An-Najah National University

Faculty of Engineering & Information Technology

Presented in partial fulfillment of the requirements for

Bachelor degree in computer Engineering

Final Project

Gofit Center Mobile Application and Website

Students:

Samah Abd Al-Rahman

Ameera Shobaki

Dr.Raed Qadi

Dr.Samer Arandi

February 5, 2024

Acknowledgment

We extend our deepest gratitude to all who played a pivotal role in making the GOFIT Center implementation possible. This achievement comes as a result of the collective dedication, experience, and steadfast support of our dear supervisors, Dr. Raed Al-Qadi and Samer Al-Arandi.

Primarily, we would like to express our deep appreciation to our esteemed project supervisors whose wise guidance propelled the project towards excellence. Their guidance has been indispensable in shaping the implementation of the GOFIT Center.

We extend our sincere thanks to the distinguished faculty members in the Department of Computer Engineering for fostering an environment of academic rigor and innovation. Their commitment to academic excellence has been a cornerstone in providing the foundation needed to develop the GOFIT Center application.

We extend our deepest thanks to our families and friends, whose unwavering encouragement and understanding have been a constant source of strength throughout this journey.

In conclusion, the completion of the GOFIT Center project is a testament to the collective efforts.

sincerely,

Samah Abdel Rahman, Amira Shobaki

Abstract

In a world where fitness meets nutrition, GoFit mobile and web apps stand out as a one-stop solution, seamlessly integrating fitness and nutrition to promote overall health and well-being.

There are four categories of users in the app, each with unique capabilities. The financial, administrative and operational components of the platform are managed by the administrator. While nutritionists provide plans and tips, and encourage a holistic approach to health and fitness, fitness professionals arrange a range of workouts, share tips, and interact with users. There are many advantages that both visitors and users can benefit from.

A variety of services are available to users, such as personal profiles, participation in social communities, BMI and basal metabolic rate estimates, and free at-home exercise programs. This feature-rich software encourages dedication to fitness goals and convenient booking by making it easy to plan and pay for simple gym sessions and monthly subscriptions. For meetings held remotely. Users can easily create and edit profiles, share and interact with posts, calculate fitness data, and schedule sessions on GOFIT, a self-improvement paradise.
Administratively, the application enables administrators to efficiently manage the platform, create workout schedules, and monitor financial aspects. This comprehensive approach enables individuals to make informed decisions regarding their lifestyles, and promotes proactive steps towards improved health and fitness.

The project process is centered on a comprehensive study of requirements, ensuring that the application successfully meets user expectations. Web and mobile applications are included in the development. The mobile app uses Flutter, ensuring a smooth and sophisticated user experience. The Node.js backend increases the functionality and responsiveness of the application. One of the databases chosen was mongodb.

GoFit differentiates itself by combining fitness, nutrition, and health into one platform, although there are many health and nutrition apps available, each with their own capabilities. With the help of cutting-edge technology solutions, this strategic integration seeks to give users a seamless and comprehensive experience that addresses many issues related to well-being.

Contents

Chapter 1

Introduction

1.1 Background

cIn a world where holistic well-being is crucial, the GOFIT Center Project stands out as a leading force at the intersection of fitness, nutrition and cutting-edge technology. GOFIT rises to the challenge by seamlessly integrating fitness and nutrition. In an age where mobile apps have become indispensable, GOFIT aspires to be more than just an app – it aims to be a lifestyle companion.

Diving into the heart of GOFIT, fitness professionals play a pivotal role in creating a vibrant tapestry of exercise. From general workouts to specialized routines, these professionals share valuable fitness tips and interact with users, fostering a dynamic and interactive environment. On the nutritional front, GOFIT offers nutritionists who provide personalized advice and plans and this ensures a holistic approach to health, recognizing the integral link between fitness and nutrition.

For users navigating the world of GOFIT, the experience is nothing short of transformative. They enjoy a number of features, including creating and editing profiles that reflect their individual fitness journeys. The social community aspect allows users to interact with others, share ideas, and support each other's progress. The platform facilitates health awareness through BMI and BMR calculations, enabling users to gain valuable insights into its fitness metrics, and access to free home workouts enhances the user experience, providing convenient and effective workout options as needed. everyone . Seamless remote session booking adds another layer of flexibility as well as monthly booking and remote payment, allowing users to schedule sessions with the fitness professionals of their choice at a time that suits them. GOFIT is more than just a fitness app; It is a comprehensive ecosystem where users can sign up for monthly training with a coach of their choice.

GOFIT Center represents a lifestyle choice. It enables users to seamlessly book sessions, access free home workouts, participate in a social community, and even book appointments with fitness and nutrition professionals.

The choice of technologies - Flutter for the interface, Node.js for the backend, and MongoDB for data storage - reflects a strategic decision aimed at delivering a seamless, efficient and robust application.

Using Flutter, a cross-platform framework, is pivotal in ensuring an attractive and consistent UI across Android and iOS platforms. This choice not only simplifies the development process, but also enhances the user experience, as it allows for the creation of a unified and attractive interface that meets the preferences of users on both mobile platforms.

Node.js, which serves as the backbone of GOFIT's backend, provides efficient data management. Its event-driven architecture enables real-time data flow, facilitating instant updates and dynamic interactions within the app. This proves essential in scenarios where timely information is crucial, such as managing user accounts, monitoring subscriptions, and handling financial aspects. Node.js excels at handling concurrent connections, ensuring GOFIT runs smoothly even during periods of high user activity.

MongoDB, chosen as the database system, plays a pivotal role in data security and accessibility. NoSQL's document-oriented structure allows for flexible and scalable storage, accommodating diverse information associated with user profiles, fitness routines, and nutritional plans. The ease of use and scalability of MongoDB is particularly beneficial for GOFIT, as it efficiently handles the diverse and evolving data requirements inherent in a fitness and nutrition application.

In summary, the strategic choice of Flutter, Node.js, and MongoDB collectively contribute to GOFIT's success by providing a visually appealing interface, real-time data management, and secure, scalable storage capabilities. This combination ensures that GOFIT not only meets but exceeds the expectations of users looking for a comprehensive solution to their fitness and nutrition journey.

The development of the GOFIT Center application faced several constraints that shaped the course of the project. There are many features that could be included, but there is a limited time to complete the project, so we have collected the most important features. Compiling the

features requires a thoughtful selection process to meet the diverse needs of users. Learning new technologies such as Dart, Flutter, Node.js, Firebase Database, and MongoDB within a limited time frame has led to Increasing complexity, previous coursework in web programming, object-oriented programming, and databases provided the foundational knowledge and building blocks. Independent learning of technologies such as Flutter, Dart, Node.js, MongoDB, and Firebase has enriched our capabilities, contributing to a smoother development process.

Adopting the Agile model methodology served as the guiding framework to guide the GOFIT project through its various development phases, promoting a systematic and iterative approach. The Agile model's focus on flexibility and collaboration has enabled seamless progress through requirements analysis, planning, design, development and testing, ensuring methodological rigor and continuous improvement.

The project commenced with an extensive requirements analysis to identify essential features, laying the foundation for understanding user needs and defining project goals. The planning phase involved creating a detailed timeline, facilitating efficient task management and ongoing reviews with supervisors. Prioritizing tasks during the design phase, daily meetings assessed progress and addressed challenges, ensuring effective solutions , Development marked the coding initiation for both front-end and back-end components. The final testing phase encompassed backend functionality verification, frontend connectivity, and rigorous testing of the machine learning model for accuracy.

the Agile model methodology ensured a methodologically rigorous and continuously improving development process for GOFIT, allowing for adaptability and systematic progress from conception to implementation.

Chapter 2

Literature Review

Maintaining a healthy lifestyle requires physical activity, although obesity is mostly caused by a number of obstacles as well as bad eating habits. Through fitness applications, the advent of mobile technology, particularly smartphones, offers public health campaigns a creative means of addressing these issues.

A pilot study investigation evaluated the effectiveness of fitness apps and compared their energy consumption with traditional workouts including brisk walking, WiiFit Plus, and resistance cycling. The results indicate the effectiveness of fitness apps compared to organized fitness activities, which may represent a paradigm shift for people who find it difficult to be physically active [1] .

In another study , mobile technology is emerging as a promising alternative, as smartphones provide a means of delivering interventions that address sedentary behavior and promote activity. Another study, which focused on first-year university students, found that mobile fitness app exercises were as effective as traditional gym exercises in enhancing health-related fitness performance. Tangible. In the battle against physical inactivity, the combination of gym workouts and smartphone fitness apps becomes essential [2] .

There has been a lot of emphasis on the significance of fitness app technology. These fitness tracker applications, personal training programs, and other resources are essential for encouraging drive and dedication to a healthy lifestyle. Studies have generally demonstrated the significance of these applications in promoting fitness clubs and physical activity among individuals, as well as in enhancing nutrition.

These apps offer a variety of inspiring workout regimens, which helps users stay motivated and get past any barriers to engaging in physical activity. These apps also give users the ability

to monitor their own development and accomplishments, which boosts their sense of dedication and motivates them to reach their sporting objectives. Through the ability to participate in online sports groups [3] , applications also play a significant social function by fostering better user communication and encouragement.

When it comes to improving nutrition , Nutrition apps provide useful advice on how to encourage proper nutrition and give customized meal planning for individuals. As well as helping people realize the significance of a good diet in reaching their fitness and health objectives, this helps them live a more balanced and healthy lifestyle.

Mobile and online applications offer a full solution to encourage physical activity, track progress, overcome hurdles, engage in a virtual sports community, and promote healthy nutrition, enabling people to commit to a healthy lifestyle.

Chapter 3

Methodology

In this section, we provide a detailed explanation of the methodology as following :

3.1 Backend Development

In order to improve organization and scalability, we implemented a specific design during the Node.js application development phase. The main parts of our system, controller, services, modules, routers and middleware are separated in different directories within the backend source code.

The initial phase of the application is represented by the standard entry point, “index.js”. It is responsible for setting up the server and importing the “app.js” file to start the program.

We have set up the basic parameters of our Express application in ’app.js’. This requires defining the routing system using a regular router, configuring the middleware, and importing the necessary modules.

One important part is the router, which arranges the endpoints and directs incoming requests to the correct control paths. Modular routers help divide responsibilities, making maintenance of the code base easier.

In order to manage the business logic of our application, controller files are necessary. Agents provide input that they receive, process, and collaborate to create relevant responses. Controllers act as intermediaries, coordinating with services and models to perform specific tasks in response to client requests.

Services - which include business logic - specialized management, and reusable functionality

such as database interfaces, data processing, and communication with third-party APIs. Controllers often use services to perform the real work required to complete a particular process.

This flow [index.js -¿ app.js -¿ Router -¿ Controller -¿ Service] is a common practice in Node.js application development, this separation helps build a well-structured, scalable and maintainable code base.

3.2 Frontend Development

We decided to use the flexible Flutter framework to create the web and mobile application. We were able to save significant development time by avoiding the need to duplicate code thanks to this wise decision.

The choice to use Flutter was made after a thorough analysis of all its features, focusing on its success. Flutter is a powerful cross-platform framework that makes it easy to share code across multiple target systems. Flutter simplifies the development process by allowing both UI and code to be shared, unlike existing application frameworks. Being an open source framework, Flutter also includes a large number of training materials and abundant documentation, which helps developers solve problems more quickly and easily. Flutter is, in any case, the perfect choice for creating cross-platform mobile applications that work quickly and efficiently.

Our top pick among programming languages was Dart. Dart was created by Google specifically for web and mobile app developers, and can be used to create apps for both iOS and Android. The language is designed to run flawlessly on contemporary web servers, mobile devices and browsers.

3.3 Database

A number of convincing arguments led us to select MongoDB as our project's database platform. Firstly, the NoSQL document structure of MongoDB offers remarkable processing flexibility. MongoDB, in contrast to conventional relational databases, does not require a pre-defined schema, making it simple to adapt and modify the data model as the project moves forward—a feature that is particularly helpful for dynamic and developing data structures.

Scalability is another key component of MongoDB. MongoDB scales horizontally by spreading data among several servers or clusters. As the project develops and user needs grow, this feature makes sure that massive data volumes and heavy traffic loads are managed efficiently,

enabling the database to meet growing requirements. When it comes to performance, MongoDB is outstanding for data retrieval tasks. Data querying is made quick and easy by its document-oriented paradigm and effective indexing methods. Enhancements to query performance and response times include support for complicated queries with numerous fields and criteria, as well as integrated caching and memory management tools.

Dealing with alterations to the data structure over time is made easier by MongoDB's scalable design. This flexibility makes it possible to quickly edit fields in a document without affecting already-existing data, which is a useful feature in situations involving rapid development when requirements may change often. Furthermore, Node.js, the project's preferred backend technology, connects with MongoDB with ease. With the official MongoDB driver for Node.js, you may use a wide range of features and methods to streamline CRUD tasks, run intricate queries, and take advantage of sophisticated MongoDB capabilities in Node.js applications.

MongoDB is a great choice for our project's database technology because of its flexibility, scalability, performance, and capacity to scale schemas. Its interaction with Node.js further enhances this pick.

3.4 Main Features

Welcome Page :

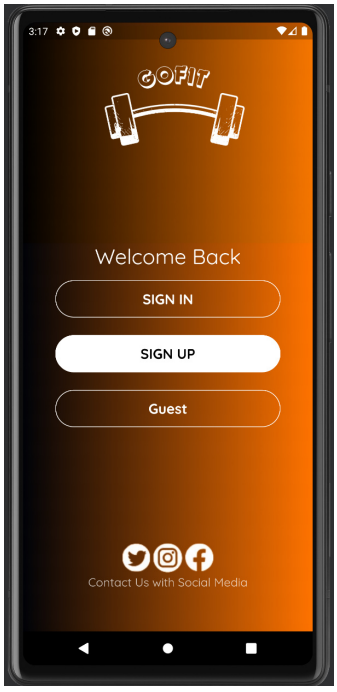

Figure 3.1: Welcome Page .

The welcome page is the first page that the user deals with when visiting our application. On this page, the user is welcomed and options such as “Sign In,” “Sign Up,” and “Guest Login” are provided.

OnBoarding :

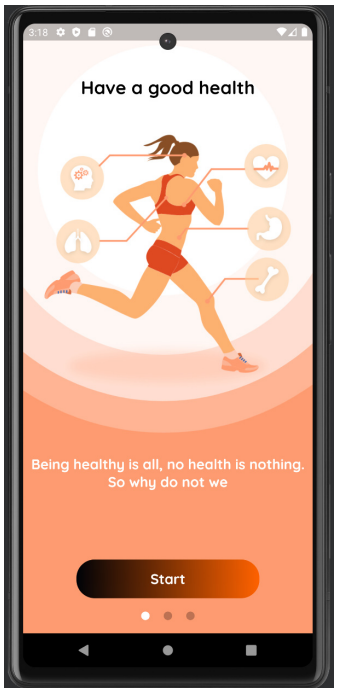

Figure 3.2: OnBoarding 1.

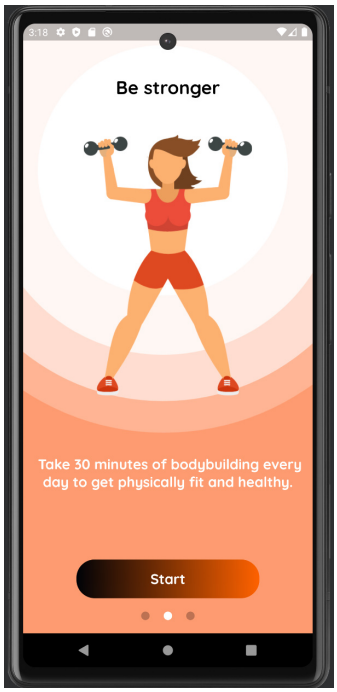

Figure 3.3: OnBoarding 2.

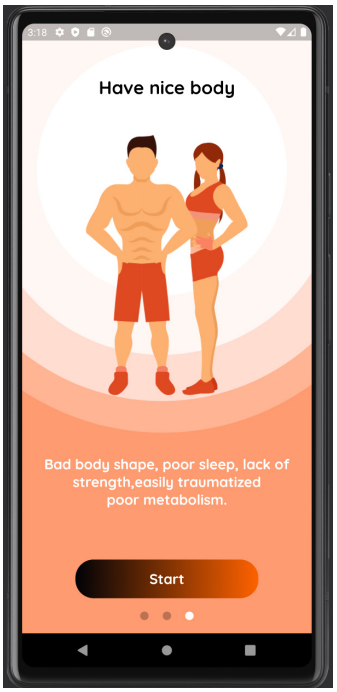

Figure 3.4: OnBoarding 3.

Our pages start with internal pages that guide users through the process of using it in an automated manner and give them a general idea of the topic as shown above.

Guest Page :

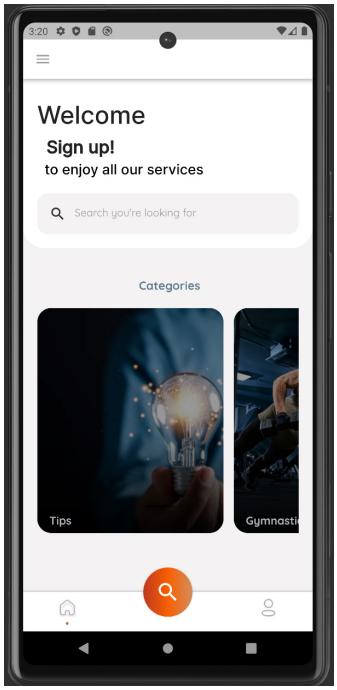

Figure 3.5: Guest Page 1.

