



An-Najah National University
Faculty of Engineering & Information Technology
Department of Computer Engineering

Software Graduation Project



Finesse

Prepared By:

Lama Loay Mletat
Mayyasa Eyad Shakaa

Supervised By:

Dr. Anas Toma
Dr. Raed Qadi

A Report Submitted in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Computer Engineering

June, 2026

Dedication

We dedicate this achievement to our families, whose endless love, encouragement, and support have been the foundation of our success. Their belief in us gave us the confidence and determination needed to overcome challenges and continue striving toward our goals.

To our parents, we offer our deepest appreciation for the sacrifices they have made throughout our lives. Their guidance, patience, and unwavering trust have inspired us to work hard and pursue our ambitions with dedication and perseverance.

We also extend this dedication to our friends and all those who supported us along the way. Their encouragement, understanding, and positive influence helped make this journey more enjoyable and rewarding.

Acknowledgments

We would like to express our sincere gratitude to Dr. Raed Qadi and Dr. Anas Toma for their valuable guidance, continuous support, and constructive feedback throughout the development of this project. Their knowledge, experience, and insightful suggestions greatly contributed to improving the quality of our work and helped us overcome many challenges encountered during the project.

We are especially thankful for their encouragement, patience, and dedication, which motivated us to stay focused and committed to achieving our goals. Their support played an important role in the successful completion of this project and provided us with a valuable learning experience that will benefit us in our future careers.

Disclaimer

This report has been prepared by Lama Mleitat and Mayasa Shakaa, students of the An-Najah National University. Except for editorial modifications introduced as part of the assessment process, the content has not been altered and may still contain minor language or content inaccuracies.

The views, findings, conclusions, and recommendations expressed in this report are solely those of the authors. The use of this report for purposes other than that for which it was prepared shall not imply any responsibility or liability on the part of An-Najah National University.

Abstract

Finesse is a web and mobile-based fitness and wellness platform designed to provide users with a personalized, intelligent, and engaging health management experience. The platform integrates fitness training, nutrition planning, progress tracking, artificial intelligence, and social interaction features into a single comprehensive system aimed at promoting healthier lifestyles and long-term wellness.

The system supports multiple user roles, including users, coaches, nutritionists, and administrators, each with dedicated functionalities and management capabilities. Through integrated communication tools, users can connect with fitness professionals for personalized guidance and support throughout their fitness journey.

Finesse provides a comprehensive exercise database containing detailed workout information, including targeted muscle groups, difficulty levels, estimated duration, calorie expenditure, and instructional videos. Based on the user's selected goal—weight loss, weight gain, or weight maintenance—the platform recommends suitable workout plans and nutritional programs. Additionally, users can generate personalized workout routines tailored to their goals, fitness levels, and preferences.

The nutrition module calculates individual calorie requirements and offers meal recommendations aligned with the user's objectives. Users can track daily calorie consumption, monitor water intake, record body weight and measurements, and visualize their progress over time through detailed tracking features.

A key component of the platform is its AI-powered assistant, which provides three primary services. First, it acts as a virtual fitness coach, offering personalized fitness and nutrition guidance and answering user inquiries. Second, it utilizes image analysis capabilities to estimate the calorie content of meals from uploaded food images, helping users make more informed dietary decisions. Third, it generates personalized weekly summaries that analyze user activity, progress, nutrition habits, and fitness achievements over the previous seven days, providing valuable insights and recommendations for improvement.

To further enhance engagement and motivation, Finesse incorporates gamification features such as experience points (XP), achievement levels, streak tracking, and personalized motivational insights. The platform also includes a community feature where users can create posts, share experiences, interact with others, and build a supportive fitness environment. In addition, a leaderboard system ranks users according to their accumulated XP points, encouraging healthy competition and sustained participation.

By combining fitness management, nutrition guidance, artificial intelligence, social interaction, and motivational tools within a unified platform, Finesse delivers a smart, adaptive, and user-centered solution that empowers individuals to achieve their health and fitness goals effectively and sustainably.

Dedication	2
Acknowledgments	2
Disclaimer	3
Abstract	4
Chapter 1	15
Introduction	15
1.1 General Background	15
1.2 Objectives	17
1.3 Significance and Importance of the Work	18
1.4 Organization of the Report	19
Chapter 2	20
Literature Review	20
2.1 Theoretical Background of the Project	20
2.2 Previous Similar Works	21
Chapter 3	23
Methodology	23
3.1.1 Economic Constraints	23
3.1.2 Environmental Constraints	24
3.1.3 Social Constraints	24
3.1.4 Political Constraints	25
3.1.5 Ethical Constraints	26
3.1.6 Health and Safety Constraints	26
3.1.7 Manufacturability (Implementation Feasibility)	27
3.1.8 Sustainability Constraints	27
3.2 Used Technologies	28

Flutter	28
Why Flutter?	29
Firebase	29
Why Firebase?	30
Node.js	30
Why Node.js?	31
MySQL	31
Why MySQL?	32
Visual Studio Code	32
Why Visual Studio Code?	32
Postman	33
Why Postman?	33
Groq AI Integration	34
3.3 Finesse as a Mobile and Web Application	35
3.3.1 Splash Screen	36
3.3.2 Onboarding Screens	36
3.3.3 Landing Page (Web Application)	41
3.3.4 Authentication and Validation	43
3.3.6 Login	47
3.3.7 Login Verification and Error Handling	48
3.3.8 Google Sign-In Authentication	49
3.3.9 Forgot Password	52

3.3.10 Reset Password	54
3.3.11 User Information and Fitness Profile Management	56
3.4 Profile Management	59
3.5 Settings Management	63
3.5.1 Finesse Pro and Subscription Management	65
3.5.2 Notifications and Reminders Management	68
3.5.3 Support Chat & Multi-Role Communication Management	71
3.5.3.1 Communication Hierarchy: Target Contexts	71
3.5.3.2 Modalities of Interaction: Dual Asynchronous & Synchronous Channels	72
3.5.3.3 Client-Side Media Execution and Hardware Handshake	73
3.5.3.4 Thread Constraints and Lifecycle Safeguards	74
3.6 Notifications Feature	77
3.7 Home Screen (Dashboard)	79
3.7.1 Quick Actions Section	86
3.7.2 Dynamic Workout Generation Flow (AI Coach Engine)	88
3.7.3 Workout Plan and Execution Directory	92
3.7.4 Goal-Aligned Meal Planning and Calorie Tracking	96
3.7.5 Gamified Challenges System	99
3.7.6 Hydration Tracking System	103
3.7.7 Weight & Progress Tracking System	107

