University of Mumbai

Smart Fitness Assistant using Artificial Intelligence and Django

Submitted in partial fulfillment of requirements

For the degree of

Bachelor of Technology

by

Pushti Ranpura (Roll No: 1813044)

Vidhi Sejpal (Roll No: 1813048)

Akshit Tayade (Roll No: 1813106)

Guide

Prof. Ankit Khivasara

Department of Electronics and Telecommunication Engineering

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Batch 2018 -2022

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate

This is to certify that the dissertation report entitled **Smart Fitness Assistant using Artificial Intelligence and Django** is bona fide record of the dissertation work done by Pushti Ranpura, Vidhi Sejpal and Akshit Tayade in the year 2021-22 under the guidance of Prof. Ankit Khivasara of Department of Electronics and Telecommunication Engineering in partial fulfillment of requirement for the Bachelor of Technology degree in Electronics and Telecommunication Engineering of University of Mumbai.

Guide

Head of the Department

Principal

Date:

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate of Approval of Examiners

We certify that this dissertation report entitled **Smart Fitness Assistant using Artificial Intelligence and Django** is bona fide record of project work done by Pushti Ranpura, Vidhi Sejpal and Akshit Tayade. This project is approved for the award of Bachelor of Technology Degree in Electronics and Telecommunication Engineering of University of Mumbai.

____

Internal Examiner

____

External Examiner

Date:

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

DECLARATION

We declare that this written thesis submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

______________________ **Signature of the Student** ______________________ **Roll No.**	______________________ **Signature of the Student** ______________________ **Roll No.**
______________________ **Signature of the Student** ______________________ **Roll No.**	

Date:

Place: Mumbai-77

Abstract

The ultimate goal of every human being is to be physically and mentally fit. The pioneers of fitness application development recognized this requirement. Nowadays, virtual assistants play a significant role in our daily lives and have become an inseparable part of them. People who workout at home by themselves, have a higher risk of getting injured because of their incorrect posture and form. This is because of a lack of proper training. With this project, we have used Artificial Intelligence to explore of AI-based workout assistants.

In our project, we propose Fitlife.ai, an application that recognizes the user's workout posture, counts the reps accomplished, and provides detailed recommendations on how to improve their workouts. The application uses the MediaPipe to detect a person's posture, then analyses the posture's geometry and maps it to the threshold angles for exercise correction. After a period of exercise, a machine learning model can forecast diet type based on the number of calories expended during exercise, the quantity of calories consumed through food, and a variety of other factors. A variety of breakfast, lunch, snack, and supper meal plans are also supplied for each individual based on their eating habits and body statistics. It also provides the user's progress in the form of data and charts, as well as the user's BMI, BMR values, and diet regimens.

In the future, the scope of the project could be expanded by adding a greater number of exercises with a training schedule on the basis of the user's body type and weight. Additionally, to provide a convenient user experience the website can be converted to a mobile application, and recipes of the recommended meals of the diet plan can be provided.

Keywords: Image Processing, Posture Correction, Machine Learning, Diet Recommendation, Django

Contents

List of Figures

List of Tables

Chapter 1
Introduction

This chapter presents the introduction of the thesis topic selected that is, the background and it defines motivation behind our topic selected and scope of thesis work. It also provides brief description of the project undertaken and organization of the report.

1.1. Background

In the scenario of the pandemic, many people who were routinely pursuing their fitness practices in gyms or on the ground were severely affected after the news of a sudden lockdown. People have been compelled to stay at home due to the closure of fitness centers and public parks, disrupting their daily fitness routines. However, there has been an intriguing shift in the Indian consumer's fitness habit.

A survey conducted by Gympik [1], having responses of over 50,000 fitness enthusiasts suggests that, the extended lockdown drove a massive surge in the demand for virtual classes for yoga (87%) and high-intensity interval training (HIIT) cardio workouts such as Zumba (72%), aerobics (67%), and pilates (22%). Moreover, an overwhelming 84% of fitness enthusiasts tried live-streaming fitness classes at least thrice during the lockdown – marking a significant increase over the corresponding numbers in 2019, which stood at 29%. During the lockdown, 77% of Indians also tried to stay fit by combining household chores with virtual classes and DIY home workout routines.

More people are following these routines independently, gradually diminishing their reliance on gym-based workouts, thanks to the quantity of workout videos online and popular fitness apps having a dedicated training programmed area. Because of their hectic lifestyles and financial constraints, many people have taken to working out at home on their own. Furthermore, because of sanitization difficulties, people are hesitant to enter a gym during covid. Going to the gym can result in more serious situations if the workouts and equipment are not performed correctly or under professional supervision. The most prevalent problem is a backbone problem caused by misdirection, overtraining, and poor posture.

1.2. Motivation

To avoid injury, proper technique and form are essential when exercising out. Although certain exercises, such as squats, deadlifts, and shoulder presses, are helpful for health and fitness, they can cause significant damage to muscles and ligaments if performed improperly. Incorrect form can result from a lack of professional instruction through courses or a personal trainer, as well as utilizing too much weight (body pose alignments). To grasp proper form and methods, you must be taught while training.

Currently, only a trainer or other gym-goers offer assistance in following the correct posture. Personal trainers are expensive, around £50 per hour, and most people who wish to train cannot afford one. Furthermore, many gym goers are too embarrassed to ask a member of staff in a gym for advice. It's not uncommon for us to feel nervous about the gym because we don't know what we're doing. While at-home workouts are convenient, they may cause severe injuries if not performed properly. Despite having plenty of potential competitors in the market, there is no current way to inform a user how accurately a particular workout is being followed.

For this research, we want to help people complete workouts in the proper posture. Our objective is to help people avoid injuries, increase the quality of their exercises, and support them in performing these activities on their own. In addition, we want to supply them with a healthy nutrition plan in order to preserve their general well-being.

1.3. Scope of Project

This project aims to provide fitness enthusiasts of all ages with a personal fitness AI trainer to help them perform workouts on their own. Going for a desktop-first approach, we'd create a web application that anyone with an email address could use to get started. As part of this project, a full-fledged user interface and user experience will be created, supported by user research. The project will include, at a bare minimum, the ability to provide real-time performance feedback in the form of in-built captions and adapt the workout accordingly, guidance to improve posture/form, personalized workout recommendations, diet plans, and a workout tracker to summarize your performance improvement. The Smart fitness assistant should have a variety of workouts that focus on upper-body, lower-body, and core activities.

1.4. Brief Description of project undertaken

We present a system that analyzes the user's body posture during a workout and provides posture corrections to help fix this problem and offer visual feedback while conducting an exercise. Fitness AI, the smart fitness assistant, is what we call it.

Our fitness application counts reps, corrects technique, and tracks practically any workout using your device's camera and a 3D capture technology. Captions are provided as real-time feedback tailored to your performance. It's like if you had a personal trainer at your side, encouraging you to work harder and reach higher goals. It transforms your device's camera into a relentless motivator, assisting you in getting the greatest results and avoiding damage by counting squats, timing planks, and evaluating your pace. It would also gather detailed information for each workout and track your progress over time, allowing you to see real results.

We suggest a two-step method for analyzing exercise posture, which includes a count of repetitions and a recommendation for inappropriate posture. The real-time body tracking model, which extracts a total of 32 body coordinates, is the initial stage in Fitness AI. Furthermore, these body coordinates are fed into a statistical system that offers a person with exercise count and posture correction recommendations. The device will not only track the number of exercises completed, but will also monitor body posture, assisting in staying fit and healthy. The application also checks the accuracy of a user's projected stance for a particular exercise. We use machine learning models to accomplish this, with the poses and teaching of certified professionals serving as the ground truth for perfect form. The suggested system employs the K Nearest Neighbor (KNN) model, Random Forest, Deep Neural Networks (DNN), and Linear Regression for classification.

The second portion of our application allows users to calculate BMI (Body Mass Index) and BMR (Basal Metabolic Rate) from their input data using the Mifflin-St Jeor Equation. In addition, the project intends to develop a personalized diet advice system based on the user's input. Machine learning model was created using a random-forest classifier based on input values for clustering and classification of six types of diet according to their particular preferences and building a healthy lifestyle behavior.

Chapter 2

Literature Survey

This chapter presents the literature survey of the report, that is, the reason behind development of our Smart fitness Assistant using Django and Artificial Intelligence. It outlines the results from our background research and the role of Artificial intelligence in fitness and its potential challenges in our solution.

Several applications are available on the market that guide the user about the exercises they should perform. However, through our fitness website, we guide users about correct posture in addition to providing them with diet suggestions to promote a healthy lifestyle. This website can be considered as a fitness assistant that identifies a person's posture in real time and offers suggestions for improving their form while exercising. It is built for people exercising at home and can also contribute to reducing the human intervention in gyms by increasing the project scope.

2.1. Primary Research

Our research study was carried out using a simple random sample technique. Personal interviews and questionnaires were used to collect primary data from users. Secondary data were gathered from a variety of websites, journals, and publications.

With the goal of defining the target users and understanding their mental models, we surveyed 49 people who were into fitness in the last few months to over a year. The majority participants belonged to the age group of 18-35 years. This helped us validate the survey results from our white paper research and helped us elucidate the problem space. Furthermore, we conducted another survey to understand target users' dietary habits and scope of improvement in the current methods. We collected responses from 30 people who were between the ages 18-35 years old. Presented below is some interesting data we gathered.

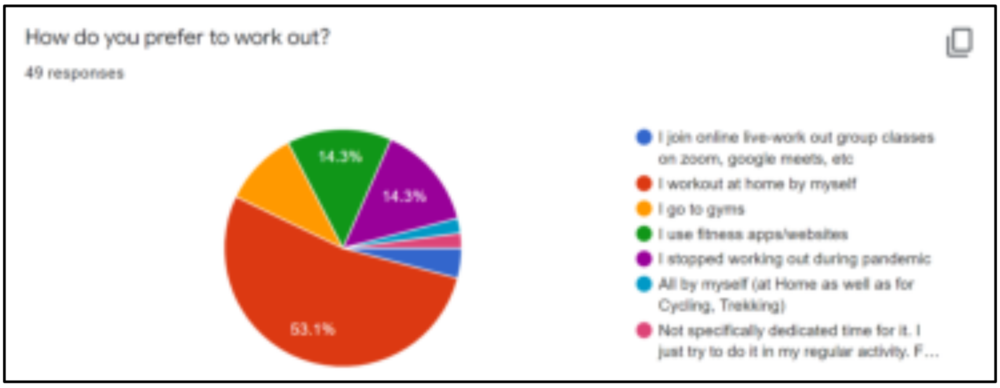


Fig 2.1 Workout preferences

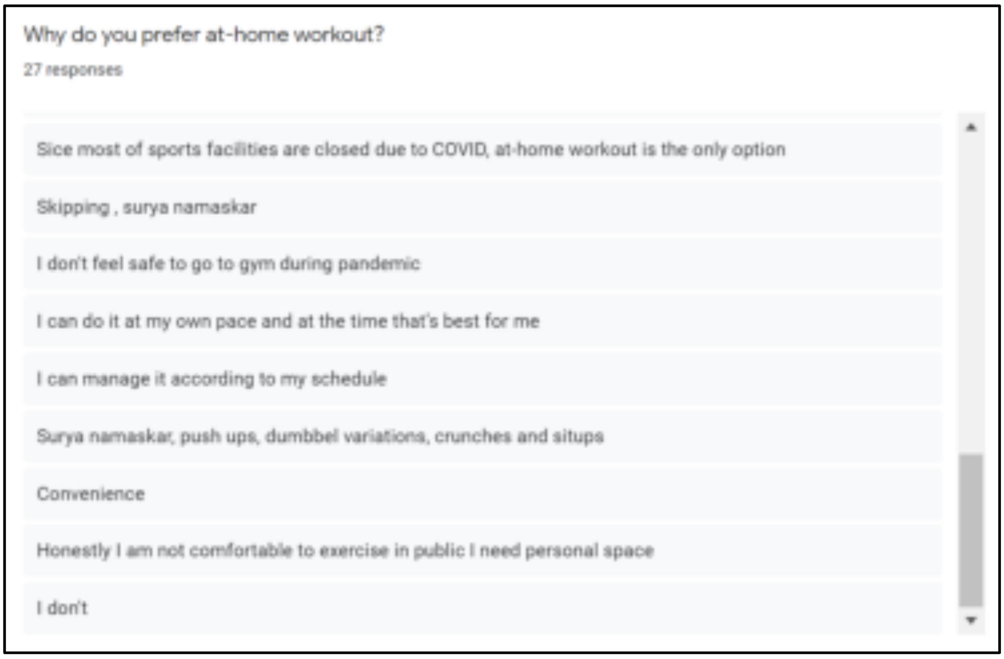


Fig 2.2 Home-Workout preferences

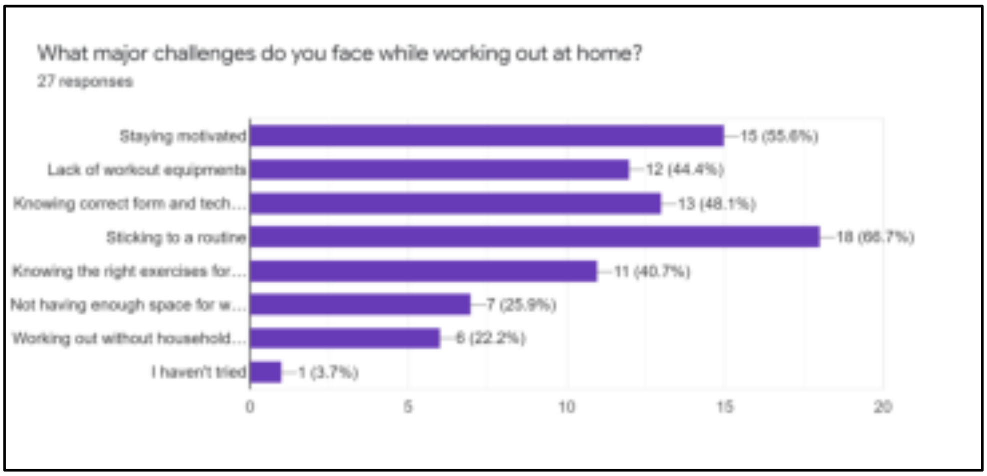


Fig 2.3 Bar chart of challenges faced at home workout

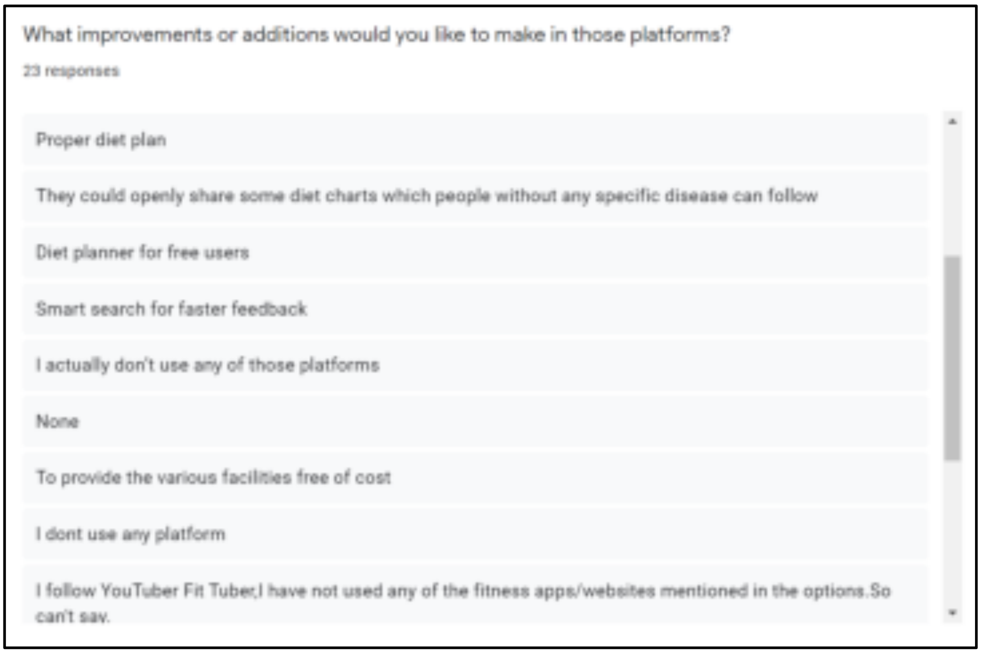


Fig 2.4 Suggestions for online platforms

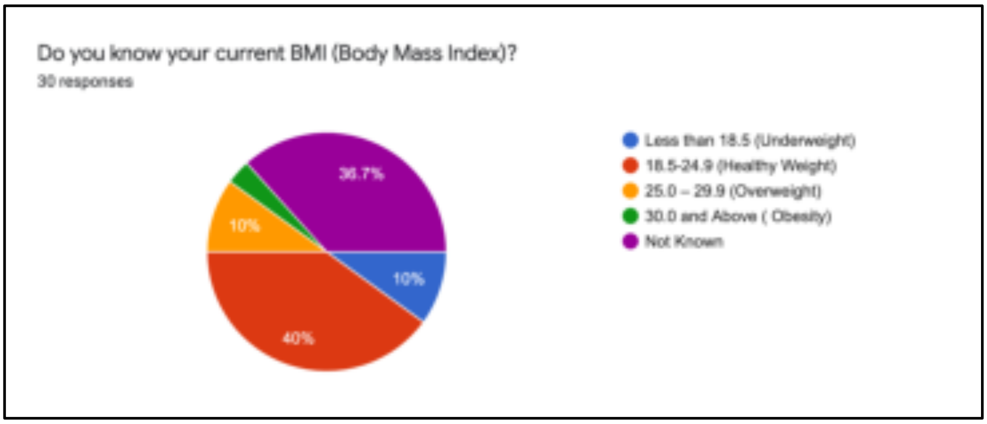


Fig 2.5 Knowledge about BMI (Body Mass Index)

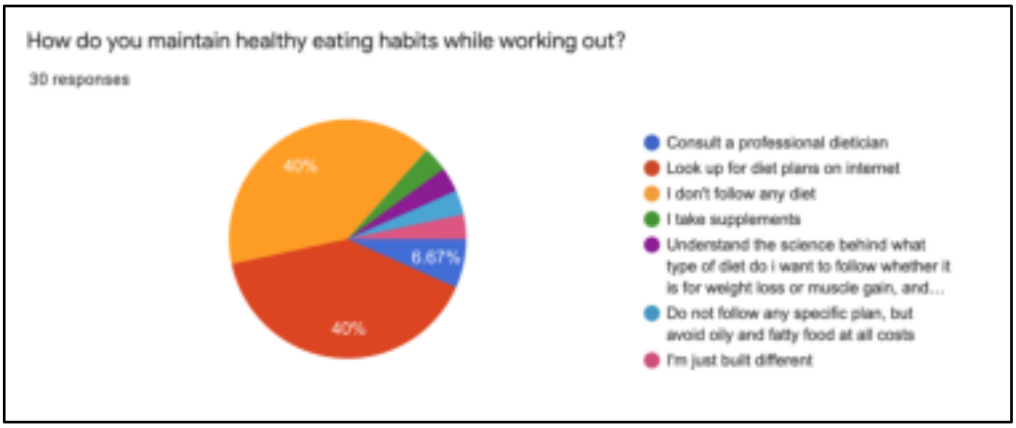


Fig 2.6 Personal healthy routine

2.2. Artificial Intelligence in fitness Industry

Fitness is a trend today. A report by Wellness Creatives suggests that fitness industry revenue grows by 8.7% every year, and fitness apps are no exception.

Technologies are used in many ways to help improve our bodies, from tracking our exercise to adjusting our nutrition. The question remains that how much better such apps can assist in improving the physical training performances exercises compared to human instructors?

Artificial Intelligence has penetrated nearly every aspect of business for a long time. Pose estimation determines the position and orientation of the human body from an image showing a person and it is one of the most used by many AI solutions. A powerful tool like this can be used for many things, like avatar animation for Artificial Reality, and employee pose analysis, among other applications.

In recent years, with the advent of pose estimation technology, personal trainer apps based on artificial intelligence have been flooding the fitness technology market. These technologies make use of computer vision, human pose estimation, and natural language processing algorithms to guide its users through various workouts and provide feedback in real-time.

2.3. Challenges of AI in fitness

Let us say that a human trainer is replaced with an AI-based app for fitness. Would it have the same capabilities of estimating exercises as human trainers? Below are some of the possible errors that we came across while learning about human pose estimation technology:

(a) Specifics of Physiology:

Humans do not have ideal body proportions. There are disproportions in everyone, whether it be the length of their arms, the legs, or the back. We need to understand that only those images of people used for training will be used to analyse the user's body for pose estimation. This means, the model's perception of the body of the user depends on people's bodies from the training images. It is not assured that the training dataset consists of images of the same body structure.

(b) Detection of start of the exercises

In order to estimate the exercise duration period, the system is expected to detect the exercise start and end. For example, while squatting with a barrel, the system can analyse the user's hand and shoulder positions through arbitrary hard-coded thresholds. Errors might occur if the arm angles briefly exceed this threshold.

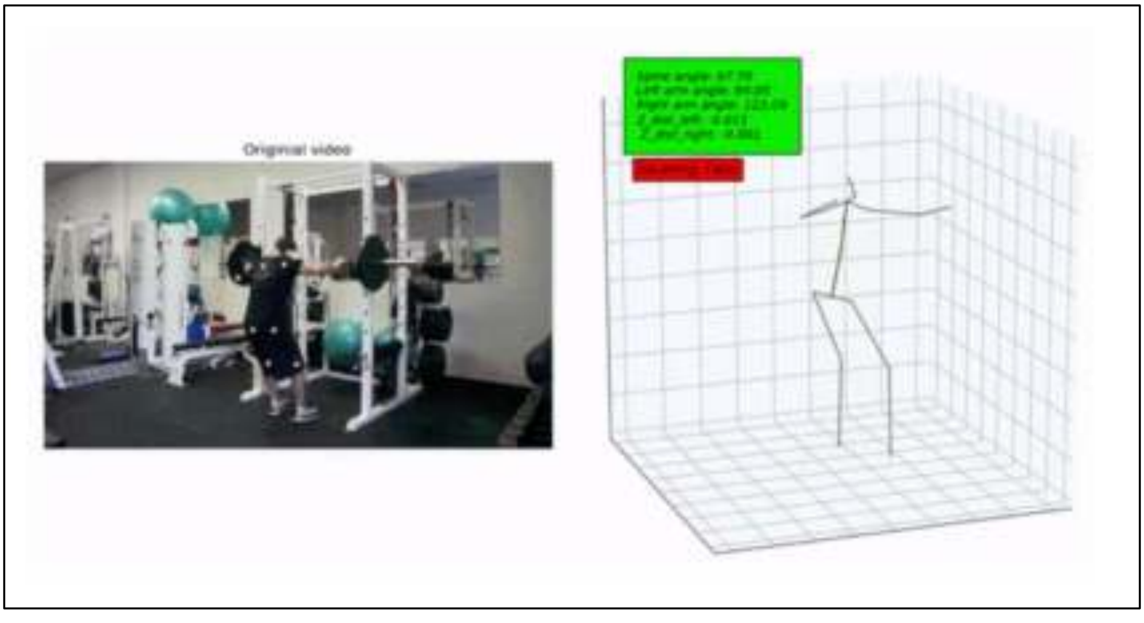


Fig 2.7. Error in detecting the start of exercise - this condition randomly interrupts after beginning the exercise.

(c) Quick Movements of the Lower Body Part:

Another example where these AI-based apps may cause errors is in Martial arts while using legs for kicking. In the case of a quick kick, the deep learning model might be unable to record this action. This is because the fast leg-transition could lead to the blurring of key points of the leg.

Additionally, the 2D key point dataset may not include images of such limb actions. The 2D detections which act as an input for 3D pose prediction were detected wrongly. As a result, 3D predictions for the lower-body will not reflect real actions.

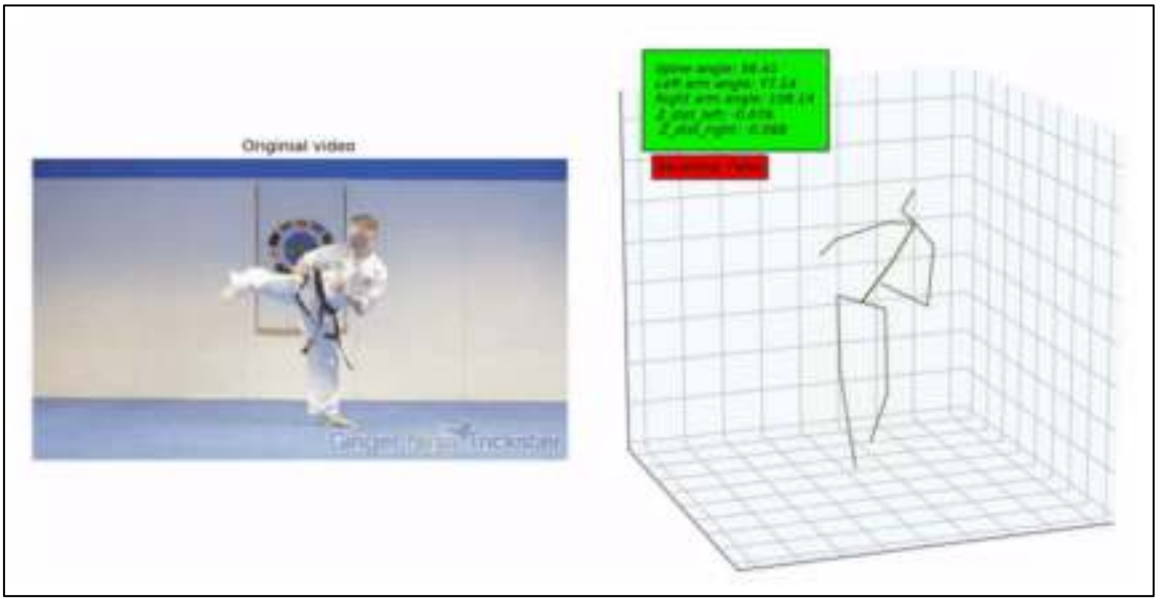


Fig 2.8 . Incorrect 2D detections that lead to wrong 3D detections.

