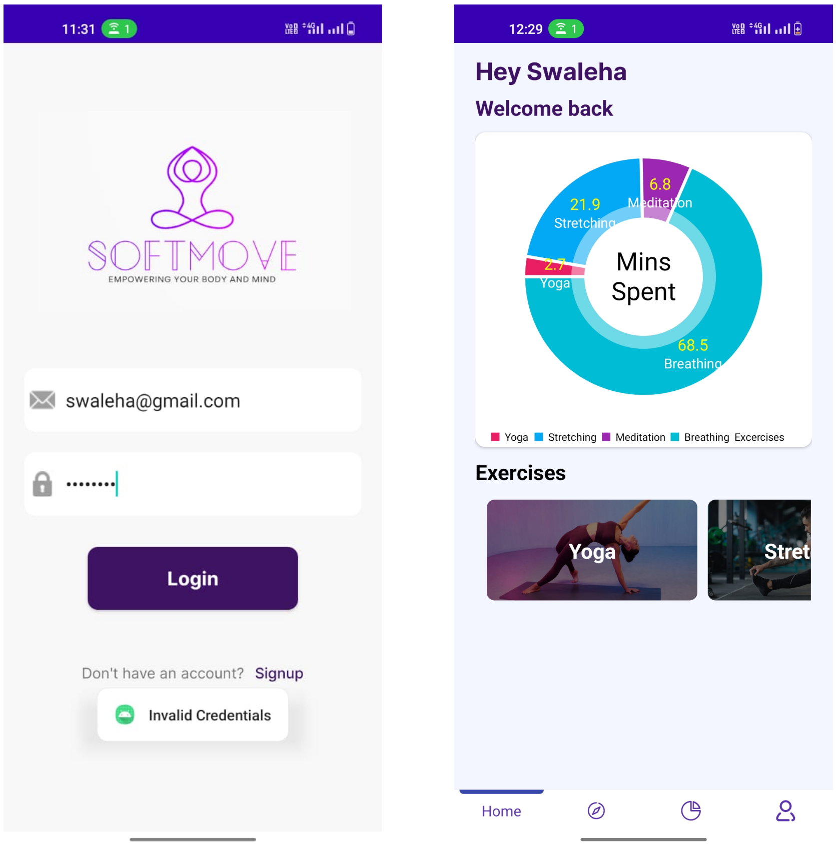


Figure 6.3: Login and Home Screen

6.2 Result Analysis

The app is capable of detecting whether the arm is folded or not. It utilizes pose detection techniques to analyze the positions and orientations of the user's body parts, including the arms. By analyzing the detected pose, the app can determine if the user's arm is folded or in an unfolded position. This information can be used to provide feedback and guidance on correct exercise execution or yoga pose alignment