3.7.8 Progress Analytics & PDF Report Export	112
3.7.9 AI Coach Integration (Powered by Groq)	117
3.7.10 Social Community & Interactive Engagement Hub	122
3.8 Admin Screens & Management Infrastructure	131
3.8.1 Admin Panel Overview & System Metrics	131
3.8.2 Pro Version Settings & Configuration	135
3.8.3 Support Inbox & Ticket Management	138
3.8.4 Members Directory & User Profile Management	141
3.8.5 Content Library & Catalog Management	147
3.9 Nutritionist Portal (Nutrition Hub)	155
3.9.1 Members Directory (Nutritionist View)	155
3.9.2 Client Profile Deep-Dive & Read-Only Governance	155
3.9.3 Nutritionist Content Library & Isolated Support	158
3.10 Coach Portal (Coach Hub)	163
3.10.1 Members Directory & Assessment Registry	163
3.10.2 Exercises Database & Training Library (Training)	165
3.10.3 Isolated Coach-Level Support Inbox	166
Results and Analysis	173
Discussion	175
Conclusion and Recommendation	176
6.1 Summary	176
6.2 Recommendations	177

6.3 What We Have Learned	178
6.4 Future Work	178
Figure 1 :Splash screen	36
Figure 2 :Track your weight Screen	37
Figure 3 :Nutrition that fits you Screen	38
Figure 4 : Workouts & hydration Screen	39
Figure 5 : AI coach in your pocket Screen	40
Figure 6 : Landing Screens	43
Figure 7 : Sign Up Screen	47
Figure 8 : Login Screen	48
Figure 9 : Login Screen	49
Figure 10 : Google Sign-In Screen	52
Figure 11 : Profile Screen with Google Sign-In	53
Figure 12 : Forgot Password	54
Figure 13 : Reset Password Email	56
Figure 14 : Reset Password Page	56
Figure 15 : Reset Password Page	57
Figure 16 : User Info Screen	59
Figure 17 : User Info Screen	59
Figure 18 : Profile Screen	61
Figure 19 : Profile Screen	62
Figure 20 : Add Post Screen	63

Figure 21 : Add Post Screen	63
Figure 22 : Add Post Screen	64
Figure 23 : Edit name Screen	64
Figure 24 : Dark mode Screen	66
Figure 25 : App language Screen	67
Figure 26 : Finesse Pro Screen	69
Figure 27 : Finesse Pro Screen	69
Figure 28 : Pay for the Pro screen	70
Figure 29 : End of trial period notification	70
Figure 30 : End of trial period notification	71
Figure 31 : Reminders screen	73
Figure 32 : Reminders screen	73
Figure 33 : Support chat screen	75
Figure 34 : chatting screen	78
Figure 35 : Video call and notification followed screen	79
Figure 36 : Notification screen	81
Figure 37 : Home screen	82
Figure 38 : Home screen	83
Figure 39 : Home screen	84
Figure 40 : Generate workout screen	94
Figure 41 :Workout screen	97
Figure 42 :Workout screen	98
Figure 43 :Meal plan screen	100
Figure 44 :Log meal screen	101
Figure 45 :Warning notice screen	101

Figure 46 :Challenges screen	104
Figure 47 :My challenges screen	105
Figure 48 :Hydration screen	108
Figure 49 :Hydration screen	109
Figure 50 :weight and track weight screen	113
Figure 51 :weight and track weight screen	114
Figure 52 :Weight progress screen	118
Figure 53 :Weight progress PDF file screen	119
Figure 54 :Ai coach screen	122
Figure 55 :AI coach screen	123
Figure 56 :AI coach screen	124
Figure 57 :Scanning calories screen	124
Figure 58 :Community screen	129
Figure 59 :Invittation screen	129
Figure 60 :Community screen	130
Figure 61 :Post order screen	131
Figure 62 :Post writing screen	131
Figure 63 :Leaderboard screen	132
Figure 64 :Profile screen	132
Figure 65 :Follwore screen	133
Figure 66 :Profile screen	133
Figure 67 :Home screen	136
Figure 68 :Admin panel screen	137
Figure 69 :Admin panel screen	137
Figure 70 :Admin panel screen	140

Figure 71 :Pro setting screen	140
Figure 72 :Pro setting screen	141
Figure 73 :Admin panel(support inpox button) screen	143
Figure 74 :Support inpox screen	144
Figure 75 :Members screen	148
Figure 76 :Member screen	148
Figure 77 :Member screen	149
Figure 78 :Member screen	149
Figure 79 :Member screen(role)	150
Figure 80 :Meals screen	153
Figure 81 :Meals screen	154
Figure 82 :Exercises screen	154
Figure 83 :Add exercises screen	155
Figure 84 :Challenges screen	155
Figure 85 :Add challenges screen	156
Figure 86 :Food category screen	156
Figure 87 1:Add food cats screen	157
Figure 88 :Add food cats screen	157
Figure 89 :Members screen	160
Figure 90 :Member screen	160
Figure 91 :Meals screen	163
Figure 92 :Add meal screen	163
Figure 93 :Members screen	167
Figure 94 :Member screen	167
Figure 95 :Exercises screen	170

Figure 96 :Add exercises screen	170
Figure 97 :Support inbox screen	171
Figure 98 :Smart watch screen	172
Figure 99 :Smart watch(Calories) screen	173
Figure 100 :Smart watch(Steps) screen	173
Figure 101 :Smart watch (Water)screen	174
Figure 102 :Smart watch(Ai Coach) screen	174
Figure 103 :Smart watch(generate byAi) screen	175

Chapter 1

Introduction

1.1 General Background

In recent years, the increasing awareness of healthy lifestyles and physical well-being has led to a growing demand for digital fitness and nutrition solutions. Many individuals seek effective ways to manage their fitness activities, monitor their nutritional habits, and achieve personal health goals such as weight loss, weight gain, or weight maintenance. However, maintaining a healthy lifestyle often requires continuous guidance, personalized planning, and consistent motivation, which may not always be available through traditional fitness methods.