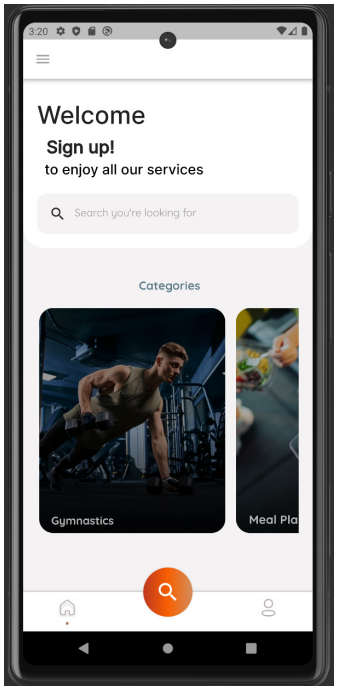

Figure 3.6: Guest Page 2.

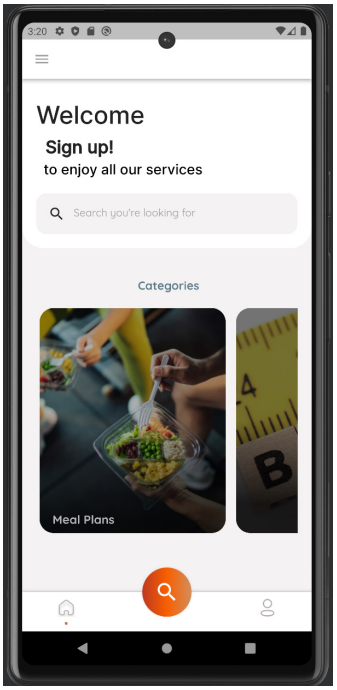

Figure 3.7: Guest Page 3.

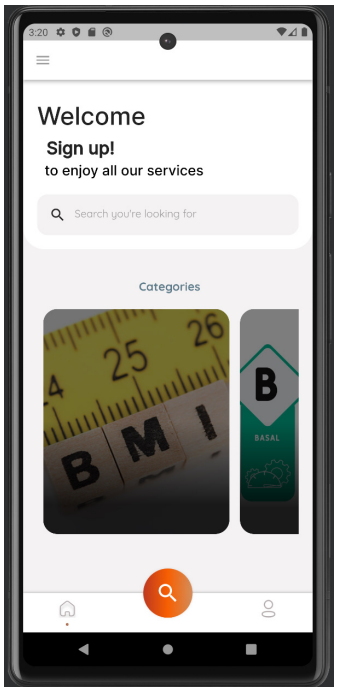

Figure 3.8: Guest Page 4.

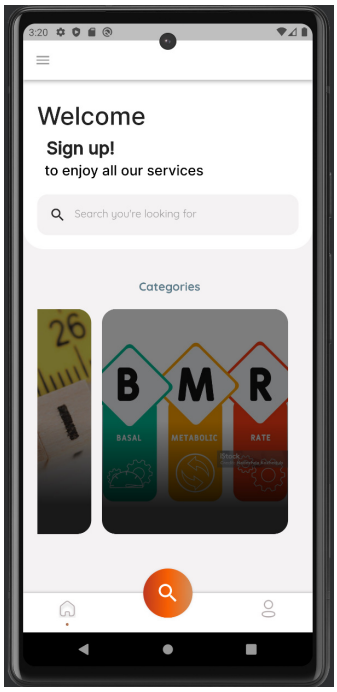

Figure 3.9: Guest Page 5.

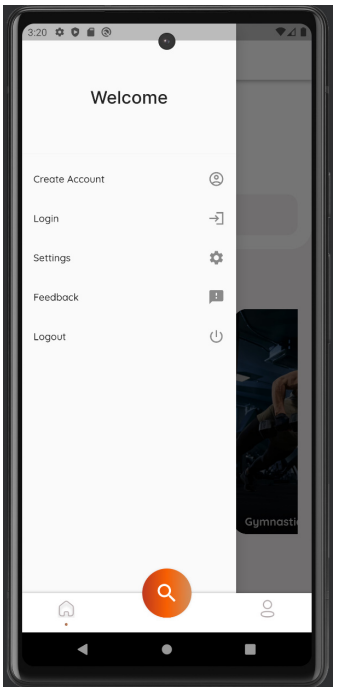

Figure 3.10: Guest Page 6.

Guest who want to explore the app or site without creating an account. It allows the guest to access specific content or perform some activities without having to log in. This option is offered for the purpose of demonstrating the benefits of the app and encouraging users to register. The guest user can search for whatever he wants

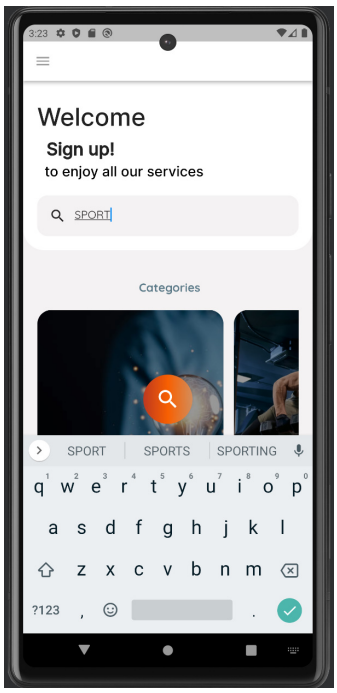

Figure 3.11: Guest Page 7.

Exercises on the guest page :

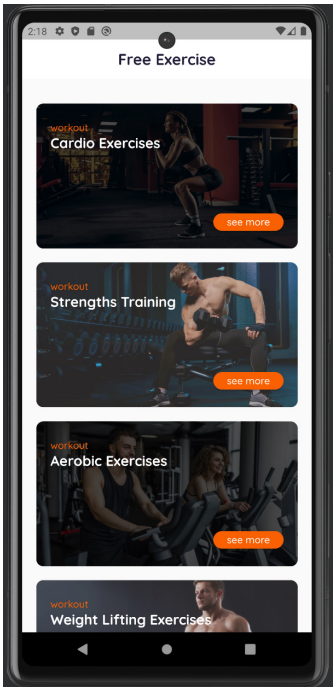

Figure 3.12: Guest Exercises 1.

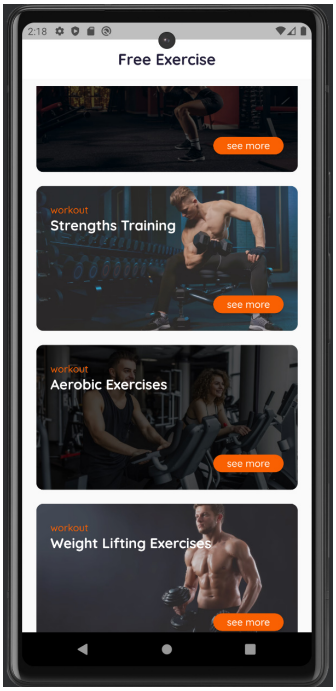

Figure 3.13: Guest Exercises 2.

coaches add general types of exercises Cardio, Strengths, Aerobic and Weight Lifting Exercises , Provided free to guests .

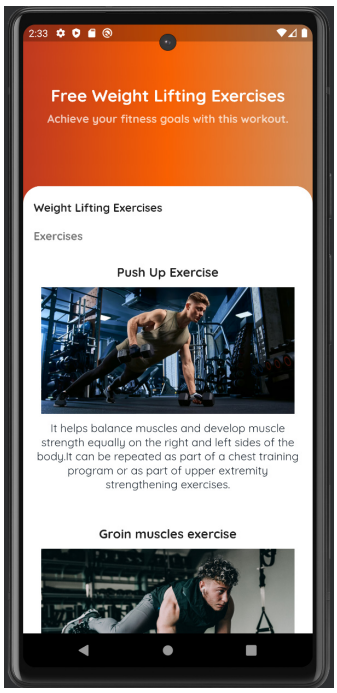

Figure 3.14: Guest Exercises 3.

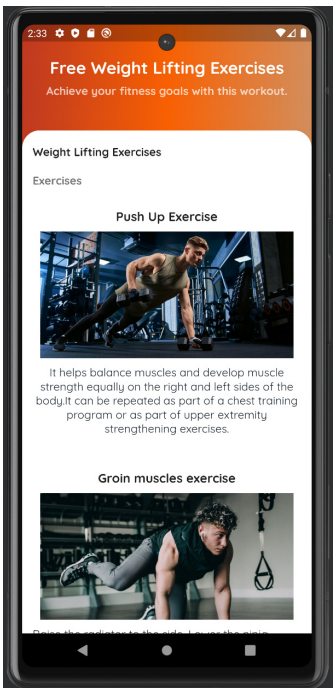

Figure 3.15: Guest Exercises 4.

Meals on the guest page :

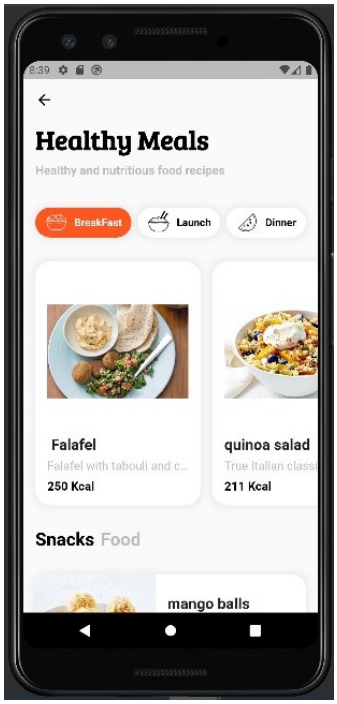

Figure 3.16: Guest Meals 1.

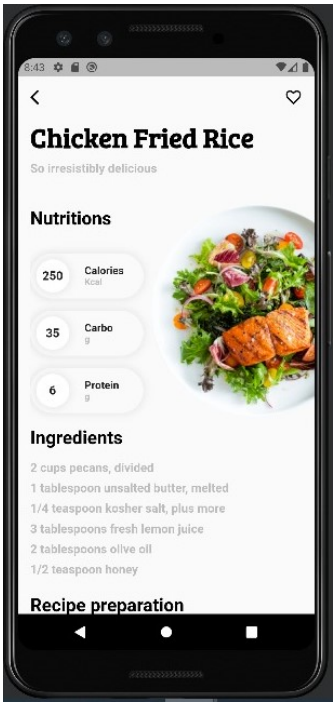

Figure 3.17: Guest Meals 2.

The nutritionist adds meals with complete information about them, such as their ingredients,

how they work, the expected time to complete them, and the number of calories, carbohydrates, and protein in each meal. Here, the user chooses the meal: breakfast, lunch, or dinner. Meals will be suggested to him with full details and information.

There are more features that the guest can enjoy, such as: Body Mass Index (BMI) and Basal Metabolic Rate (BMR), which we will discuss later .

Guest Web Pages :

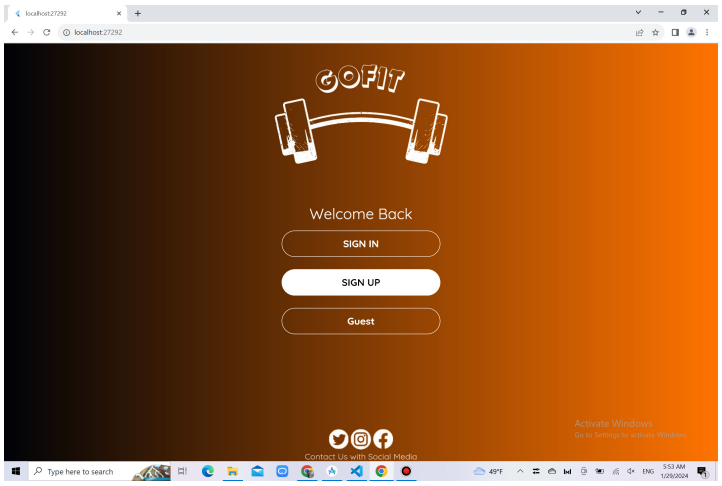

Figure 3.18: Welcome Page.

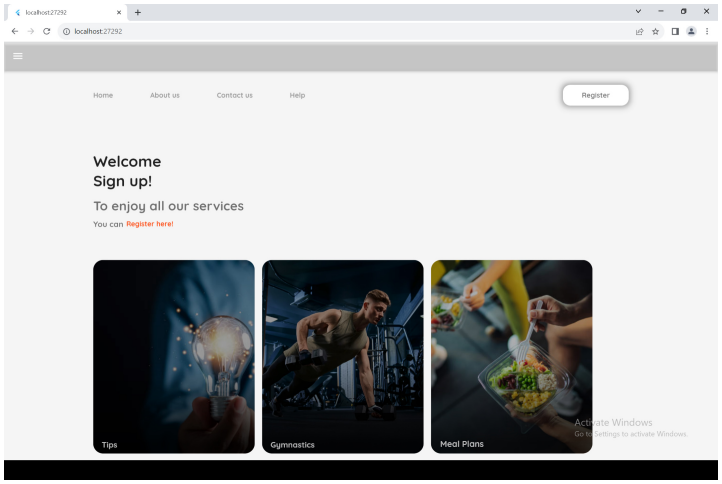

Figure 3.19: Guest Home Page.

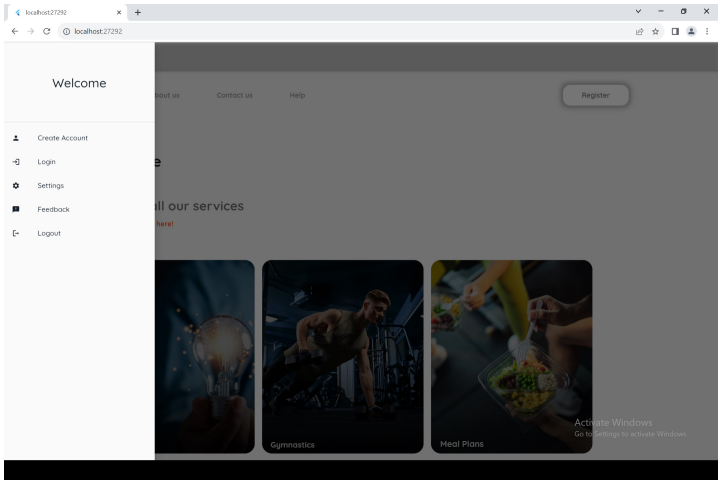

Figure 3.20: Guest Menu.

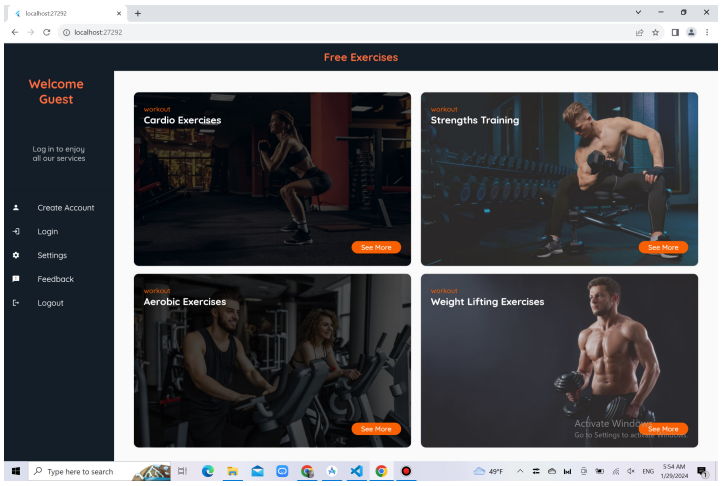

Figure 3.21: Guest exercises 1.

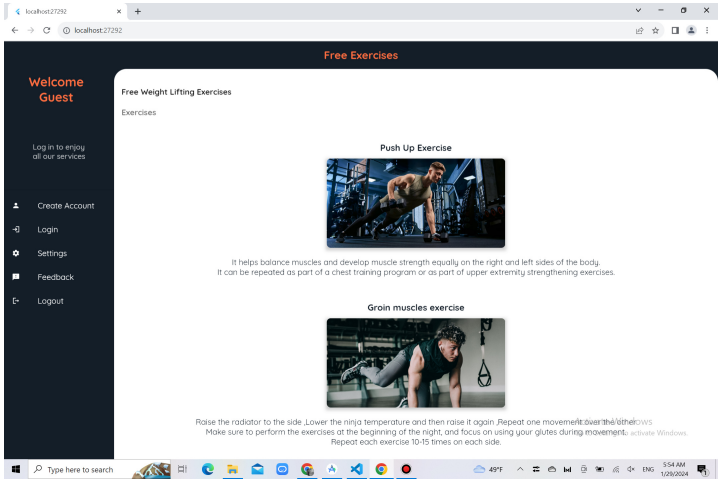

Figure 3.22: Guest exercises 2.

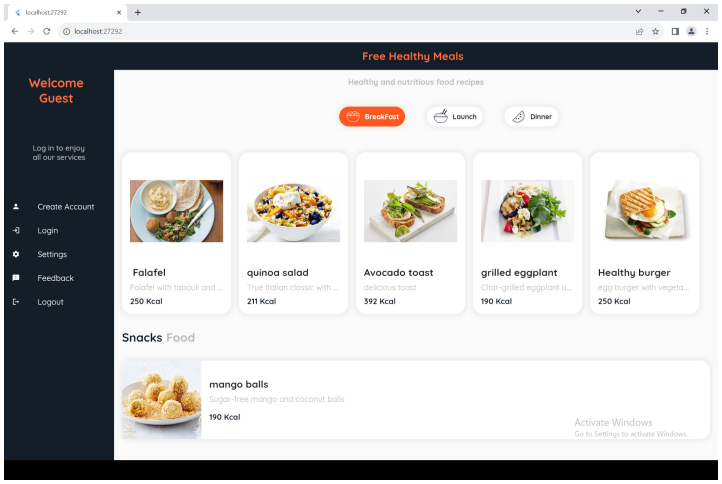

Figure 3.23: Guest Meals 1.