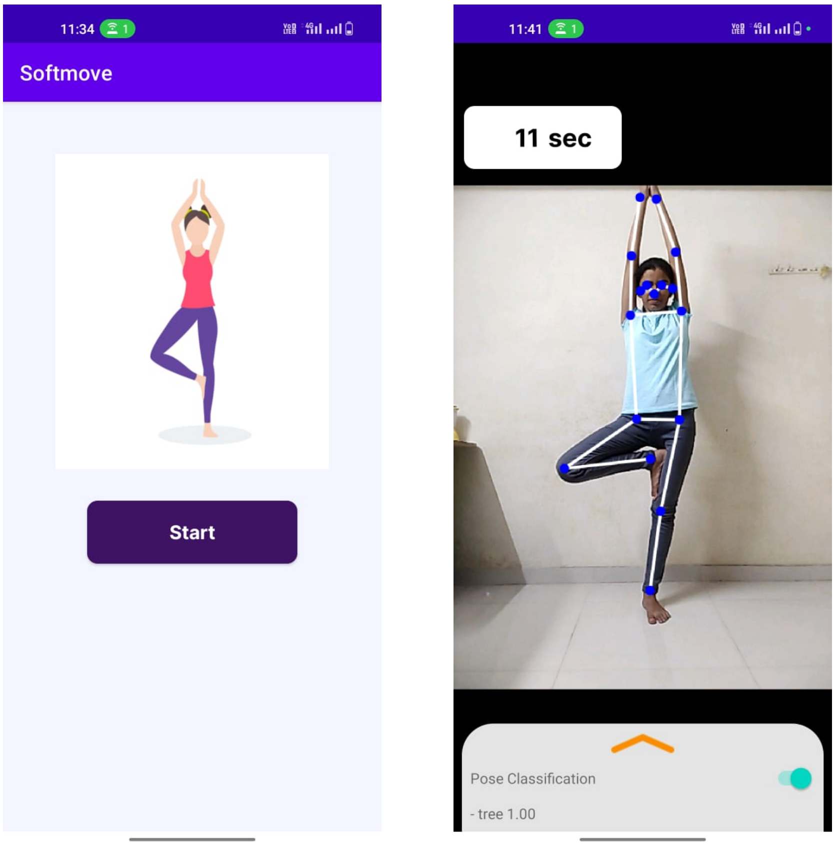


Figure 6.4: Tree pose performed by human as per instruction

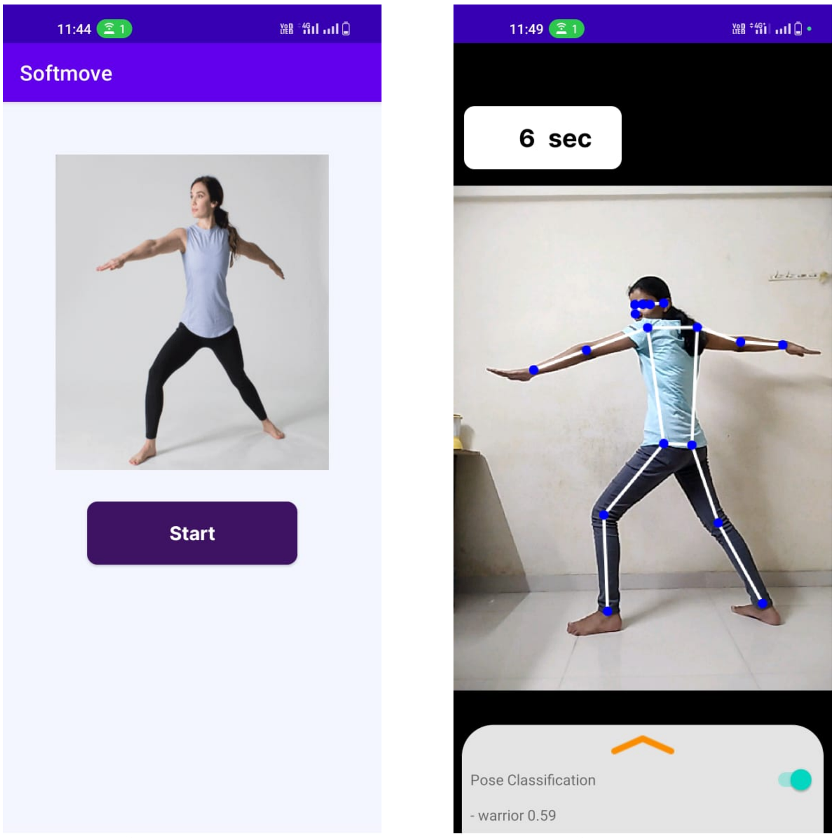


Figure 6.5: Warrior pose performed by human as per instruction