(d) Horizontal position:

A human pose estimation model may also have difficulty estimating push-ups. A large number of errors are returned by the model when detecting 2D key points of arms and legs of this athlete doing push-ups in the video. However, rotating the video vertically to examine the athlete's movements worked wonderfully. This problem proved that visual data in such open datasets was not sufficient.

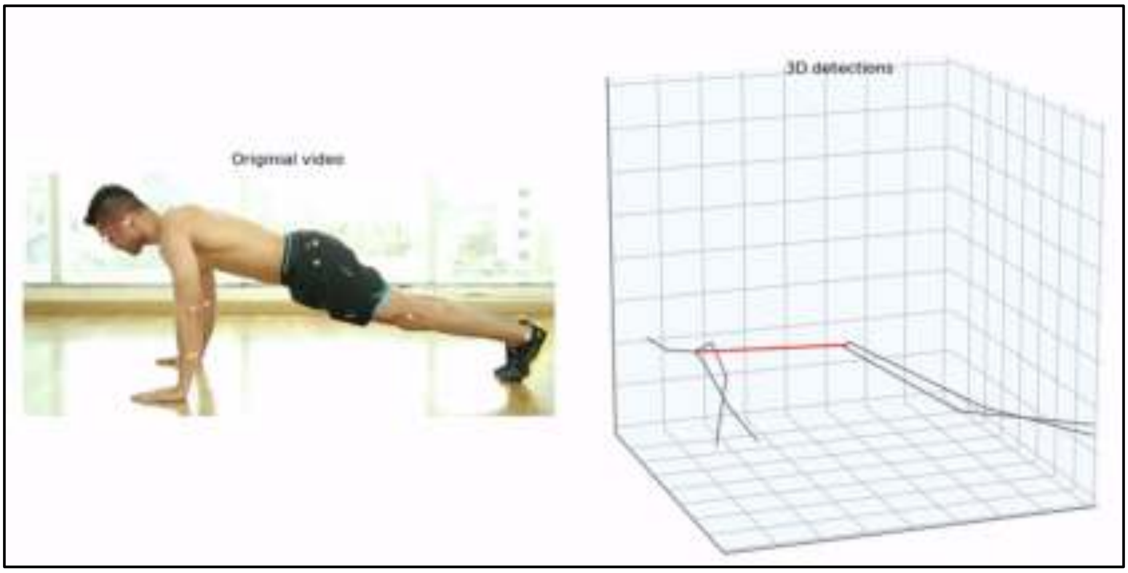


Fig 2.9 . Detection error when some joints are blocked

Note that these challenges are just examples of what problems could be encountered when building AI-based fitness apps using human pose estimation. This technology in a fitness app depends on a variety of parameters.

Chapter 3

Project Design

This chapter presents the project design, that is, the introduction to the project, the problem statement of our project deal, objectives, description of the libraries used and finally different tools and technologies we made use of in developing our solution.

3.1. Problem Statement

Physical activity, particularly structured exercises, not only improves physical function but has also been linked to improved social and mental wellbeing. However, for some people, regular training, particularly at a gym or outside, may be inconvenient or impossible. Currently, the only way to achieve this is through interaction with a trainer or other gym-goers. Personal trainers are expensive, and many people who wish to exercise cannot afford one.

44% of gym users are too embarrassed to seek help from a gym employee" (pure gym, 2019). During the COVID-19 epidemic, the majority of people have looked into some type of at-home exercise. "***40% OF CONSUMERS ANTICIPATING TO EXERCISE MORE AS A RESULT OF THE ONGOING PANDEMIC***" (LSN GLOBAL, 2020). Additionally, fast-food consumption is frighteningly increasing, which has resulted in the consumption of unhealthy foods. As a result, it has become critical for people to have a well-balanced nutritionally sound diet in addition to engaging in physical activity. However, in this fast-paced world, not everyone has the time or money to spend on a personal dietitian and nutritionist who will monitor and care for their health by counseling them on a balanced food plan.

In this study, we looked at how technology can help with home training, how it can persuade people to start and maintain an active lifestyle, and how it may be helpful in attaining better strength and balance. We reviewed individual personal information in this study and attempted to propose a diet type and meal plans for a healthy lifestyle.

3.2. Objectives

- To understand user perceptions of fitness applications and its impacts on them.
- To assess the efficiency of a fitness workout in enhancing the fitness of its users.
- To create workouts using artificial intelligence and track progress in real time.
- To demonstrate the summary of an user's performance when the workout is completed
- To comprehend a user's diet based on a variety of inputs
- To compute the user's BMI and BMR values, as well as a graphical representation of weight reduction and calories burned.

3.3. Libraries Used

3.3.1. OpenCV

OpenCV essentially stands for Open-Source Computer Vision Library. OpenCV is a Python open-source library, which is used for computer vision in Artificial intelligence, Machine Learning, face recognition, etc. In OpenCV, the CV is an abbreviation form of a computer vision, which is defined as a field of study that helps computers to understand the content of the digital images such as photographs and videos. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features. The purpose of computer vision is to understand the content of the images. It extracts the description from the pictures, which may be an object, a text description, a three-dimensional model, and so on.

Features of OpenCV:

- Image/video I/O, processing, display (core, imgproc, highgui)
- Object/feature detection (objdetect, features2d, nonfree)
- Geometry-based monocular or stereo computer vision (calib3d, stitching, videostab)
- Computational photography (photo, video, superres)
- Machine learning & clustering (ml, flann)
- CUDA acceleration (gpu)

3.3.2. Pandas

Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.Pandas is an open-source, BSD-licensed Python library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc. It's most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named NumPy, which provides support for multi-dimensional arrays.

Features of Pandas:

- Pandas allows us to analyze big data and make conclusions based on statistical theories.
- Pandas can clean messy data sets, and make them readable and relevant.
- Easy handling of missing data (represented as NaN) in floating point as well as non-floating point data

- Automatic and explicit data alignment: objects can be explicitly aligned to a set of labels, or the user can simply ignore the labels and let Series, DataFrame, etc. automatically align the data for you in computations

3.3.3. PyOTP

The Python One Time Password Library is a Python library for generating and verifying one-time passwords. It can be used to implement two-factor (2FA) or multi-factor (MFA) authentication methods in web applications and in other systems that require users to log in.

3.3.4. JSON

JSON is an acronym for JavaScript Object Notation, is an open standard format, which is lightweight and text-based, designed explicitly for human-readable data interchange. It is a language-independent data format. It supports almost every kind of language, framework, and library. It means that a script (executable) file which is made of text in a programming language, is used to store and transfer the data. Python supports JSON through a built-in package called JSON. To use this feature, we import the JSON package in Python script. The text in JSON is done through quoted-string which contains the value in key-value mapping within { }. It is similar to the dictionary in Python.

Features of JSON:

- Simplicity
- Openness
- Self-Describing
- Internationalization
- Extensibility
- Interoperability

3.3.5. NumPy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays NumPy (Numerical Python) is an open-source core Python library for scientific computations. It is a general-purpose array and matrix processing package. NumPy stands for Numerical Python. NumPy is a Python library that provides a simple yet powerful data structure. Features of NumPy:

- High-performance N-dimensional array object
- In Python we have lists that serve the purpose of arrays, but they are slow to process.

- NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.
- The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.

3.3.6. Scikit-Learn

Scikit-learn is a Python module integrating a wide range of state-of-the-art machine learning algorithms for medium-scale supervised and unsupervised problems. This package focuses on bringing machine learning to non-specialists using a general-purpose high-level language. Emphasis is put on ease of use, performance, documentation, and API consistency. It has minimal dependencies and is distributed under the simplified BSD license, encouraging its use in both academic and commercial settings.

Features of Scikit-Learn:

- Supervised learning algorithms
- Unsupervised learning algorithms
- Feature extraction
- Cross-validation
- Dimensionality Reduction
- Clustering
- Ensemble methods

3.3.7. BeautifulSoup

Beautiful Soup is a python library for pulling data out of HTML and XML files. It works with a parser to provide idiomatic ways of navigating, searching and modifying the parse tree. The main purpose of using this library is that it saves programmers hours or days of work as it provides a multiple list of commands which can be used to pull data from the web pages. Beautiful Soup is a Python library designed for quick turnaround projects like screen-scraping.

Three features make it powerful:

- Beautiful Soup provides a few simple methods and Pythonic idioms for navigating, searching, and modifying a parse tree.
- Beautiful Soup automatically converts incoming documents to Unicode and outgoing documents to UTF-8.
- Beautiful Soup sits on top of popular Python parsers like lxml and html5lib, allowing you to try out different parsing strategies or trade speed for flexibility.

3.4. Tools and Technologies

3.4.1. Figma

Figma is a browser-based UI and UX design application, with excellent design, prototyping, and code-generation tools. It's currently (arguably) the industry's leading interface design tool, with robust features which support teams working on every phase of the design process.Itis a versatile program that can be used to design websites, apps, and many more digital products.

Features of Figma:

- Real-Time Updating of the Project
- Intuitive and Straightforward Prototyping
- Accessibility and collaboration.
- Versatility to be customized and improved

3.4.2. Bootstrap

Bootstrap is the most popular CSS Framework for developing responsive and mobile-first websites. It is a potent front-end framework used to create modern websites and web apps. It is a free and open source front end development framework. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps.

Features of Bootstrap:

- Easily Customizable
- Responsive Utility Classes
- Browser compatibility
- Simple Integration
- Extensive list of Components
- Bundled JavaScript plugins

3.4.3. Python

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in the Software Industry. Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development. Python supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles. Python makes the development and debugging fast because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

Features of Python:

- Interpreted Language
- Object-Oriented Language
- Extensible
- GUI Programming Support
- Dynamic Memory Allocation
- Wide Range of Libraries and Framework

3.4.4. Django

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It's free and open source. Django is a web application framework written in Python programming language. It is based on the MVT (Model View Template) design pattern. Django is designed in such a manner that it handles much of the configuration automatically, so we can focus on application development only.

Features of Django

- Rapid Development
- Secure
- Scalable
- Versatile
- Open Source
- Vast and Supported Community

3.4.5. SQLite

SQLite is an embedded relational database management system. It is self-contained, serverless, zero configuration and transactional SQL database engine. SQLite is free to use for any purpose commercial or private. In other words, "SQLite is an open source, zero-configuration, self-contained, stand alone, transaction relational database engine designed to be embedded into an application". By default, Django works with SQLite, database and allows configuring for other databases as well.

Chapter 4

Methodology

This chapter presents the detailed explanation of concepts like Pre-processing of images for multi-pose landmarks, angle calculation, calorie burned estimation used in the algorithm that we have implemented. We have elaborated on the dataset taken for Diet recommendation and analysed various data visualisation graphs to draw conclusions for them. Finally, we have discussed the machine learning algorithm and the system architecture and the user flow of our website.

4.1. Exercise Correction

4.1.1. Pre-processing of Images to get Multi-Pose Landmarks

MediaPipe [2] is a framework that enables developers to build cross-platform multi-modal (video, audio, and any time series data) machine learning pipelines. MediaPipe features a huge collection of human body detection and tracking algorithms that were trained on Google's massive and diversified dataset. They use a skeleton of nodes and edges, or landmarks, to identify crucial points on various areas of the body. All coordinate points are normalized in three dimensions. Models created by Google developers using TensorFlow lite make information flow more adaptive and customizable using graphs. Pipelines in MediaPipe are made up of nodes on a graph that are often described in a pbtxt file. C++ files are related to these nodes. The base calculator class in MediaPipe is based on these files. This class, like a video stream, receives media stream contracts from other nodes in the graph and guarantees that they are linked. The class creates its own output processed data after the remainder of the pipeline nodes are joined. To convey each stream of data to each calculator, packet objects encoding a variety of various forms of data are utilized. Side packets, in which a calculator node is inserted with auxiliary data such as constants or static attributes, can also be forced onto a graph. This streamlined structure in the dataflow pipeline makes it easier to add or modify data, and the flow of data becomes more precisely managed.

The backend of the Human Pose Estimation solution is an ML pipeline that consists of two models that run alongside: a) ML Kit Pose Detection Model (BlazePose) and b) Pose Landmark Model. The Pose Detection Model generates a cropped human picture, which is then handed on to the landmark model. This method reduces the usage of data augmentation (i.e., Rotations, Flipping, and Scaling) in Deep Learning models and instead devotes the majority of its processing capacity to landmark localization. Detecting the hand from the frame and then doing landmark localization across the current frame is the standard method. However, a different method is used in this Pose Detector utilizing ML pipeline challenge. Hand detection is a time-consuming operation since it requires image processing and thresholding, as well as working with a range of hand sizes. Instead of recognizing the human body directly from the current frame, the detector is first taught to estimate bounding boxes around hard items like the face and palm, which is easier than detecting the entire body. Second, for larger scene context, an encoder-decoder is used as an extractor. Within the identified locations, this model precisely localizes 33 3D pose coordinates (i.e., x, y, and z-axis). Because the model is so well-trained and reliable in its detection, it can even map locations to a partially visible hand.

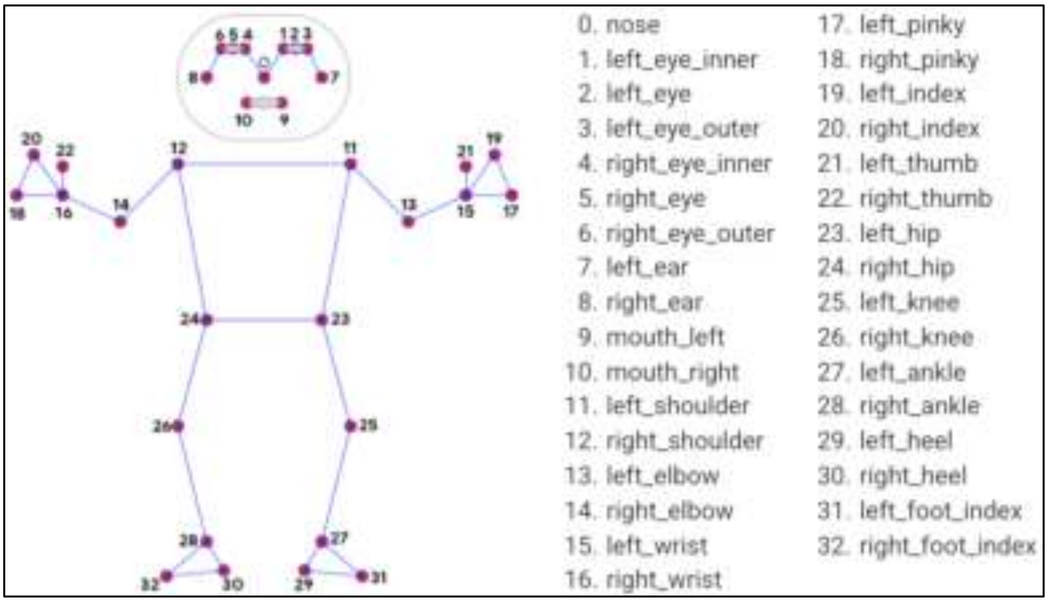


Fig 4.1 33 pose landmarks

We wish to examine only the x and y coordinates for any given landmark point now that the BlazePose model has recognised 33 3D landmarks from live feed. To accomplish this, the model returns 33 landmark points for a single image. Then we traverse over the points, just looking at the id number, x and y coordinates. The x and y numbers are then multiplied by the width and height of the current window. Finally, the information is stored as an array.

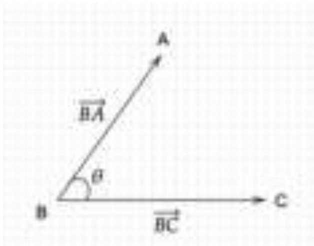
4.1.2. Angle Calculation using Landmarks

All of our landmarks are stored in the form of an array, so to access any of the body point simply use the following syntax:

$$x', y' = List[id][1:]$$

where id = body point (example left knee: 25)

To calculate the angle of any given joint part, we need 3 landmark points. Let's consider we have A, B, C landmark points as shown below.



where,

$$\overrightarrow{BA} = B - A$$
$$\overrightarrow{BC} = C - B$$

The scalar product or the dot product of two vectors is represented by the formula:

$$\overrightarrow{BA} \, . \, \overrightarrow{BC} = ||\overrightarrow{BC}|| \, ||\overrightarrow{BA}|| \cos(\theta)$$

where || * || is the magnitude of the vector and θ is the angle made by the two vectors. From the above formula we can represent the angle using the formula:

$$\theta = \cos^{-1}\left(\frac{\overrightarrow{BA} \, . \, \overrightarrow{BC}}{||\overrightarrow{BC}|| \, ||\overrightarrow{BA}||}\right)$$

4.1.3. Posture Correction

Once the required angles have been calculated during the above-mentioned method, we can approximate a person's posture by mapping the joint's angles with threshold angles. Below is a detailed table of exercises used in this project.

Body Joints	Correct Posture	Ideal Posture
Squats		
Hips	Θ < 100	Θ >= 100
Knee	Θ < 100	Θ >= 100
Jumping Jacks		
Dist b/w Legs	Dist = 40	Dist >= 150
Abdominal Crunches		
Hips	110 < Θ < 145	Θ > 145
Knee	80 < Θ < 90	80 < Θ < 90
Knee Pushups		

Knee	70 < Θ < 100	70 < Θ < 100
Elbow	Θ < 120	Θ > 140
Side Arm Raises		
Shoulders	Θ > 100	Θ < 30
Dist b/w Legs	Dist > Dist b/w Shoulders	Dist < Dist b/w Shoulders
Backward Lunges (Turn by Turn)		
Left Knee	Θ = 90	Θ > 170
Right Knee	Θ = 90	Θ > 170
Cobra Stretch		
Elbow	Θ >= 150	Θ < 30
Leg	Θ >= 150	Θ >= 150
Head	Θ >= 150	Θ < 90

Table 4.1: Posture Correction Threshold Angles

4.1.4. Calories Burned Estimation

We consume calories anytime energy is ingested or converted in our bodies because calories are effectively energy. When we think about calories, we usually think of the purposeful act of exercising and expending additional energy in compared to typical or baseline metabolic levels.

Each task, as you might think, needs a particular amount of energy. Walking burns less calories than either running or cycling. This energy expenditure is often expressed as MET — the metabolic equivalent of a task. This statistic measures the number of calories burned per hour of activity per kilogram of body weight.

MET is defined as the ratio of energy spent per unit time during a specific physical activity to a reference value of 3.5 ml O_2/(kg·min)

$$calories = T \ x \ MET \ x \ 3.5 \ x \ W \ / \ 200$$

Where, T = duration of activity in minutes, W = person's weight in kg

This will give you the number of calories you burned, or the thermic effect of physical activity.

Fig 4.2 Metabolic equivalents values comparison

The MET value of an exercise is higher the more energy an activity requires. For example, sleeping has a value of 1 MET while running has a MET of 7.5, so much higher [3].

While the person is performing the exercise, time from the first rep to the last rep is noted down in seconds. And then the estimated calorie burned is calculated using the above formula.

4.2. Diet Recommendation

4.2.1. Dataset

The dataset was taken from Kaggle [4]. This dataset contains 10,0081 rows and 19 columns as shown below.

Fig 4.3 Overview of the dataset

As there are 19 columns, "Name" column is not required and hence removed from the dataset. And "Suggestion" column is our response variable. The rest of 17 columns are highly dependent on each other. And we will explore each of them.

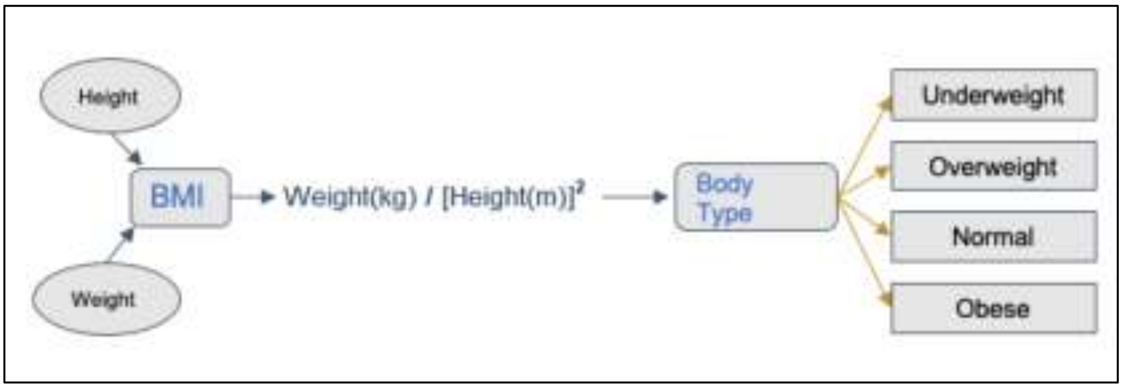


Fig 4.4 Interrelation of BMI (Body Mass Index)

From the figure, we can see that to calculate the BMI, height and weight are necessary. And by using BMI we can estimate the person's body type, i.e., Underweight, overweight, normal or obese.

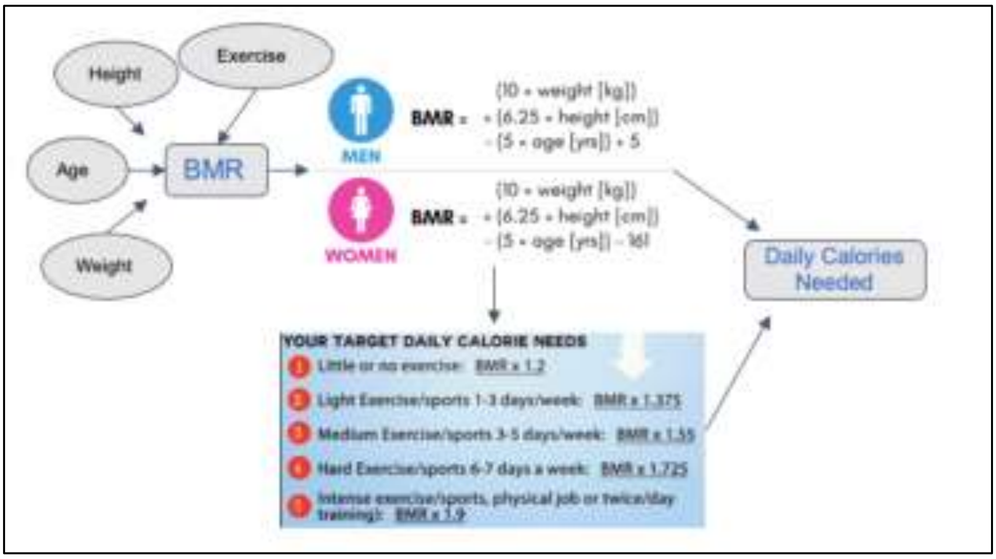


Fig 4.5 Interrelation of BMR (body mass ratio)

From the figure, height, weight, exercise type, age is required for BMR. There are different formula depending on the gender.

$$Men = (10 * W) + (6.25 * H) - (5 * A) + 5$$

$$Women = (10 * W) + (6.25 * H) - (5 * A) - 16$$

where,

W = weight in kg,

H = height in cm,

A = age in yrs.

Once we have the BMR value, it can be multiplied by a constant according to the type of exercise to calculate Daily calories needed.

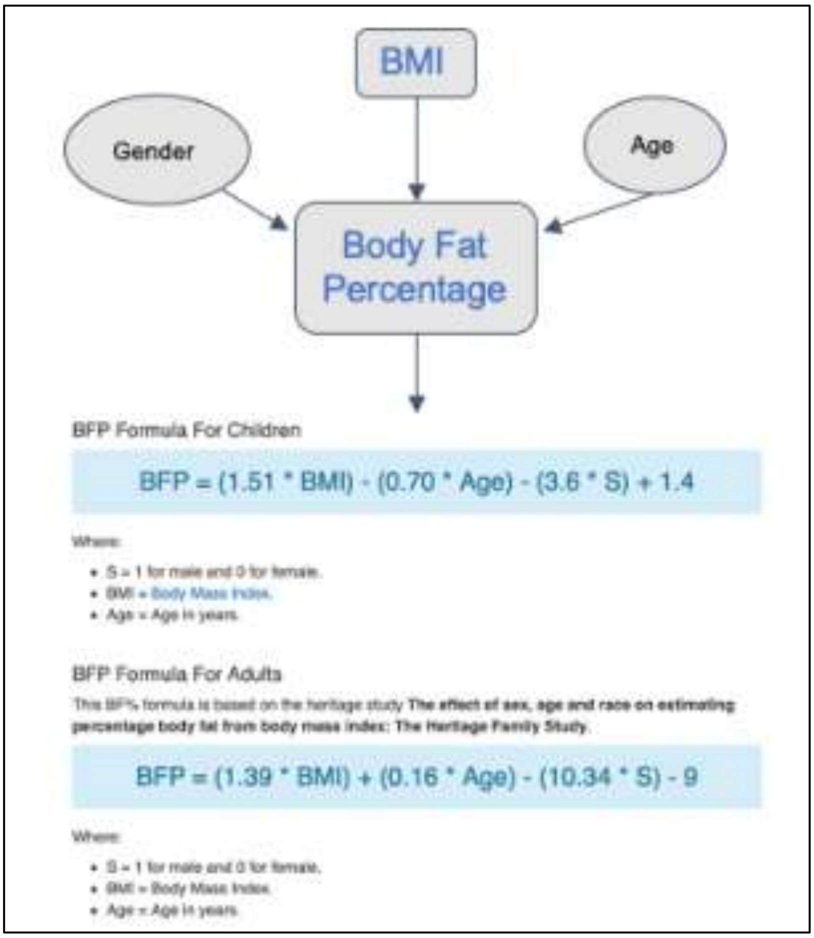


Fig 4.6 Interrelation of BPF (Body fat percentage)

From the figure, gender, BMI and age are required to calculate Body Fat Percentage. Different formulae is used depending upon the age group.

Age 18-21:

$$BFP = (1.51 * BMI) - (0.70 * Age) - (3.6 * S) + 14$$

Age 22 and above

$$BFP = (1.391 * BMI) + (0.16 * Age) - (10.34 * S) - 9$$

where,

S = 1 for male and 0 for female

4.2.2. Exploratory Data Analysis

We have plotted a few of the graphs to understand the nature of data and understanding the user's perspective.