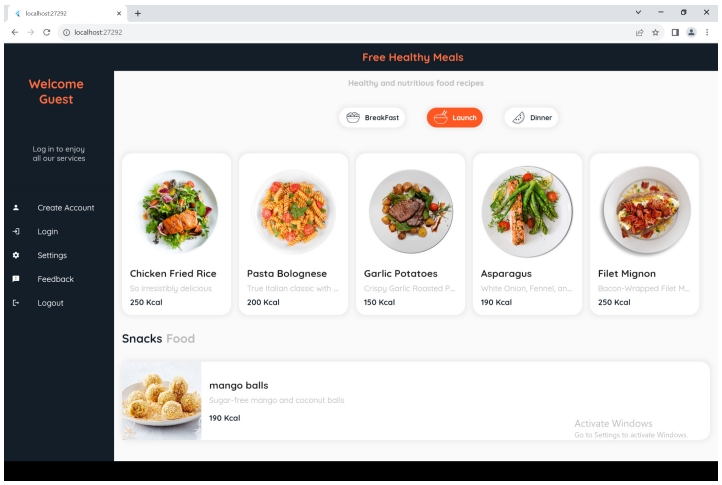

Figure 3.24: Guest Meals 2.

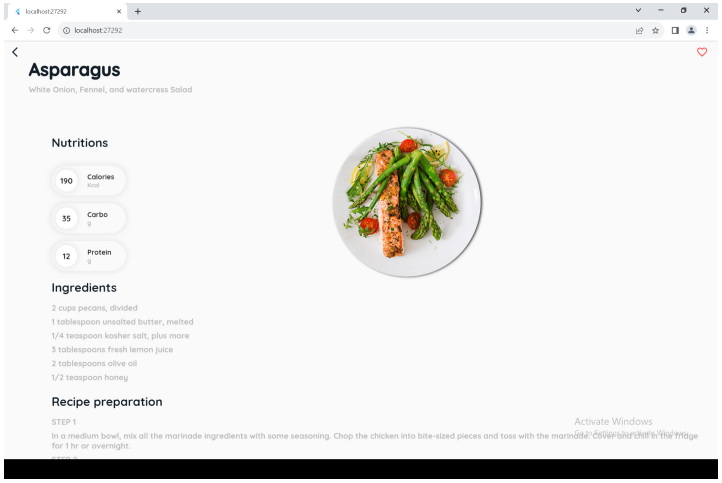

Figure 3.25: Guest Meals 3.

Signup and Login :

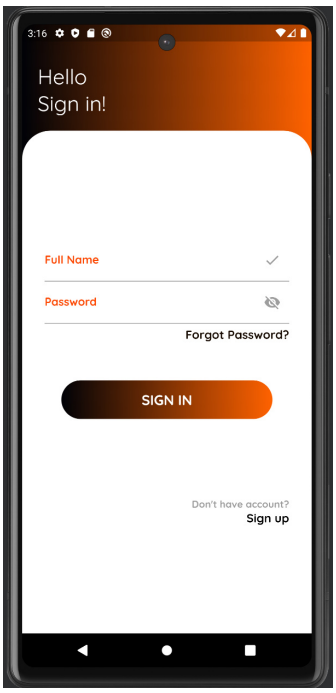

Figure 3.26: Login.

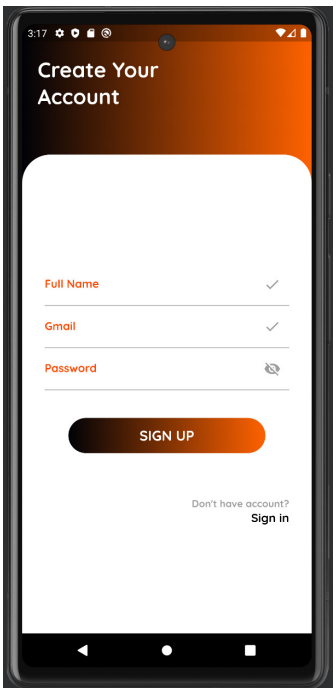

Figure 3.27: Signup.

Users need to enter the correct password and email address to log in. During the login attempt, the software verifies that the user has filled in all required fields and that the information is accurate. The user will be able to access their account and it will be restored if the information provided is accurate. The system blocks access to the user and displays a warning message alerting him of his unsuccessful attempt to log in. If the user provides incorrect information or leaves any mandatory field blank, the system redirects him to his personal pages. Users are advised to fix any issues with their login credentials via notification.

People who do not currently have an account can choose to create a new account. In order to access more capabilities within the system, users must first register an account through the registration process. They are sent to a sign-up page if they don't already have one. The MongoDb database securely stores all user-provided data. Email addresses, hashed passwords, and usernames are among the details contained in this data.

One security mechanism that protects user credentials from illegal access and potential data breaches is password hashing. A hash is a one-way encryption algorithm that creates a fixed-length string of characters, or value hash, from input data (the user's password in this case). The basic requirement for a hash function is that it must be mathematically impossible to invert. This process makes obtaining the original password from the hash very difficult.

Password hashing improves security by ensuring that hackers cannot immediately access users' plain text passwords, even if the database is stolen. Hashed values are the only thing hackers will have access to in the event of a data breach. Reverse calculation .

The same method and salt are used to hash the password entered when a user attempts to log in as when they first registered or updated their password. Next, the just generated hash is compared with the hash value saved in the database. If the hash matches, the password is considered authorized access because the entry is accurate.

The bcrypt algorithm was used, which is one of the commonly used hashing algorithms. This algorithm is designed to be computationally intensive, making it more difficult for attackers to perform brute force or dictionary attacks.

Password hashing is a crucial step in protecting user credentials. It ensures that, even in the event of a breach, the amount of sensitive data exposed is kept to a minimum, adding another degree of security to user accounts. By adhering to these procedures, the system maintains user privacy and data integrity while ensuring a simple and secure user authentication process.

User Mobile Pages :

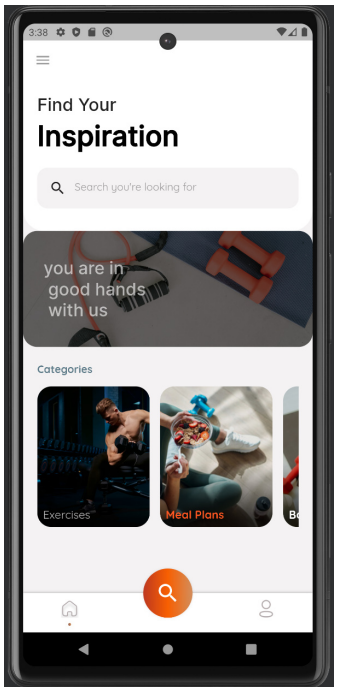

Figure 3.28: User home page 1.

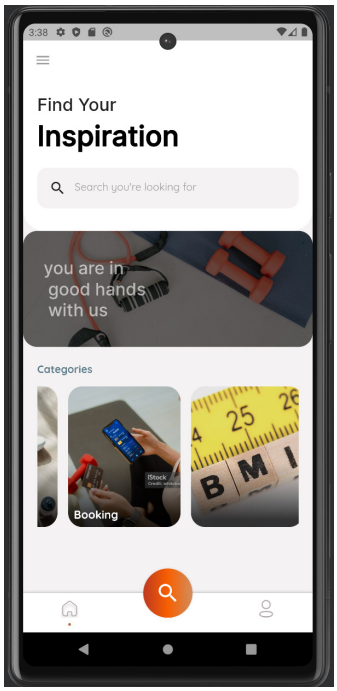

Figure 3.29: User home page 2.

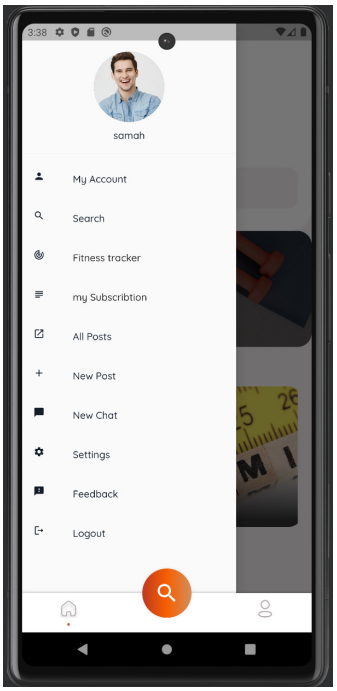

Figure 3.30: User Menu .

After logging in correctly, the user goes to the home page. If he does not have an account, he can create his own account by adding his personal information: photo, name, education, date of birth, about and bio.

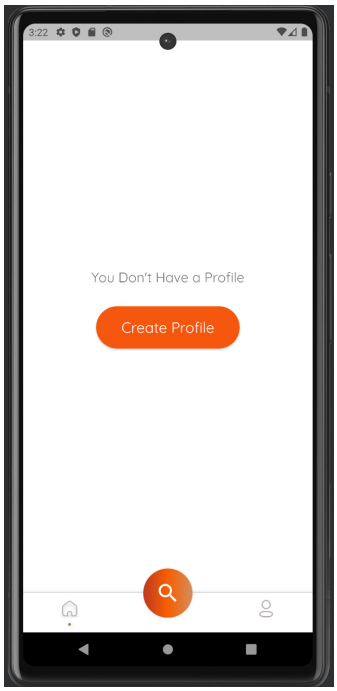

Figure 3.31: User Account 1.

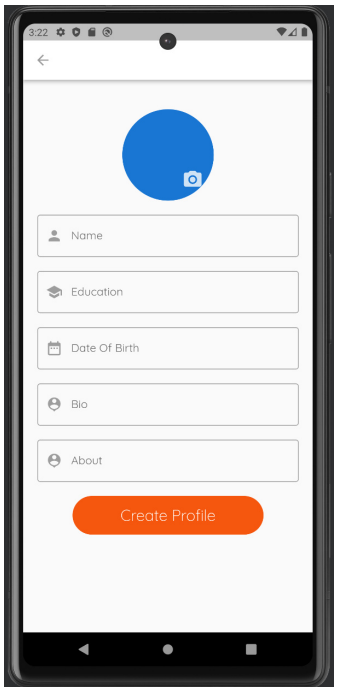

Figure 3.32: User Account 2.

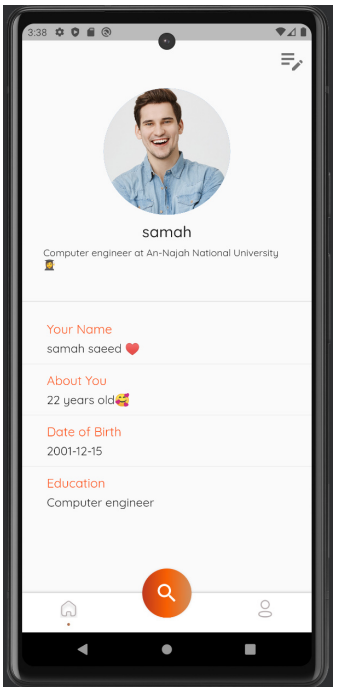

Figure 3.33: User Account 3.

The user can modify his personal account information .

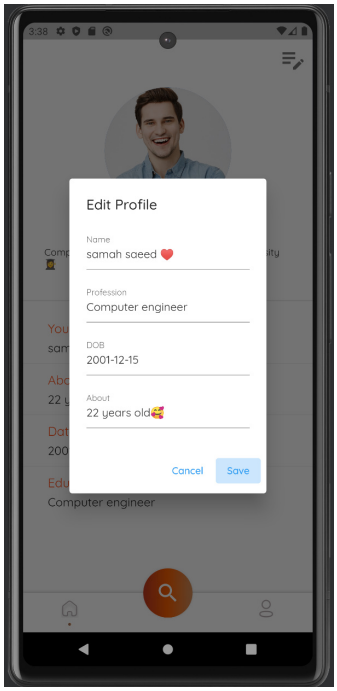

Figure 3.34: Edit Account .

User Exercises :

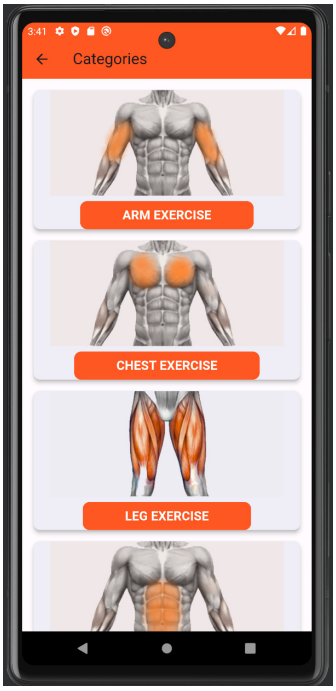

Figure 3.35: User Exercises 1.

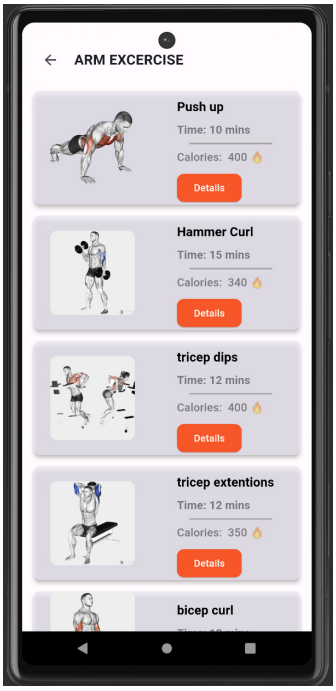

Figure 3.36: User Exercises 2.

The trainer adds more specialized exercises for each muscle, such as the thigh muscle, chest muscle, shoulder muscle, and others, by adding animated images that show how to perform each exercise and also an explanation explaining the exercise, its benefit, the estimated duration of its work, and the approximate number of calories that are burned during the exercise. The user can get it for free and benefit from it and apply it at home. There is ease of use and other features that help the user in performing exercises, such as a timer. After completing the exercise and pressing "Done," a notification appears to the user, and data is saved about the calories completed for the day, and then the user can see his daily calorie achievement and track it continuously.

Daily tracking of calorie intake per week. Showing each day how many calories have been burned, this graph helps the user stay motivated and committed

User Daily Meals Planner :

Here we use external API ,Here we'll explain how to use spoonacular API , API (Application Programming Interface) is an interface that allows two different applications to interact and talk to each other, used in this context to obtain information about meals using the spoonacular API

The API key is used to authenticate and allow access to spoonacular services. It shows the unique identity of the application. The API key must be kept confidential to ensure secure

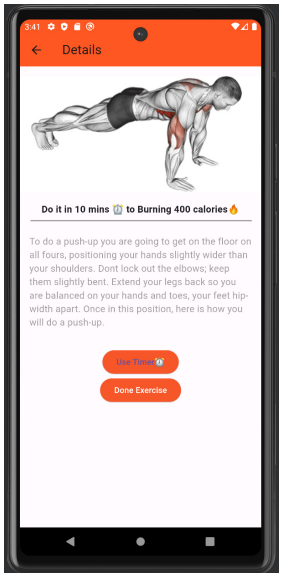

Figure 3.37: User Exercises 3.

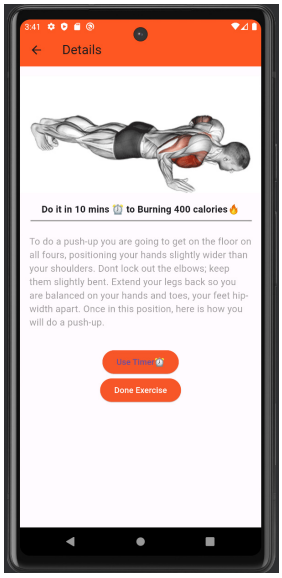

Figure 3.38: User Exercises 4.

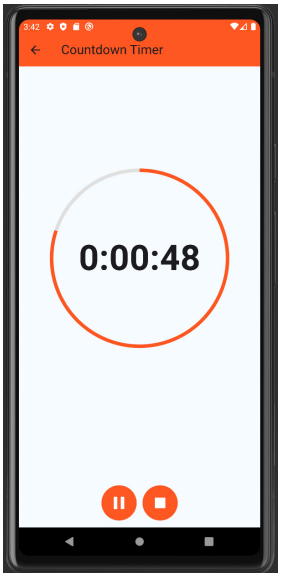

Figure 3.39: User Exercises 5.

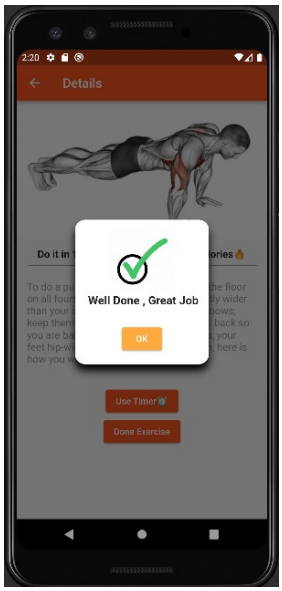

Figure 3.40: User Exercises 6.

access and prevent unauthorized use. The content can be updated regularly through frequent inquiries to ensure the most up-to-date information.

The app uses HTTP GET requests to interact with the Spoonacular API, fetching meal and ingredient details. Parameters like target calories and diet type are included. JSON responses from the API are analyzed, converted into local models, and displayed in the user interface. Users can search for meals, view details, and interact seamlessly for an efficient experience.