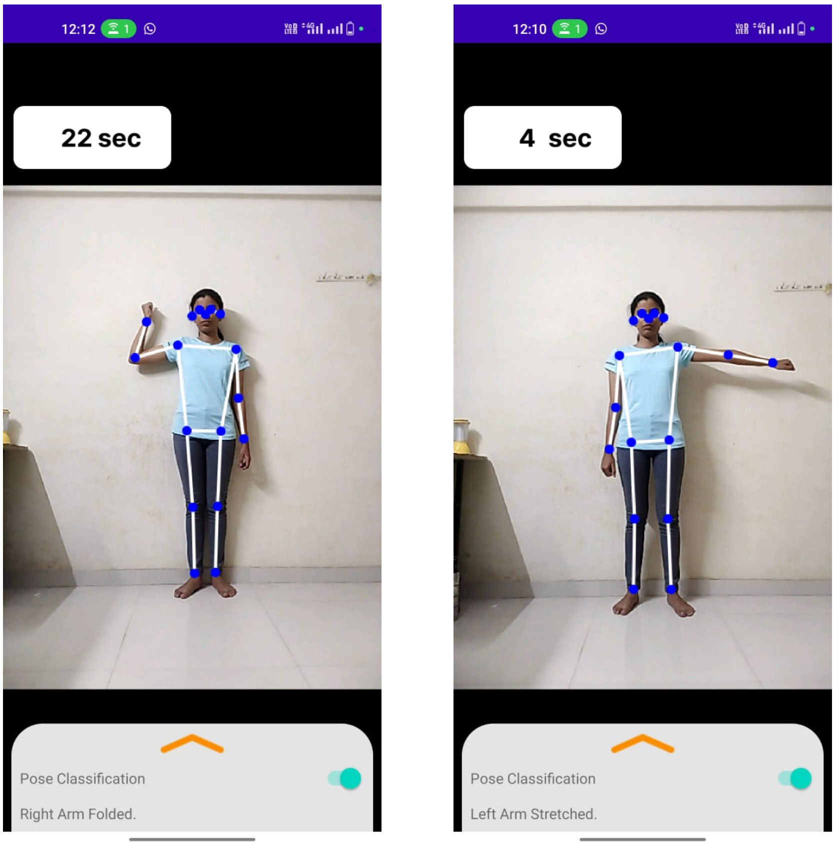


Figure 6.6: Hand stretching exercise

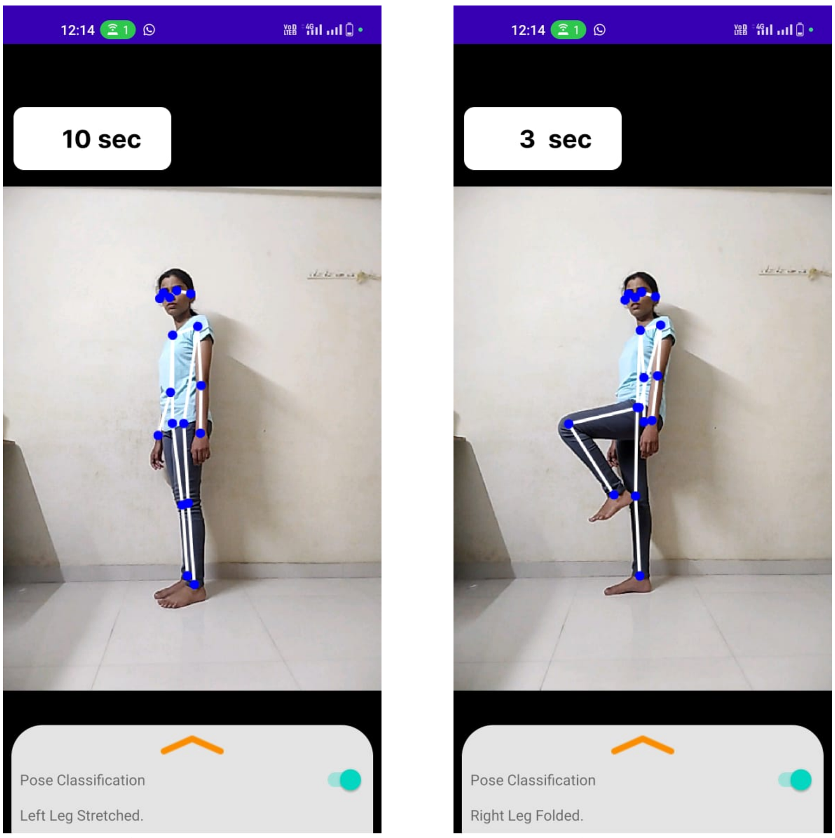


Figure 6.7: Leg stretching exercise

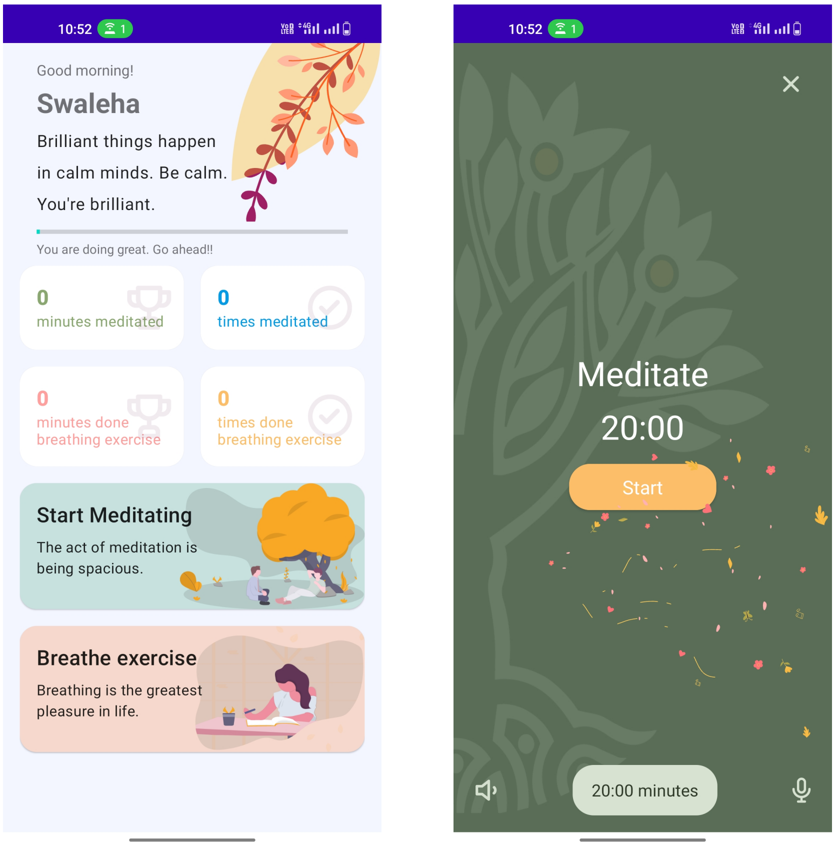


Figure 6.8: Dashboard and Meditation screen