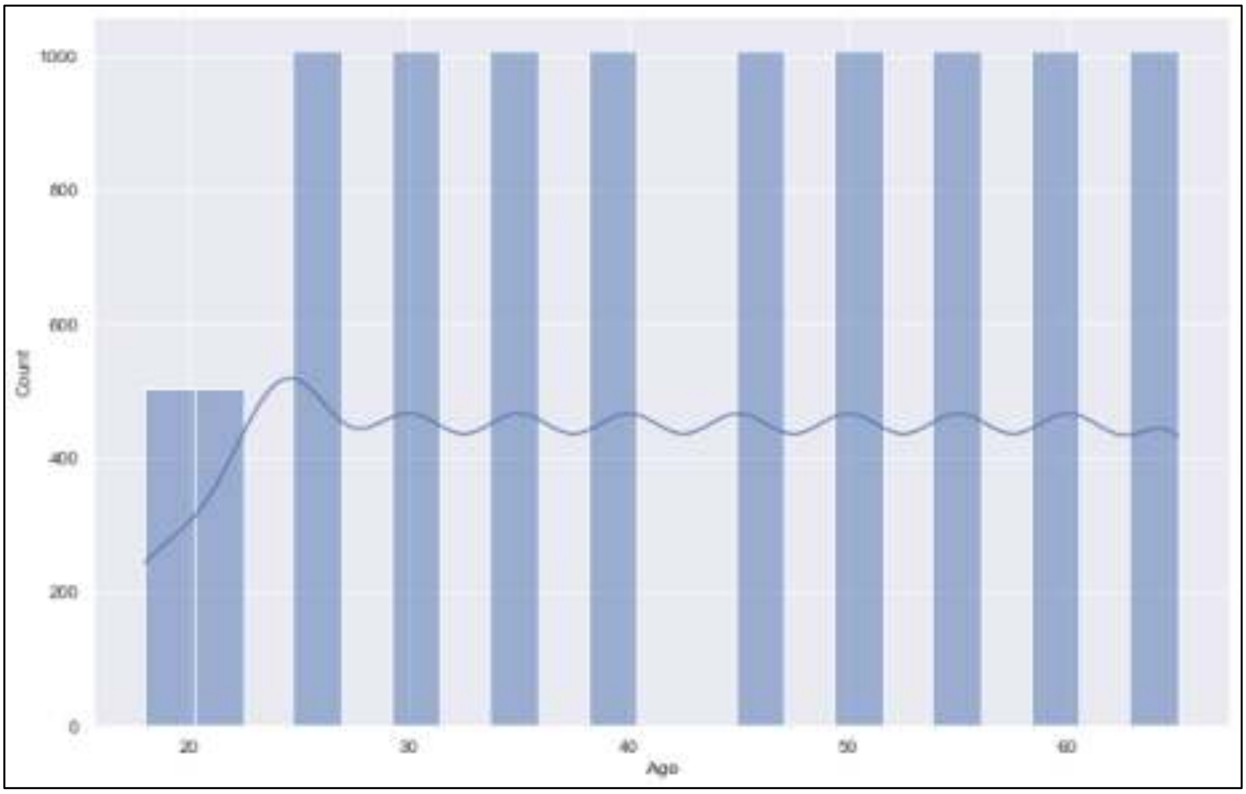


Fig 4.7 Bar graph with KDE of Age Distribution

The figure demonstrates is a bar graph which shows the different age groups spread throughout the dataset. After careful analysis of the kde, we can conclude that age group of 18-25yrs had the highest count and probability density. The data was well spread between the two-age group.

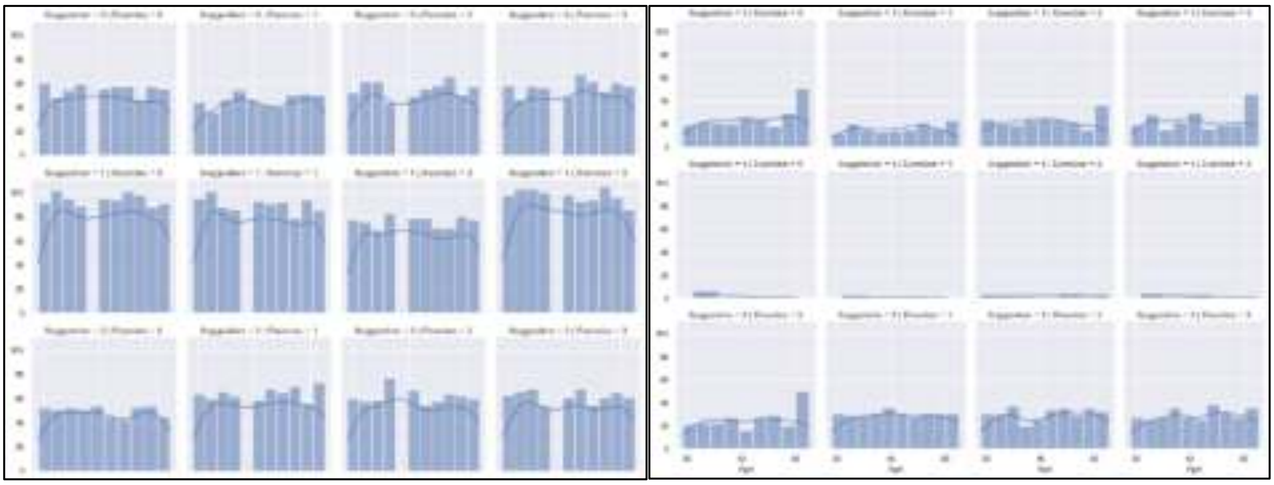


Fig 4.8 Multiplot grid for plotting conditional relationship of Suggestion, Exercise and Age

where,

- LOW CARB DIET: 0, BALANCE DIET: 1, ZONE DIET: 2, KETOGENIC DIET: 3, DEPLETION DIET: 4, HIGH CARB DIET:5
- Sedentary: 0, Lightly active: 1, Moderately active: 2, Very active: 3

The figure is a facetgrid graph showing relation between Suggestion, Exercise Type and Age groups. We can see huge data distribution for Balance diet & different types of exercise. And least user data for Depletion diet. The age is well balanced across the age group of 18-25 yrs.

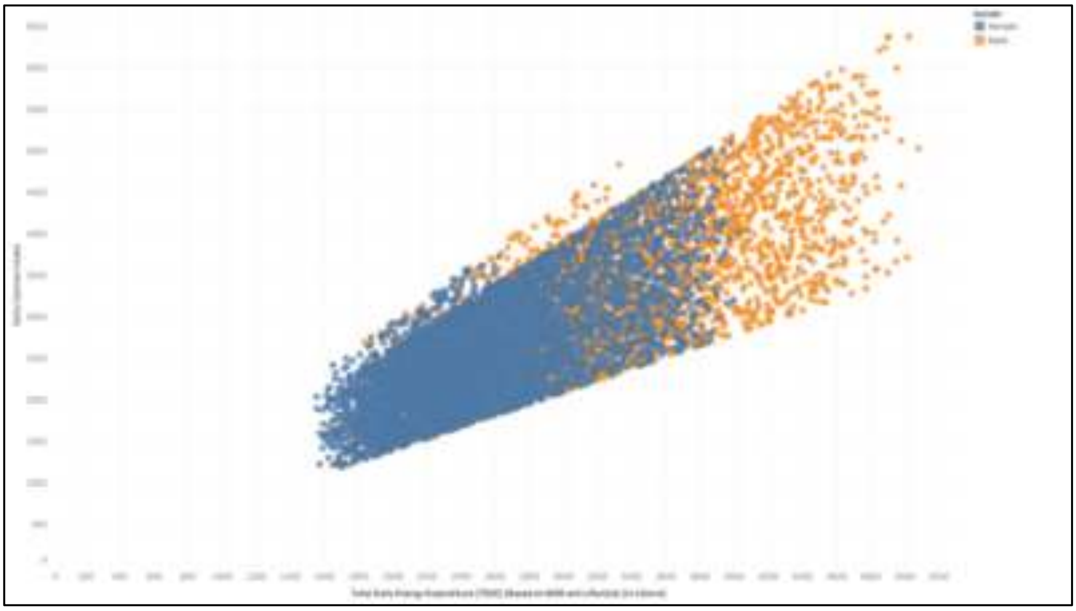


Fig 4.9 Scatterplot of Calories Burned and Daily Calorie Intake

The figure highlights an important aspect of this project and enlightens the calorie distribution between intake and burn. A linear pattern could be visualized stating that amount of calories burned is approximately equal to daily calorie intake.

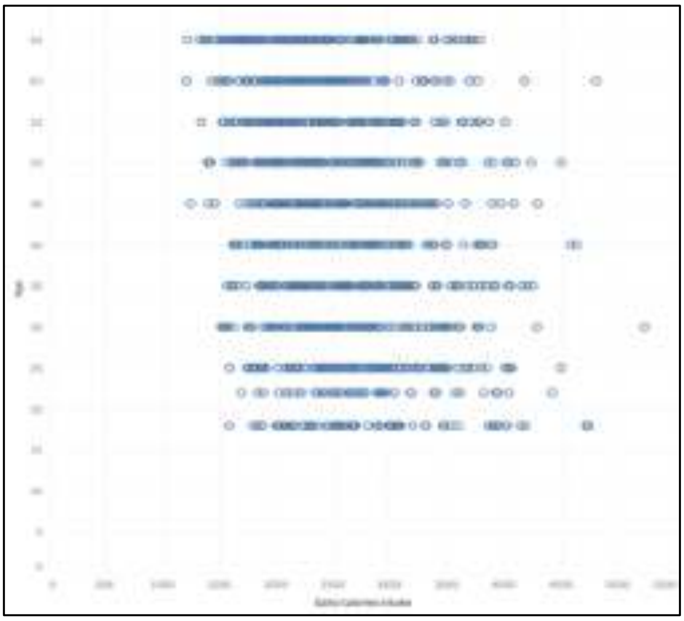


Fig 4.10 Target - Gain Muscle, Suggestion - High Carb Diet

From the figure we can infer that user targeting to gain muscle is mostly like to get High Carb Diet as their suggestion.

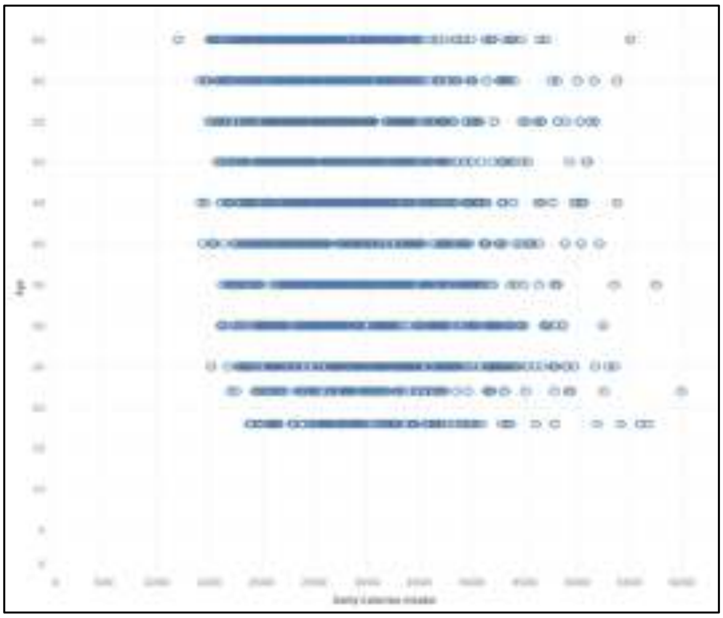


Fig 4.11 Target - Loose Weight, Suggestion - Low Carb Diet

From the figure we can infer that users targeting to lose weight are mostly likely to get a Low Carb Diet as their suggestion.

4.2.3. Data Preprocessing

Firstly, the data was read using Pandas [5] library. After reading the dataset, "Name" column was required as it was not required. As most of the columns are categorical in nature, it is necessary to convert them to digital language for the computer to understand. Mapping was performed on each column to convert it to its significant integer values. As the data amongst the columns were widespread, i.e., range of age is from 18-65yrs whereas Daily calories burned ranges from 1600 - 5000 cal. Therefore, scaling operation was applied on the dataset using StandardScaler [6]. It standardizes a feature by removing the mean and then scaling to unit variance. Unit variance is calculated by dividing all of the values by the standard deviation. Then the data was split into training and testing (80-20 ratio).

4.2.4. Machine Learning Algorithms

Random Forest, K-nearest Neighbors, Support Vector Machine, and Logistic Regression were all employed. When compared to other methodologies, the random forest approach was shown to be more accurate. Among the tools available for usage with a random forest technique, the scikit-learn Python module has the advantage of requiring fewer major parameters to be provided by the learner. Table 2 highlights the findings of the comparisons of features between a random forest classifier and the most often used classifier, the support vector machine (SVM). When normalised parameters and kernels are specified, an SVM requires kernel tweaking.

Method	Data Number	Main Parameters		
		Regularization	Strength of Regularization	Type of Classifier
Random Forest	Many	unnecessary	unnecessary	Non-linear Classifier
SVM	Little	necessary	K (Constant)	Linear or Kernel

Fig 4.12 Comparison of SVM and Random Forest

The random forest was programmed with the settings shown below.

A: The number of binary decision trees used in machine learning. A greater number of binary decision trees improves classification accuracy, but also necessitates more computation. In this investigation, we implemented 400 binary decision trees, as recommended by Breiman (2001) [7].

B: The maximum number of feature descriptors utilized in each binary decision tree for learning. According to Breiman, M is an explanatory variable, thus we employed it (2001) [7].

C: The greatest possible depth in any binary decision tree.

Random forest is an ensemble learning technique that use binary decision trees. The variances of the learning model are affected by the depth of each decision tree. According to Breiman (2001) [7], increasing the maximum depth increases the likelihood of over-fitting. However, the appropriate depth varies based on the data collection. In this work, we set the maximum depth to 8 and chose useful features after hyperparameter tweaking with the GridSearchCV [8] Python module.

```
GridSearchCV(cv=5, estimator=RandomForestClassifier(),
             param_grid={'criterion': ['gini', 'entropy'],
                         'max_depth': [4, 5, 6, 7, 8],
                         'max_features': ['auto', 'sqrt', 'log2'],
                         'n_estimators': [100, 200, 300, 400, 500]})
```

(a)

```
{'criterion': 'entropy',
 'max_depth': 8,
 'max_features': 'auto',
 'n_estimators': 400}
```

(b)

Fig 4.13 Hyperparameter tunning (a) and Best fit parameters for the model (b)

4.3. System Design and Architecture

4.3.1. System Design

The plan for system design was to describe and model the system using a Use Case Diagram, the application of use case diagram can be seen in Figure. According to the Use Case Diagram, the user system consists of potential users, users, and administrators engaging with the system menu. The use case diagram depicts the interaction between the system, external systems, and the user graphically. It specifies who will use the system and how the users intend to interact with it.

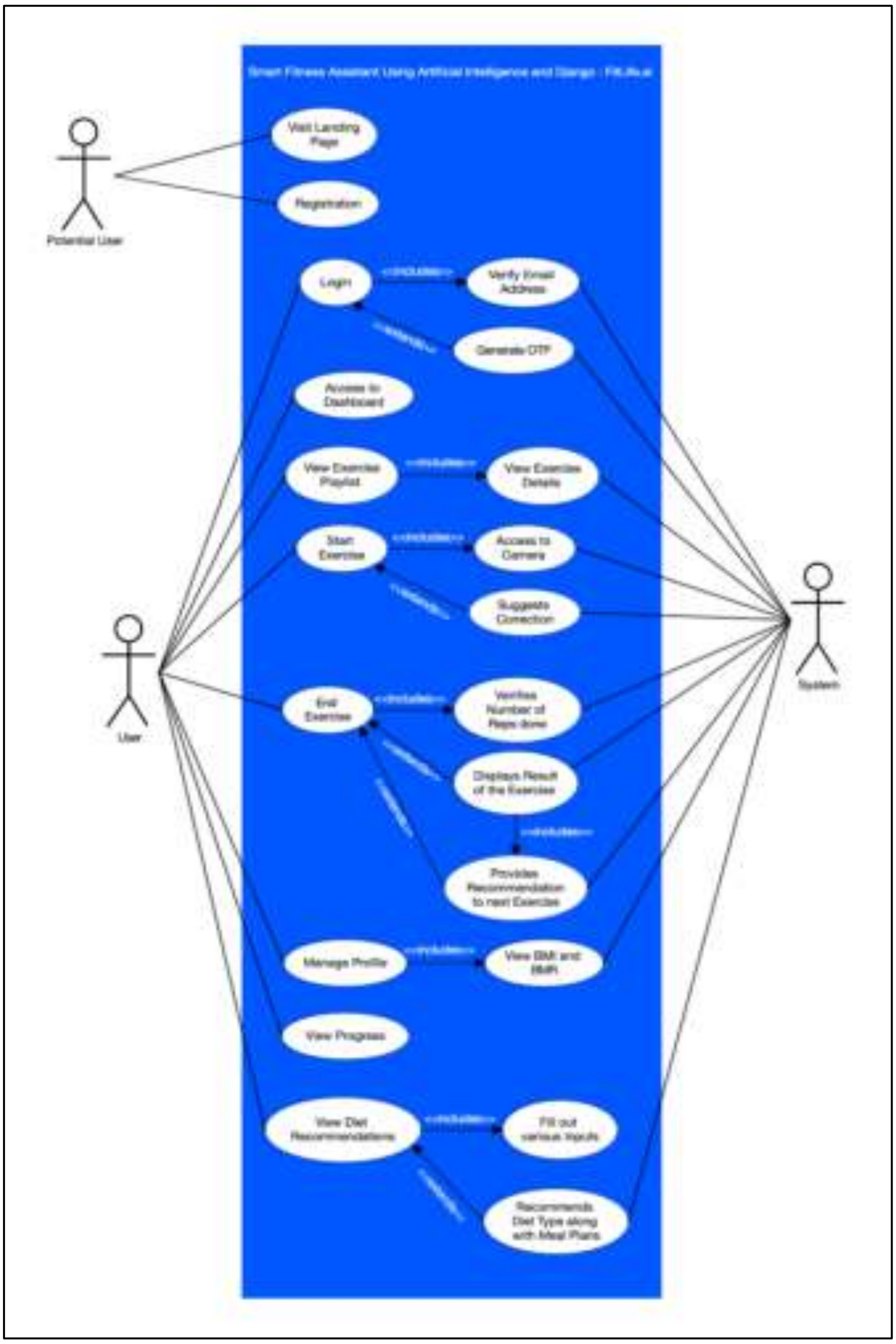


Fig 4.14 Use case diagram

4.3.2. Database Architecture

In order to effectively store the user's data, we decided to use a SQLite database, which is already incorporated into the Django Framework's ORM which includes the user's information, exercise types and exercise performed progress, history of calories burned and weight loss, and meal plans for various diet types. The first stage in building the database together was to develop objects that could be used to store and retrieve data for each data type. We designed a User Info class, a User Exercise Info class, a Playlist performed class, and a meal plan class.Each class has various variables, as well as methods for setting and retrieving them.

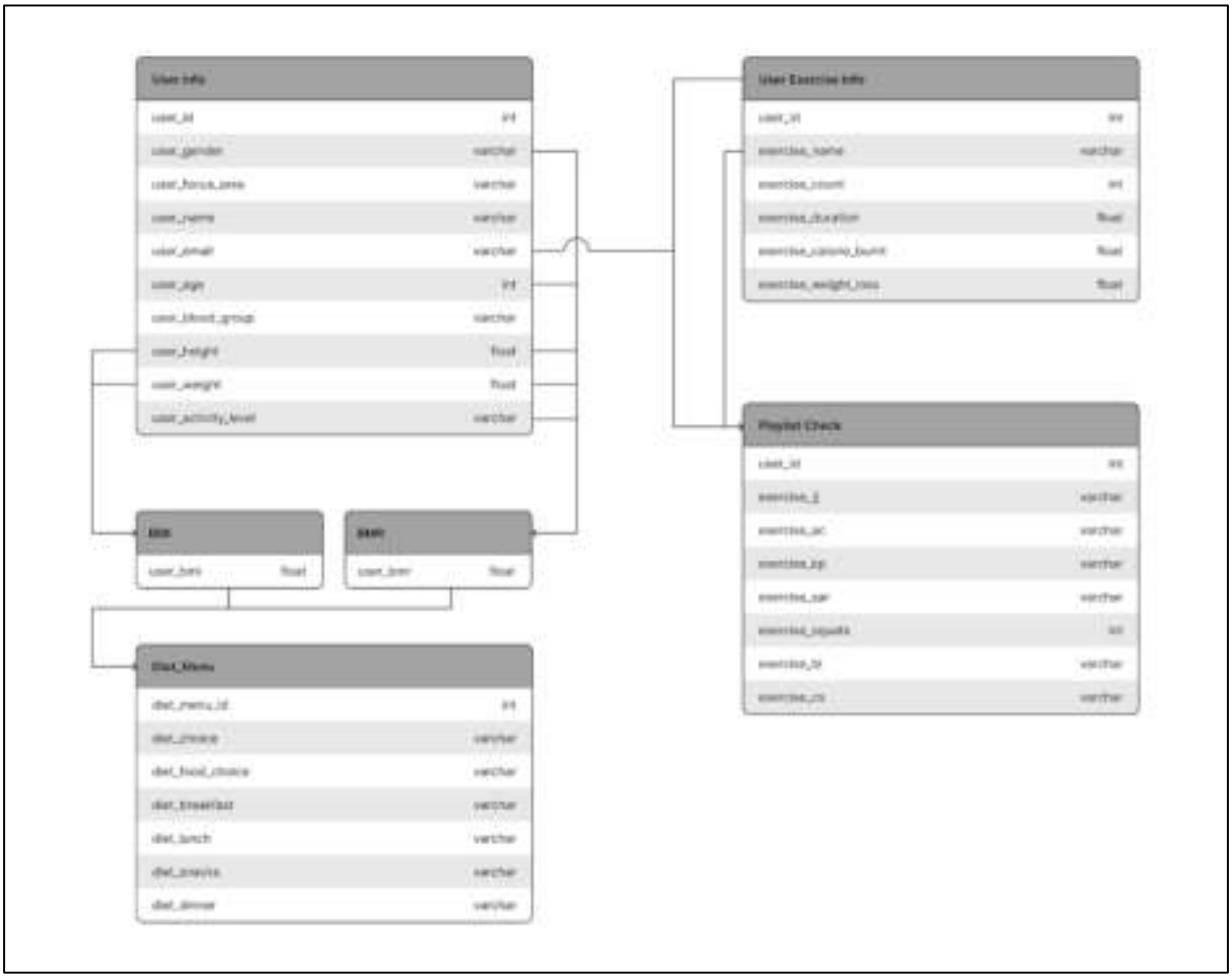


Fig 4.15 Database architecture

4.3.3. System Architecture

Part A: Perform workout exercise

- A registered user can log-in using their email address and OTP received. For non-registered users, they need to create their account. Next, they answer onboarding questions regarding their gender, age, height, weight, body-focus area, activity level, etc.
- The system uses this information to calculate the BMI and BMR and stores it in the database. Based on the focus area selected, the exercises are shown on the dashboard.
- The users can choose any exercise of their choice from the recommended exercises, workout challenges or playlist.
- After selecting the exercise, the system analyzes the user's posture using the exercise correction algorithm [Methodology]. For every rep performed in perfect posture, the count is increased. The algorithm measures the angles between the relevant joints and compares it with the correct angle for that exercise. If both match, the vertex is marked with green color, otherwise it is red. The system provides real-time instructions in terms of caption for maintaining correct posture.

- The system displays the performance summary with approximate calories burnt, total time taken and no. of reps completed once the user ends the workout. The completed exercises are marked with a tick mark on the dashboard.
- Further, the user can opt to repeat the exercise or move on to the next exercise in the playlist. If a user wants to select exercises other than those, they can go back to the dashboard and choose to continue their workout.

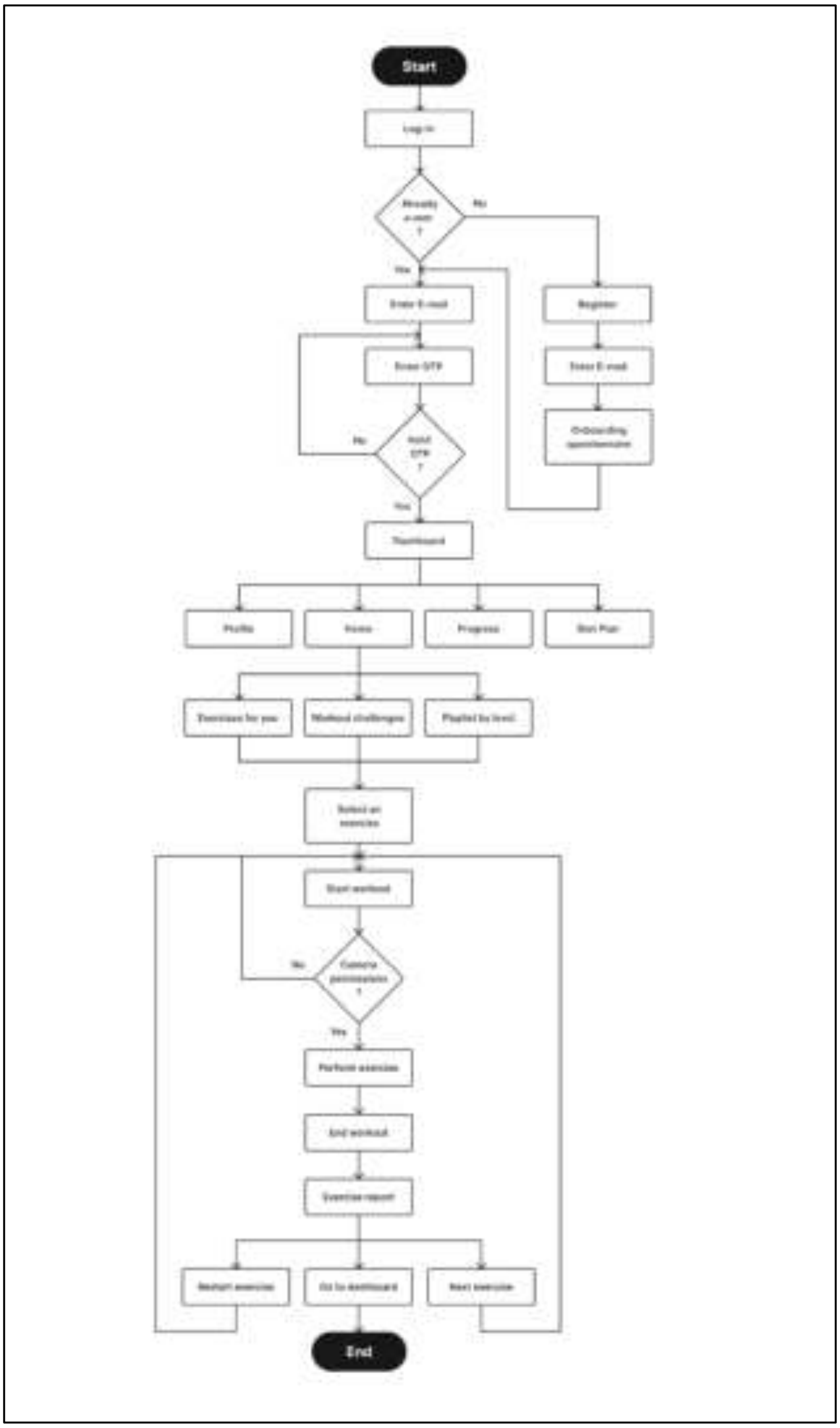


Fig 4.16 Userflow diagram - A

Part B : To view meal items for diet and check BMI / BMR

(A) Viewing meals:

- The user selects their food preferences (vegetarian or non-vegetarian), their physical activity level, diet goal, etc.
- The system takes this information based on which it recommends a type of diet from a total of 6 diets (Zone, keto, balanced, depletion, low-fat and high-carb).
- The user can view a list of food items for the recommended diet along with the breakdown of meals according to breakfast, lunch, snacks and dinner provided by the application.