Although numerous fitness applications exist today, many provide generic workout plans and nutrition recommendations that do not adequately adapt to the unique needs, preferences, and goals of individual users.

Additionally, users often need to switch between multiple applications to track workouts, monitor nutrition, communicate with experts, and analyze their progress, which can reduce convenience and engagement.

To address these challenges, this project introduces **Finesse**, an intelligent fitness and wellness platform that integrates workout planning, nutrition

management, AI-powered assistance, progress tracking, social interaction, and communication with fitness professionals into a single unified system. The platform aims to provide users with a more personalized and engaging fitness experience while promoting long-term healthy habits and sustainable lifestyle improvements.

Finesse enhances user experience through an AI-powered assistant that functions as a virtual fitness coach, analyzes food images to estimate calorie intake, and generates weekly summaries based on user activity and progress. In addition, the platform includes a community feature that allows users to share posts and interact with each other, as well as a leaderboard system that ranks users based on experience points (XP), encouraging motivation and healthy competition.

By combining fitness management, nutrition guidance, artificial intelligence, and social and motivational features within one platform, Finesse aims to deliver a comprehensive, adaptive, and user-centered solution that supports individuals in achieving their health and fitness goals effectively.

1.2 Objectives

The main objective of this project is to design and implement an intelligent fitness and wellness platform that supports users in achieving their health and fitness goals through personalized recommendations, progress tracking, social interaction, and professional guidance.

The specific objectives of the project are as follows:

- To provide users with personalized workout plans tailored to their fitness goals, experience level, and individual preferences.
- To develop an AI-powered recommendation system that generates adaptive workout suggestions based on user behavior and needs.
- To build a comprehensive exercise database that includes detailed descriptions, targeted muscle groups, difficulty levels, estimated duration, calorie burn, and instructional videos.
- To calculate personalized daily calorie requirements and provide suitable meal recommendations aligned with user objectives.
- To enable users to track key health metrics such as calorie intake, water consumption, body weight, and body measurements over time.
- To integrate an AI fitness assistant capable of acting as a virtual coach, answering user queries, and providing personalized health and fitness guidance.
- To support AI-based food image analysis for estimating calorie intake from uploaded meal images.
- To generate weekly personalized summaries that analyze user progress, activity, and nutrition habits over time.
- To support real-time communication between users, coaches, nutritionists, and administrators through chat and video communication features.
- To implement gamification elements such as XP points, streaks, achievement levels, and motivational feedback to enhance user engagement and consistency.
- To provide dedicated dashboards and management tools for coaches, nutritionists, and system administrators.
- To support integration with wearable devices, such as smartwatches, for real-time fitness and activity tracking.

1.3 Significance and Importance of the Work

The significance of this project lies in its ability to provide a comprehensive and personalized fitness management solution that integrates multiple health-related services within a single platform. By combining workout planning, nutrition guidance, AI-powered assistance, progress tracking, social interaction, and professional communication tools, Finesse simplifies the process of managing fitness activities and reduces the need to use multiple separate applications.

The platform promotes healthier lifestyles by encouraging continuous monitoring, personalized feedback, and motivational gamification features such as XP points, streaks, and achievement levels. In addition, the integration of artificial intelligence enhances the user experience by delivering adaptive recommendations, intelligent coaching, and personalized insights based on user behavior and progress.

From a practical perspective, the system benefits multiple stakeholders, including users seeking improved health and fitness, coaches providing training guidance, nutritionists offering dietary support, and administrators managing the platform. The system is also designed with high scalability, enabling integration with wearable devices such as smartwatches for real-time fitness tracking, as well as supporting a robust AI engine for more accurate and personalized recommendations.

As the demand for digital health and fitness solutions continues to grow globally, Finesse represents a modern and intelligent approach to enhancing personal wellness through technology-driven personalization, data analysis, and smart decision-making support.

1.4 Organization of the Report

This report is organized into several chapters that describe the design, development, and evaluation of the proposed system. Chapter One presents an introduction to the project, including background information, objectives, and significance. Chapter Two reviews related work and existing fitness and wellness solutions. Chapter Three discusses the system requirements, analysis, and architectural design. Chapter Four describes the implementation details, technologies, and development process. Chapter Five presents system testing, evaluation, and results. Finally, Chapter Six summarizes the project conclusions and discusses possible future improvements and extensions.

Chapter 2

Literature Review

2.1 Theoretical Background of the Project

Digital health and fitness platforms have become an essential part of modern society, enabling users to manage their physical activity, nutrition, and overall well-being through web and mobile applications. These

platforms play a significant role in helping individuals achieve personal health goals such as weight loss, weight gain, or maintaining a healthy lifestyle.

Such systems are typically based on several core concepts, including user personalization, data tracking, recommendation systems, progress monitoring, and user engagement mechanisms. These components work together to provide a tailored experience that adapts to each user's fitness level, preferences, and health objectives.

From a system design perspective, fitness platforms rely heavily on personalized recommendation engines that suggest workout plans and nutrition programs based on user input and historical data. In addition, artificial intelligence has become a key component in modern fitness applications, enabling features such as virtual coaching, intelligent feedback, and automated analysis of user behavior and progress.

Another important concept in digital fitness systems is gamification, which is used to increase user motivation and consistency. This is achieved through features such as experience points (XP), achievement levels, streak tracking, and performance rewards. These mechanisms encourage users to maintain long-term engagement and develop sustainable healthy habits.

Furthermore, many modern platforms integrate social interaction features such as community posts, user engagement, and leaderboards. These features enhance motivation through social comparison and support, creating a more interactive and engaging user experience.