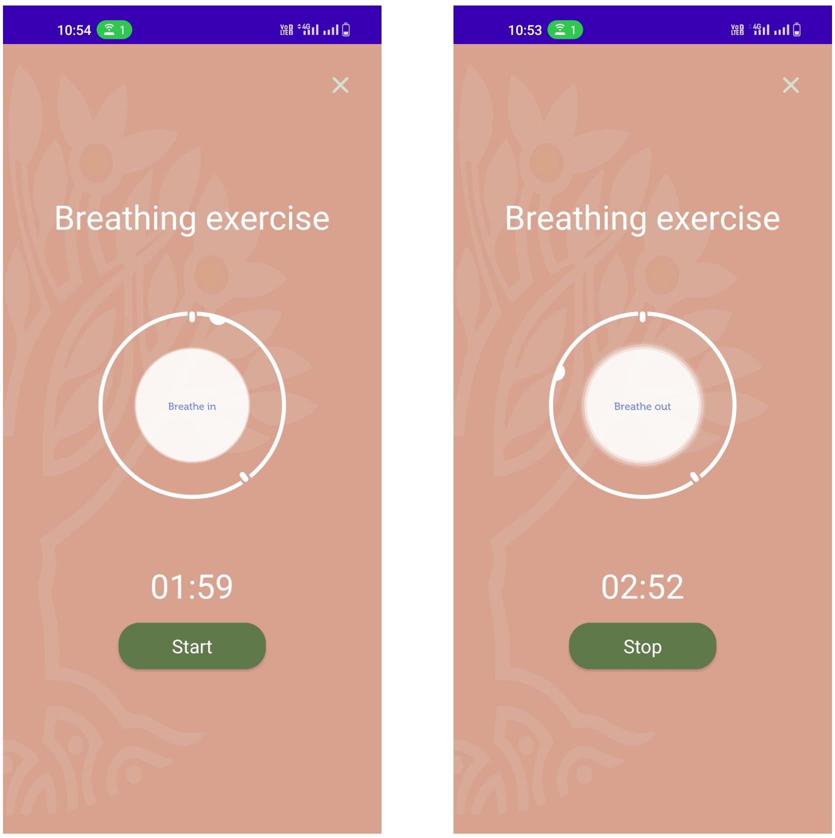


Figure 6.9: Breathing Exercise

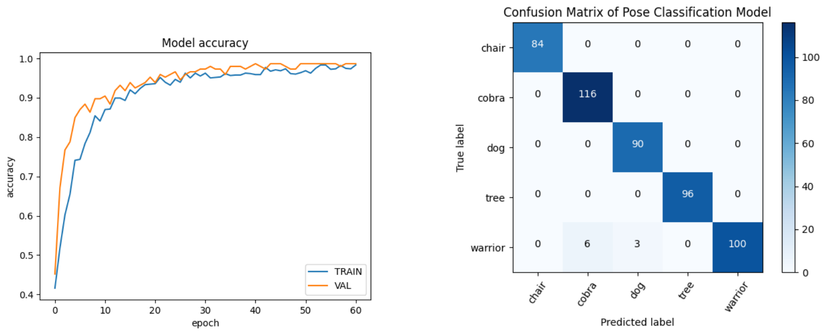


Figure 6.10: Analysis of Code

6.3 Testing

White Box

Unit testing on different modules :