(B) Computing BMI and BMR:

- In the profile section, users can edit their profile or calculate their BMI and BMR.
- For calculating the BMI, users enter height and weight as input.
- The system calculates plugs these values in the formula and displays their BMI value along with the suggested weight-range.
- For calculating the BMR, the user enters height, weight and age as inputs. The user selects gender and activity level.
- After calculating the BMR value in the system displays it along with the daily calorie need corresponding to their BMR.

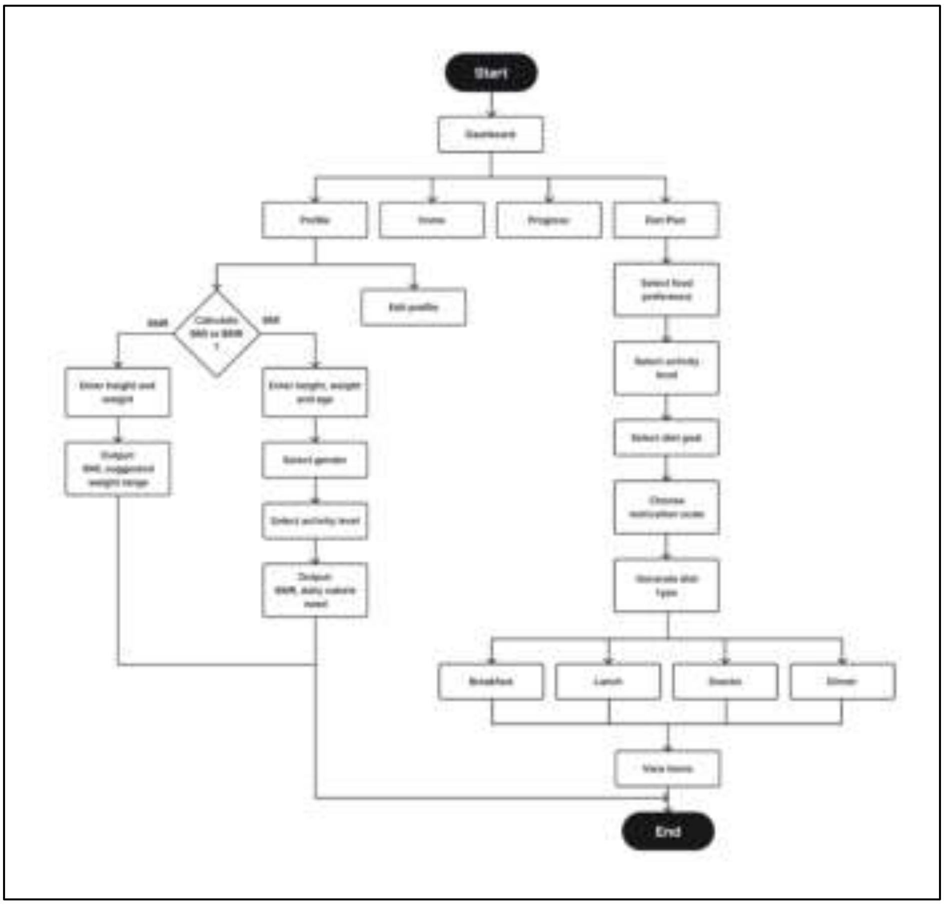


Fig 4.17 Userflow diagram - B

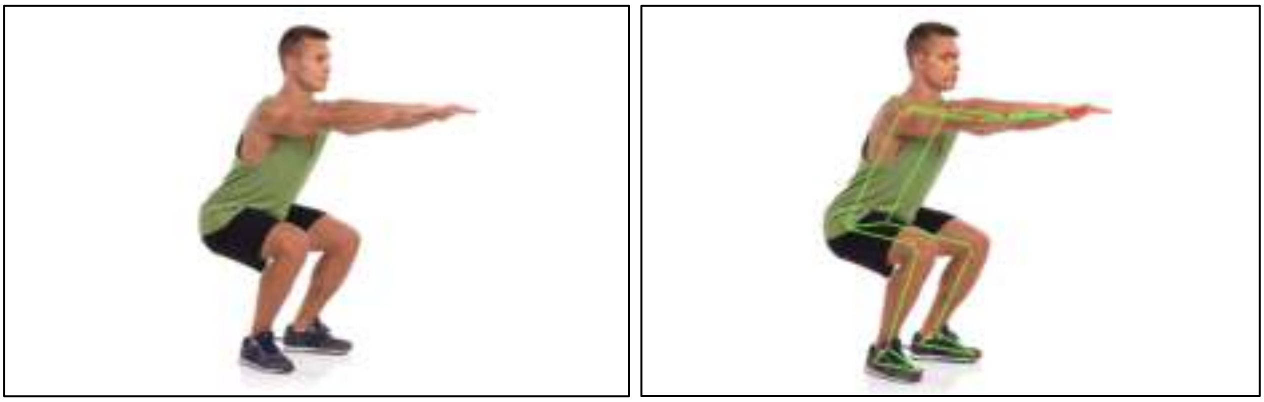
Chapter 5

Implementation

This chapter presents the mathematical calculations behind the algorithm for seven exercises that we have implemented. We have also illustrated how these exercises are implemented in our project. Further, we describe the diet management system and the machine learning model we made use of for its implementation.

5.1. Posture Correction

Let us understand this implementation of Exercise Correction [] using Squat exercise:



(a) (b)

Fig 5.1 Squats pose (a) and landmark recognition (b)

Once we have acquired the landmarks through the BlazePose Landmark model (mentioned in Methodology), let's find the Hip's angle (left-hand side of body). To do so, we need landmark points 11 (A), 23 (B) and 25 (C).

A = (454, 148)

B = (367, 307)

C = (490, 340)

$$||BA|| = \sqrt{(454-367)2+(148-307)2} = 15\sqrt{146}$$

$$||BC|| = \sqrt{(490-367)2+(340-307)2} = 127.35$$

$$cos\,\theta = BA\,.\,BC\,/\,||BA||\,.\,||BC|| = 0.2363$$

$$\therefore\ \theta = cos-1(0.2363) = 76°$$

Hence the hip's angle while doing squats in 76°.

Comparing with the table [], we could comment that 76° < 100°, hence the person has done the exercise correctly.

5.1.1. Side Arm Raises

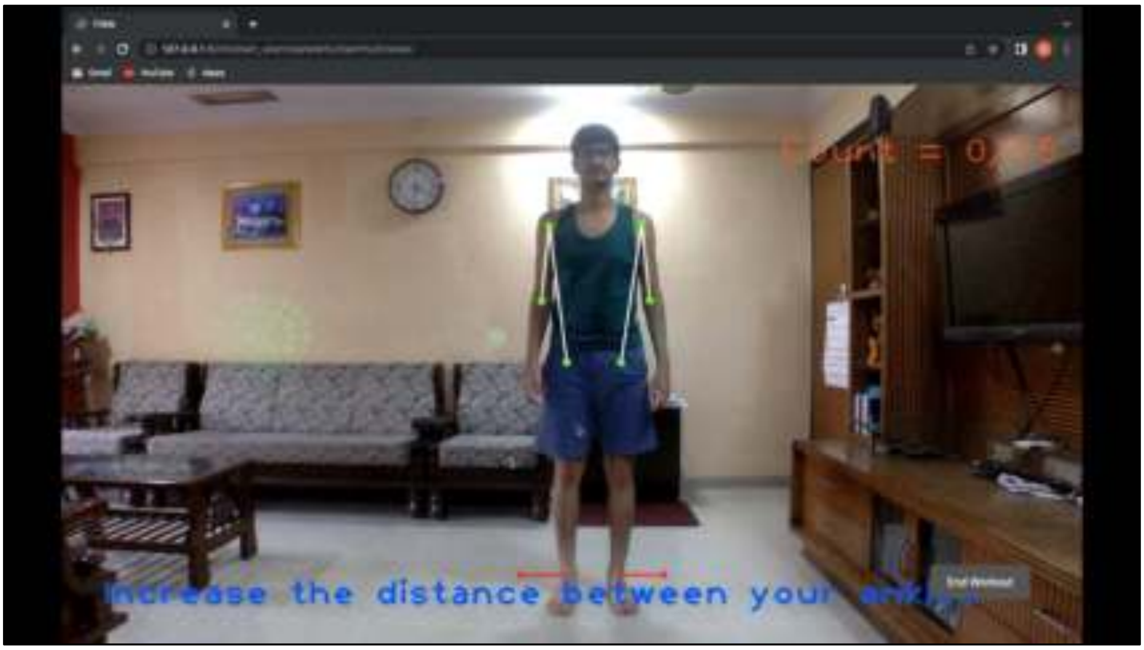


Fig 5.2 Side Arm Raises corrections-1

In the figure, the exercise won't begin unless the distance between the toes is greater than distance between the shoulders, which can be seen in the screenshot with red arrows across the toes.

Fig 5.3 Side Arm Raises corrections-2

From the figure, if the distance between the toes is sufficiently greater than the shoulders, then only the exercise count will increase. If the shoulder angle exceeds the threshold angle, as shown in the figure, the model will highlight it in red. As the hands have lifted way up, it needs to go down a bit according to the arrow indicating downwards in the screenshot. This means the person has done the exercise wrongly and needs to follow the caption provided in the screenshot.

5.1.2. Jumping Jacks

Fig 5.4 Jumping jack step 1

Fig 5.5 Jumping jack step 2

In the given figure, it is the first stage of starting the exercise. The distance between the toes should be minimum. The next step is to jump and increase the distance between the legs as shown in the figure. If the distance is greater than threshold distance, then the count will increase.

5.1.3. Squats

Fig 5.6 Squats step 1

In the figure, this is the first step where the user has to stand diagonal to the camera and the arrow towards the right side of the screen indicates whether the user is to go down and come up. The hips and knee are highlighted in red, as this isn't the squat posture but the first stage.

Fig 5.7 Squats step 2

From the figure, this is the second step of the exercise, where the user goes down provided it does not exceed the threshold knee angle. Until the user does the exercise correctly, it will be highlighted in green. And according to the arrow, the user has to come up and repeat the exercise for given counts.

5.1.4. Backward Lunges

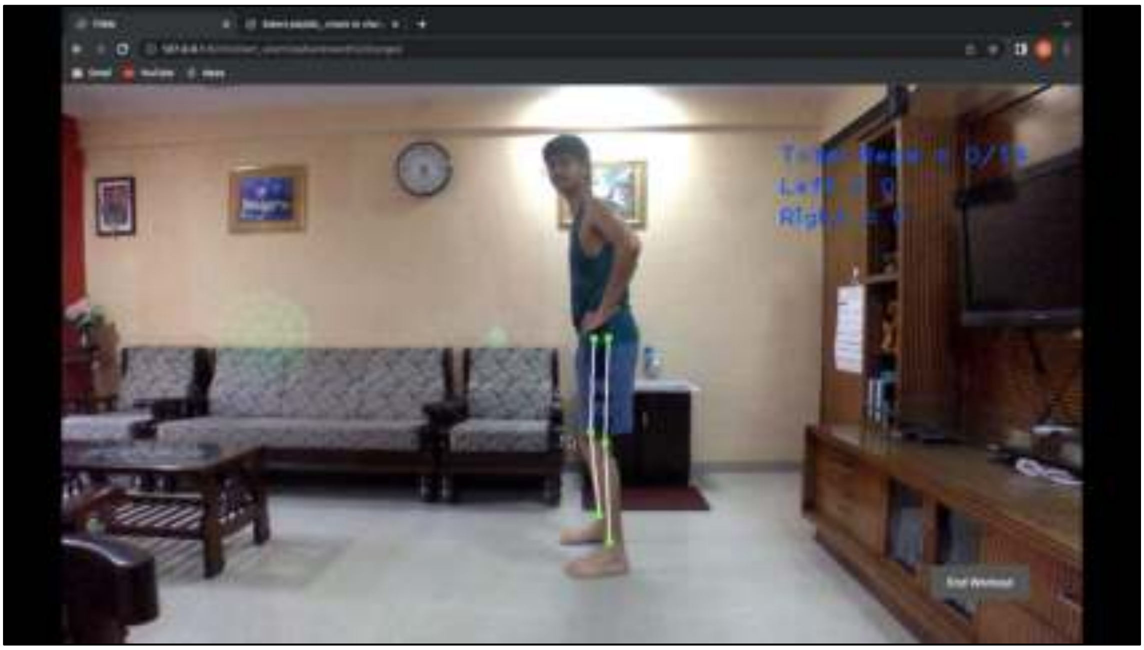


Fig 5.8 Backward Lunges step 1

In the figure, this is the first step where the user has to stand either diagonally or sideways facing the camera. Firstly, the user will move their left leg and then the right leg.

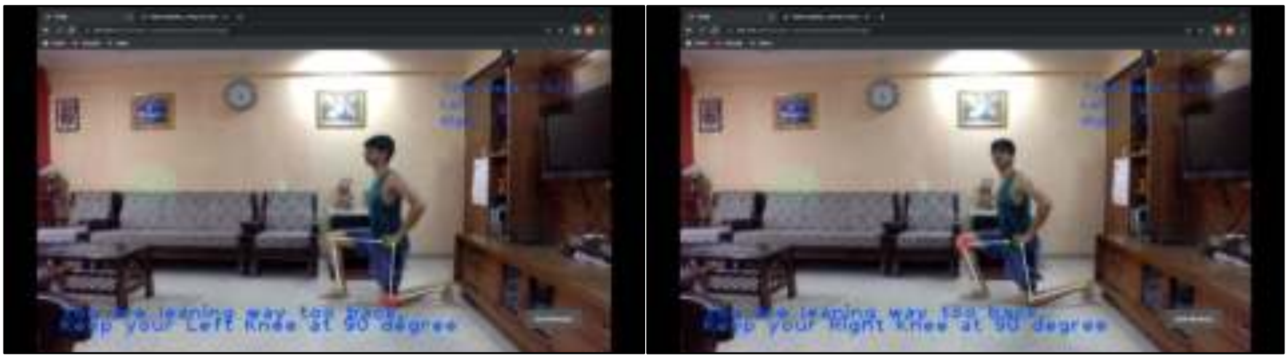


Fig 5.9 Backward Lunges step 2 – corrections

While performing the exercise, if the user goes wrong anywhere it will be indicated in red. In the figure, while bending the left leg behind, the user leaned way to the back and needed to shift the left leg behind and keep his spine straight. Same thing has happened in figure, where the user has leaned way to forward and needs to correct their right knee angle by moving the leg forward.

5.1.5. Knee Pushups

Fig 5.10 Knee Pushups – Perfect Pose

If the user performs the exercise correctly, i.e., the knee and elbow angle are correct as shown in the figure, then the count will increase. While bending down, the knee angle is not sufficiently correct, it will be highlighted in red and hence won't count as perfect pose.

Fig 5.11 Knee Pushups – corrections

As soon as you come back to the normal posture, the model will indicate to bend down and continue doing the exercise as shown in the figure.

5.1.6. Abdominal Crunches

Fig 5.12 Abdominal Crunches – Step 1

The first step of the exercise is to lift up your body from the ground such that you get little pressure on the lower body. If you have not lifted the body sufficiently up from the ground, it will be highlighted in red as shown in the figure.

Fig 5.13 Abdominal Crunches – Knee correction

While performing the exercise, if the knee is not close enough to the thigh, it will be indicated in red as shown in the figure.

Fig 5.14 Abdominal Crunches – Perfect Pose

If the hips and knee angles are correct while performing the exercise, it will be indicated by green and then only the count will increase as shown in the figure.

5.1.7. Cobra Stretch

Fig 5.15 Cobra Stretch – corrections 1

While performing this exercise the legs has to be straight and if its wrong, it will be highlighted in red as shown in the figure. And the user has to keep their face inline with the hand (i.e look straight), if done wrongly it will be highlighted in red as shown in figure.

Fig 5.16 Cobra Stretch – corrections 2

5.2. Diet Plan Management

The system operates in a Machine Learning Environment, where it calculates the user data and provides the recommended Diet type and meal plan to work on. As a result, we train the ML model with various inputs to achieve the desired results for the user. We mainly used four algorithms in this case, which are as follows:

- K nearest Neighbors
- Random Forest (best fit)
- Support Vector Machine
- Logistic Regression

Regardless of whether the user selects to grow muscle, maintain weight, or lose weight, the model will offer a diet plan depending on the data and category selected. In addition, for meal plans, we employed web scraping with attractive soup libraries [9] from various diet-related blogs and professional diet-related lifestyle websites. After web scraping, four arrays with different meal timings were created: Breakfast, Lunch, Snacks, and Dinner. After that, the arrays were saved in a SQLite database. Based on the user's input, all of the information saved in each row is retrieved in the selected category, which includes all of the food items and all of the servings.

Dietary requirements have increased substantially, leading to the development of several diet regimens such as the Keto Diet, Low Carb Diet, High Carb Diet, Zone Diet, Balanced Diet, and Depletion Diet. Each diet plan has its own set of benefits that people can use to gain or lose weight or stay fit, as outlined below:

- Keto Diet: The ketogenic diet is a very low carbohydrate, high fat diet that is similar to the Atkins and low carb diets. It involves significantly reducing carbohydrate consumption and replacing it with fat. A ketogenic diet can drastically reduce blood sugar and insulin levels.
- Low Carb Diet: A low-carb diet limits carbohydrates such as those found in grains, starchy vegetables, and fruit while emphasizing foods high in protein and fat. A low-carb diet is an effective way to lose weight [10].
- High Carb Diet: Diets high in carbohydrates reduce body weight and body fat and improve insulin function in overweight individuals. Complex carbs are naturally high in fibre, which is a component found in plant foods that provides bulk to the diet without adding calories.
- Zone Diet: It is a diet plan that emphasizes the intake of low-carbohydrate foods. The Zone diet is designed to minimize inflammation and promote healthy insulin levels. It may assist people in balancing their protein and carbohydrate intake [11].
- Balanced Diet: A well-balanced diet includes essential elements such as carbs, fats, vitamins, minerals, proteins, and fibre. A balanced diet includes sufficient and healthy food that promotes good health. A healthy, well-balanced diet lowers the risk of disease and enhances general health.
- Depletion Diet: Carbohydrate depletion diets limit either all dietary carbohydrates or certain types of carbohydrates. The majority of these diets are designed to help you lose weight.

Chapter 6

Working

This chapter presents the screenshots of various sections of our smart fitness assistant like landing page, onboarding procedure, user-login, dashboard, exercise playlists, diet plans, progress chart, workout performance report and, finally, user profile.