The user calculates his calorie needs through BMR in our application, then enters it here and determines the type of diet he follows. Then our application suggests three meals for the day

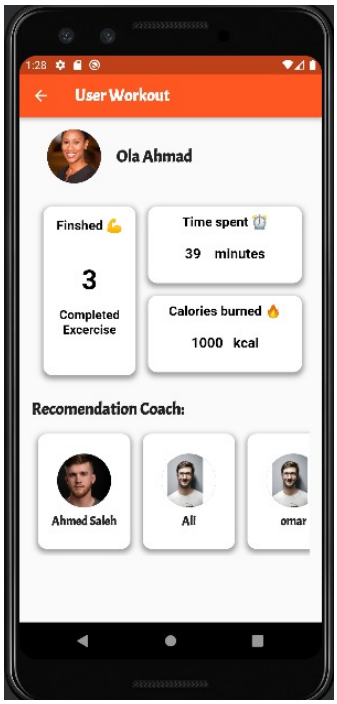

Figure 3.41: User Exercises 6.

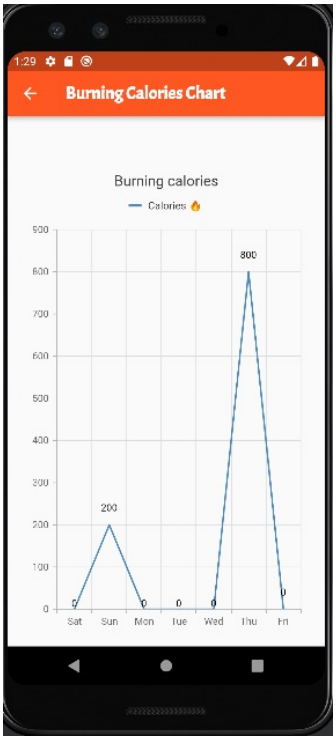

Figure 3.42: User Exercises 7.

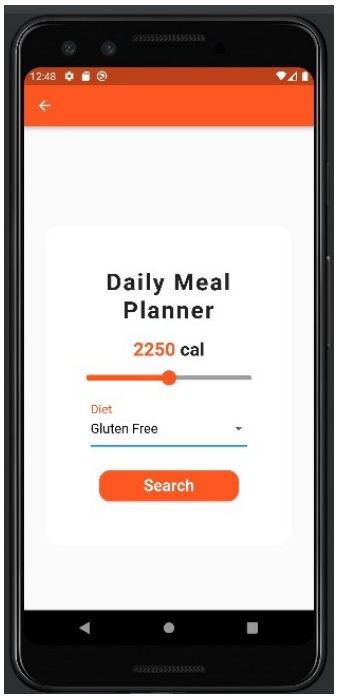

Figure 3.43: User Meals 1.

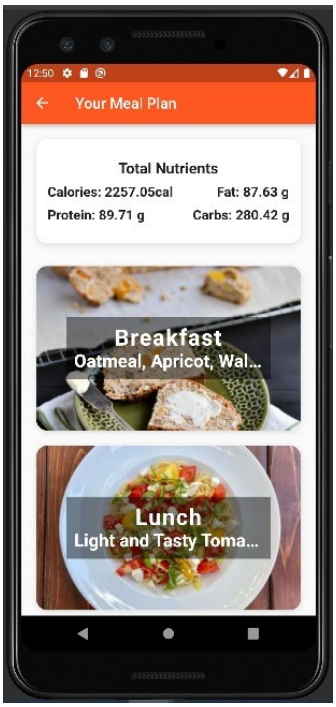

Figure 3.44: User Meals 2.

(breakfast, lunch, and dinner) with calculated calories and other information such as the amount of protein and carbohydrates in the meals. The user can re-enter his data to suggest more daily systems for him to choose from among them the system he prefers.

Each meal contains all the important details that the user needs to know, its ingredients and

Figure 3.45: User Meals 3.

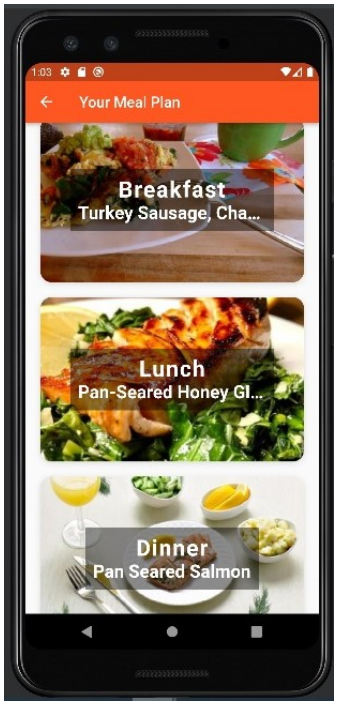

Figure 3.46: User Meals 4.

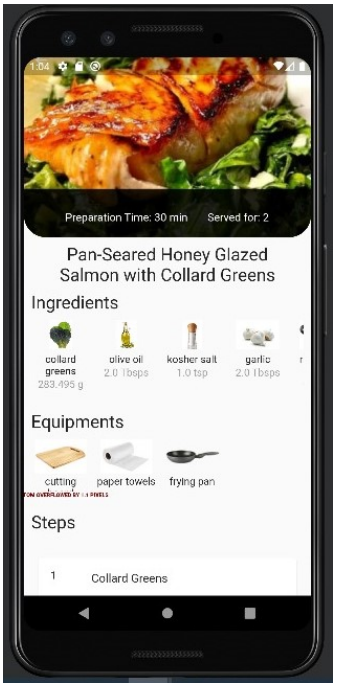

Figure 3.47: User Meals 5.

how to make it step by step.

BMI :

Body mass index (BMI) is a measure used to evaluate a person's relative weight based on their height. BMI is calculated by dividing a person's weight (in kilograms) by their height (in metres) squared. Mathematically, it is expressed in the following formula:

$$BMI = \frac{\text{Weight (kg)}}{\text{Height (m)}^2}$$

Where a person's weight is in kilograms and his height is in metres.

- Less than 18.5: Weight loss
- 18.5 to 24.9: normal weight
- 25 to 29.9: Slight weight gain (overweight)
- 30 to 34.9: Obesity grade 1
- 35 to 39.9: Obesity grade 2
- 40 and above: Obesity degree 3 (morbid obesity)

BMR :

Basal metabolic rate (BMR) ,The quantity of energy (calories) the body must burn during total rest to maintain fundamental physiological functions is known as basal metabolic rate, or BMR. BMR is computed using variables including height, weight, gender, and age.

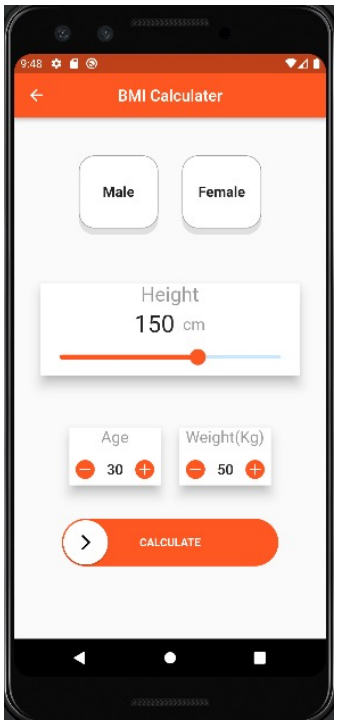

Figure 3.48: BMI .

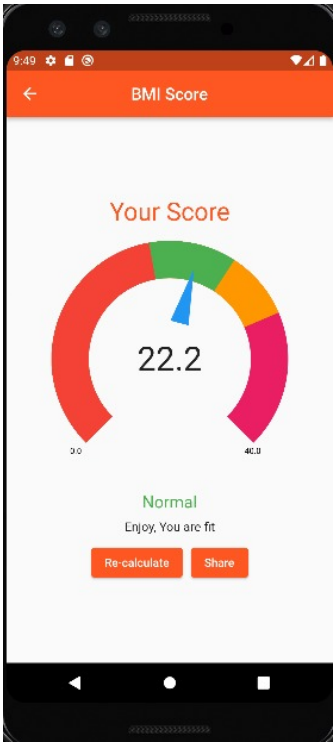

Figure 3.49: BMI Result.

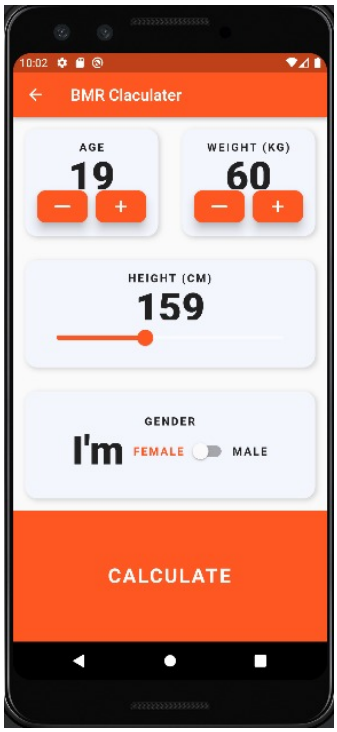

Figure 3.50: BMR .

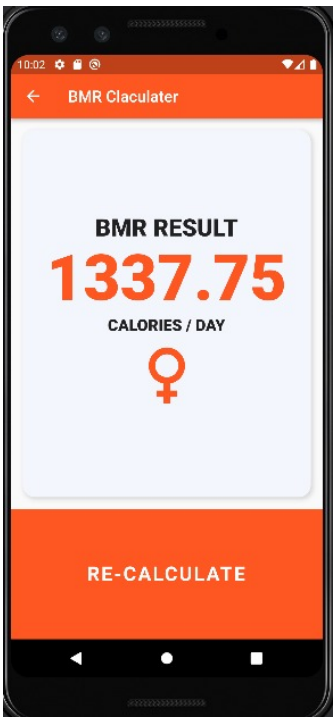

Figure 3.51: BMR Result.

- For men :

$$BMR_{\text{male}} = 10 \times \text{weight (kg)} + 6.25 \times \text{height (cm)} - 5 \times \text{age (years)} + 5$$

- For women :

$$BMR_{\text{female}} = 10 \times \text{weight (kg)} + 6.25 \times \text{height (cm)} - 5 \times \text{age (years)} - 161$$

where weight in kilograms , height in centimeters and age in years.
The number of calories the body needs is estimated using these formulae. In order to identify the right amounts of food and exercise to meet health or weight loss goals, BMR can be used to calculate basic daily calorie demands.

Search : Users can search for any person he wants or a nutritionist, or a coach, and open the personal page, communicate with him by chat in our app, and see his personal posts.

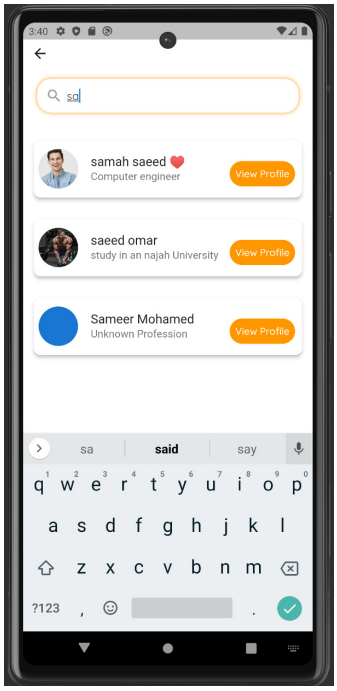

Figure 3.52: Search.

Posts : Users can publish their interests and share their opinions and health achievements, and other users can interact with them by liking or commenting.

Users can add a post by adding an image, title, and description, and he can review the post as shown.

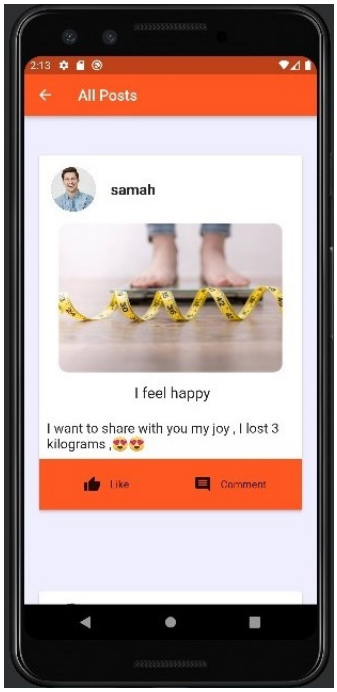

Figure 3.53: Posts 1.

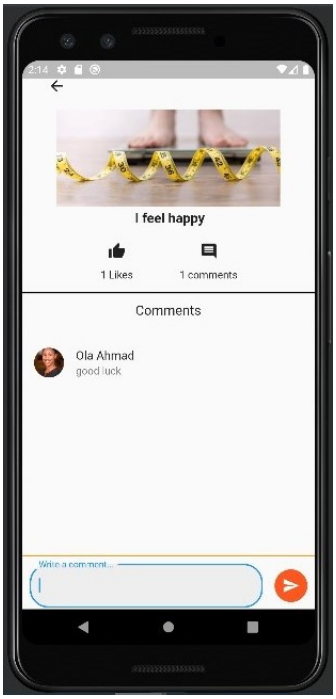

Figure 3.54: Posts 2.

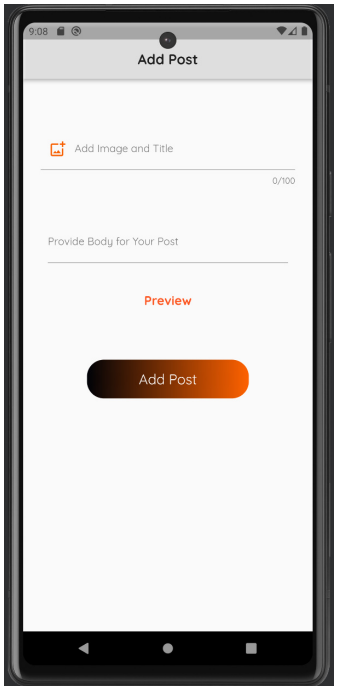

Figure 3.55: Add Post 1.

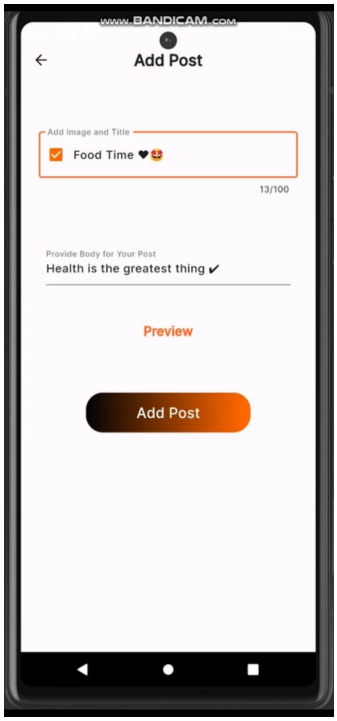

Figure 3.56: Add Post 2.

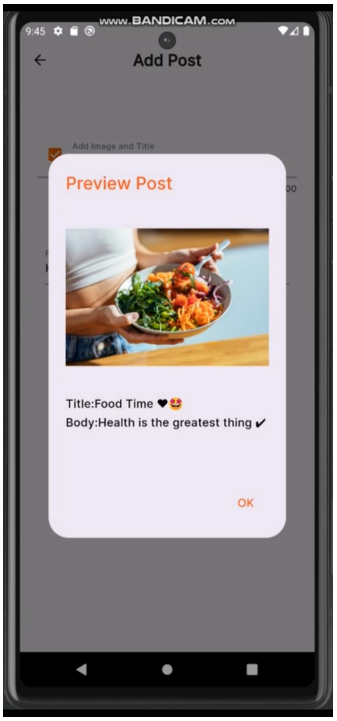

Figure 3.57: Add Post 3.

Booking Features :

Figure 3.58: Booking Type

newline There are two options for the user to book: he can book a monthly reservation or a daily reservation (booking for one class only).

First, we will talk about the user's monthly reservation feature. Here the user can choose the type of sport he wants to train in while at the center, such as weight loss, body building, general fitness, and curved body "These types are added and modified by the coach" , As shown in the figure above.

After that, the user By choosing the coach he wants to train (the AI Recommendations feature helps him choose a coach because it suggests coaches to him according to his interest in any type of sport, as each coach is interested in a specific type of sport, which means that body building trainers, for example, are different from yoga trainers), as shown below ..

After the user chooses the coach, he will be presented with a list of times when this trainer is available for booking, as shown below .
User chooses the payment method if he wants to pay in cash or by Visa to confirm the reservation, and after completing the reservation and payment, the user receives a notification on his phone informing him that the reservation and payment have been made and informs him of the period available for reservation and the end time of the reservation, as shown in Figure .

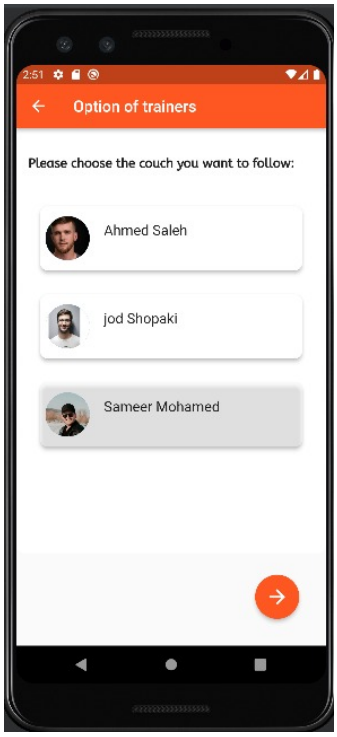

Figure 3.59: Choosing Coach

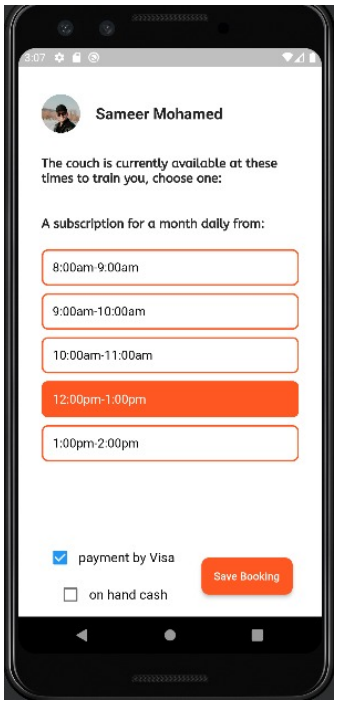

Figure 3.60: Available times .