Finally, wearable device integration has become an important aspect of fitness systems, allowing real-time tracking of physical activity and health

metrics. Combined with AI-driven insights and personalized recommendations, these technologies form the theoretical foundation of intelligent fitness and wellness platforms such as Finesse.

2.2 Previous Similar Works

Several existing fitness and health applications have been developed to help users manage their physical activity, nutrition, and overall wellness. These systems provide useful features; however, each has its own limitations in terms of personalization, integration, and user engagement.

MyFitnessPal is one of the most popular nutrition and calorie tracking applications. It allows users to log meals, track daily calorie intake, and monitor macronutrients. While it is highly effective in dietary tracking, it offers limited personalization in workout planning and lacks advanced AI-driven coaching capabilities.

Nike Training Club is a fitness application that provides a wide range of workout programs designed by professional trainers. It offers high-quality exercise routines and instructional videos; however, it does not provide integrated nutrition tracking or personalized AI-based recommendations that adapt dynamically to user progress.

Fitbit is a health and activity tracking platform that integrates with wearable devices to monitor steps, heart rate, sleep quality, and physical activity. Although it provides strong real-time health monitoring, its dependency on wearable hardware and limited interactive coaching features reduce its flexibility as a standalone fitness solution.

In comparison, the proposed Finesse platform combines the strengths of these systems while addressing their limitations. It integrates workout planning, nutrition tracking, AI-powered coaching, food image calorie estimation, social interaction features, and gamification elements within a single unified system. This makes Finesse a more comprehensive and adaptive solution that enhances user engagement and supports personalized fitness journeys.

Chapter 3

Methodology

This chapter describes the methodology followed in the design and development of the proposed intelligent fitness and wellness platform, Finesse. It explains the system constraints, technical considerations, and non-technical factors that influenced the design and implementation decisions. The methodology focuses on building a scalable, user-friendly,

and intelligent system that integrates fitness tracking, nutrition management, artificial intelligence features, and user interaction within a single platform.

3.1.1 Economic Constraints

Budget Limitations:

This project is a non-profit, student-based graduation project. Therefore, development was carried out using open-source tools and free-tier services whenever possible. Paid services were avoided during development, and external APIs such as Stripe and Groq were used in their free or limited usage modes for testing and development purposes only.

Maintenance Costs:

The system was designed with a modular architecture to minimize future maintenance costs. This approach ensures that individual components, such as the AI services, authentication, and tracking modules, can be updated or improved independently without affecting the entire system, allowing for easier maintenance and scalability.

3.1.2 Environmental Constraints

Power Consumption:

The Finesse platform is designed to be lightweight and performance-optimized to minimize unnecessary server load and reduce client-side resource consumption. Efficient data handling and optimized API calls are used to ensure smooth performance across web and mobile platforms.

Environmentally Friendly Hosting:

Although the system has not been deployed on a final production server, the architecture is designed to support deployment on cloud hosting providers that utilize energy-efficient and scalable data centers. This approach helps reduce the overall environmental impact while maintaining system availability and performance.

3.1.3 Social Constraints

Information Security:

Ensuring the security of user data is a core design constraint. The system applies secure authentication mechanisms, encrypted communication, and controlled access to sensitive information. Health-related data, user profiles, and activity records are protected to prevent unauthorized access.

Privacy:

User privacy is strictly maintained. Personal information, fitness records, nutrition data, body measurements, and AI interaction history are accessible only to authorized users and administrators when necessary. Public exposure of sensitive user data is prevented.

Social Communication:

The platform facilitates structured communication with coaches and nutritionists through controlled chat and communication features. This helps ensure effective guidance, reduces

miscommunication, and maintains a professional and supportive environment throughout the user's fitness journey.

3.1.4 Political Constraints

Inclusivity:

The platform is designed to be inclusive and accessible to all users regardless of gender, age, fitness level, or background. The system provides equal access to features and services while supporting diverse health and fitness goals.

Regulatory Compliance:

The system respects applicable digital regulations and data protection principles. Personal and health-related information is handled securely, and users are informed about how their data is collected, stored, and used within the platform.

3.1.5 Ethical Constraints

Honesty and Transparency:

The platform is designed to provide accurate and transparent fitness and nutrition information. AI-generated recommendations

are intended to support users in making informed decisions regarding their health and wellness. Users are provided with clear information about the platform's features, recommendations, and data usage practices.

Intellectual Property:

The system respects intellectual property rights by using original content and legally available resources. Any external materials used within the platform are properly credited and utilized in accordance with applicable copyright regulations.

3. 1. 6 Health and Safety Constraints

User Safety:

The platform is designed to support healthy lifestyle habits through fitness and nutrition guidance. AI-generated recommendations are intended for informational purposes and should not be considered a substitute for professional medical advice, diagnosis, or treatment.

Digital Wellbeing:

Notification and reminder features are designed to encourage healthy habits without overwhelming users. The system avoids excessive notifications and promotes a balanced user experience to reduce digital fatigue and maintain long-term engagement.

3.1.7 Manufacturability (Implementation Feasibility)

The system is developed using widely adopted technologies, including Flutter, Node.js, and MySQL, making it practical to implement and deploy using existing software infrastructure. The platform does not require specialized hardware and can operate on standard web servers and mobile devices, ensuring ease of deployment and accessibility.

3.1.8 Sustainability Constraints

Scalability:

The modular architecture of the platform allows easy expansion and integration of new fitness features, enhanced community functionalities, and additional health-related services without major system modifications. The system also supports further improvements in the existing smart watch integration to extend its capabilities.

Future Upgradability:

The platform supports future enhancements such as more advanced AI capabilities, improved personalization, enhanced

community features, and additional health and wellness services, ensuring long-term sustainability and adaptability.

3.2 Used Technologies

Flutter

Flutter is an open-source UI software development kit developed by Google, used to build natively compiled applications for mobile and web platforms from a single codebase. It uses the Dart programming language and provides a rich set of customizable widgets that enable fast and consistent user interface development.