Landing Page

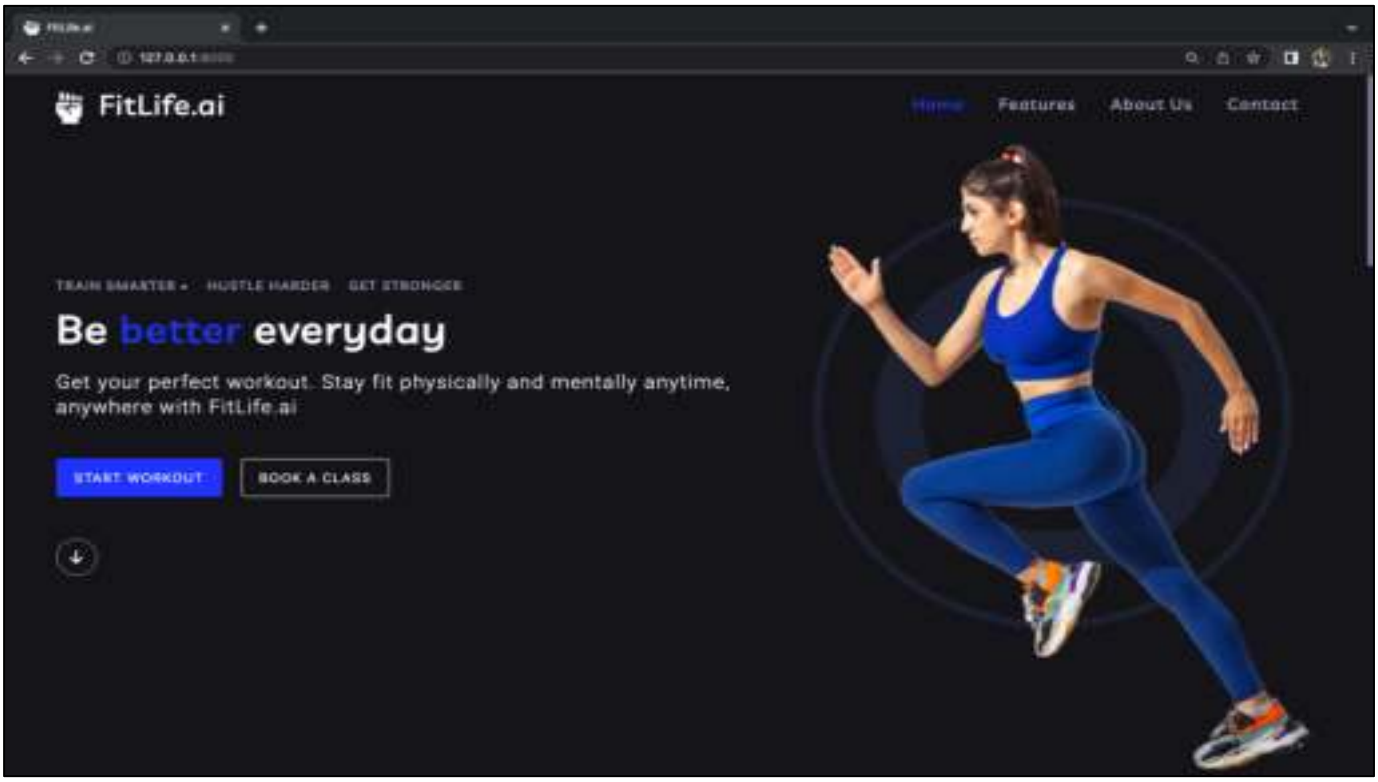


Fig 6.1 Home Page

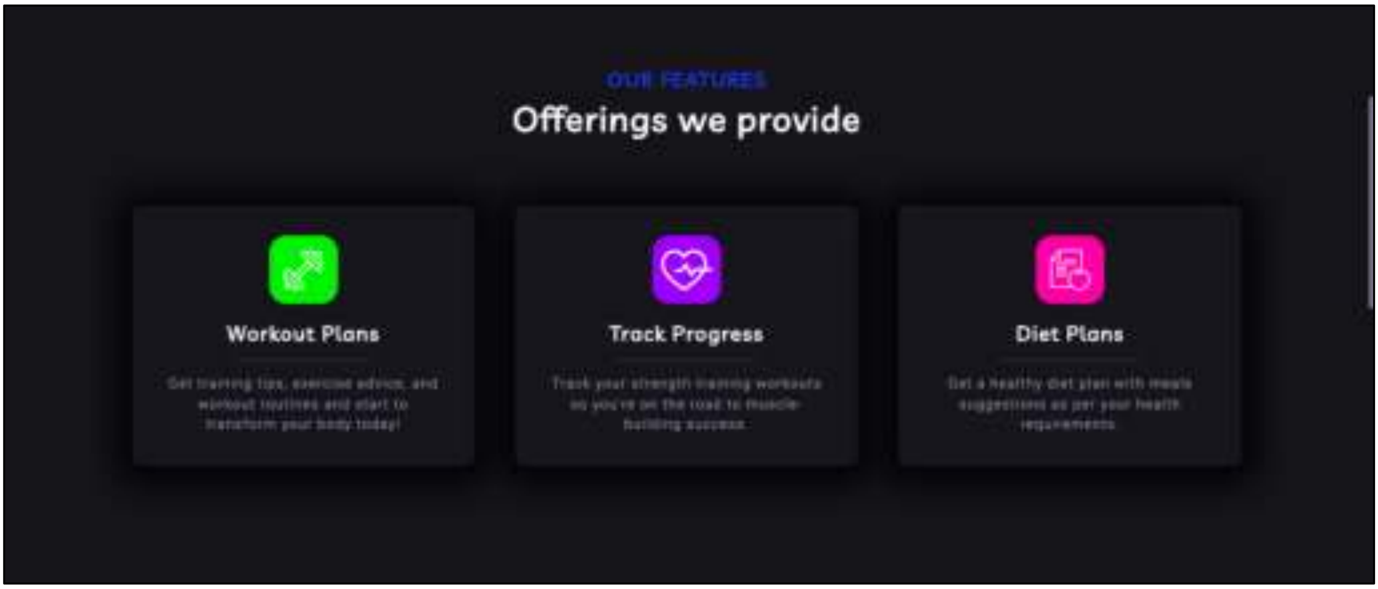


Fig 6.2 Our Features

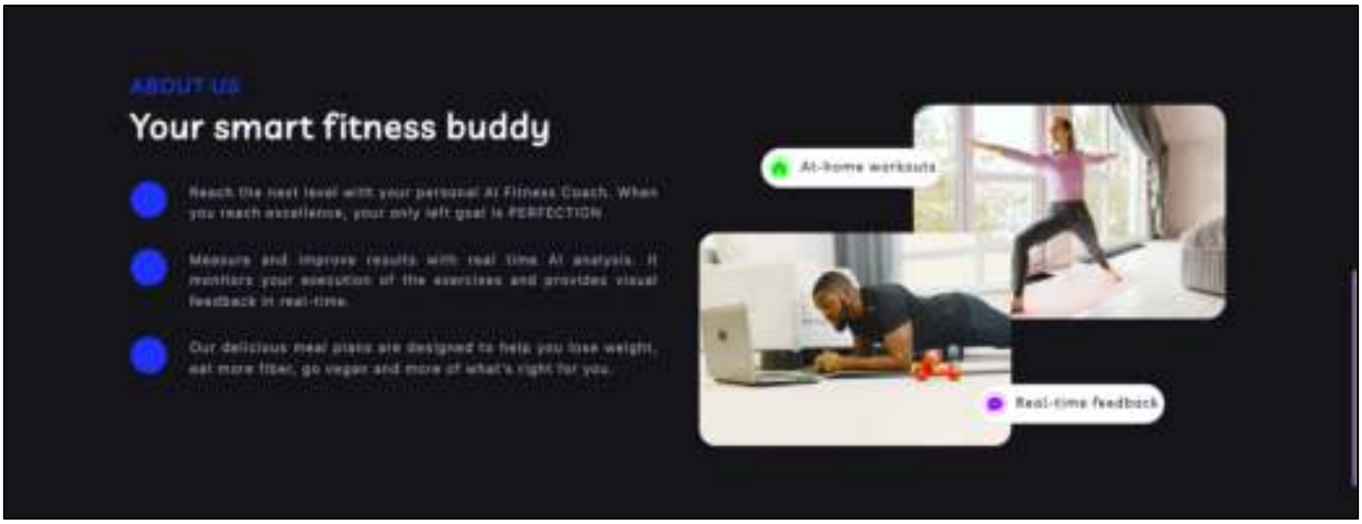


Fig 6.3 About Fitlife.ai

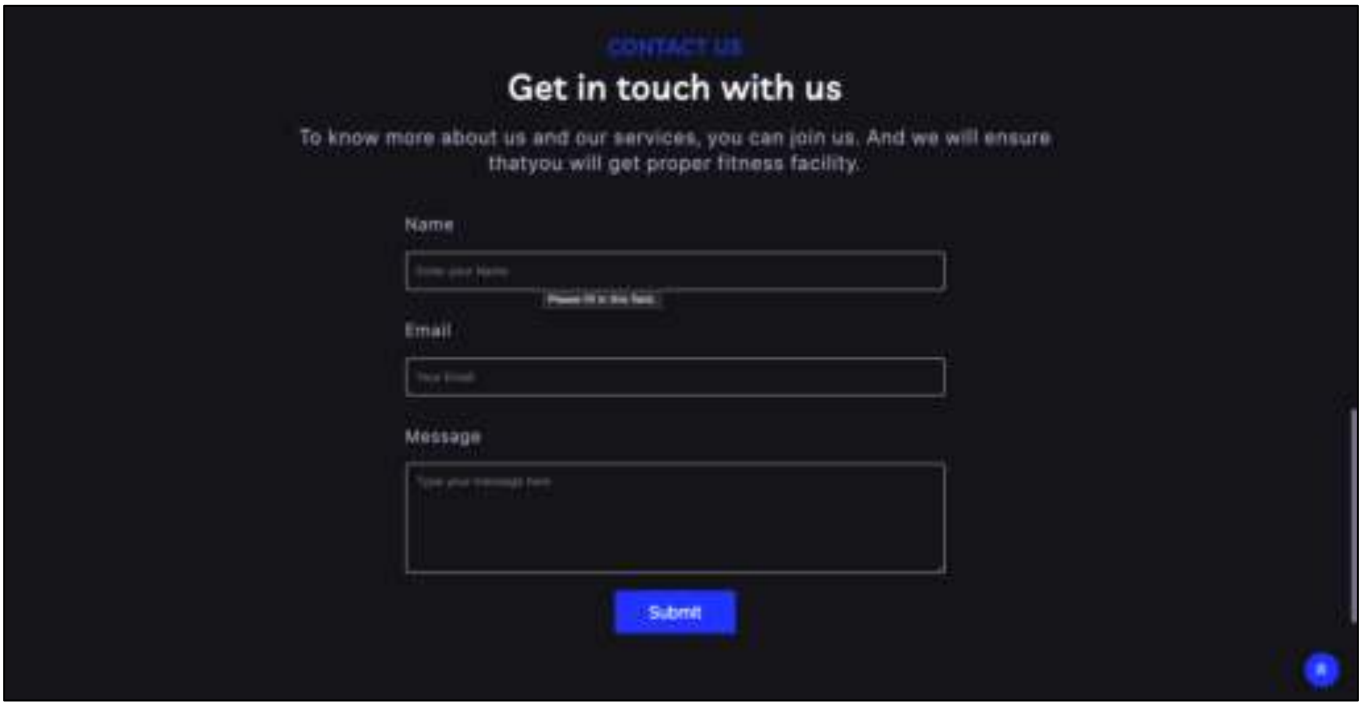


Fig 6.4 Contact Us

Login

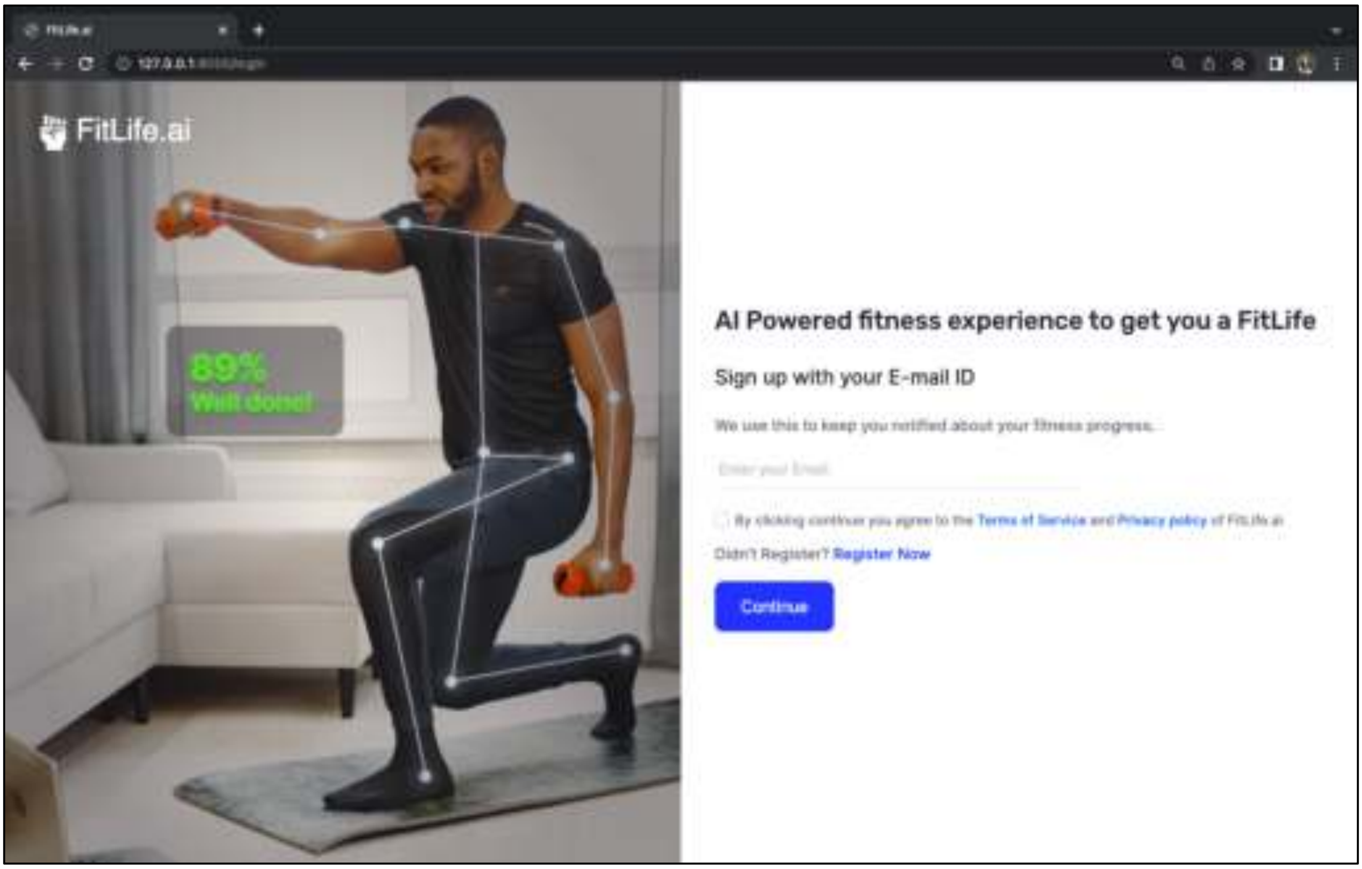


Fig 6.5 Login Page

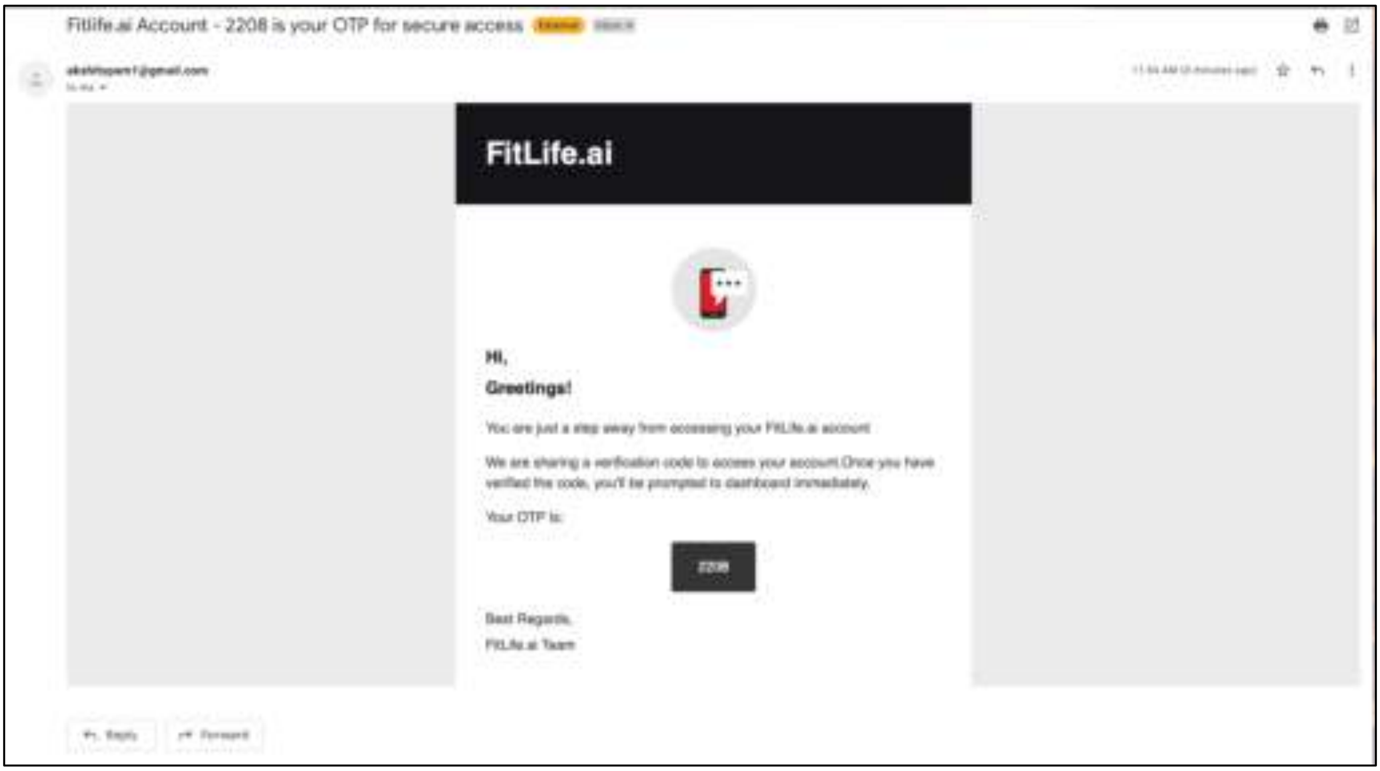


Fig 6.6 Email of OTP generation

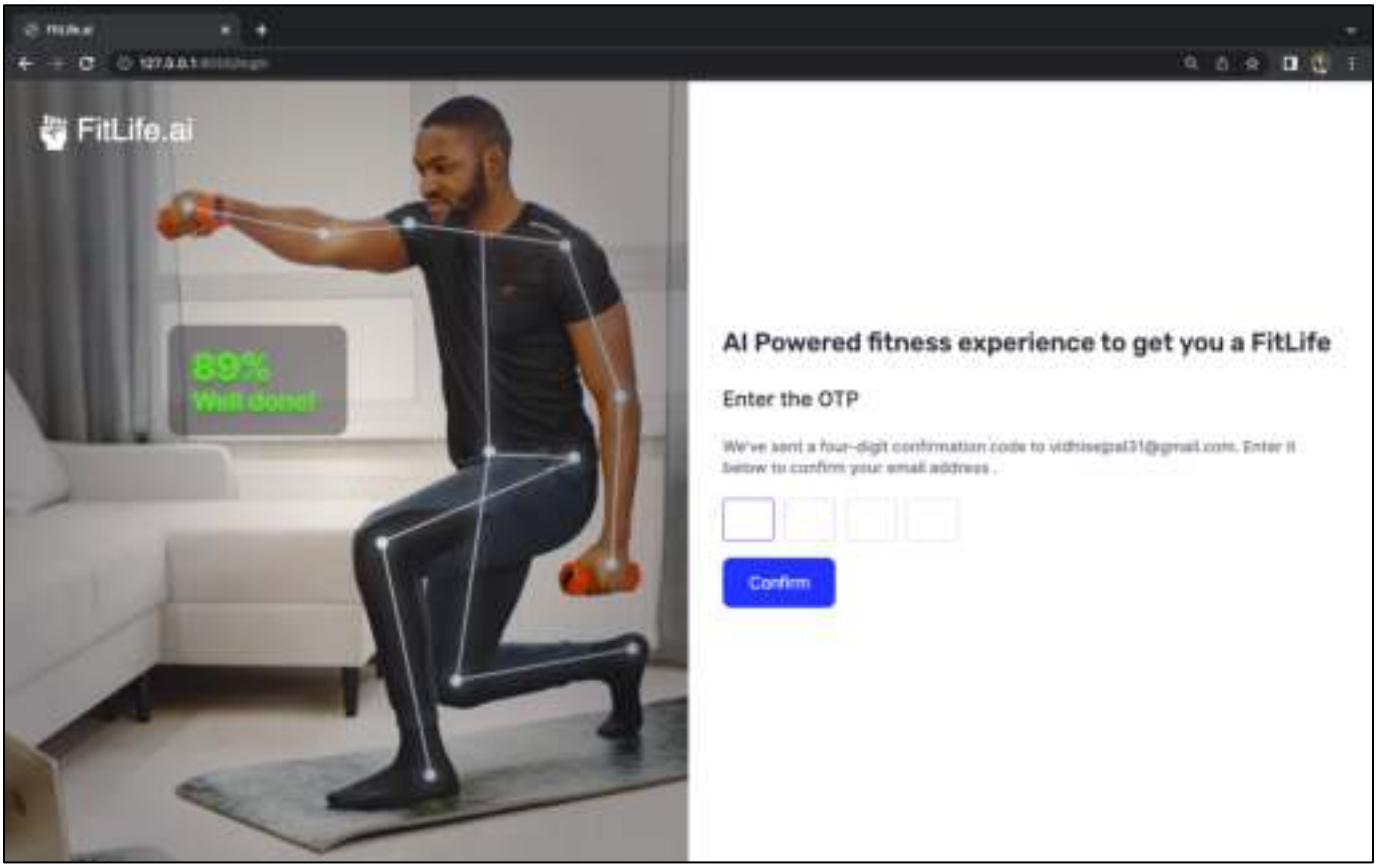


Fig 6.7 Page for inputting the OTP

Onboarding Process

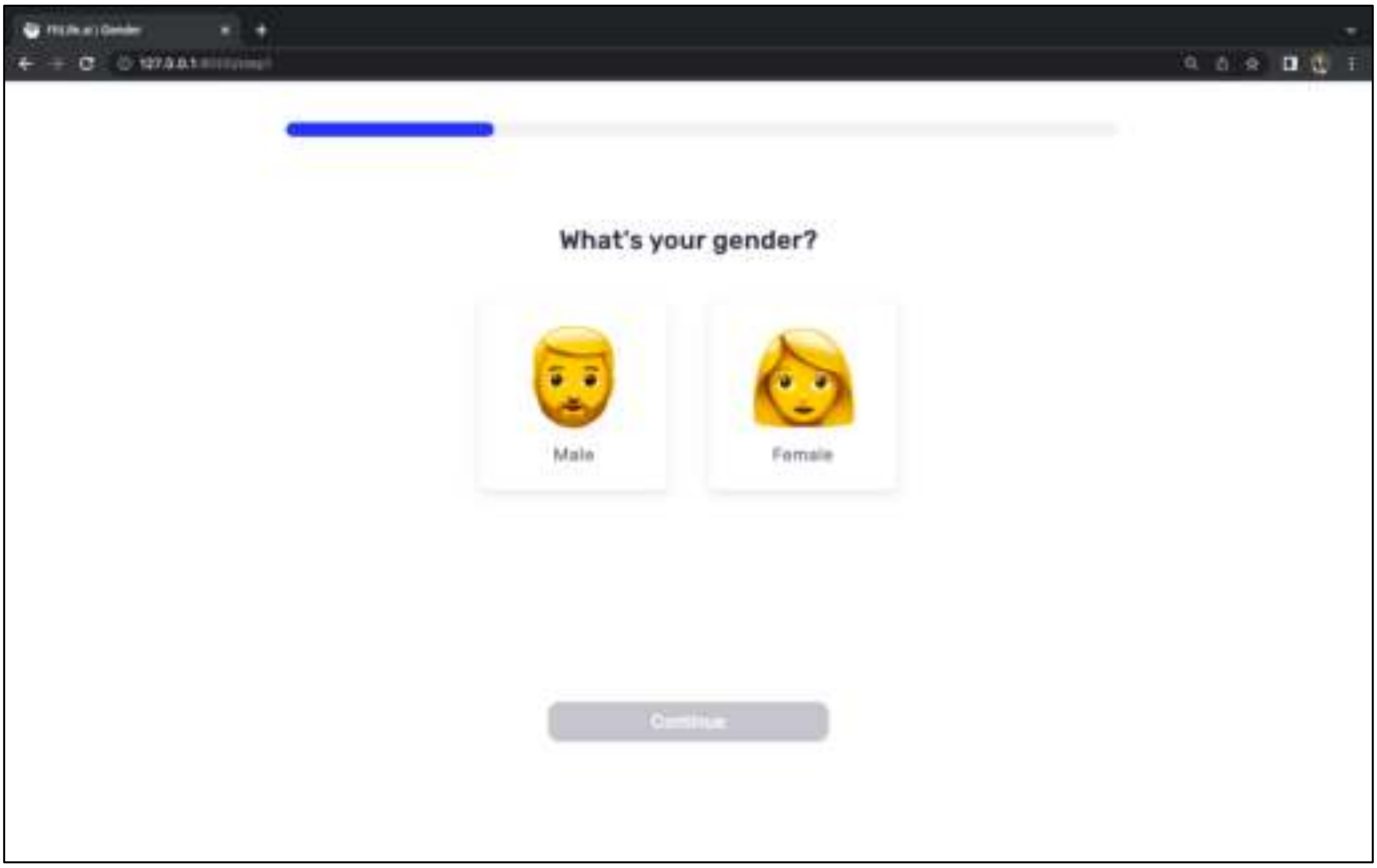
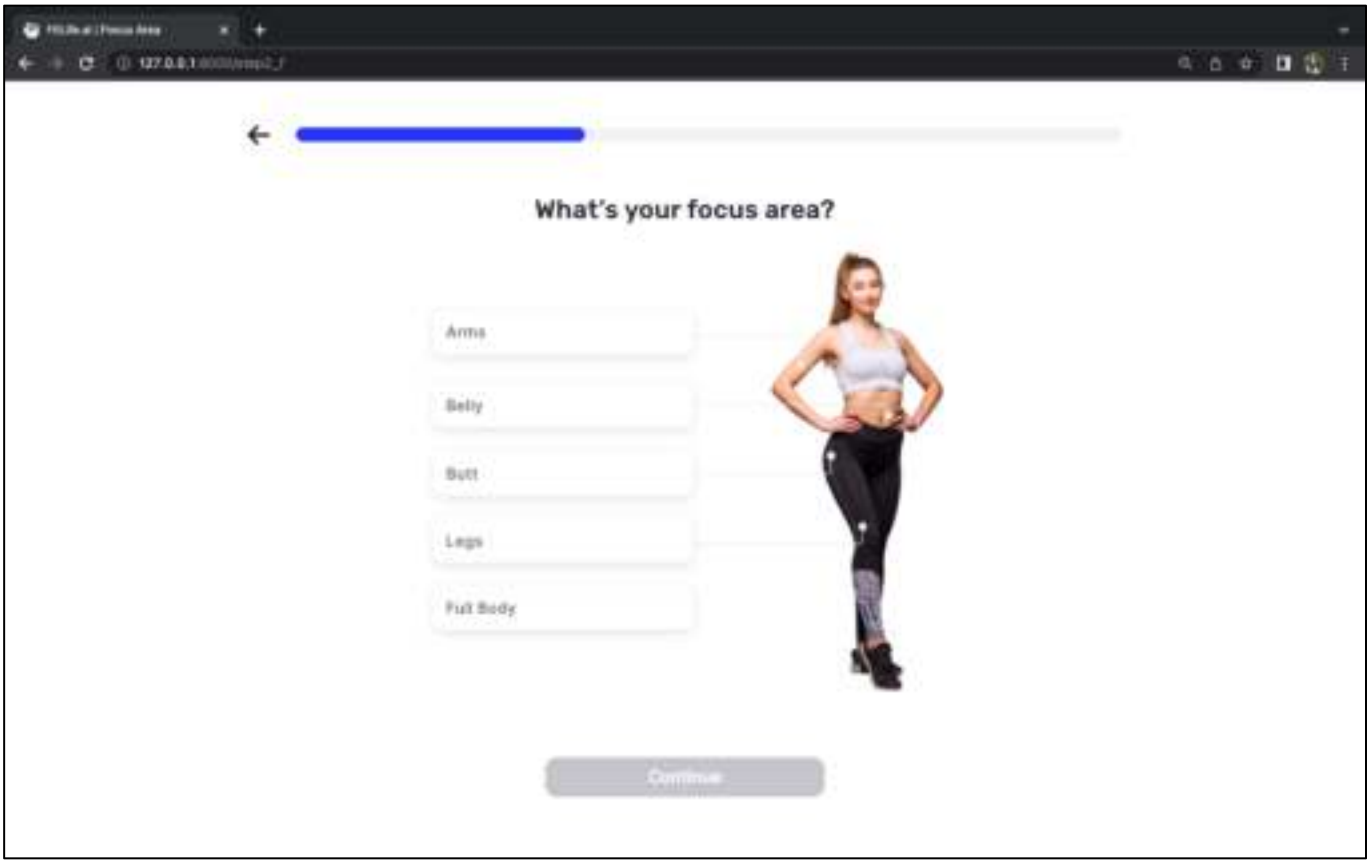