For payment shown , we used Stripe payment integration by creating an account on the Stripe website, then using the publishable key and secret key from my Stripe dashboard. The publishable key is used on the client-side, while the secret key is kept secure and used on the server.

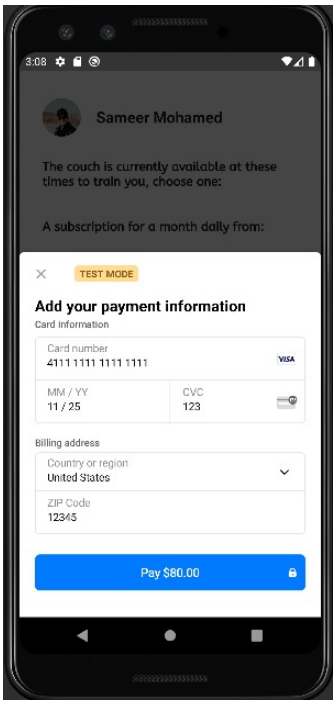

Figure 3.61: Online Payment.

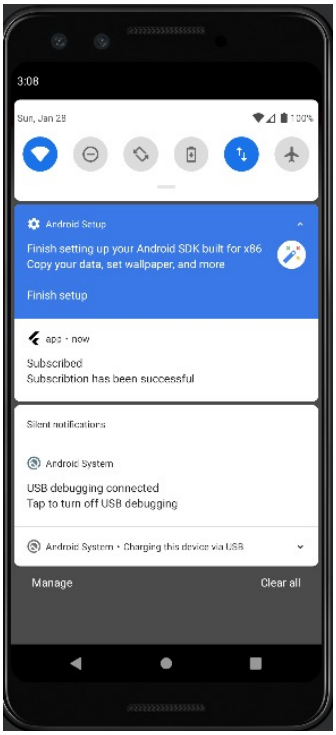

Figure 3.62: Notification.

With Stripe, we test payments without completing actual transactions thanks to its test mode functionality. To prevent actual processing of money while in development, we use test mode.

The user's calendar shows the available class every day, its time, and the name of the trainer. After attending the class, the user clicks on the box and a check mark will appear so the trainer can track the user's attendance, as shown.

User has two options : monthly or daily booking In daily booking, the coaches add classes (one-time only), users will see details about the booking:class name(yoga, Zumba, etc.), name of the trainer, date of the class, time, its price and the number of participants in this quota. After completing the reservation, the user will receive a notification on his phone

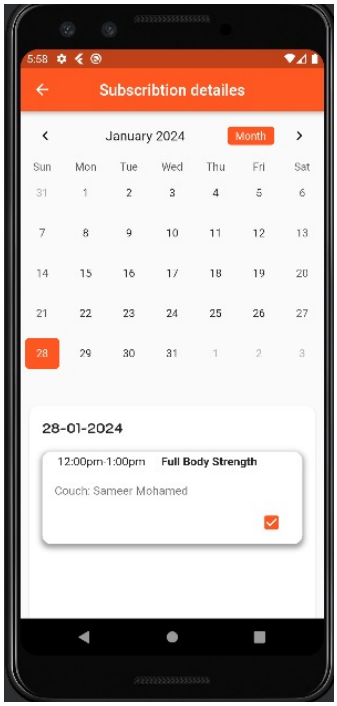

Figure 3.63: User's Calendar .

Figure 3.64: Book a class 1.

Figure 3.65: Book a class 2.

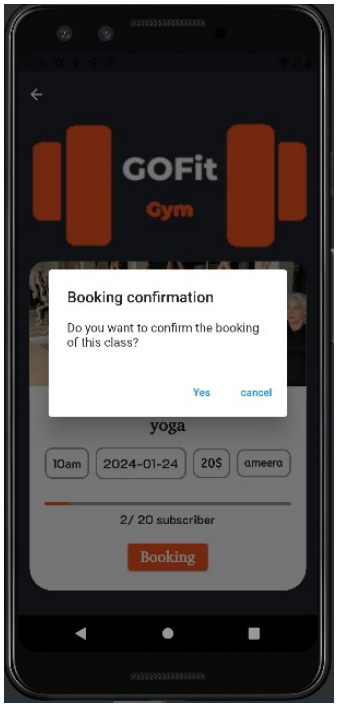

Figure 3.66: Book a class 1.

Figure 3.67: Book a class 2.

User Web Pages :

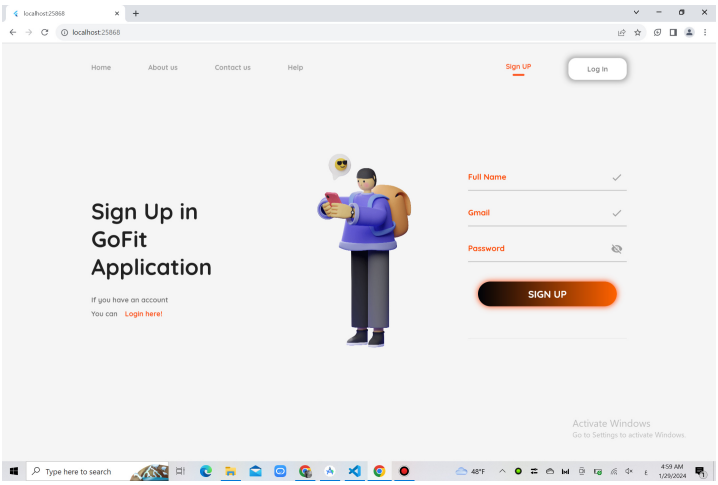

Figure 3.68: Sign Up.

.

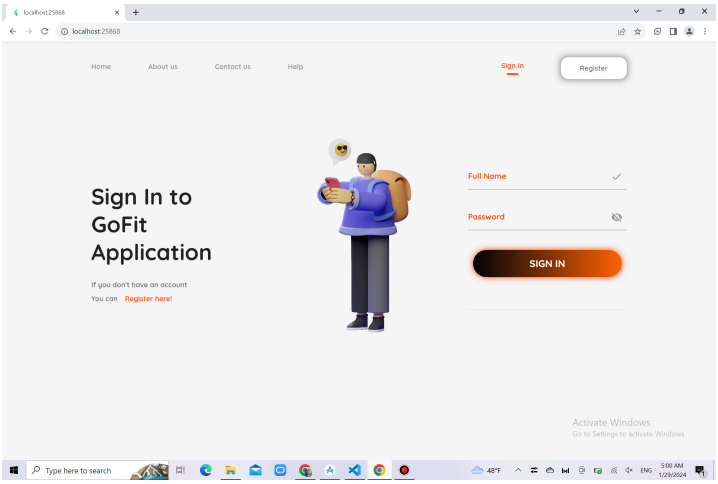

Figure 3.69: Log In.

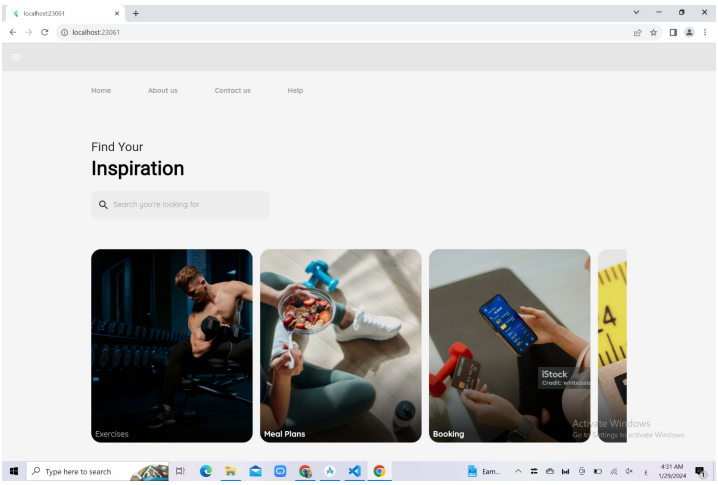

Figure 3.70: Home Page.

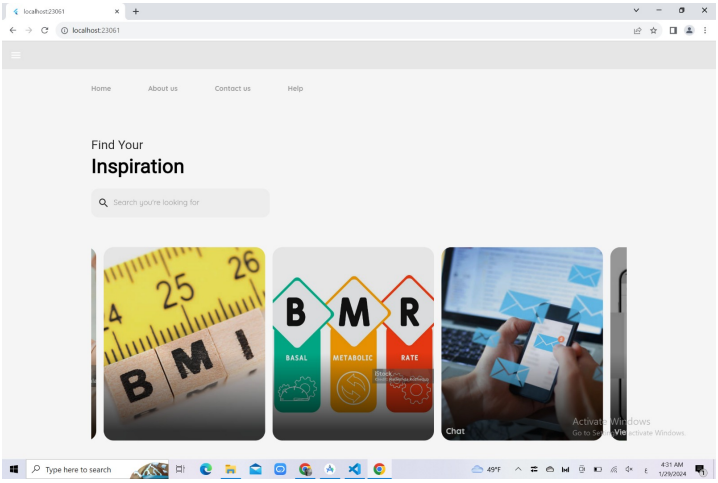

Figure 3.71: Home Page.

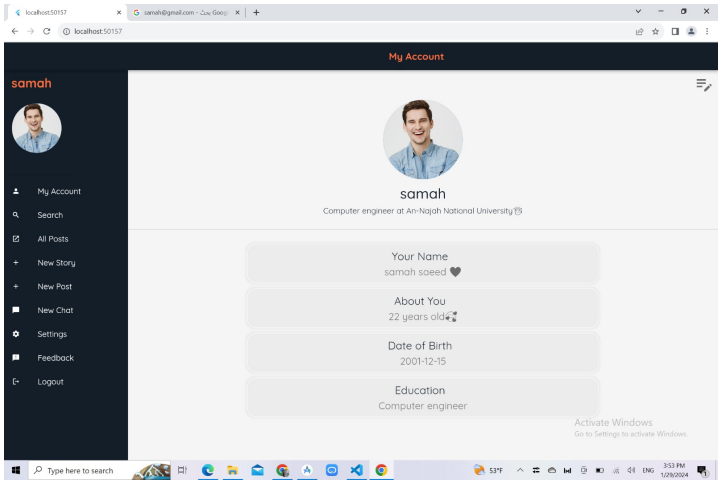

Figure 3.72: User's Account.

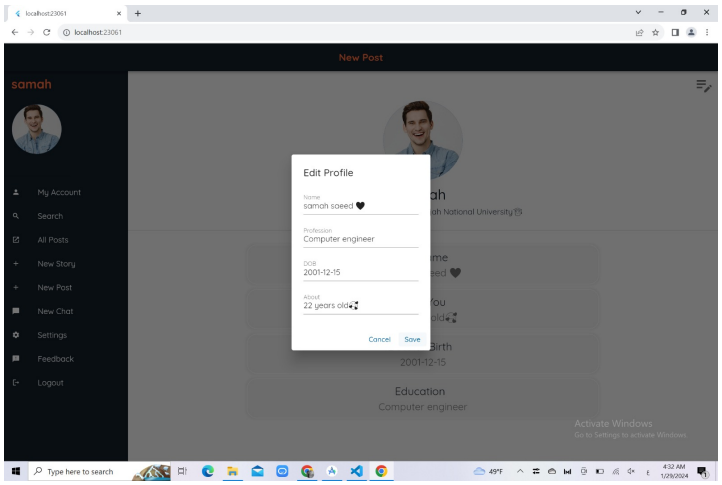

Figure 3.73: Edit Account.

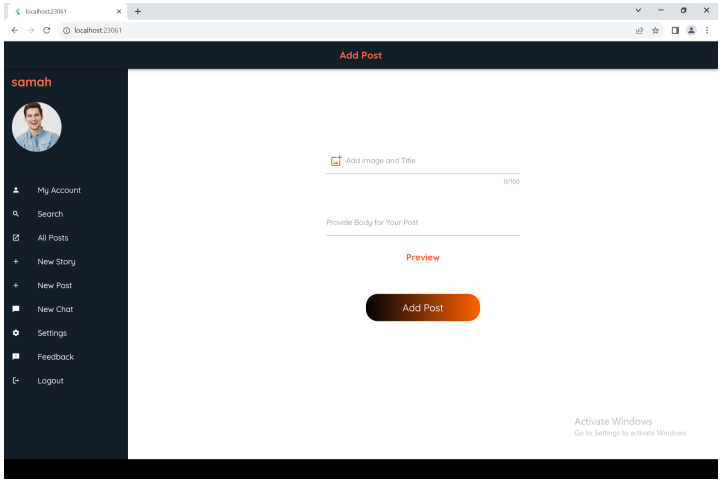

Figure 3.74: Add Post.

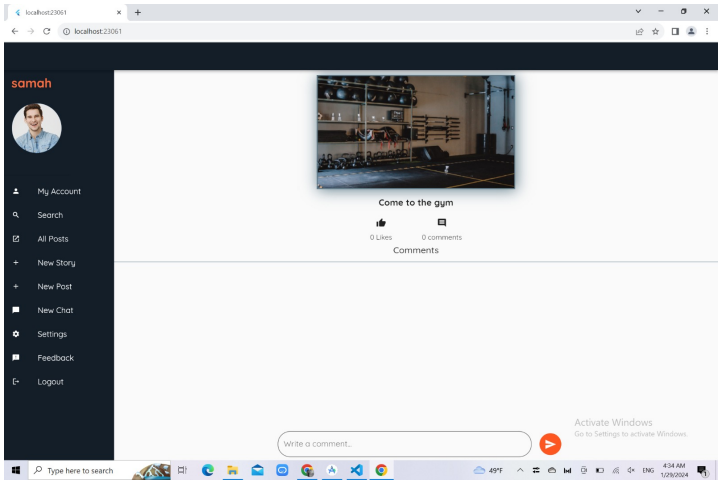

Figure 3.75: Posts.

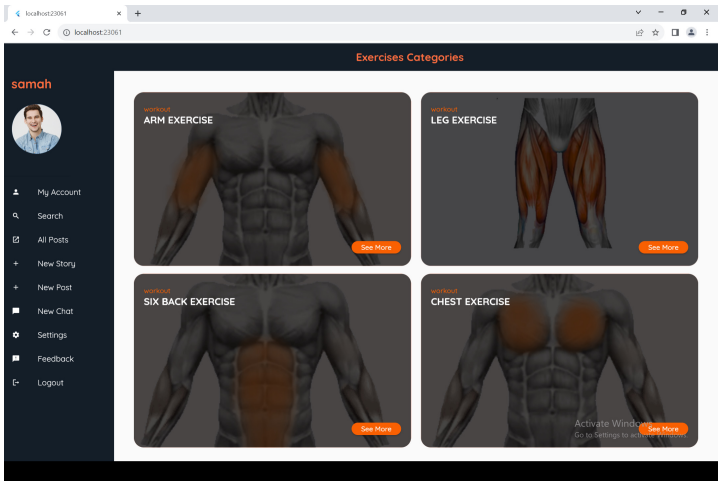

Figure 3.76: User exercises 1.

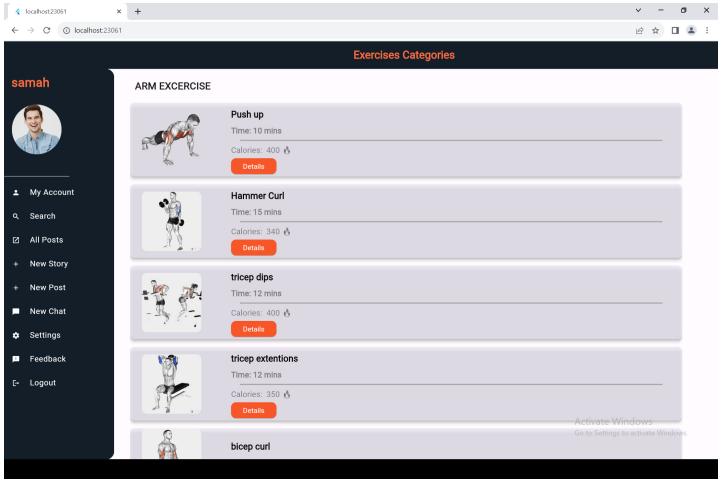

Figure 3.77: User exercises 2.

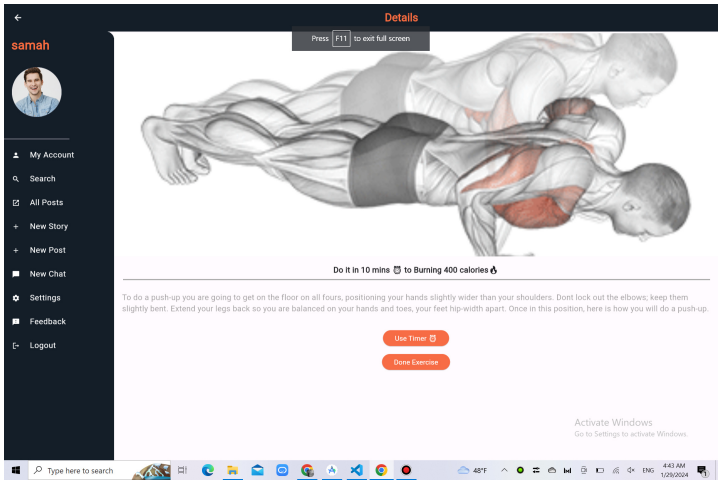

Figure 3.78: User exercises 3.