Unit Testing

Sr. No	Description	Expected result	Actual result	Result
1	Speed of detection of poses	Within seconds	Fast	PASS
2	Points detected accurately	Yes	Yes, With lines	PASS
3	If human not present, does it detect points	No	No	PASS

Figure 6.11: Table for White Box testing

Black Box

Sr. No.	Test Case	Input	Expected output	Actual Output	PASS/FAIL
1	Exercise recommendation accuracy	User's specific body stiffness issues	AI application suggests targeted exercises and stretches that effectively address the user's stiffness issues.	It does suggests	PASS
2	Exercise progression	User's current exercise routine and progress	AI application adjusts the exercise plan, gradually increasing difficulty and intensity to ensure continued improvement and prevent plateaus in reducing stiffness.	App updates the plan accordingly with help of doctors	PASS
3	User feedback analysis	User's feedback on the effectiveness of the recommended exercises	AI application analyzes user feedback to identify patterns and make adjustments to the exercise plan for better results in alleviating body stiffness.	Runtime progress is shared with doctors to update the exercises and record the progress.	PASS
4	Initial stiffness level	User's initial stiffness level	AI application recommends appropriate exercises and stretches based on the stiffness level.	It does not suggest automatically	FAIL
5	Exercise duration and frequency	User's desired exercise duration and frequency	AI application generates a personalized exercise plan that aligns with the user's preferences and provides gradual improvement in reducing stiffness.	It aligns with doctors recommendations	PASS

Figure 6.12: Table for Black Box testing

Summary of Black box testing :

No. of test cases pass: 04

No. of test cases fail: 01

7. Conclusion and Future Scope

7.1 Conclusion

To build an application where doctors can prescribe exercises and track progress reports. Design a system that lets the user enter his/her health condition and then asks him/her to select from a list of different types of training exercises. After selecting one, the system will suggest a personalized program that includes recommended volume and intensity for each exercise. The user can also modify the program if he/she wants to change any of the variables. The program will then be saved in a database. Further we can also add blogs related to how these exercise can ease up and help certain diseases.

7.2 Limitations of the Project

While AI applications utilizing the Movenet pose detection model can offer valuable assistance in addressing body stiffness in Parkinson's disease, there are some limitations to consider:

1. Sensitivity to Environmental Factors: Movenet and similar pose detection models can be affected by various environmental factors, such as lighting conditions, camera quality, and occlusions. If the input image quality is compromised, it may impact the accuracy of the pose detection results, leading to potential errors in exercise detection and feedback.

2. Limited Contextual Understanding: Pose detection models like Movenet primarily focus on tracking joint positions and movements. While they can identify specific body postures and movements associated with body stiffness, they may lack a deeper understanding of the overall context and subtleties of Parkinson's disease. This limitation may restrict the model's ability to provide nuanced feedback or adapt to individual patient needs.

3. Lack of Individualization: AI applications using Movenet typically rely on general

models trained on diverse datasets. However, each Parkinson's patient experiences unique symptoms and movement patterns. The lack of personalized training data can limit the model's ability to adapt to individual variations in body stiffness and exercise requirements. Personalized exercise plans may require additional customization beyond what pose detection models alone can provide.

4. Inability to Capture Internal Sensations: Body stiffness in Parkinson's disease is not solely manifested through external movement patterns. Patients may experience internal sensations related to stiffness and rigidity that are not directly detectable through pose detection models. Therefore, AI applications based on Movenet may not fully capture the subjective experiences of patients, which are important for comprehensive assessment and management.

5. Lack of Domain Expertise: AI models, including pose detection models like Movenet, lack the expertise and clinical judgment of healthcare professionals specializing in Parkinson's disease. While AI can assist with exercise guidance and monitoring, it should not replace the role of healthcare providers who can consider a broader range of factors, conduct comprehensive assessments, and make informed decisions regarding treatment plans.