In this project, Flutter is used to develop the frontend of the Finesse fitness and wellness platform. It allows users to access workout plans, nutrition tracking, AI coaching features, progress monitoring, community posts, and notifications efficiently across multiple platforms.

Why Flutter?

- **Cross-Platform Development:** Enables building Android, iOS, and web applications using a single codebase, reducing development time and effort.
- **High Performance:** Provides smooth and responsive user interfaces through an efficient rendering engine.

- **Hot Reload:** Allows real-time updates during development, improving productivity and speeding up testing.
- **Customizable UI:** Offers flexibility to design modern and interactive interfaces suitable for fitness tracking, AI interaction, and user engagement features such as dashboards and community sections.

Firestore

Firestore is a cloud-based platform developed by Google that provides a set of backend services for web and mobile applications. It supports features such as authentication, real-time databases, cloud messaging, and storage.

In this project, Firestore is used specifically to handle push notifications within the Finesse platform. It enables the system to send real-time alerts and updates to users regarding activities such as reminders, progress updates, and system notifications.

Why Firestore?

- **Real-Time Notifications:** Enables instant delivery of push notifications to users without requiring manual refresh.
- **Cross-Platform Support:** Works seamlessly across Android, iOS, and web platforms.
- **Integration with Flutter:** Easily integrates with the Flutter framework for efficient notification handling.

- **Scalability:** Supports a large number of users and ensures reliable delivery of notifications.

Node. js

Node.js is an open-source, cross-platform JavaScript runtime environment that allows developers to build scalable and efficient server-side applications. It is widely used for developing backend APIs and handling asynchronous operations efficiently.

In this project, Node.js is used to implement the backend API of the Finesse platform. It handles core functionalities such as user authentication, data management, workout and nutrition tracking, AI service integration, and communication between the frontend and backend systems.

Why Node.js?

- **High Performance:** Built on an event-driven, non-blocking I/O model, making it suitable for handling multiple simultaneous requests efficiently.
- **Scalability:** Well-suited for building scalable backend systems that can handle increasing user activity.
- **JavaScript Full Stack:** Allows using the same language (JavaScript) for both frontend and backend development, improving consistency.

- **Large Ecosystem:** Provides a wide range of libraries and packages through npm, accelerating development.
- **Flexibility:** Supports building RESTful APIs and integrating with databases and external services such as AI APIs.

MySQL

MySQL is an open-source relational database management system (RDBMS) that uses structured tables to store and manage data. It is widely used for its reliability, performance, and strong support for structured query operations.

In this project, MySQL is used as the primary database for the Finesse platform. It stores and manages all structured data, including user accounts, authentication details, workout plans, nutrition information, progress tracking records, AI interaction logs, and community posts.

Why MySQL?

- **Relational Structure:** Organizes data into structured tables, ensuring consistency and integrity across the system.
- **Data Integrity:** Supports relationships, constraints, and normalization to maintain accurate and reliable data.
- **Performance:** Efficient in handling complex queries and large datasets typical of fitness tracking systems.

- **Scalability:** Capable of supporting growing amounts of user and activity data.
- **Wide Support:** Easily integrates with Node.js and other backend technologies used in the project.

Visual Studio Code

Visual Studio Code (VS Code) is a lightweight yet powerful source-code editor developed by Microsoft. It provides extensive support for debugging, intelligent code completion (IntelliSense), and built-in Git version control. In this project, Visual Studio Code was used as the main development environment for both the frontend using Flutter and the backend using Node.js.

Why Visual Studio Code?

- **Cross-Platform Support:** Runs on Windows, macOS, and Linux, making it suitable for multi-environment development.
- **Extension Support:** Provides rich extensions for Flutter, Dart, JavaScript, and Node.js, which enhances development speed and productivity.
- **Debugging Capabilities:** Offers advanced debugging tools for both frontend (Flutter) and backend (Node.js) components.
- **Git Integration:** Simplifies version control, collaboration, and code management within the development team.

Postman

Postman is a widely used API development and testing tool that allows developers to create, test, and document RESTful APIs efficiently.

In this project, Postman was used to test backend APIs developed using Node.js, including authentication, workout management, nutrition tracking, AI services integration, notifications, and community features.

Why Postman?

- **API Testing:** Enables quick and effective testing of backend endpoints such as login, data retrieval, AI requests, and progress tracking.
- **Ease of Debugging:** Provides a clear and user-friendly interface for inspecting API requests and responses.
- **Collaboration:** Allows sharing API collections and documentation among team members.
- **Automation Support:** Facilitates automated testing of API responses to ensure reliability and consistency.

Groq AI Integration

Groq was integrated into the Finesse system as the main artificial intelligence provider to support multiple core features within the application. It is used to deliver fast and accurate AI-driven responses with low latency, improving the overall user experience in real-time interactions.

In this project, Groq is utilized in three main functionalities:

1. **AI Coach (Chat Assistant):**

Groq powers the in-app AI Coach that provides personalized fitness and nutrition guidance. The coach analyzes the user profile, daily logs, and progress data to generate tailored recommendations based on the user's goals such as weight loss, weight gain, or maintenance.

2. **Food Image Calorie Estimation:**

Groq vision models are used to analyze food images uploaded by users. The system identifies food items, estimates portion sizes, and calculates approximate calorie values. This feature helps users track their meals automatically without manual entry.

3. **Weekly Performance Summary:**

Groq is also used to generate weekly summaries based on the user's last 7 days of activity. It analyzes trends in calories, water intake, and progress, then provides a concise report with insights and personalized suggestions for improvement.

The system communicates with Groq through secure API requests using an API key stored in environment variables (`GROQ_API_KEY`). Depending on the feature, either text-based chat models or vision models are used.

Why Groq?

- **High Speed Performance:** Enables real-time AI responses with minimal delay.
- **Multi-Feature Support:** Handles chat, image analysis, and data summarization in one system.

- **Context-Aware Responses:** Uses user profile and dashboard data to generate personalized advice.
- **Easy API Integration:** Simple integration with Node.js backend using OpenAI-compatible endpoints.
- **Improved User Experience:** Provides intelligent, interactive, and automated fitness assistance.