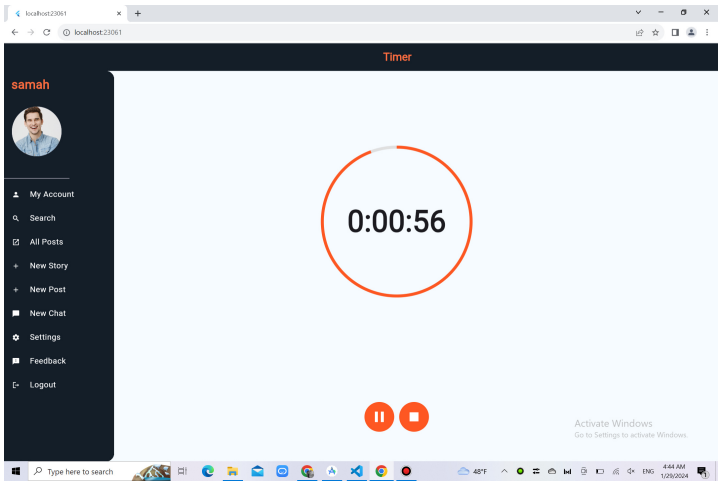

Figure 3.79: Timer.

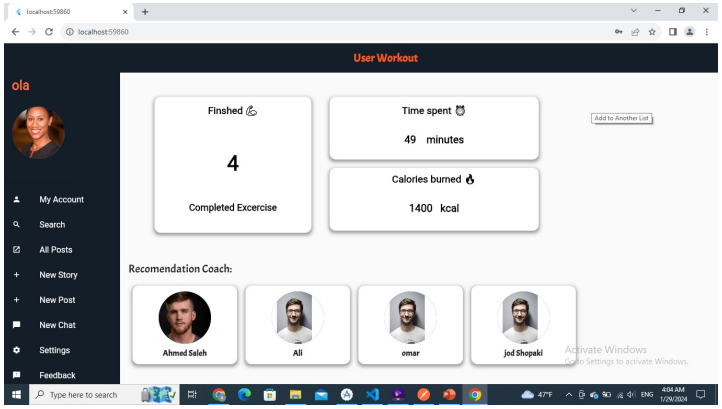

Figure 3.80: Calorie tracking

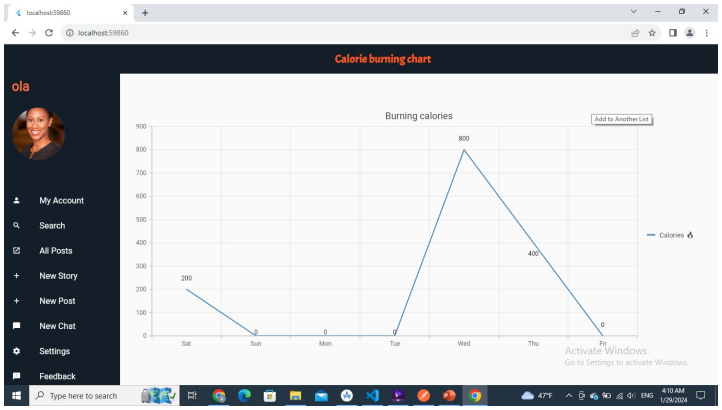

Figure 3.81: Calorie Tracking .

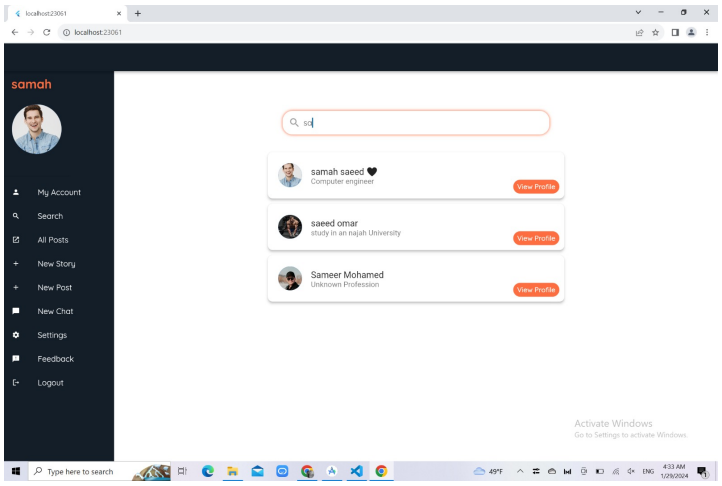

Figure 3.82: Search.

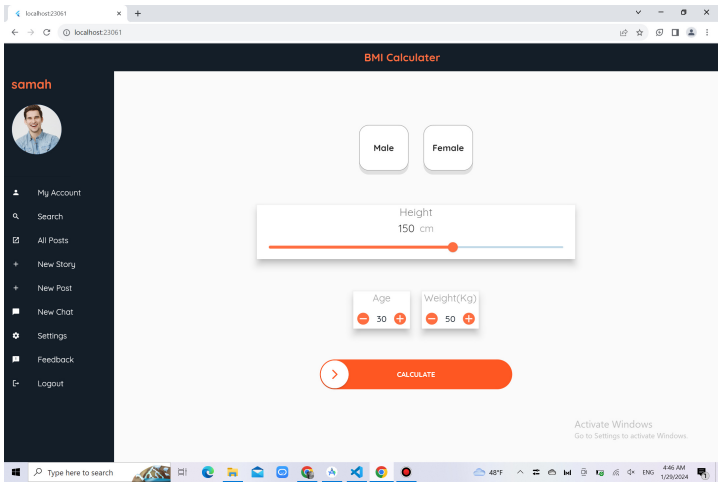

Figure 3.83: BMI.

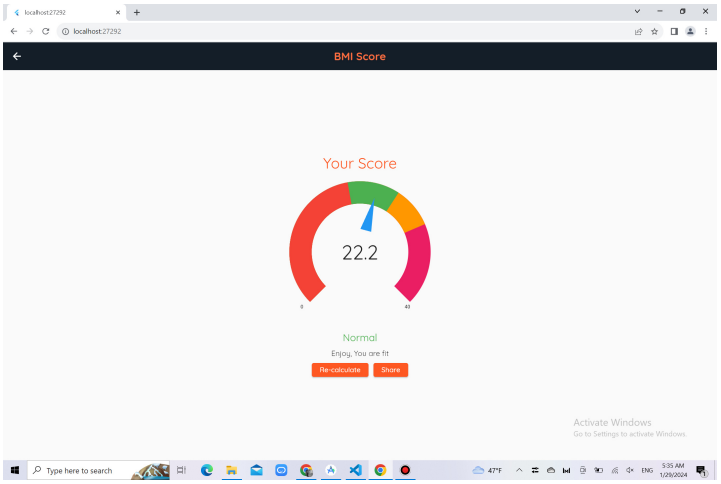

Figure 3.84: BMI Result.

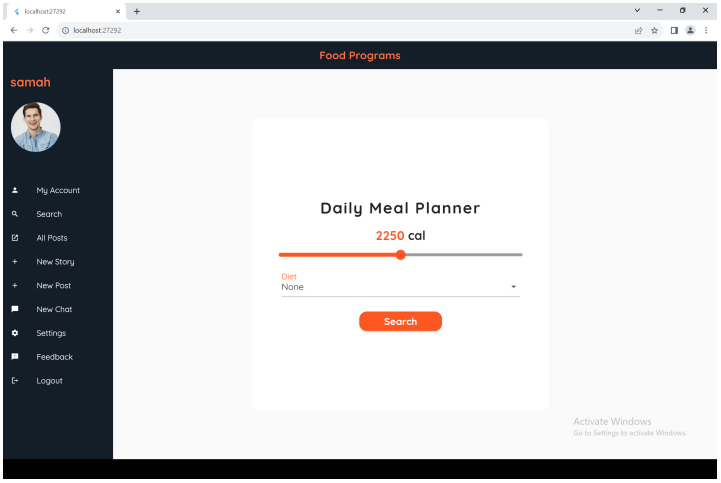

Figure 3.85: User Meals 1.

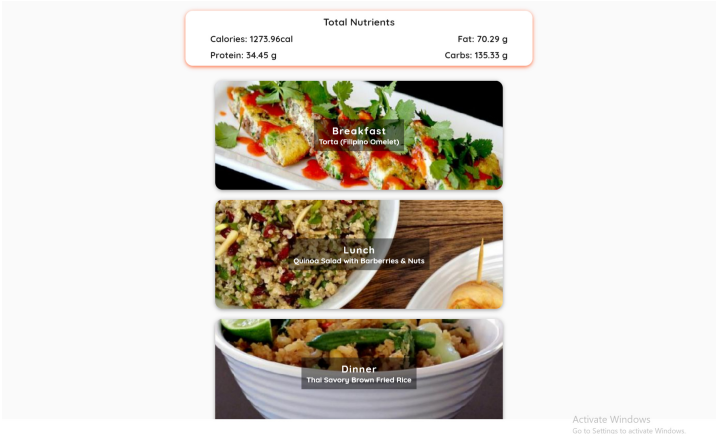

Figure 3.86: User Meals 2.

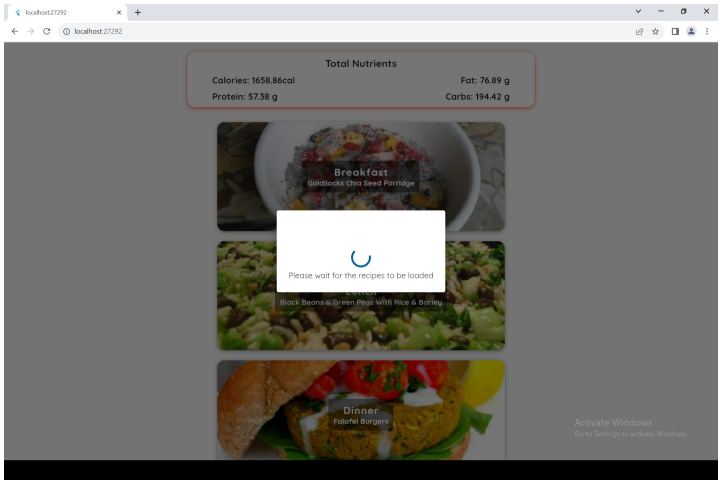

Figure 3.87: User Meals 3.

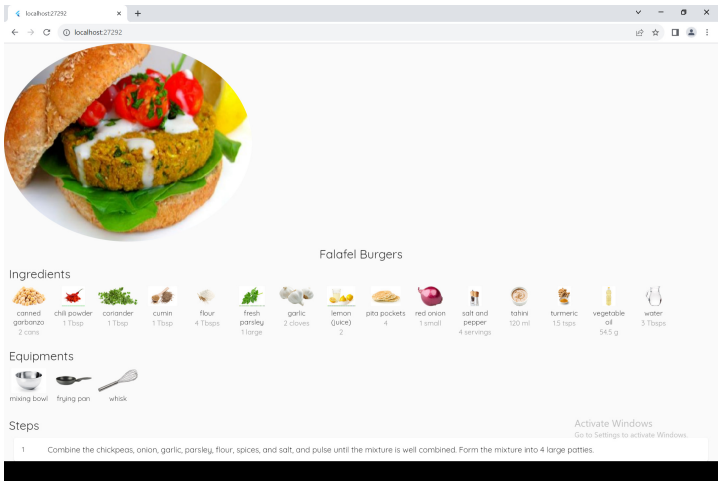

Figure 3.88: User Meals 4.

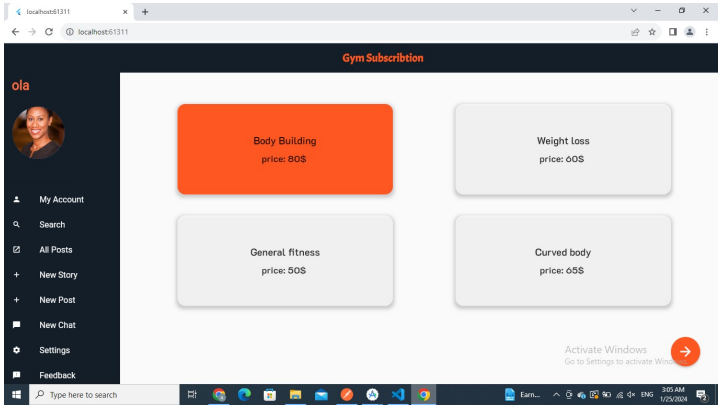

Figure 3.89: Monthly subscription.

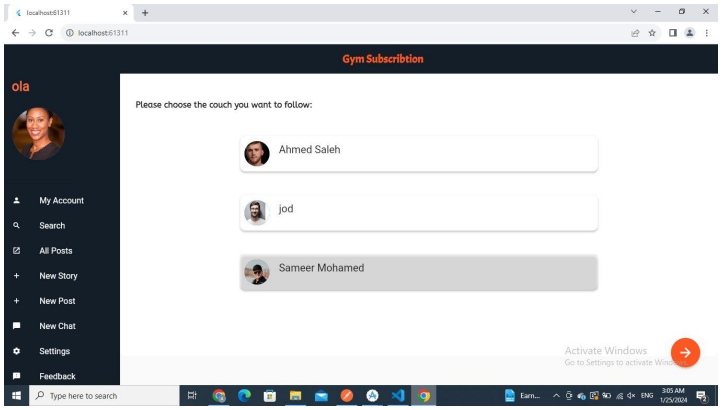

Figure 3.90: Monthly subscription

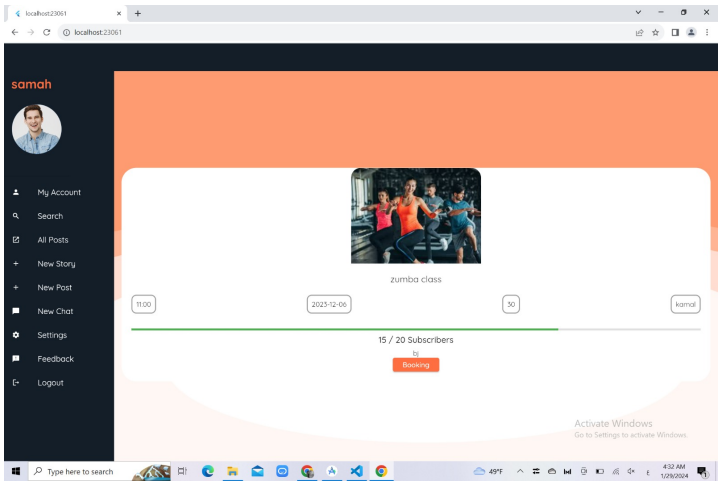

Figure 3.91: Book a Class 1.

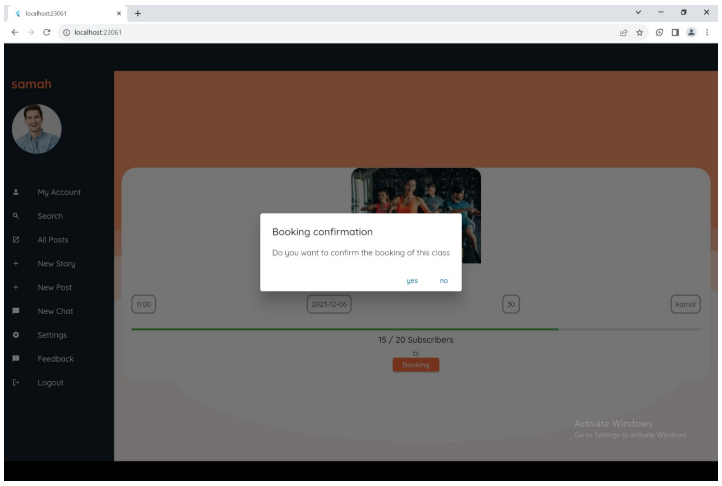

Figure 3.92: Book a Class 2.

Admin Mobile Pages :

The center director has several features that enable him to manage the center and the center's data, including trainers, nutritionists, users, and subscribers, easily.

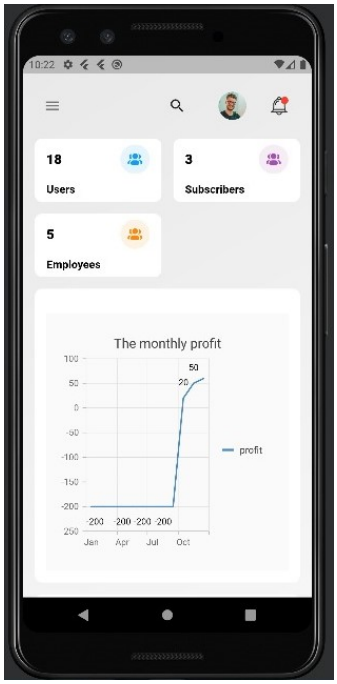

Figure 3.93: Home Page 1.

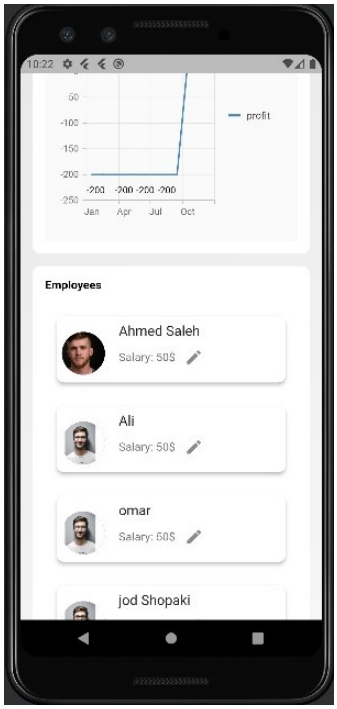

Figure 3.94: Home Page 2.