6. Technical Limitations: The performance of AI applications utilizing Movenet or similar pose detection models can be affected by computational limitations on mobile devices. The processing power, memory, and battery constraints of smartphones may limit the real-time capabilities and accuracy of the model. Additionally, continuous improvements and updates to the model are necessary to address any limitations and enhance performance.

7.3 Future Scope

In future scope we aim to add more functionalities like counting steps, Aerobics where it can very effective application if we will add the below features and consider the situations such as:

1. Improved Accuracy and Reliability: As AI technology advances, pose detection models such as Movenet are likely to become more accurate and reliable. Future iterations may overcome limitations such as environmental sensitivity and occlusion issues, leading to more precise detection of body stiffness and exercise performance.

2. Integration with Wearable Devices: Wearable devices, such as smartwatches or motion sensors, can provide continuous monitoring of movements and body posture. AI applications can leverage data from these devices in conjunction with pose detection models to provide real-time feedback, track progress, and customize exercise plans based on real-time movement data.

3. Machine Learning for Personalization: Machine learning techniques can be employed to analyze a wealth of data collected from individuals with Parkinson's disease. By combining pose detection information with patient-specific characteristics, such as disease progression, medication intake, and response to exercises, AI algorithms can personalize exercise plans and treatment strategies to better address individual needs.

4. Virtual Rehabilitation and Telemedicine: The integration of AI with virtual reality (VR) or augmented reality (AR) technologies can create immersive rehabilitation environments. Patients can perform exercises in virtual settings that provide real-time feedback, guidance, and motivation. Additionally, AI-powered telemedicine platforms can enable remote consultations and monitoring, allowing healthcare professionals to assess body stiffness remotely and make necessary adjustments to treatment plans.

5. Collaborative Healthcare: AI applications can facilitate collaboration between patients, caregivers, and healthcare professionals. By integrating pose detection models with mobile apps or web platforms, individuals with Parkinson's disease can share exercise data, receive feedback, and engage in virtual support communities. Healthcare providers can remotely monitor progress and provide timely interven-

tions, fostering a more collaborative and patient-centered approach to managing body stiffness.

Overall, the future scope of AI applications for body stiffness in Parkinson's disease using pose detection models like Movenet is likely to involve advancements in accuracy, personalization, prediction, and integration with wearable devices and immersive technologies. These developments can contribute to more effective and accessible interventions, enhanced patient engagement, and improved outcomes in the management of body stiffness in Parkinson's disease.

Bibliography

[1] I. Pernek, G. Kurillo, G. Stiglic, R. Bajcsy, “Recognizing the intensity of strength training exercises with wearable sensors,” J. Biomed. Inform., vol. 58, no. C, pp. 145-155, 2015.

[2] G. Prabhu, A. Ahmadi, N. O’Connor, K. Moran, “Activity recognition of local muscular endurance (LME) exercises using an inertial sensor,” Proc. IACSS, pp. 35-47, 2018.

[3] M. Gandomkar, R. Sarang, Z. Gandomkar: TrainingPal, “An algorithm for recognition and counting popular exercises using smartphone sensors,” Proc. ICEE, pp. 1471-1476, 2018.

[4] D. Deponti, D. Maggiorini, C. E. Palazzi: DroidGlove, “An android based application for wrist rehabilitation,” Proc. ICUMT, pp. 1-7, 2009.

[5] VAY: Vay Fitness coach, https://www.vay-sports.com/ (Accessed on 07/03/2020).

[6] Kaia Health: Personal Trainer, https://www.kaiahealth.com/ (Accessed on 07/06/2020).

[7] KP. Ma et al, ”A Deep Learning Approach for Parkinson’s Disease Diagnosis Based on Sequential Pose Estimation”, IEEE Journal of Biomedical and Health Informatics, 2020.

[8] R. S. Paes et al, ”Parkinson’s Disease Classification Using Deep Learning with Movenet Model”, 2021 IEEE Latin American Symposium on Circuits and Systems (LASCAS), 2021.

[9] [9] L. Zhao et al, "Automatic Detection of Bradykinesia and Rigidity in Parkinson's Disease Using a Pose Tracking Approach", Sensors, 2019.

[10] S. S. Nayak et al, "Automated Detection of Motor Symptoms in Parkinson's Disease Using Pose Estimation and Recurrent Neural Networks", Journal of Medical Systems, 2020.