Fig 6.8 Onboarding Step1 : Gender of User



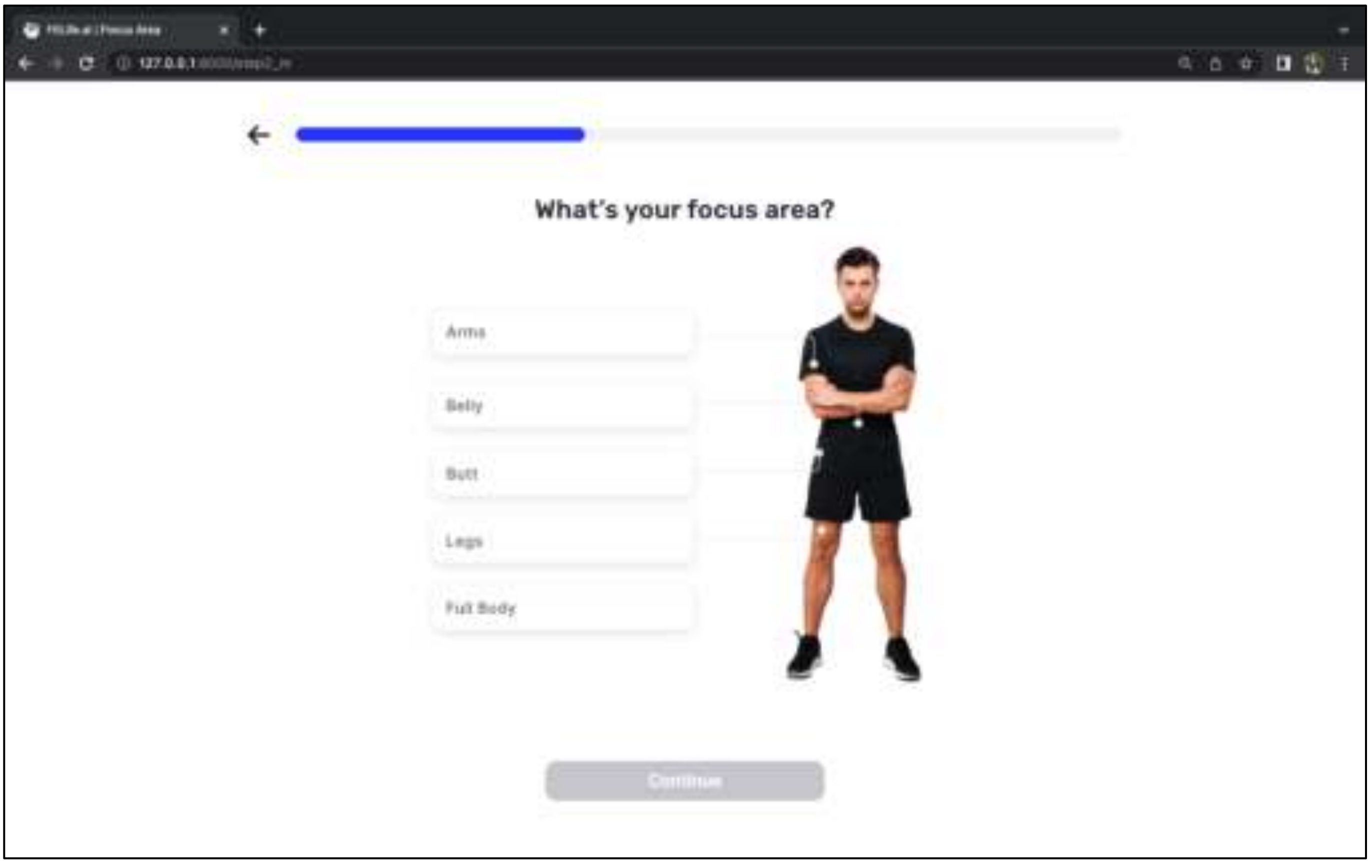


Fig 6.9 Onboarding Step2 : Focus Area of User

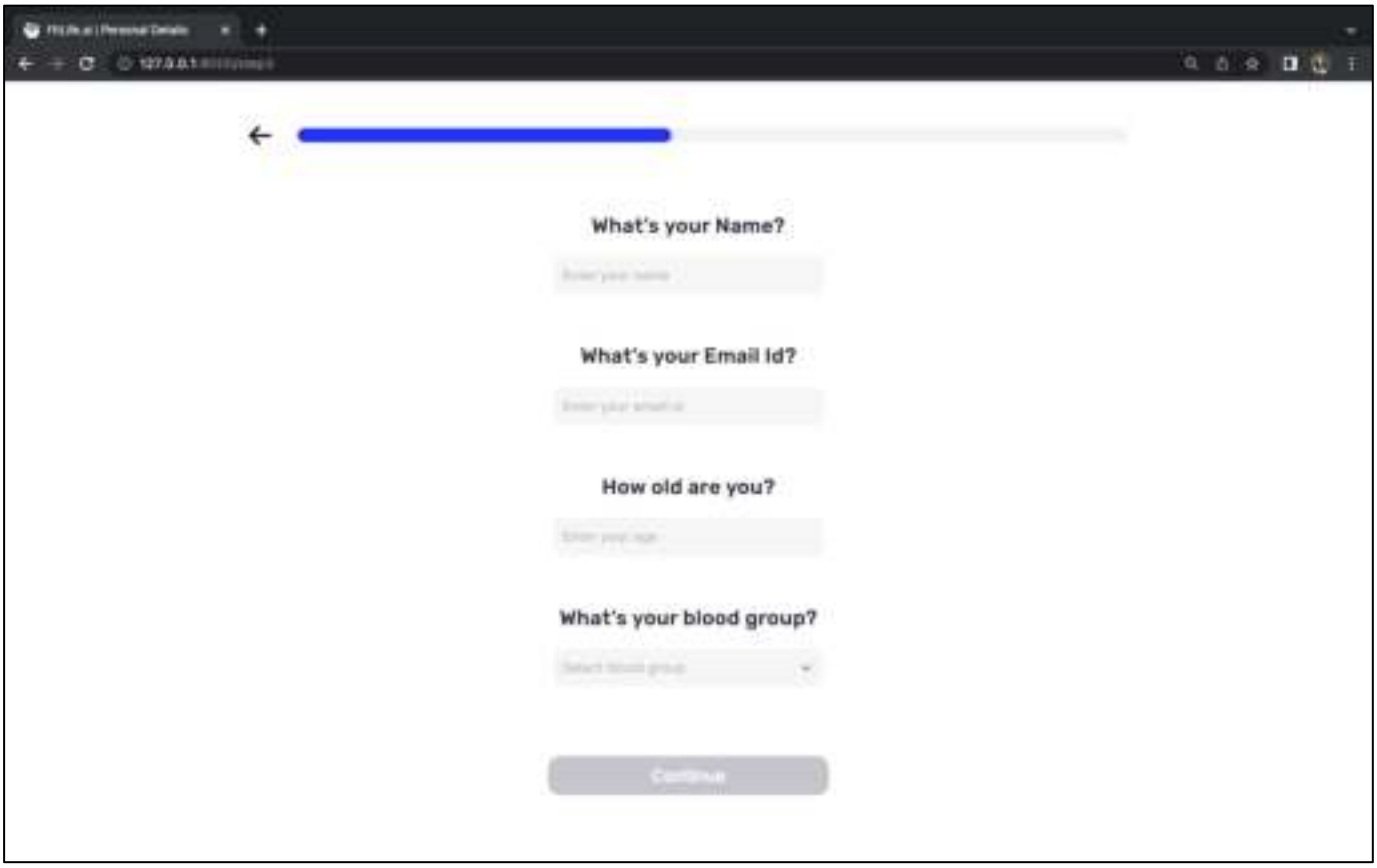


Fig 6.10 Onboarding Step3 : Personal Details of User

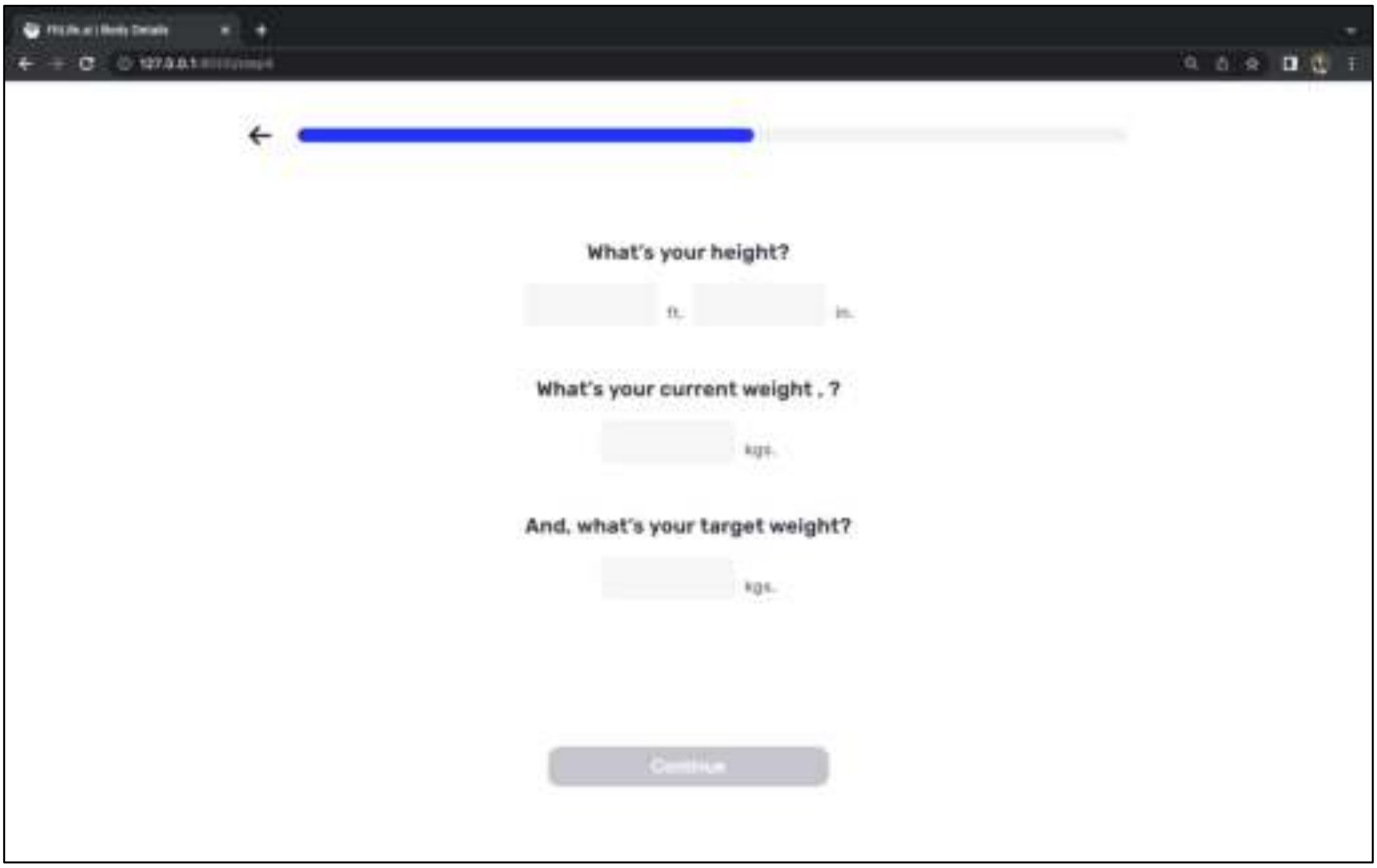


Fig 6.11 Onboarding Step4 : Body Details of User

Fig 6.12 Onboarding Step5 : Active Status of User

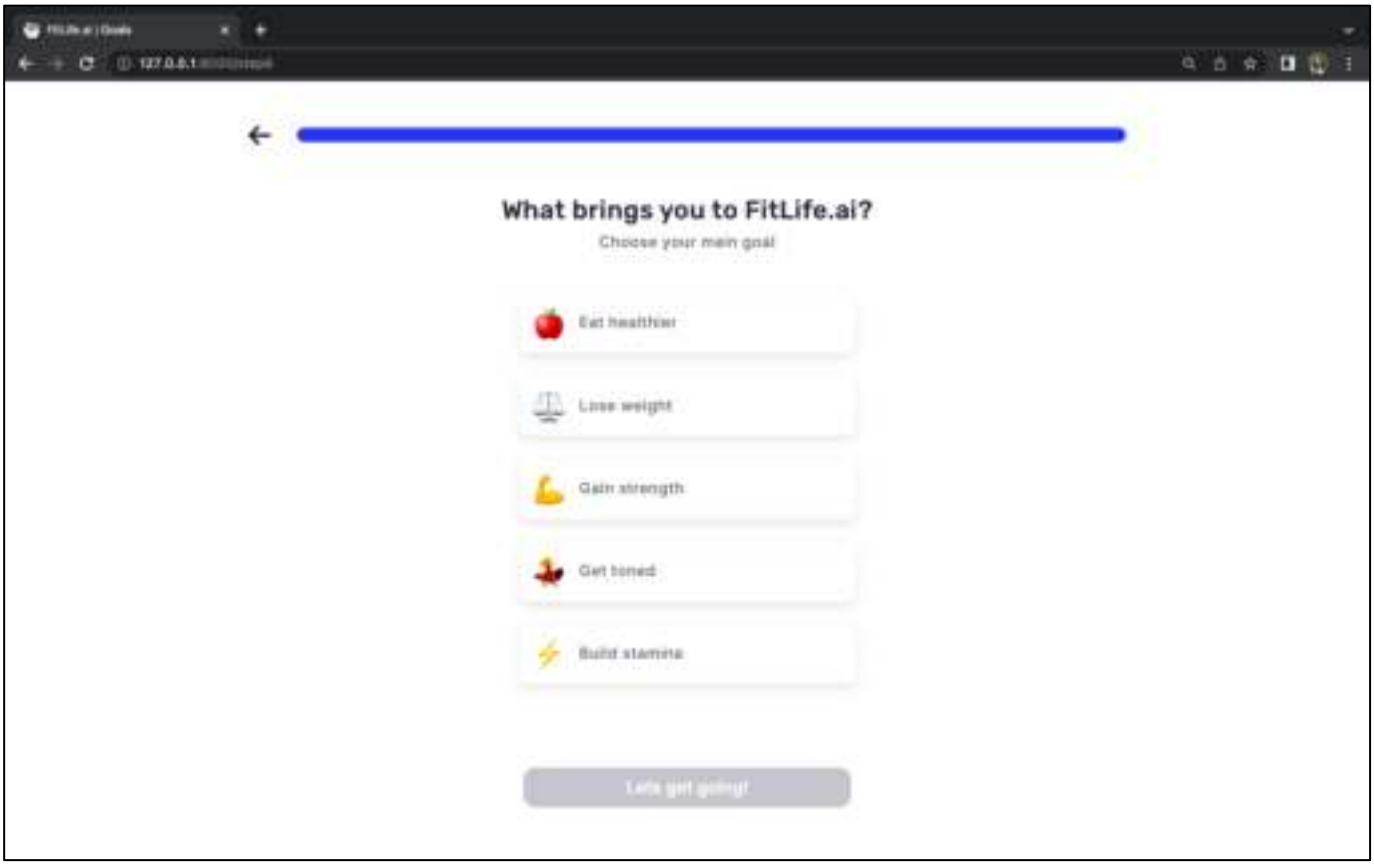


Fig 6.13 Onboarding Step6 : Main Goal of User

Dashboard

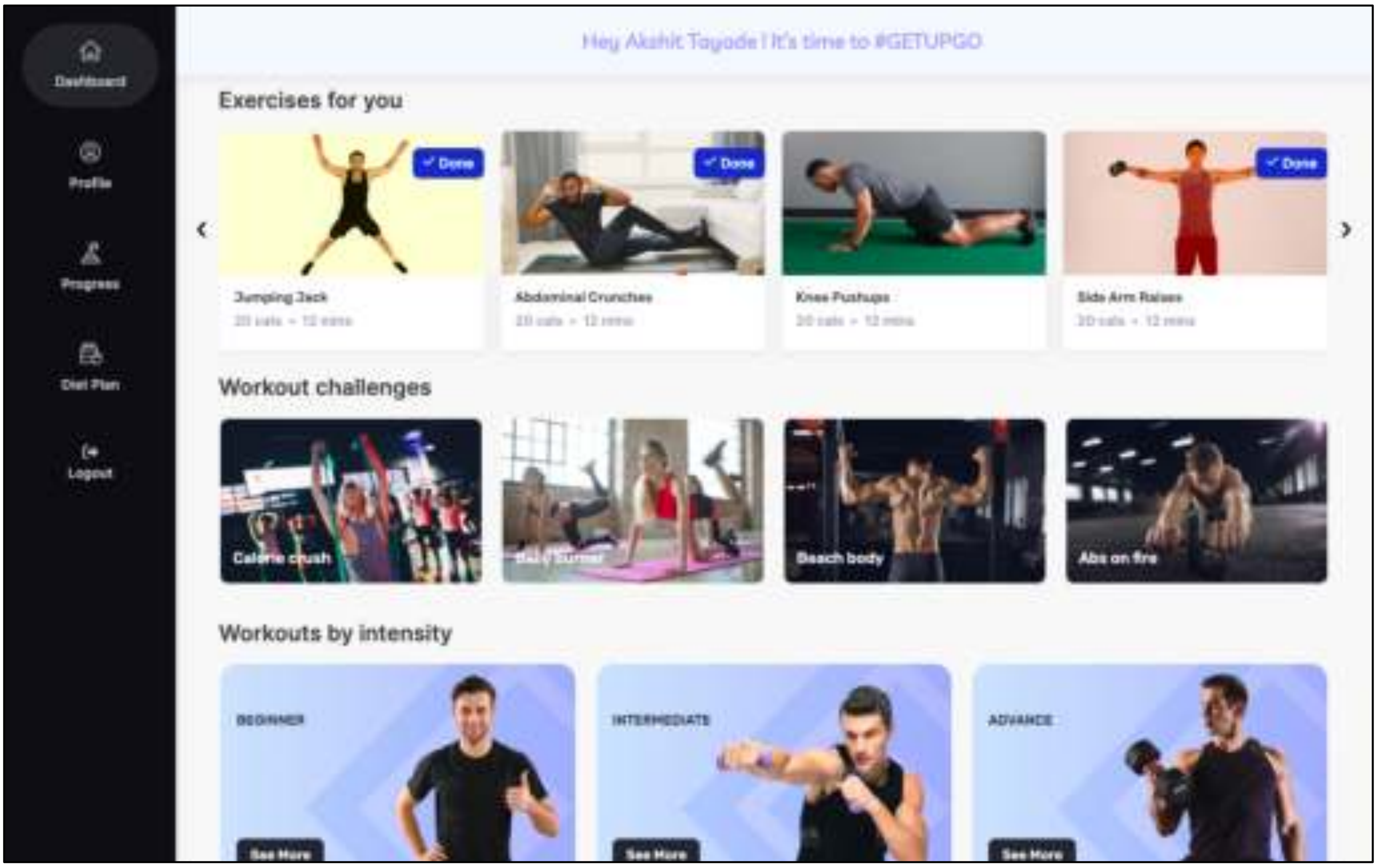
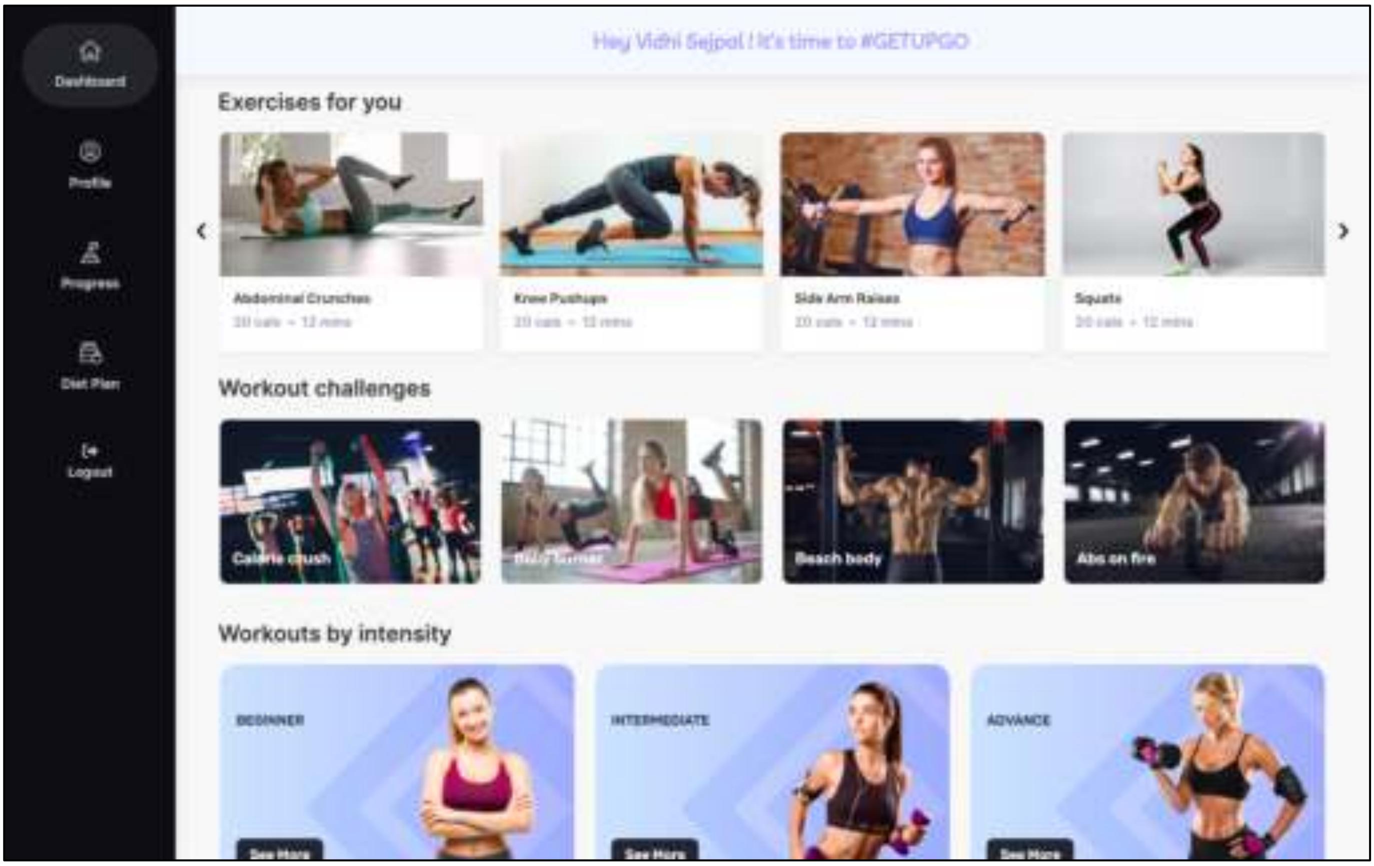