3.3 Finesse as a Mobile and Web Application

3.3.1 Splash Screen

The Splash Screen is displayed when launching the Finesse mobile application only. It provides an introductory animation that represents the application branding while performing initial loading operations.

In this project, the Splash Screen includes animated elements such as the application logo, feature highlights (Workouts, Nutrition, and Progress), and a loading indicator. These animations are implemented using Flutter animation controllers to enhance user experience and create a smooth visual transition.

During this screen, the system checks the user's onboarding status using `SharedPreferences` and navigates the user either to the Login Screen or the Onboarding Screen after a short delay.

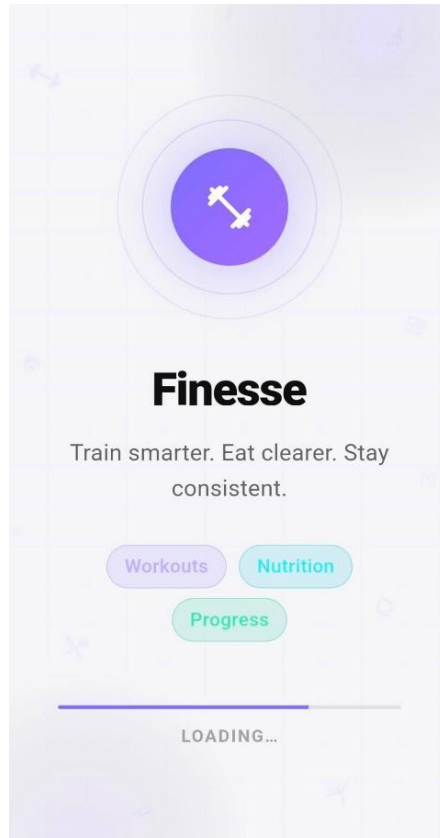


Figure 1: Splash screen

3.3.2 Onboarding Screens

The onboarding screens are displayed to introduce new users to the core features of the Finesse application before accessing the authentication process. The onboarding flow consists of three main screens, each focusing on a key functionality of the system.

This screen introduces the weight tracking feature of the application. It explains how users can log their weight measurements over time and view progress trends to monitor their fitness journey and achieve their goals.

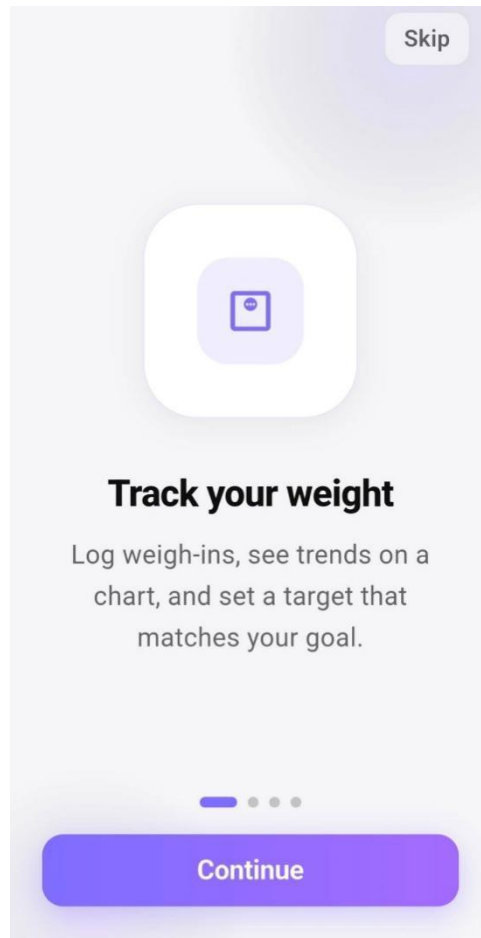


Figure 2:Track your weight Screen

This screen presents the nutrition management feature. It highlights how the system provides personalized meal plans aligned with the user's goals and allows users to track their calorie intake easily.

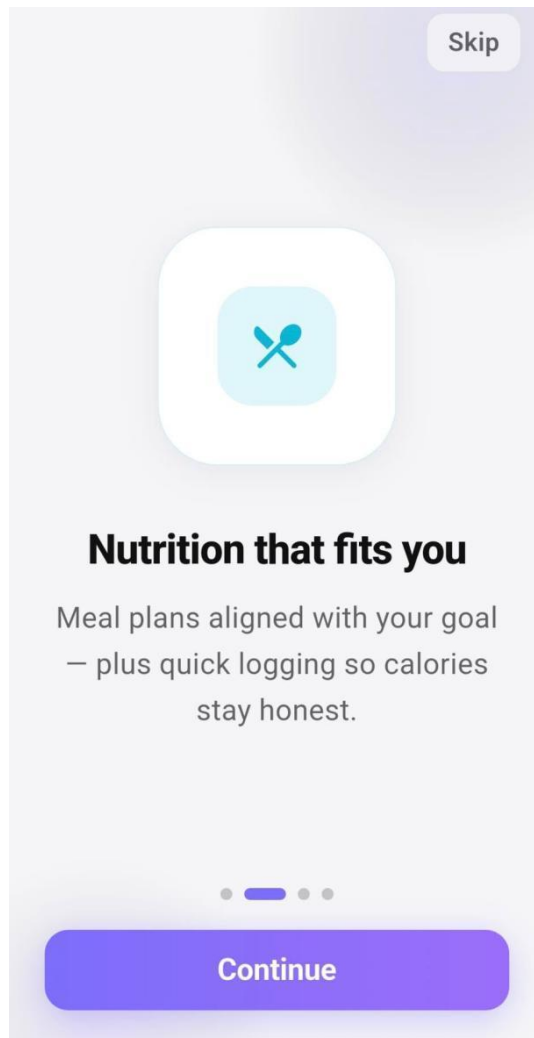


Figure 3: Nutrition that fits you Screen

This screen focuses on workout sessions and hydration tracking. It explains how users can follow goal-based workout plans and monitor their water intake to maintain healthy habits.