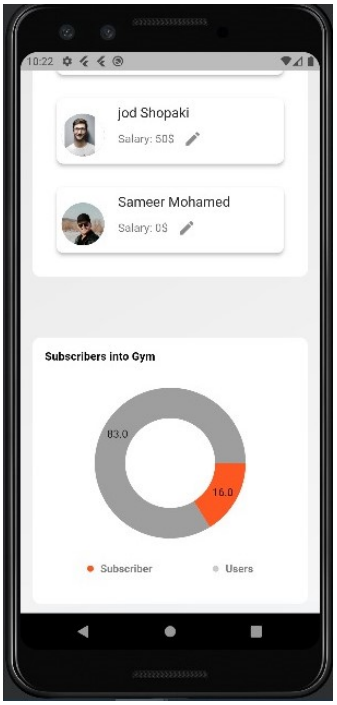

Figure 3.95: Home Page 3.

Here is shown in the pictures above the home page of the admin. The number of employees (coaches and nutritionists), the number of all users, and the number of subscribed users appear at the top of the first picture .

Then a graph is shown in which a report of the monthly profit calculated from user subscriptions is shown, and employee salaries are subtracted from it, and this is the net profit.

Figure 3.94 appear the employees, The admin can modify the monthly salary of the Employees.

Chart showing the ratio of the number of subscribed users to the number of non-subscribed users in the club.

Manager's menu, easy to access the features that the manager wants.

In the picture below , the manager displays the details of the monthly reservations and daily reservations.
We previously talked about the types of reservations that are available to the user as two options: either booking a full month or booking a single share for once.

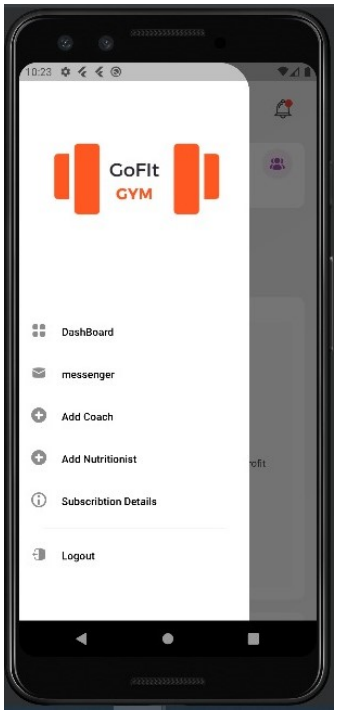

Figure 3.96: Admin Menu.

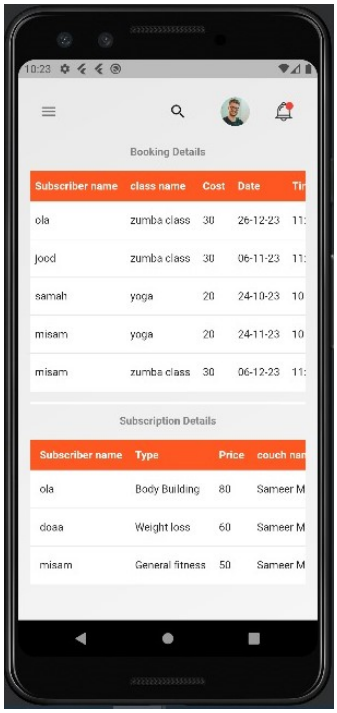

Figure 3.97: Subscriptions.

The manager also adds the accounts of coaches and nutritionists by entering name and password. The nutritionist and coach can then modify their passwords to maintain privacy.

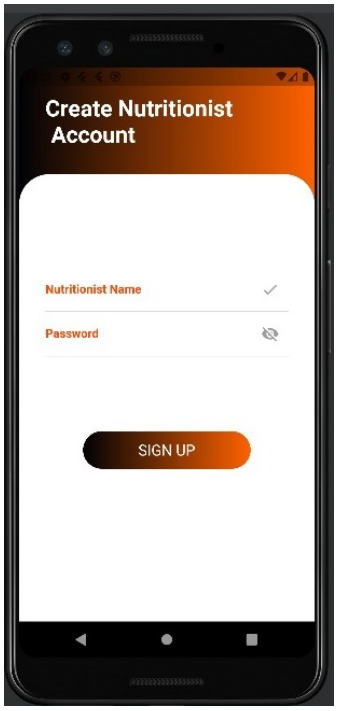

Figure 3.98: Add Nutritionist.

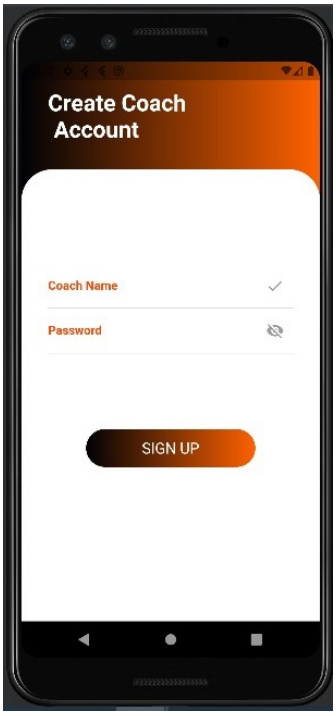

Figure 3.99: Add Coach.

Admin Web Pages :

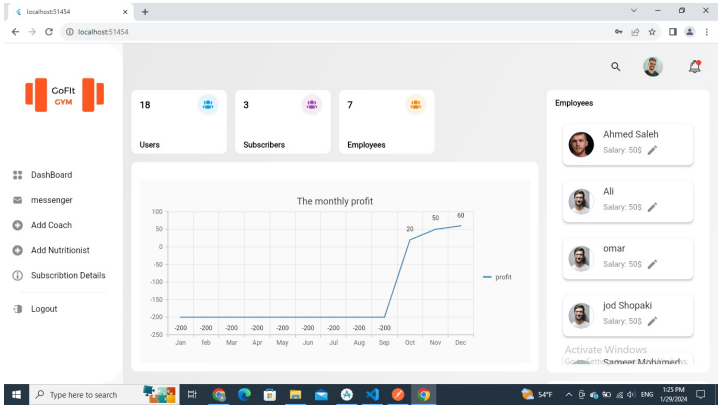

Figure 3.100: Admin Home Page 1.

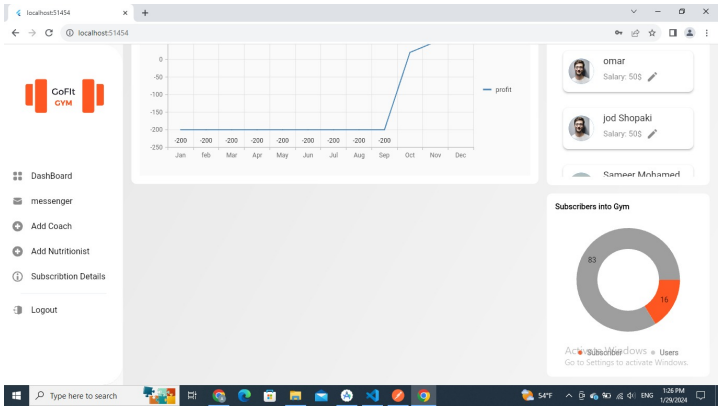

Figure 3.101: Admin Home Page 2.

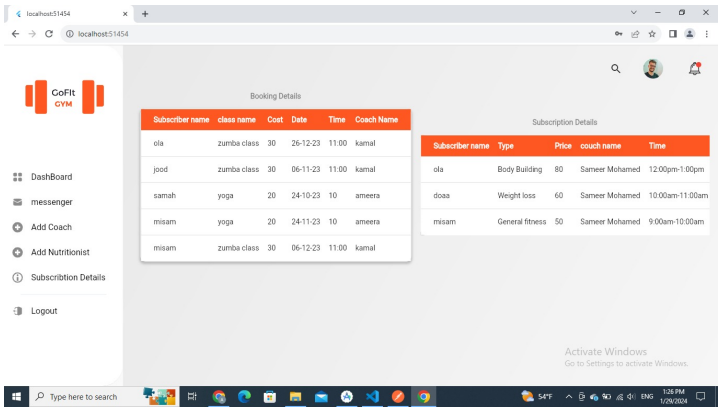

Figure 3.102: Admin Home Page 3.

Coach Mobile Pages :

The picture below shows the coach's home page and menu

Coaches Edit the content of the general exercises that appear on the guest page.

Coaches Edit the content of the private exercises (exercises according to the target muscle) that appear on the page of the user who has an account.

Here the muscle types are added

Here Coaches adds exercises according to the target muscle .

The coach adds class (It is held once), enters full details about the class, its time and date, and shows the name of the coach who added this class, the price of participating in it, and the maximum number of participants allowed.

We mentioned earlier that the user has two types of booking: booking one-time classes or a monthly subscription with a specific coach. Here on the coach pages, the coach can divide his working hours at the club, and these times appear on the user's page as available times for reservations and participation with this coach.

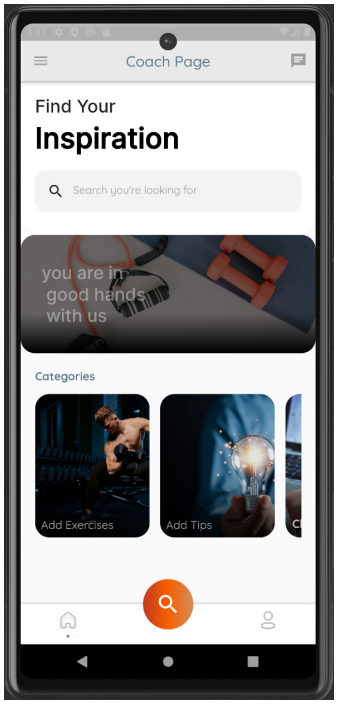

Figure 3.103: Home Page 1.

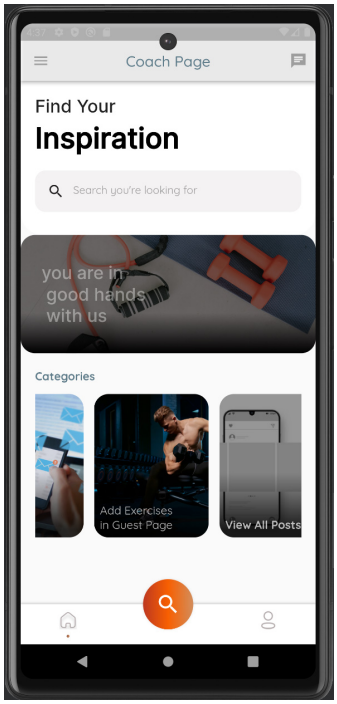

Figure 3.104: Home Page 2.

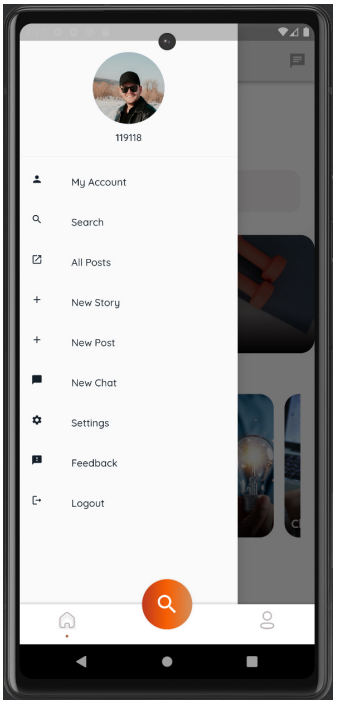

Figure 3.105: Coach Menu.

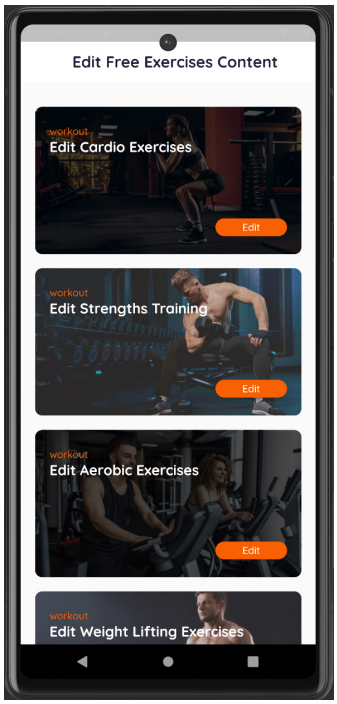

Figure 3.106: edit guest content 1.

Figure 3.107: edit guest content 2.

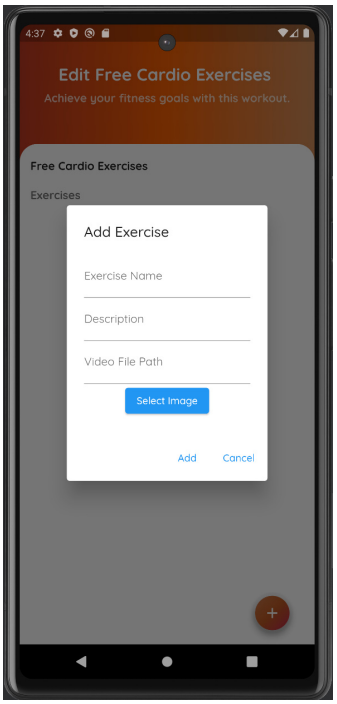

Figure 3.108: edit guest content 3.

After the user subscribes with a specific trainer -as we explained previously in the monthly subscription- , then will show in the trainer's calendar every day what classes he has and who is the subscriber, and there will be a check mark in the box if the user comes to the class. It is possible for the trainer to follow up on the participants through this.

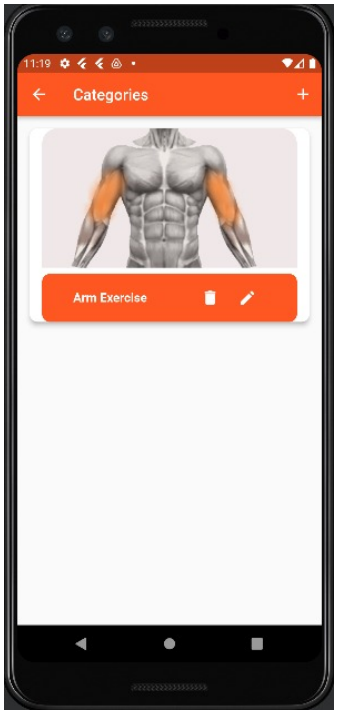

Figure 3.109: Muscle Types

Figure 3.110: Add Exercise

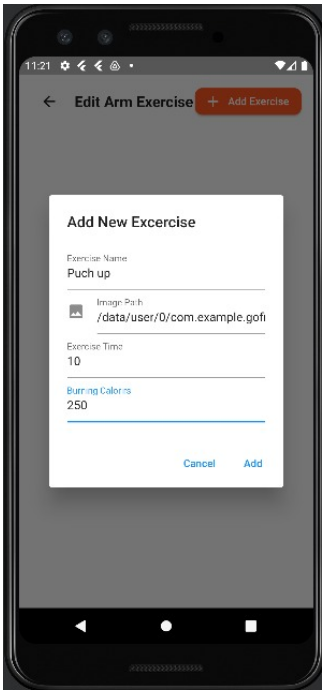

Figure 3.111: Add Exercise

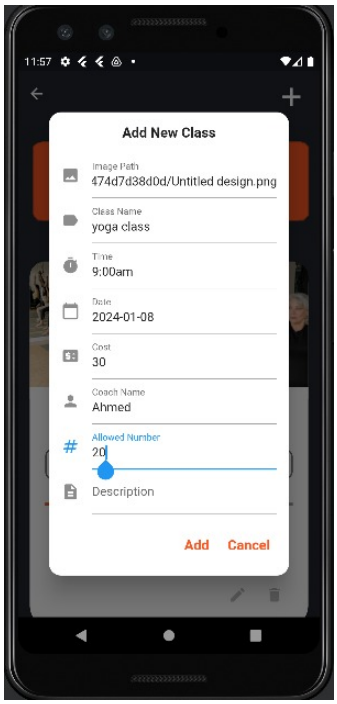

Figure 3.112: Add New Class.

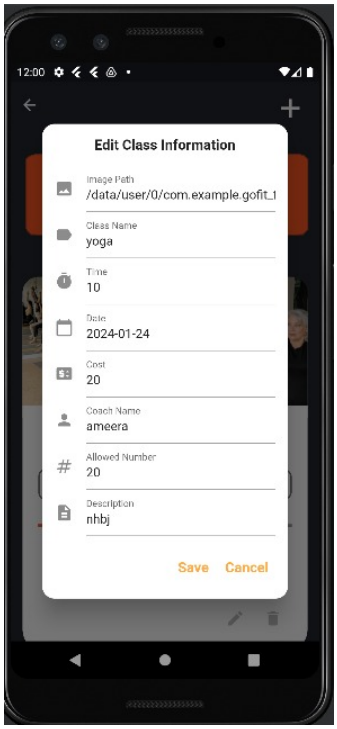

Figure 3.113: Edit Class Info

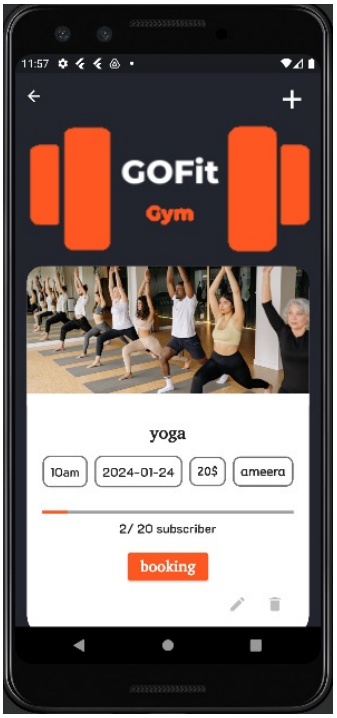

Figure 3.114: Class.