Fig 6.14 Dashboard



Beginner Playlist

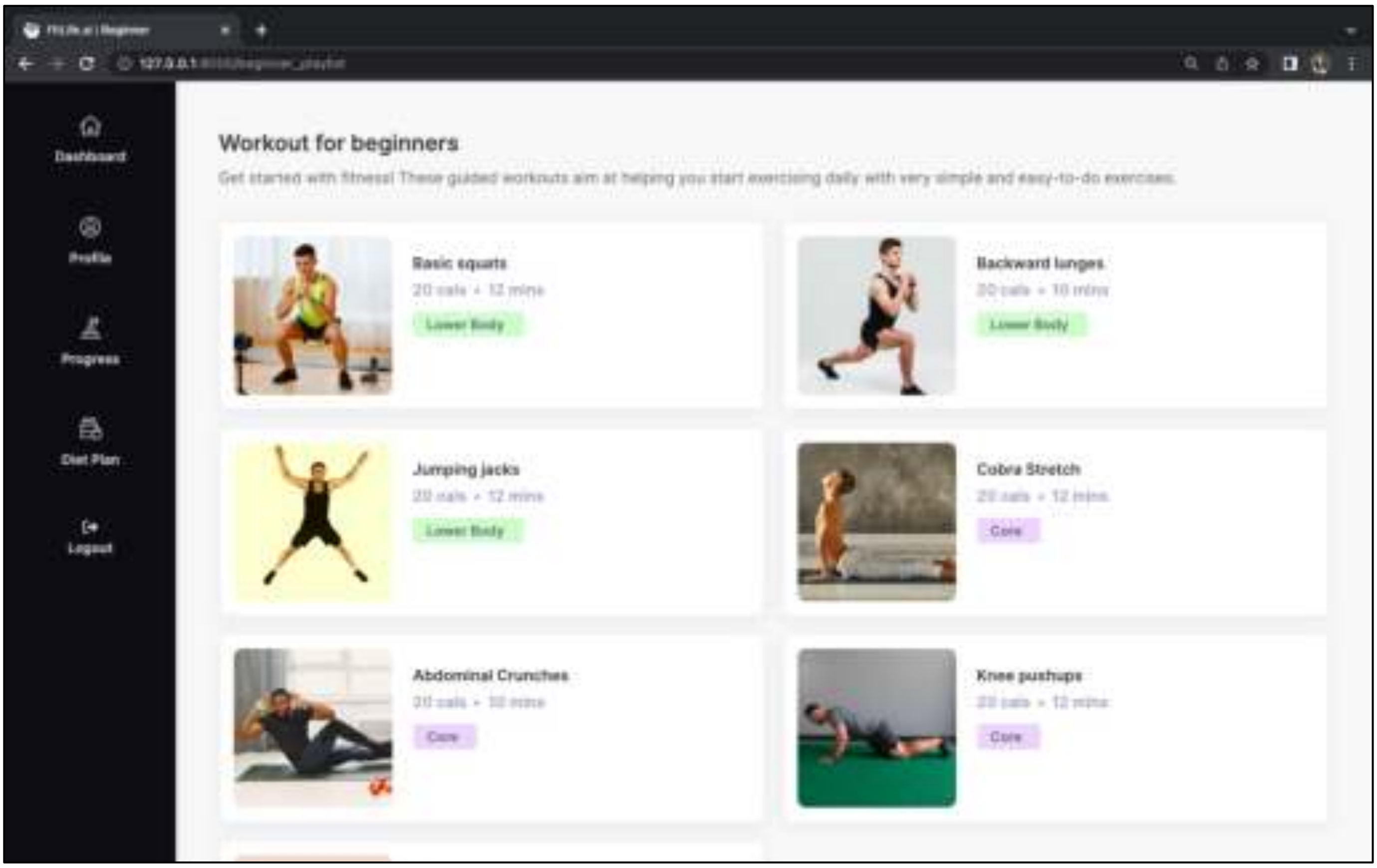


Fig 6.15 Beginner Playlist

Intermediate Playlist

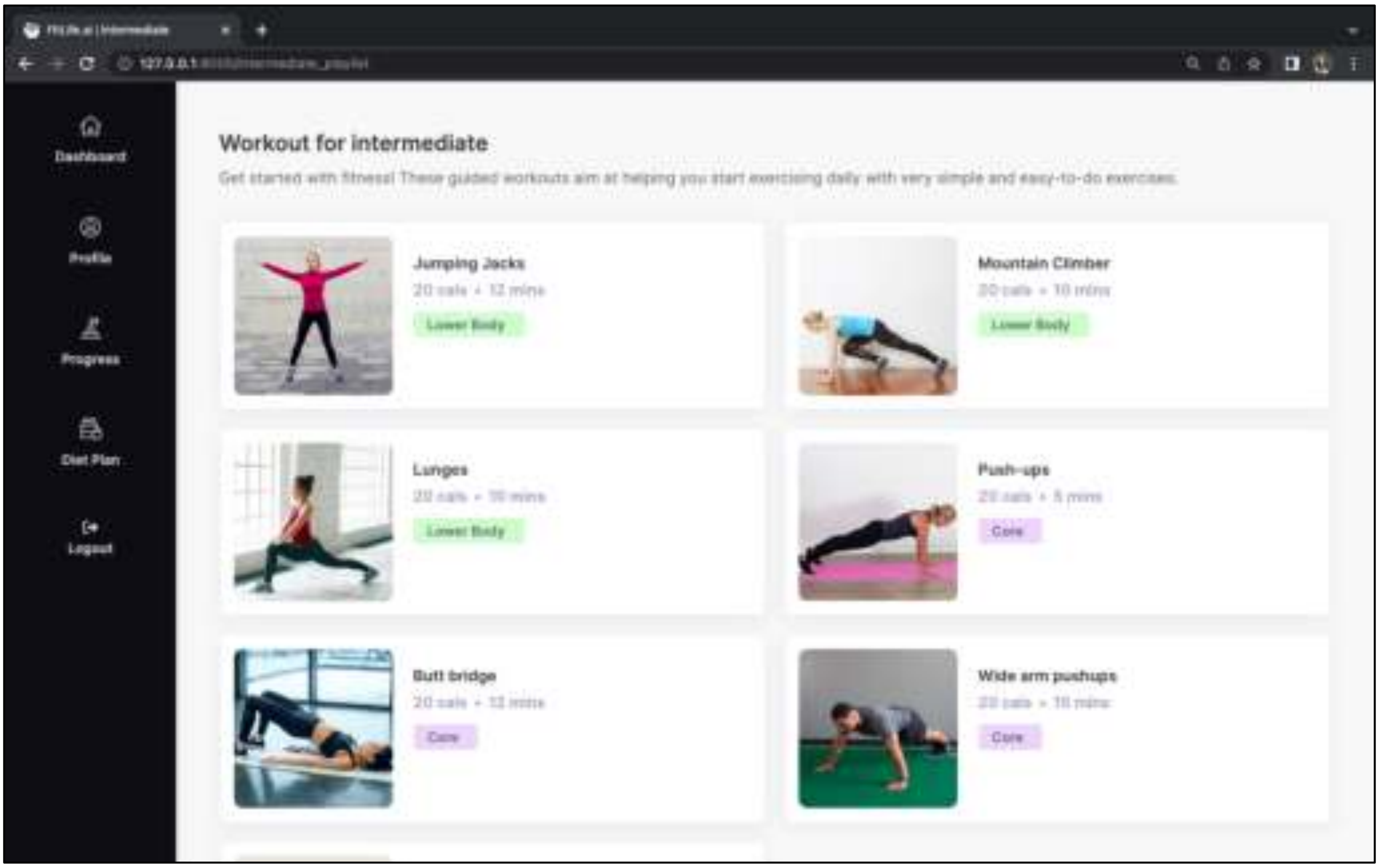


Fig 6.16 Intermediate Playlist

Advance Playlist

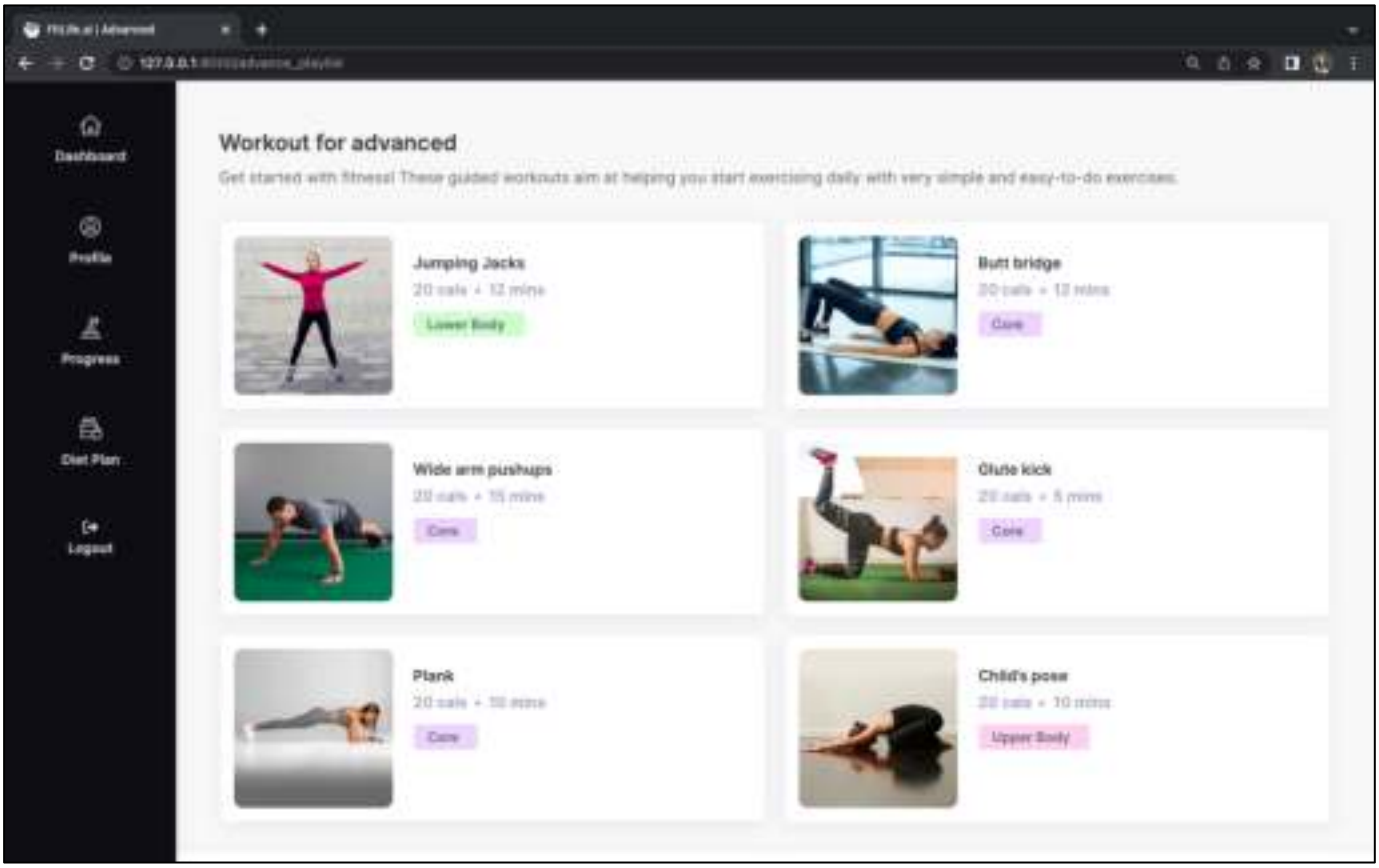


Fig 6.17 Advance Playlist

Exercise Details

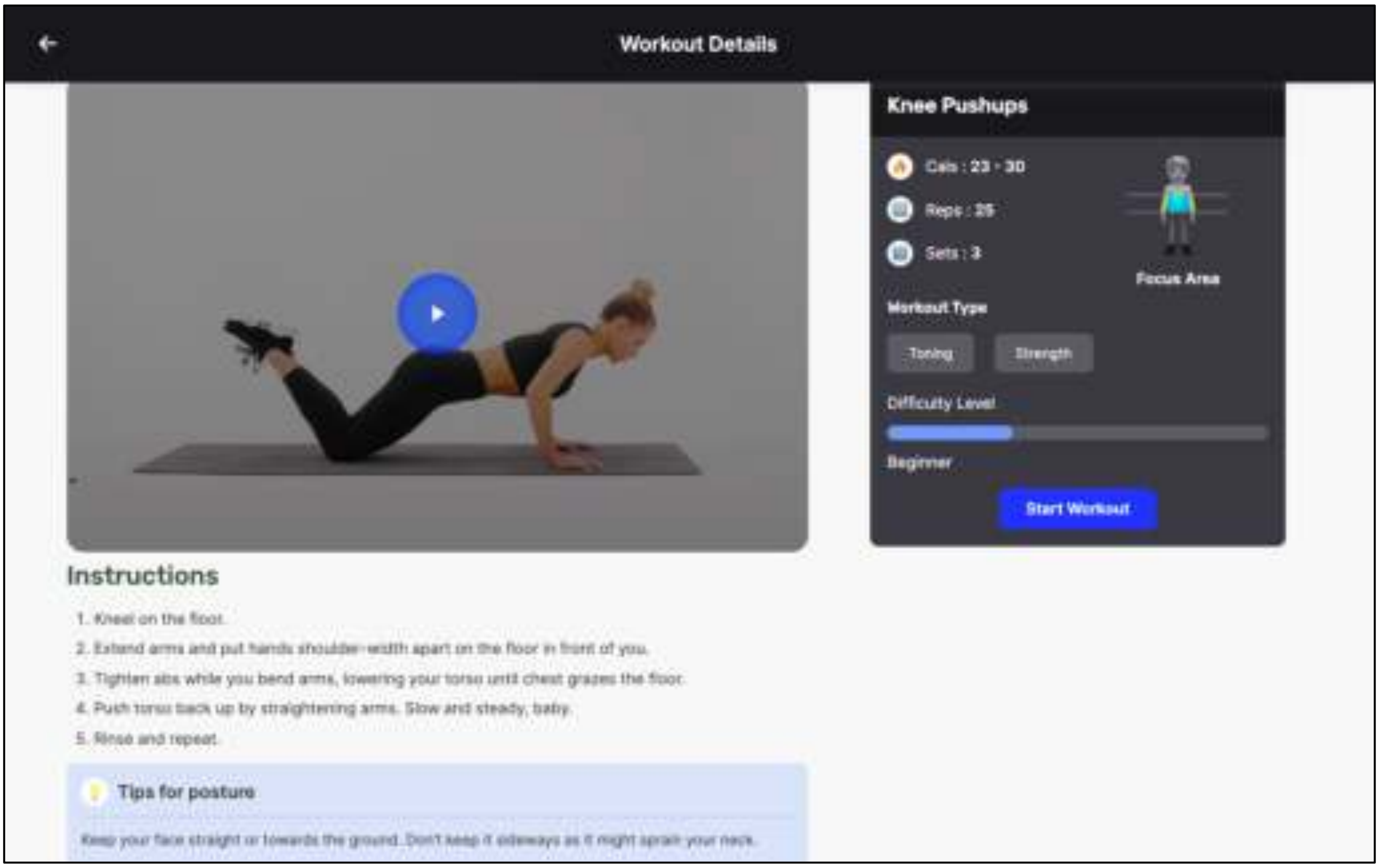


Fig 6.18 Exercise Description

End Workout

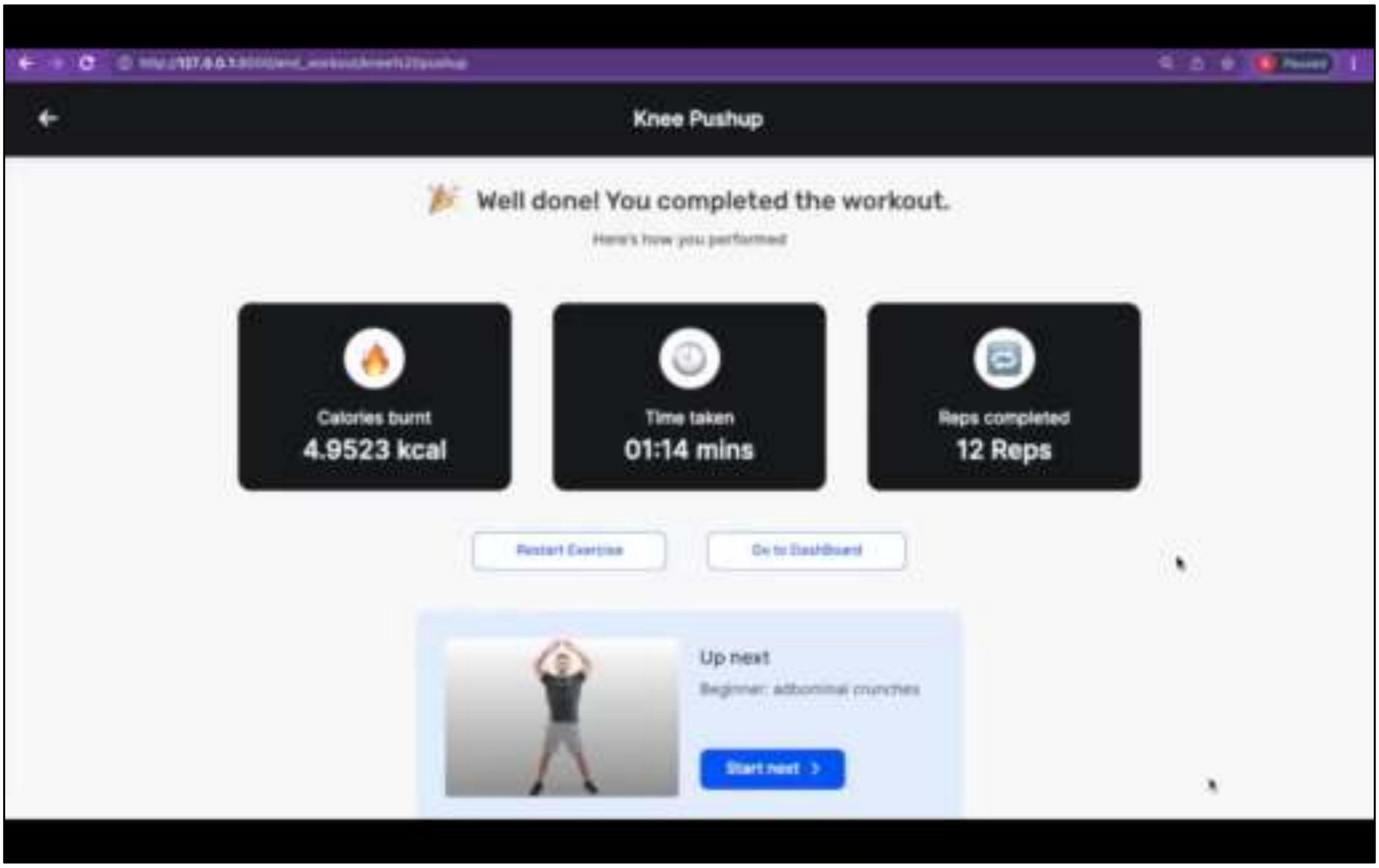


Fig 6.19 Progress after the workout

Profile

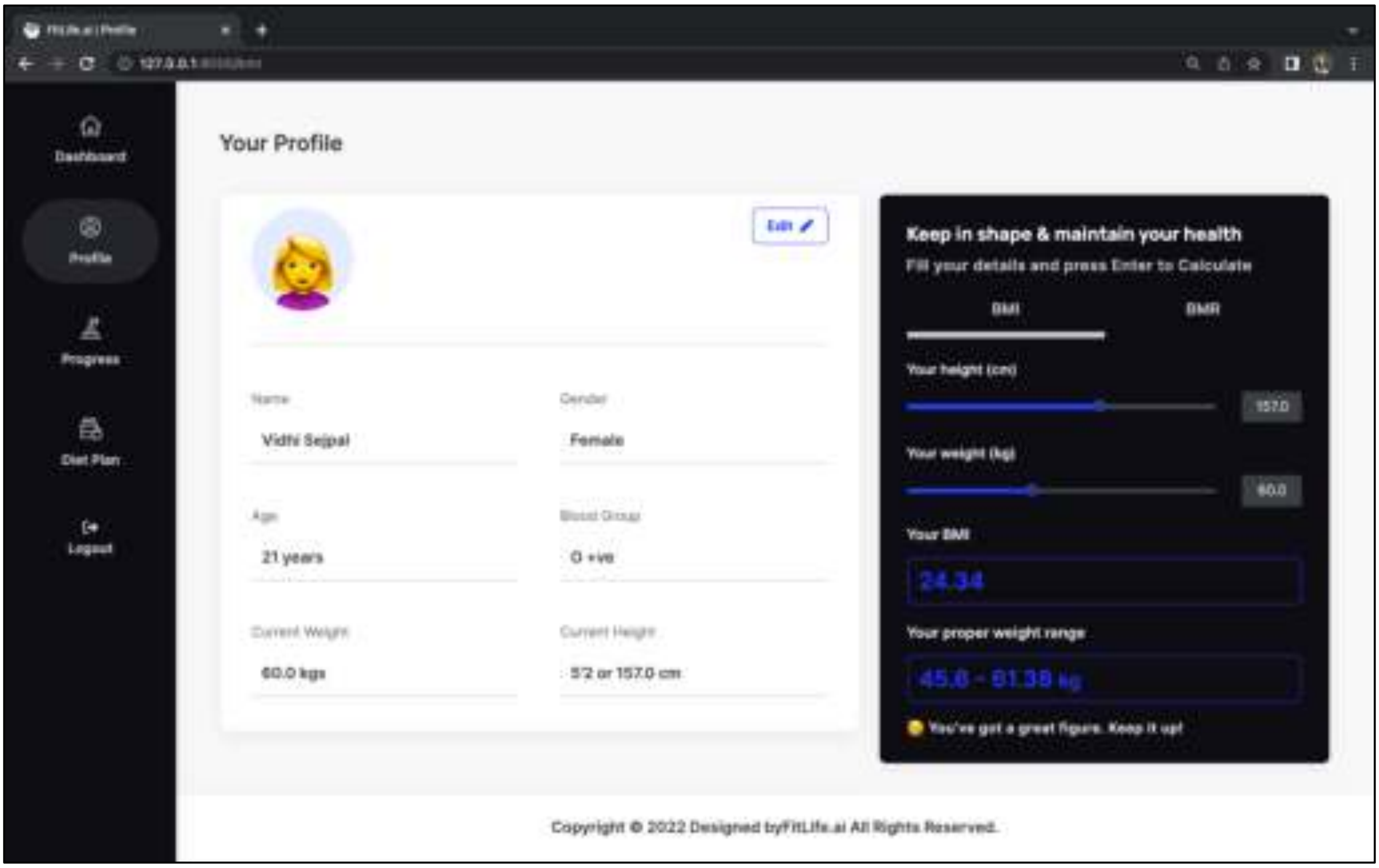


Fig 6.20 BMI Interpretation

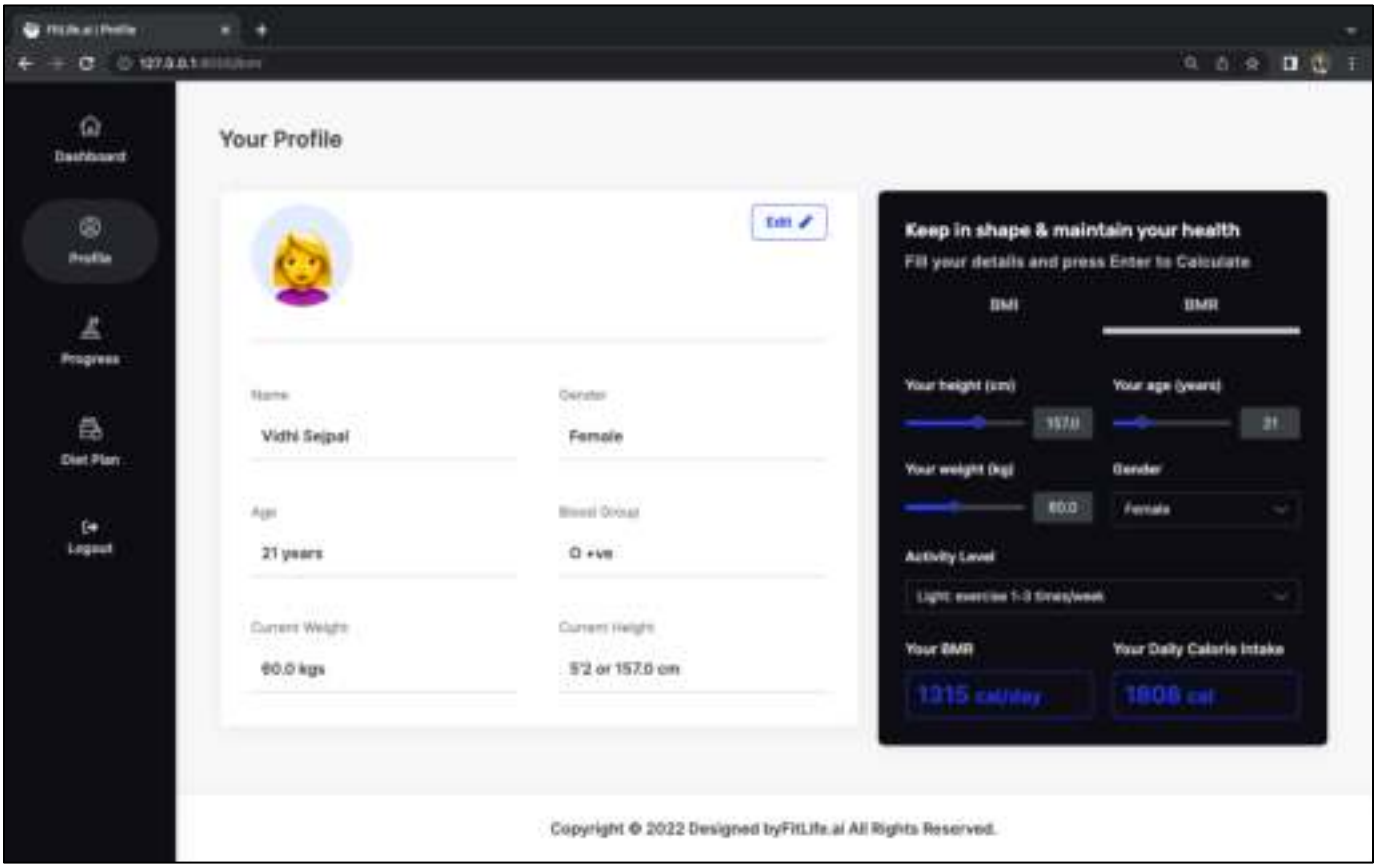


Fig 6.21 BMR Interpretation

Progress

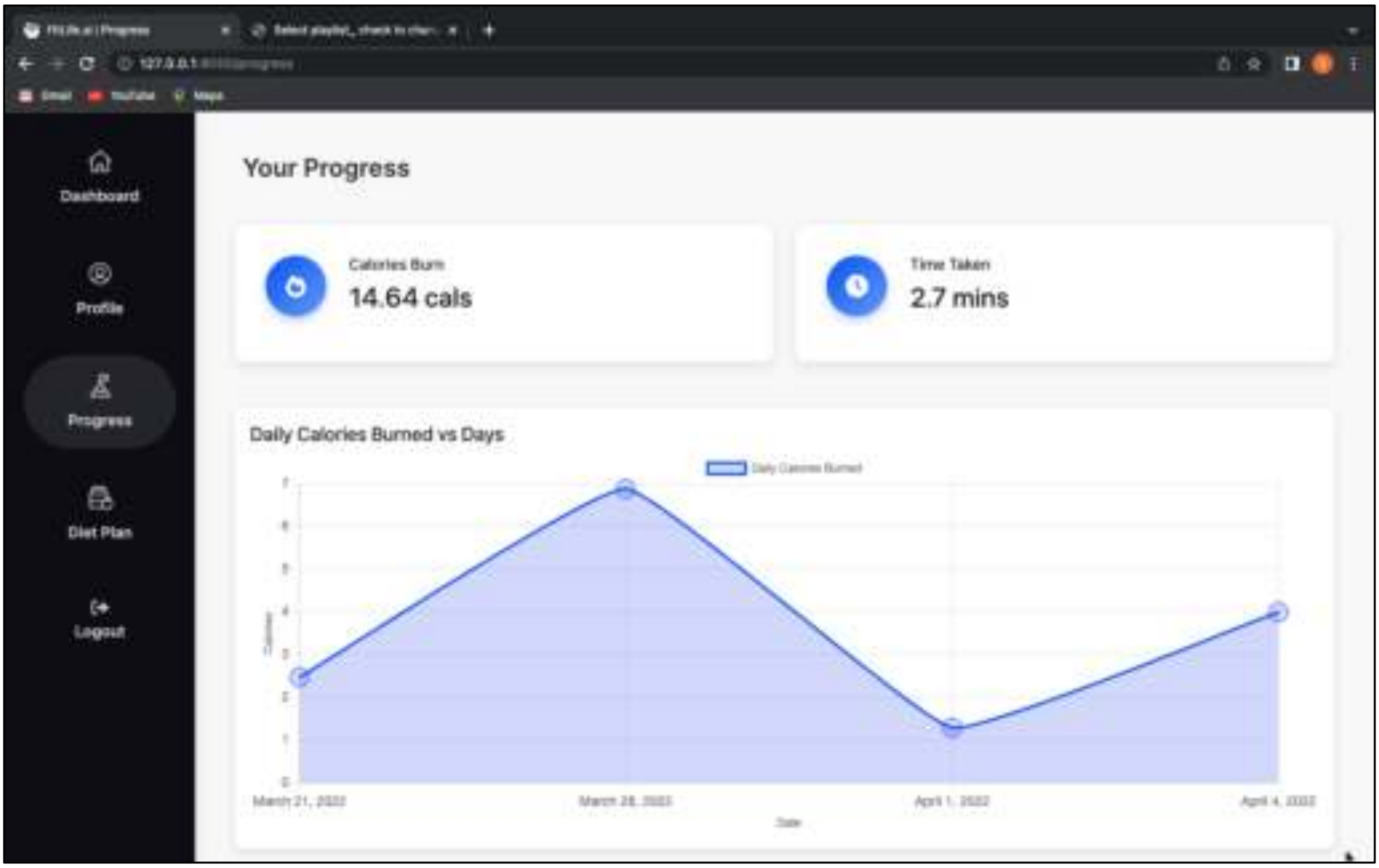


Fig 6.22 Progress of Daily calories burned

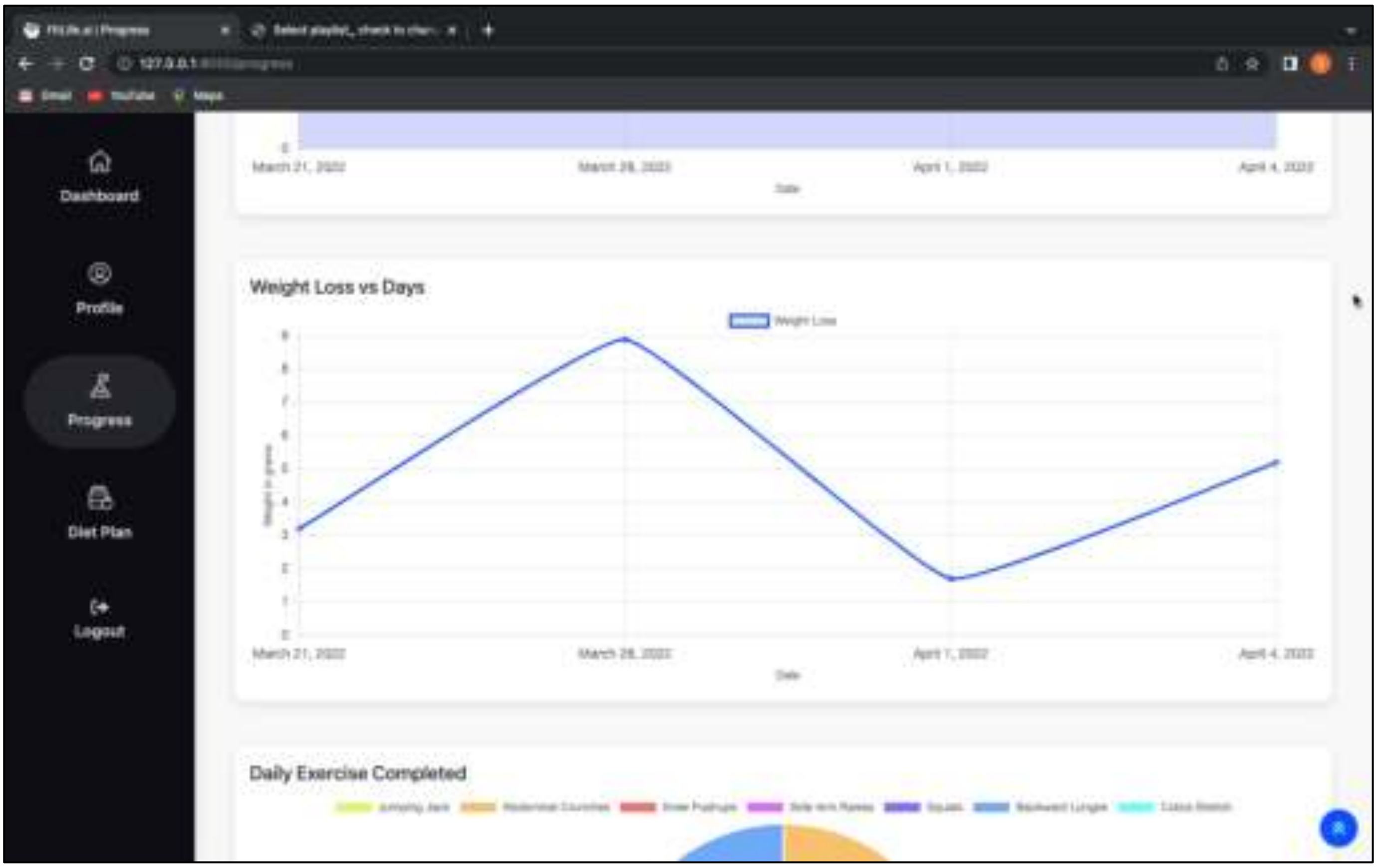


Fig 6.23 Progress of weight loss

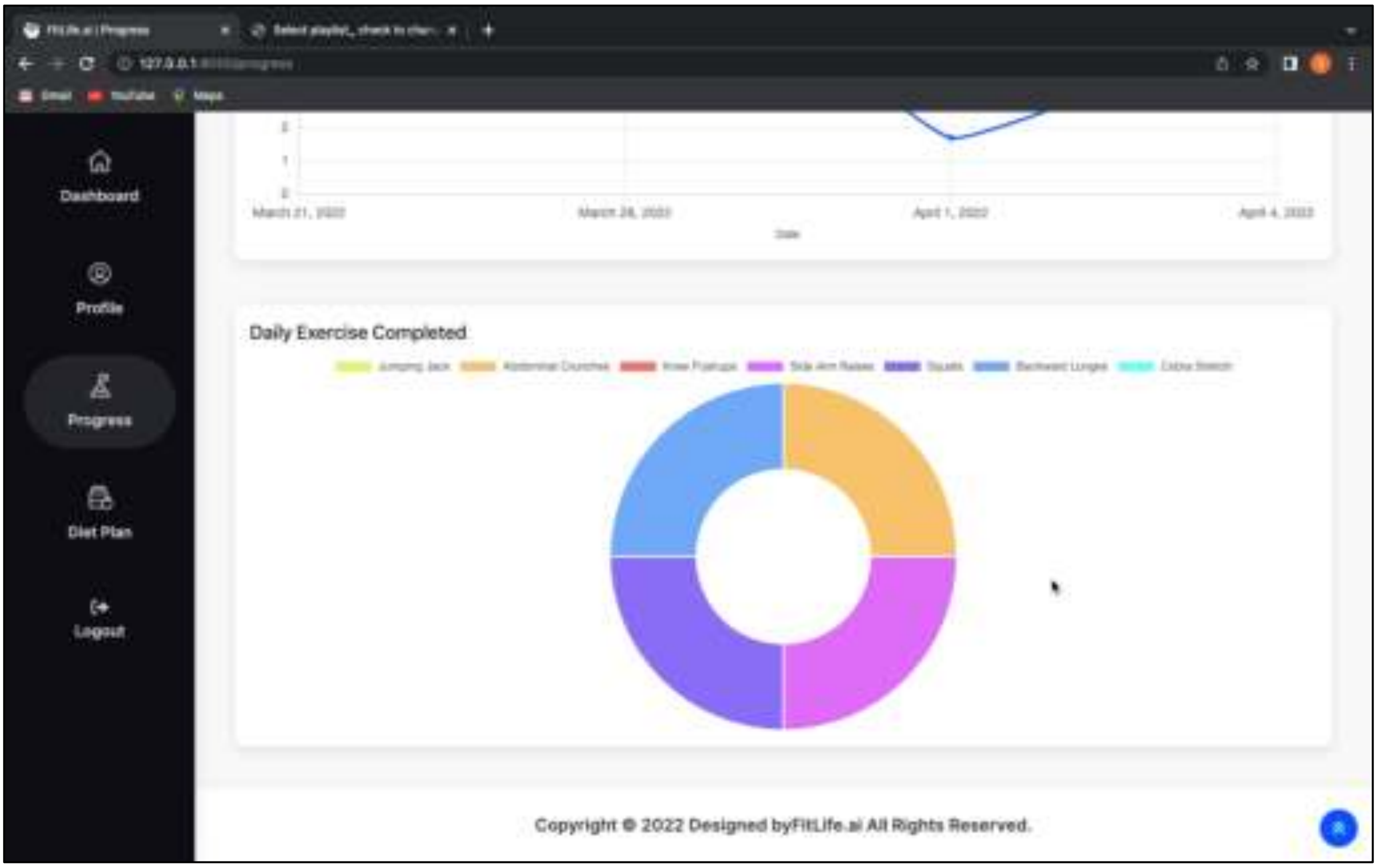


Fig 6.24 Progress of Daily Exercise Completed

Diet Plans

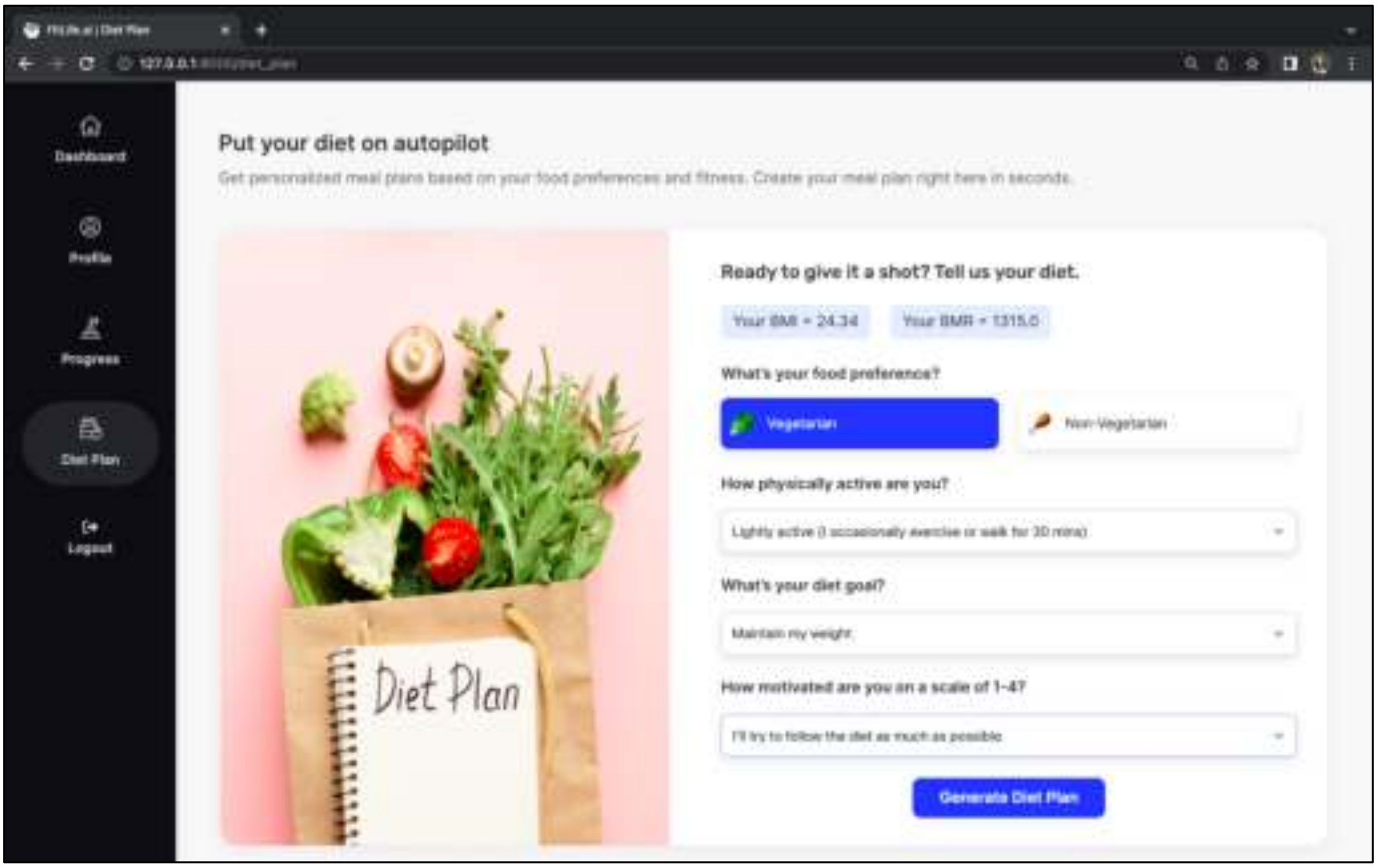


Fig 6.25 Diet Plan Generation

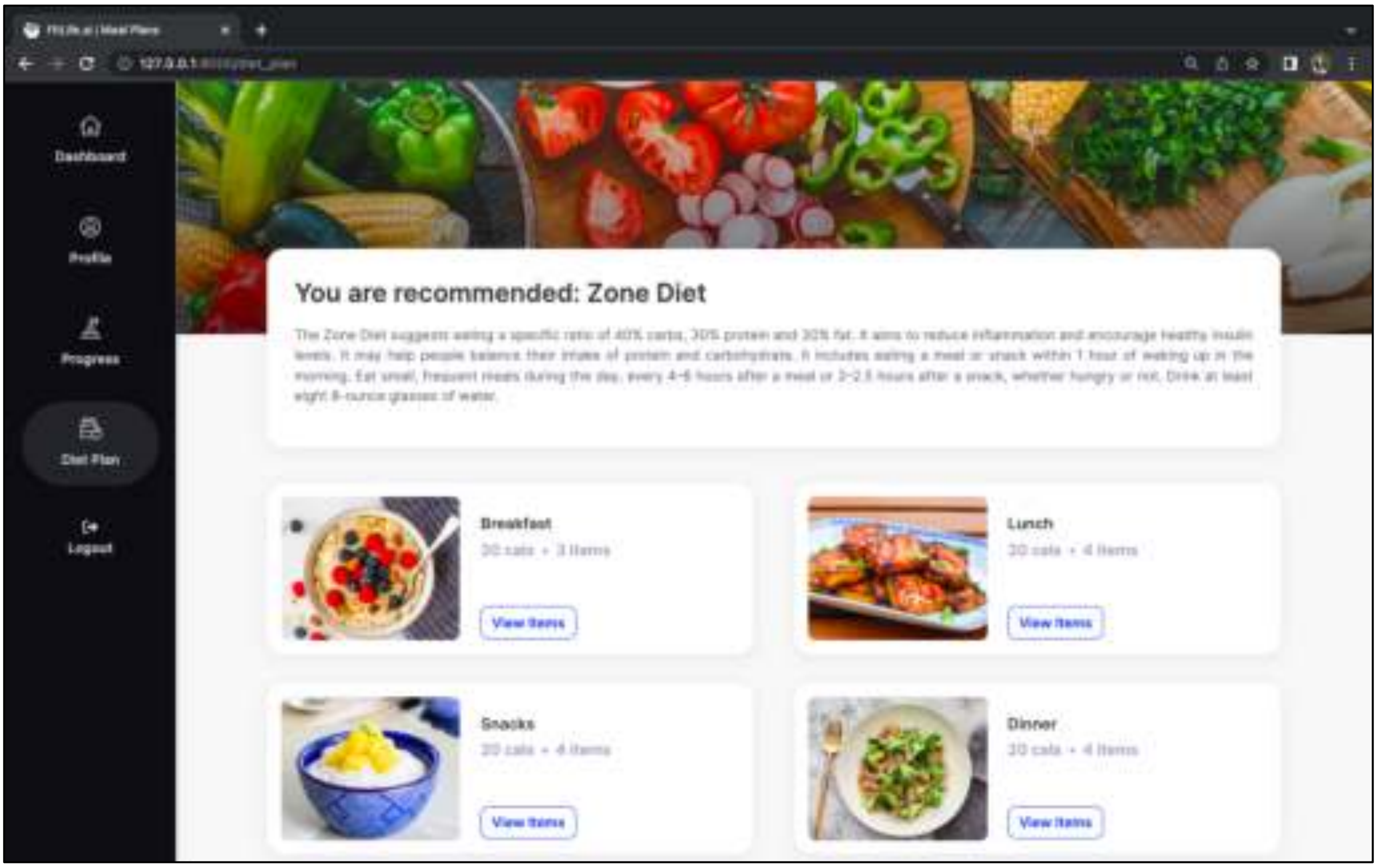


Fig 6.26 Diet Type as well as Meal Plans

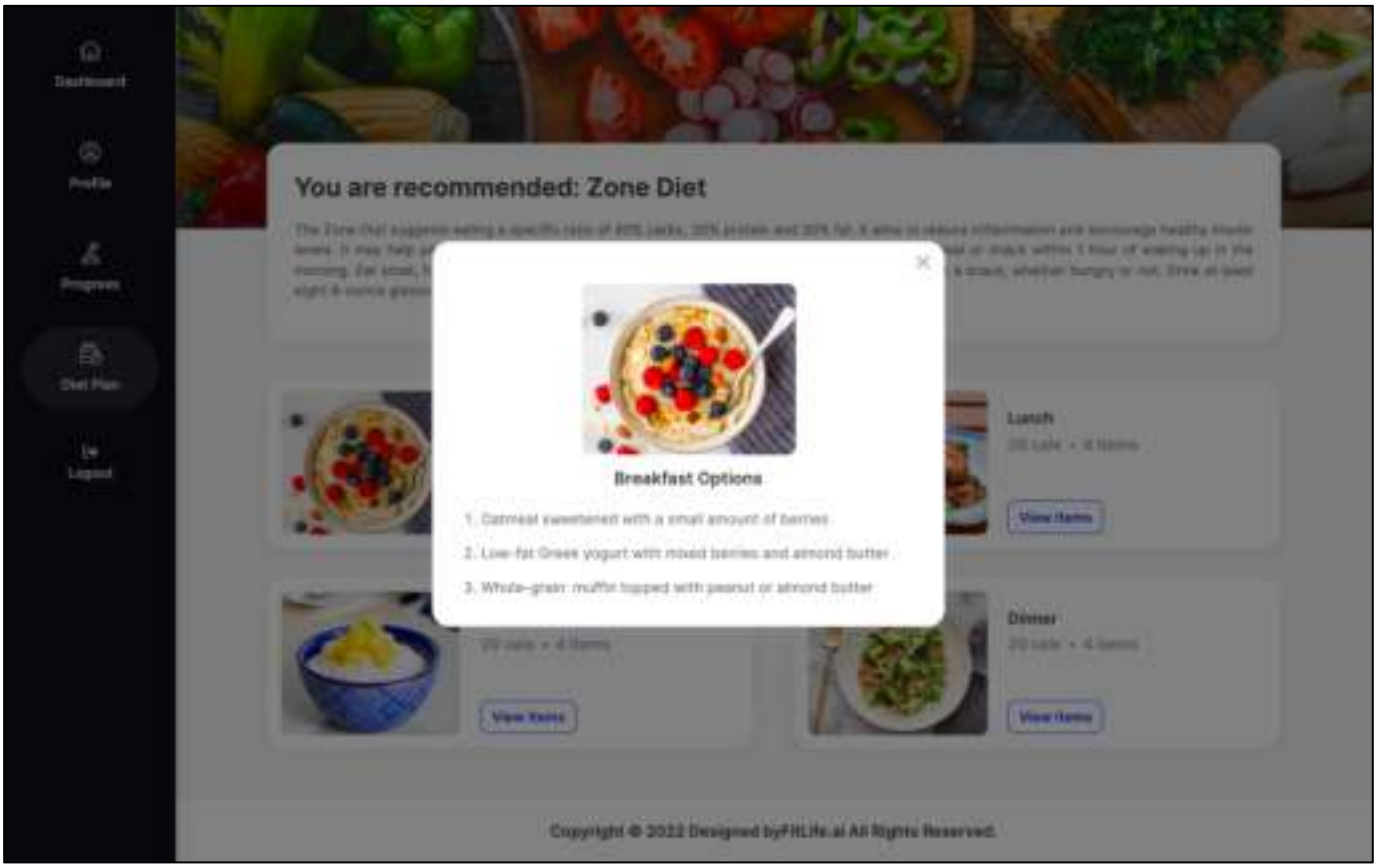


Fig 6.27 Breakfast Meal Plan

Chapter 7

Results

This chapter presents the results we obtained from training and testing various models. It explains the reason for implementing Random forest model for our diet recommendation system through the confusion matrix diagram.

We tested our data against 4 models: Random Forest, KNN, SVM and logistic regression. On observing the results, we understood that the Random Forest model had the highest training and testing accuracy of 1.0 and 0.997 respectively among all the other models and so, we went ahead with that for the recommendation of diet type to the user. Further, to evaluate how many classes were correctly predicted in this model, we plotted a Confusion matrix as shown below. We concluded that the most correctly predicted class was that of Low-carb diet. On the contrary, Depletion diet was the most incorrectly predicted class because of the lesser amount of data available [10].

Model	**Training Accuracy**	**Testing Accuracy**
Random Forest	1.0	0.997
KNN	0.843	0.864
SVM	0.955	0.945
Logistic Regression	0.85	0.86

Table 7.1: Comparison of different model's accuracies

From the table, the Random Forest classifier performed very well, giving 99% accuracy.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	429
1	1.00	1.00	1.00	713
2	1.00	1.00	1.00	511
3	0.98	1.00	0.99	148
4	1.00	0.40	0.57	5
5	0.99	1.00	0.99	210
accuracy			1.00	2016
macro avg	0.99	0.90	0.93	2016
weighted avg	1.00	1.00	1.00	2016

Fig 7.1 Classification report of Random Forest

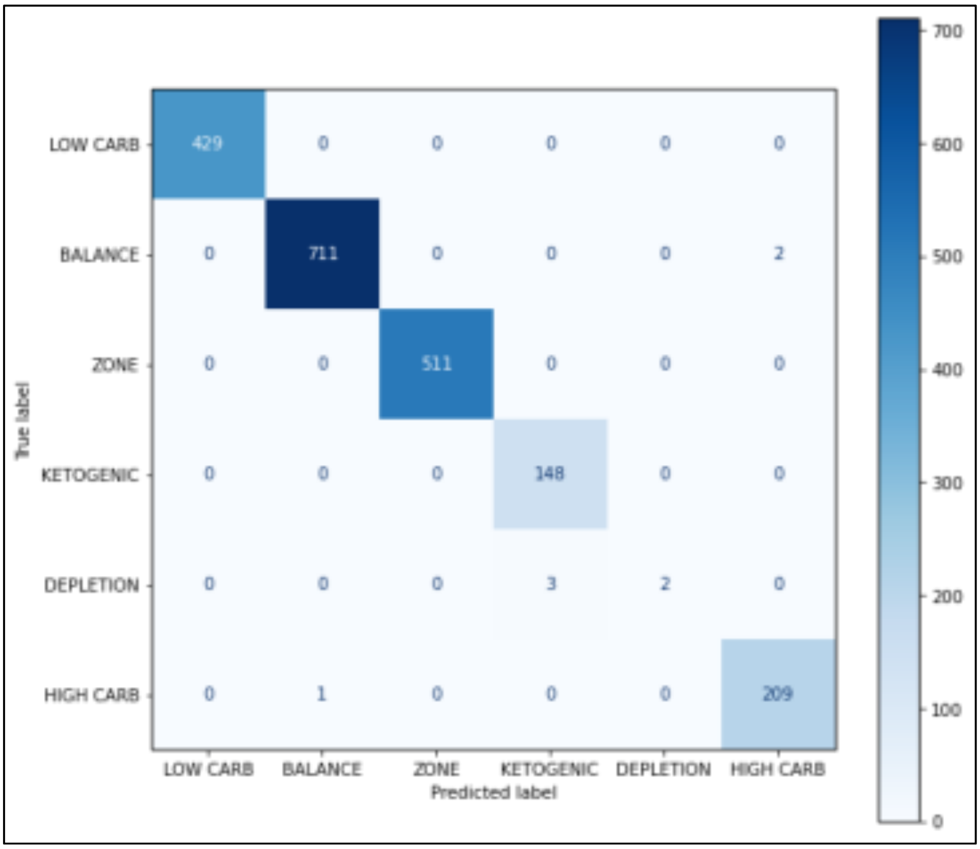


Fig 7.2 Confusion matrix

Chapter 8

Conclusion

This chapter presents the conclusions which we have drawn based on the results and discussion. We have discussed certain applications of our project.

Emerging technologies such as machine learning and artificial intelligence are influencing the growth of the IT (Information Technology) sectors. We have made use of these technologies and created a website for people who are consulted about their workouts, diet and want to lead a healthy life. Based on the design and execution of the smart fitness application, it can be inferred that the web application can provide captions while exercising, as well as a progress chart of the exercise completed and a recommendation for the next exercise to be performed. The application may save records of the user's physical fitness, which can be displayed in data or graphic form, and it can also recommend diet types and meal plans based on the user's input. In addition, the web server simulation helps the application work smoothly.