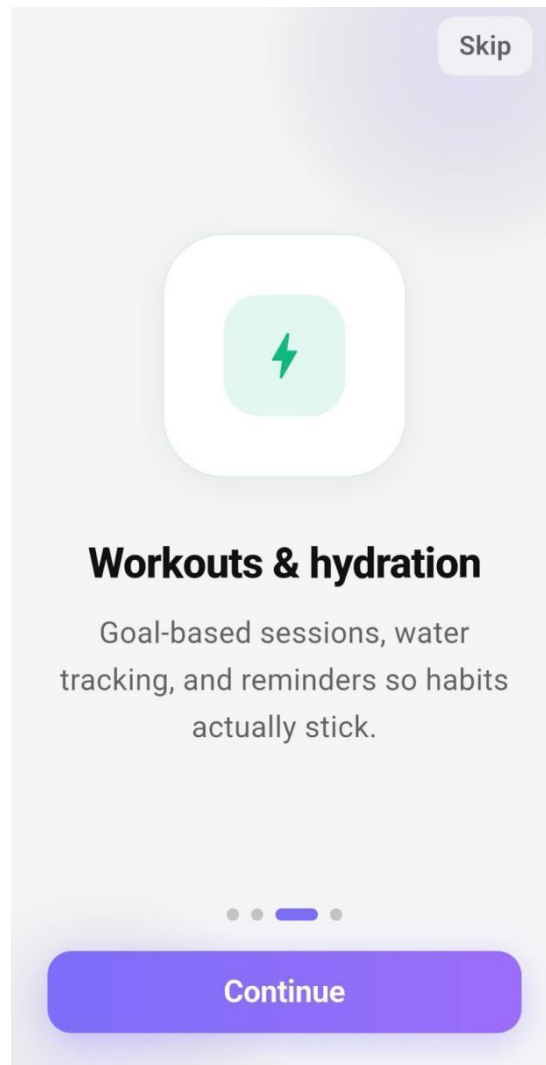


Figure 4: Workouts & hydration Screen

This screen introduces the AI fitness assistant feature. It explains how users can interact with an AI coach to ask questions related to workouts, nutrition, and motivation, with responses tailored to their profile.

After completing or skipping the onboarding process, the system stores the completion status using `SharedPreferences` and navigates the user to the Login Screen.

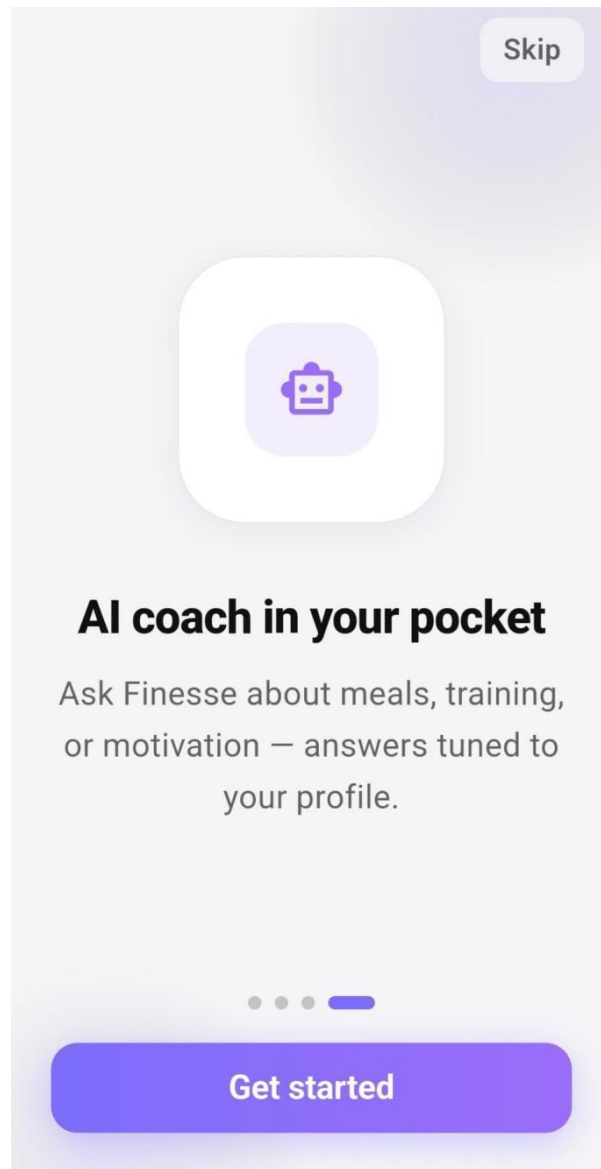


Figure 5: AI coach in your pocket Screen

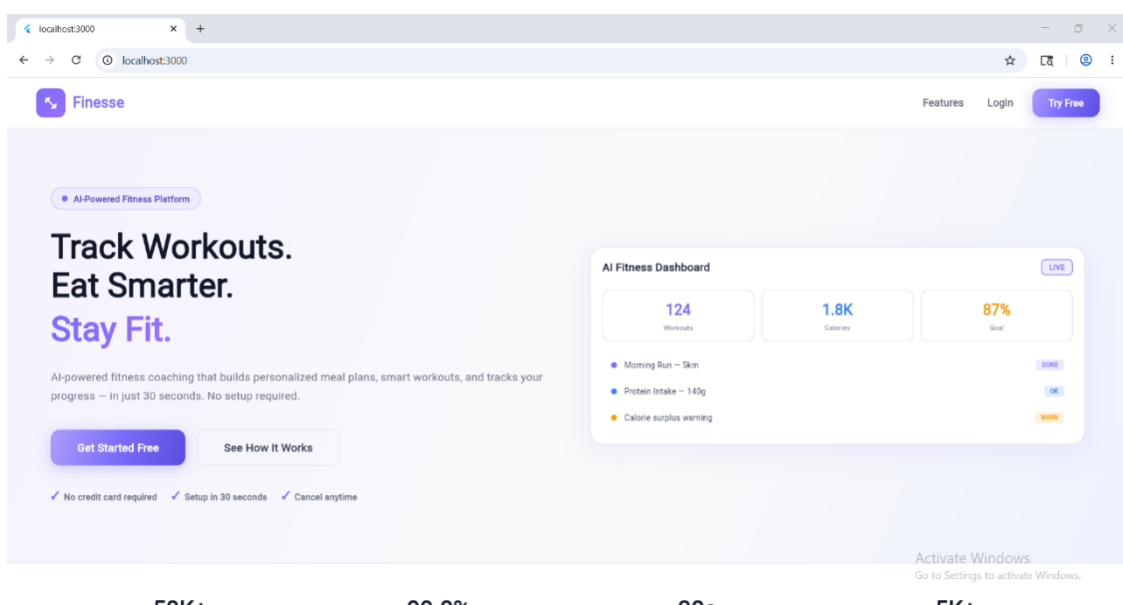
3.3.3 Landing Page (Web Application)