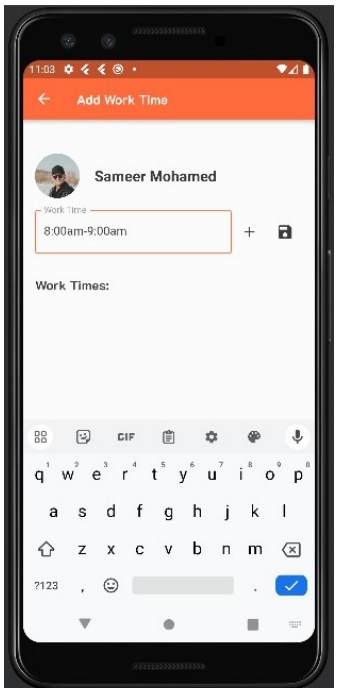

Figure 3.115: Coach program

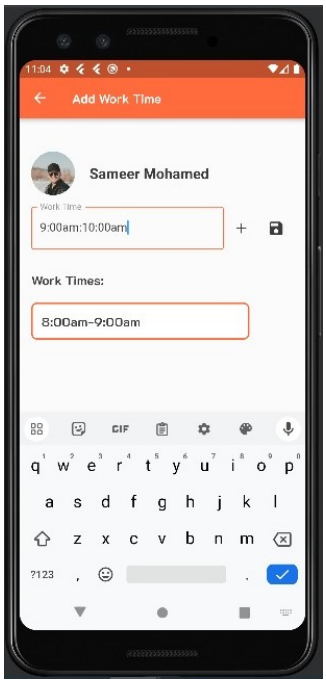

Figure 3.116: Coach program

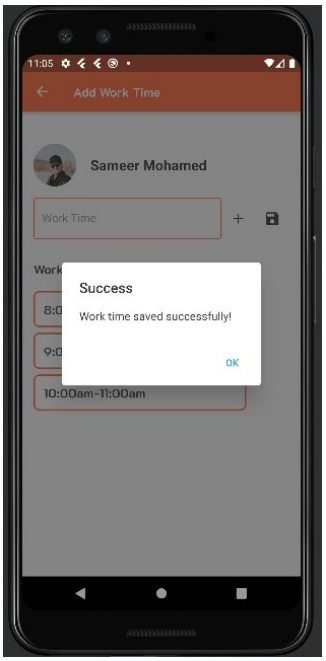

Figure 3.117: Coach program

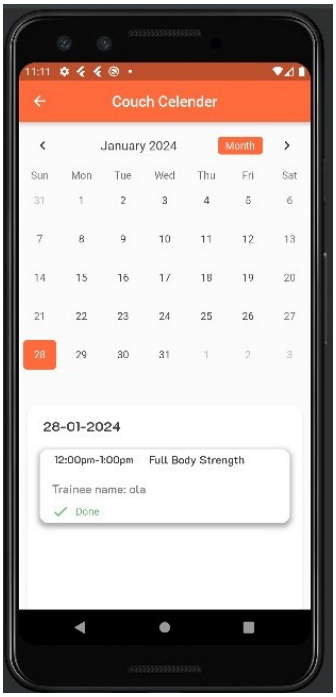

Figure 3.118: Coach calendar

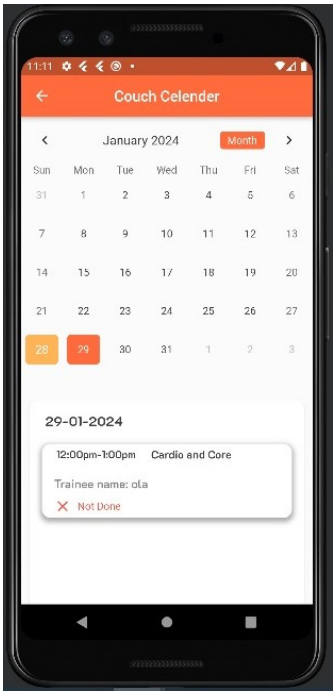

Figure 3.119: Coach calendar

Chat Feature : .

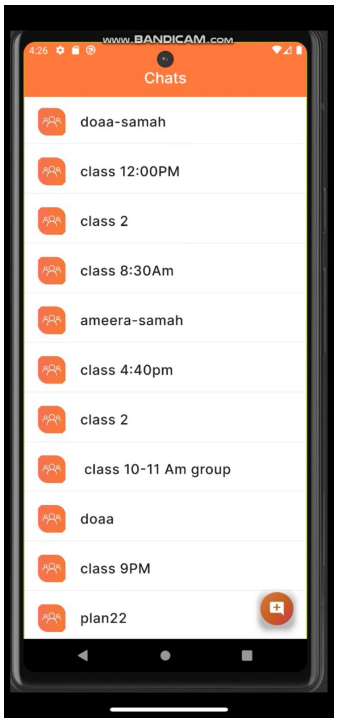

Figure 3.120: List of Chats

To create a new chat, you can chat with one or more (group) and then enter the name of the chat

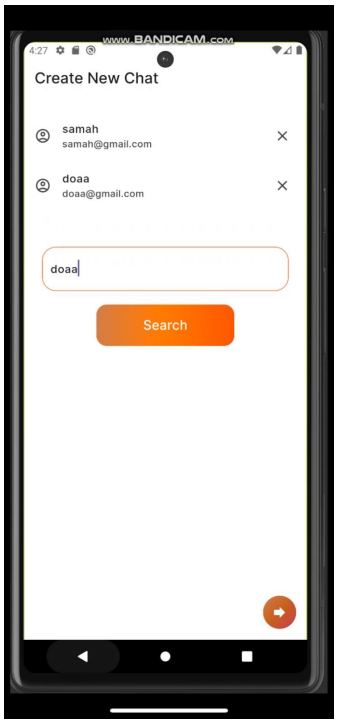

Figure 3.121: Add members

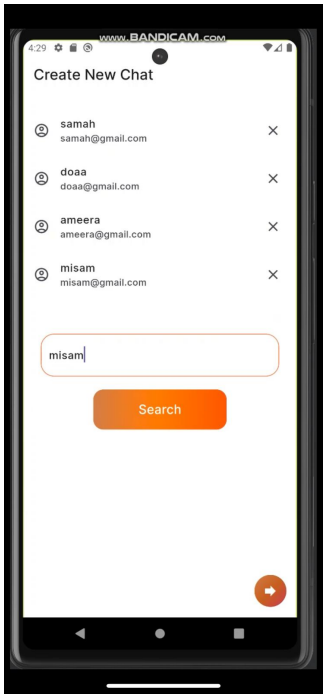

Figure 3.122: Add members

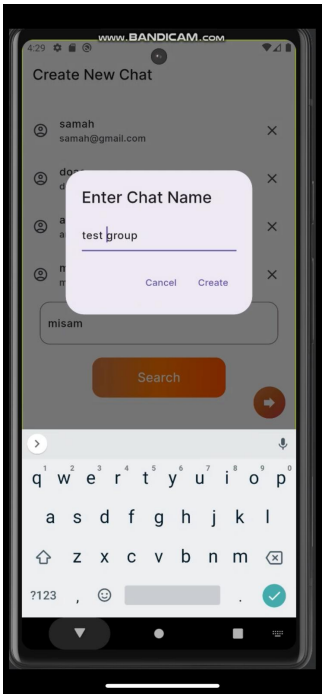

Figure 3.123: Chat naming

Figure 3.124: Chat Screen

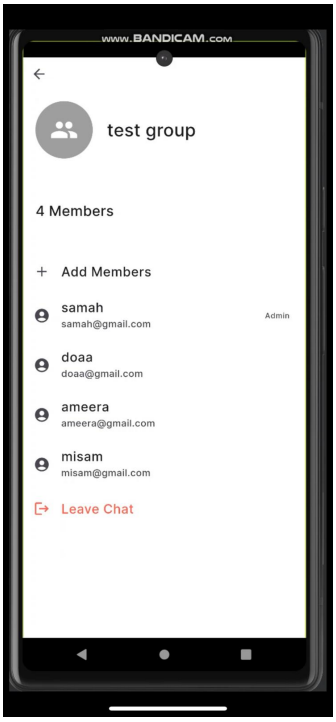

Figure 3.125: Chat Info

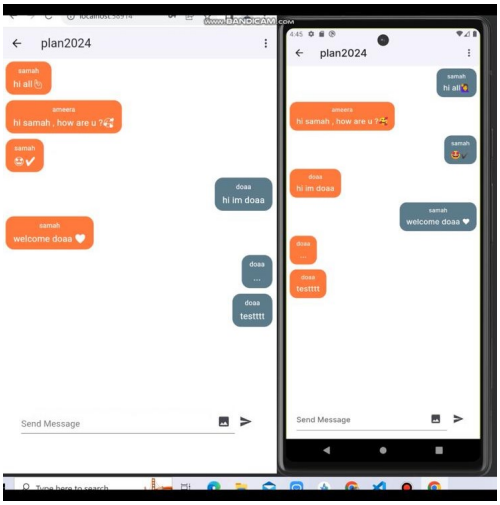

Figure 3.126: Instant messages.

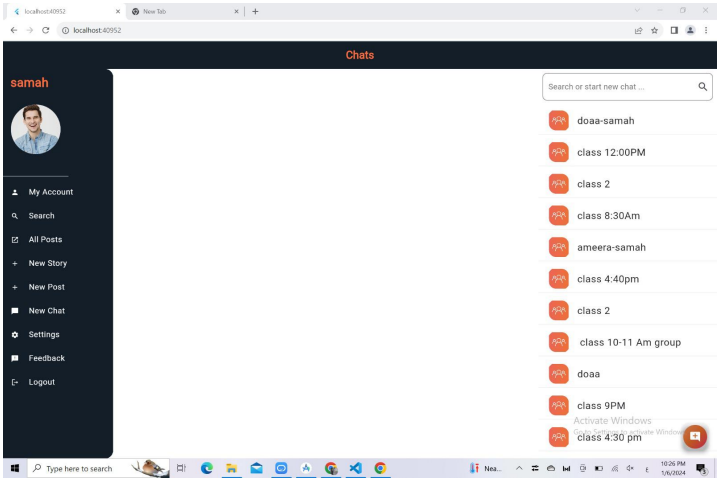

Figure 3.127: web chat.

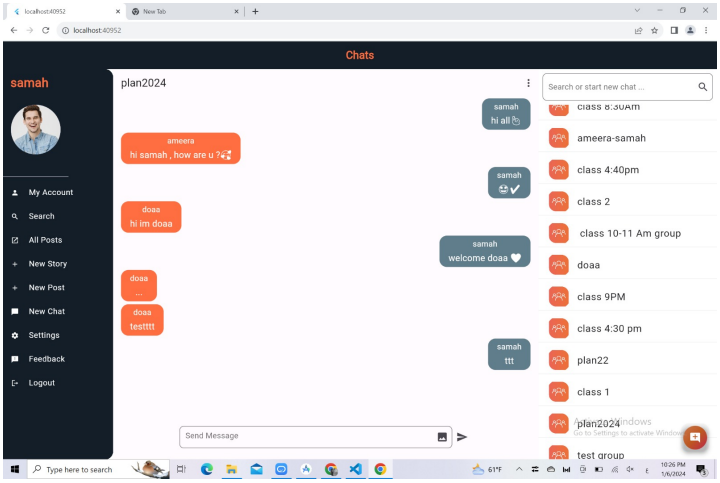

Figure 3.128: web chat.

Firebase is used in our app, where we used Firebase to create new conversations and add members to them. Firestore is used to store user details and group information. Firebase

Authentication is used by the Firebase SDK to handle login and registration processes. The current user's information is accessed using 'FirebaseAuth', and Firebase Cloud Firestore is used to store data in the database. Functions such as 'set', 'add', and 'get' are used to manage data in groups and conversations.

Data is organized into groups using Firestore groups. For example, group data is stored in the Groups group, and individual chat messages are stored in the Chats group.

Our app leverages Firebase Cloud Firestore's real-time capabilities to provide instant updates. When a new message is added to a specific group, the user interface is updated in real time (instant messaging).

One of Firebase's features is that it facilitates real-time identity management and storage, making it efficient for building chat applications

Nutritionist page :
The nutritionist has the same features as users and coaches. He sets up his personal page, publishes his interests through posts, and interacts with others through comments and likes. The nutritionist adds meals to the guest pages as follows

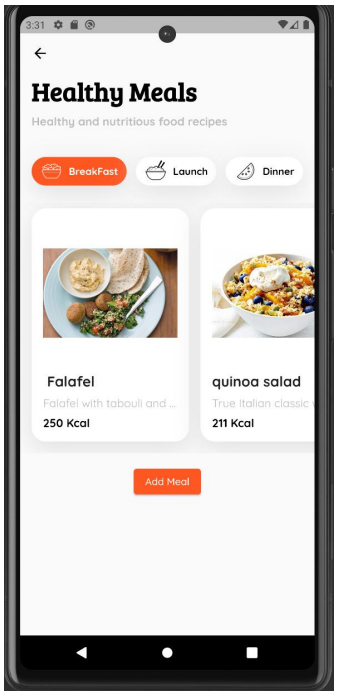

Figure 3.129: Edit Meals

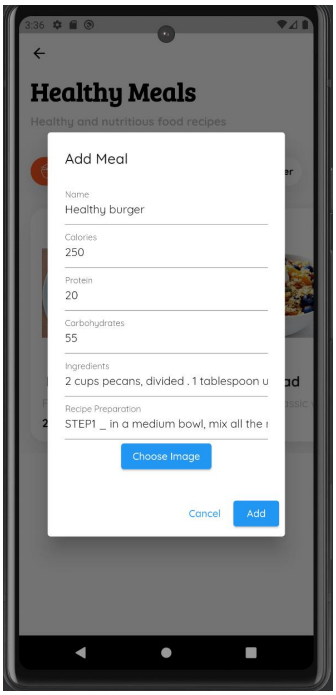

Figure 3.130: Edit Meals

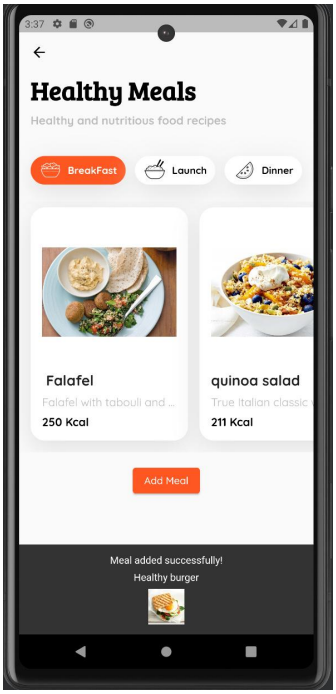

Figure 3.131: Edit Meals

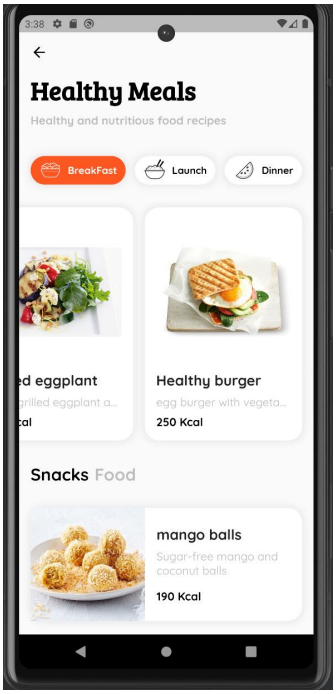

Figure 3.132: Meals Page

As can be seen in Figure 3.129, the nutritionist adds breakfast, lunch and dinner meals by adding the data as shown in Figure 3.130, and in Figure 3.131 shows the success of adding the meal, the meal will appears on the guest pages as shown in Figure 3.132.

Chapter 4

Future Work

The application can be integrated with other features such as a smart watch to monitor biometrics, which allows users to monitor heartbeat and blood pressure during exercise.

Artificial intelligence and machine language can be used to provide suggestions for accurate mathematical systems specific to each person according to their data

Periodic notifications can be added to remind you of appointments and encourage follow-up and continuity

Chapter 5

Conclusion

In Conclusion, the GOFIT Center project is an innovative endeavor at the nexus of technology, nutrition, and fitness. The dedication to offering consumers a smooth and customized journey in their quest for overall well-being served as the impetus for the creation of this extensive platform.

GOFIT caters to a wide range of customers by incorporating four distinct user types, each designed to fulfill particular needs. GOFIT is a multifaceted solution that benefits everyone involved—from the administrators who oversee the platform's operation to the nutritionists and fitness experts who advise users on their health journeys to the users themselves who take advantage of a plethora of services.

This project's technology decisions have been a major factor in its success. An interesting and uniform user experience on both Android and iOS devices is ensured by Flutter's use of the cross-platform interface. As the backend orchestrator, Node.js ensures dynamic user involvement by facilitating real-time data flow. The scalability and security of MongoDB's information storage and retrieval system are essential to the overall application's resilience.

The project was guided through its several phases by the Agile model methodology, which allowed for flexibility, cooperation, and continuous development. The development team successfully navigated requirements analysis, planning, design, development, and testing by using an iterative approach, guaranteeing a rigorous process in terms of methodology.

In spite of obstacles such as time limits and the requirement to obtain new technology, the project team showed adaptation and flexibility. A dedication to innovation and user happiness may be seen in the use of machine learning models and the emphasis on user interaction.

References

Padmasekara, G., 2014. Fitness apps, a Valid Alternative to the Gym: a pilot study. Journal of Mobile Technology in Medicine, 3(1), pp.37-45. [1].

Cabauatan, K.A. and Lopez, G.A., 2021. MOBILE FITNESS APPLICATIONS AND GYM EXERCISES OF FIRST-YEAR COLLEGE STUDENTS. Globus Journal of Progressive Education, 11(2).[2].

Maheedhar, M., Gaurav, A., Jilla, V., Tiwari, V.N. and Narayanan, R., 2016, August. StayFit: A wearable application for Gym based power training. In 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 6290-6293). IEEE.[3].