Chapter 9

Future Scope

This chapter shows how useful will this project be in the future and what other things could be added in this project to make it more viable and advanced.

This project has a lot of potential for growth. The project is now running on local storage; however, it may be moved to cloud storage providers such as AWS and AZURE for limitless storage and faster website loading. The project can be expanded to include more exercises. A daily step tracker can be added as well. The application will provide a training schedule and intensity level based on your body type and weight. For simplicity of usage, this application can be expanded into a full-fledged Android/iOS application.Along with the food items recommended for each meal, the system can be designed to develop and deliver recipes that include all of the food items recommended in the meal plan.

References

[1] Financial Express, “How COVID-19 changed fitness regime of Indians; Here’s what a survey revealed” in October 17, 2020.

[2] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, “MediaPipe: A Framework for Building Perception Pipelines”, arXiv:1906.08172, June 14, 2019.

[3] Barbara E. Ainsworth, William L. Haskell, Stephen D. Herrmann, et al., “2011 Compendium of Physical Activities: A Second Update of Codes and MET Values,” Medicine and Science in Sports and Exercise, Aug. 2011

[4] Neeraj Upadhyay, “Diet Data”, www.kaggle.com, 2021

[5] W. Mckinney, “pandas: a Foundational Python Library for Data Analysis and Statistics” in Python High Performance Science Computer, 2011

[6] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi and V. Padma, "Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, doi: 10.1109/ICSSIT48917.2020.9214160

[7] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[8] Philipp Probst, Marvin N. Wright, Anne-Laure Boulesteix, “Hyperparameters and tuning strategies for random forest”, January 28, 2019, doi: 10.1002/widm.1301

[9] Chunmei Zheng, Guomei He1, Zuojie Peng, “A Study of Web Information Extraction Technology Based on Beautiful Soup”, Journal of Computers, July 20, 2015, doi: 10.17706/jcp.10.6.381-387

[10] AYUSHMAAN WANCHOO, “Low Carb Indian Diet for Beginners – Ultimate Guide”, Crediblog

[11] Medical News Today, “The Zone diet: All you need to know”, www.medicalnewstoday.com, January 29, 2020

[12] Mohan Shrimal, Mohit Khavnekar, Shubham Thorat, Deone Jyoti, “Nutriflow: A Diet Recommendation System”, Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021), June 15, 2021

Acknowledgment

We would like to express our gratitude to our guide, Professor Ankit Khivasara as well as our principal Dr. Shubha Pandit who gave us the golden opportunity to do this wonderful project. We are immensely thankful for their constant mentoring, reviewing, and support during our implementation. His advice and support helped us to move in the right direction towards completion of the project. We would like to thank him for his time, help and efforts towards this project. This also helped us in doing a lot of research and exploring our technical knowledge.

I would also like to thank my friends and family for their constant encouragement and support throughout the project.

Thank you very much.