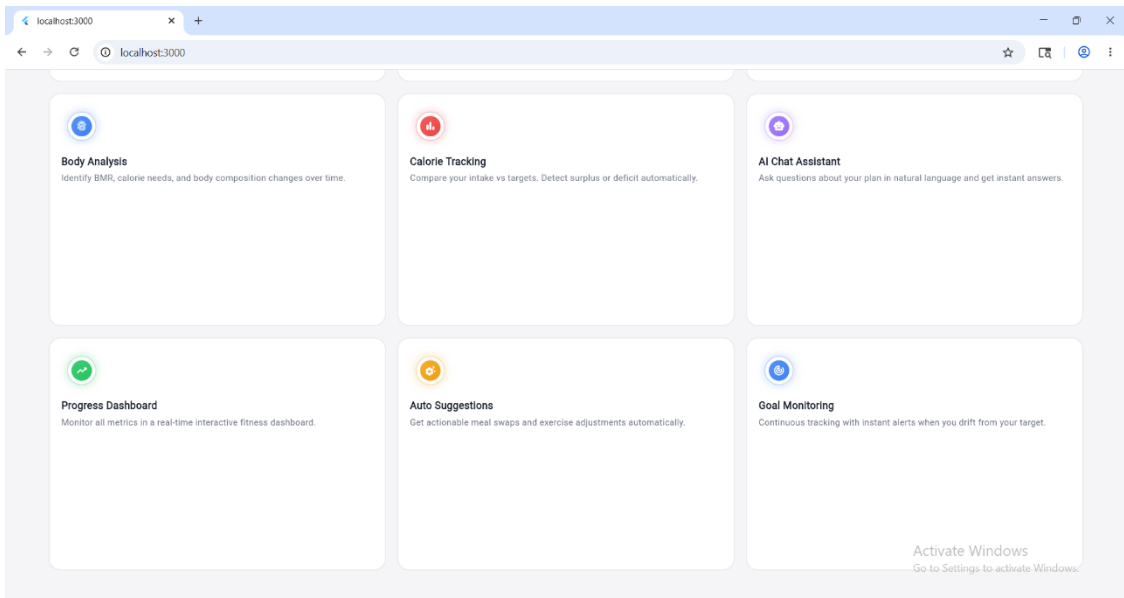
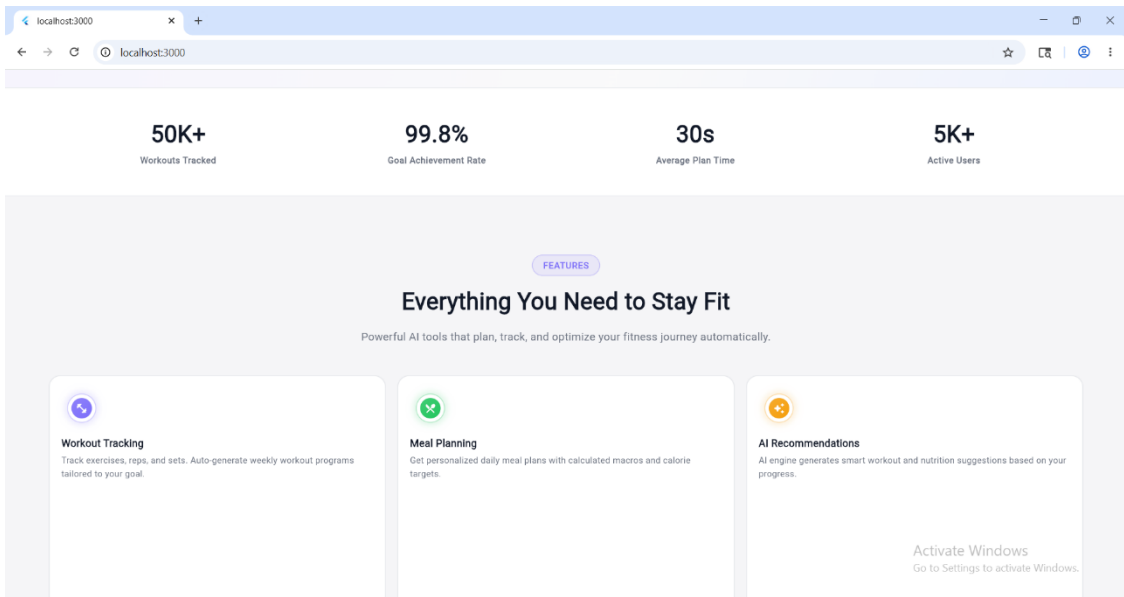
The landing page represents the web entry point of the Finesse platform. It is designed to introduce users to the system and encourage them to register or log in.

It includes several sections such as a navigation bar, hero section, statistics display, features overview, call-to-action section, and footer. The layout is responsive and adapts to different screen sizes.

The hero section highlights the core value of the platform, focusing on AI-powered fitness tracking, personalized meal planning, and smart workout recommendations. The features section presents the main functionalities of the system such as workout tracking, nutrition planning, AI recommendations, body analysis, and progress monitoring.

The landing page serves as a marketing and onboarding interface for web users, while the mobile application handles daily usage and tracking functionalities.





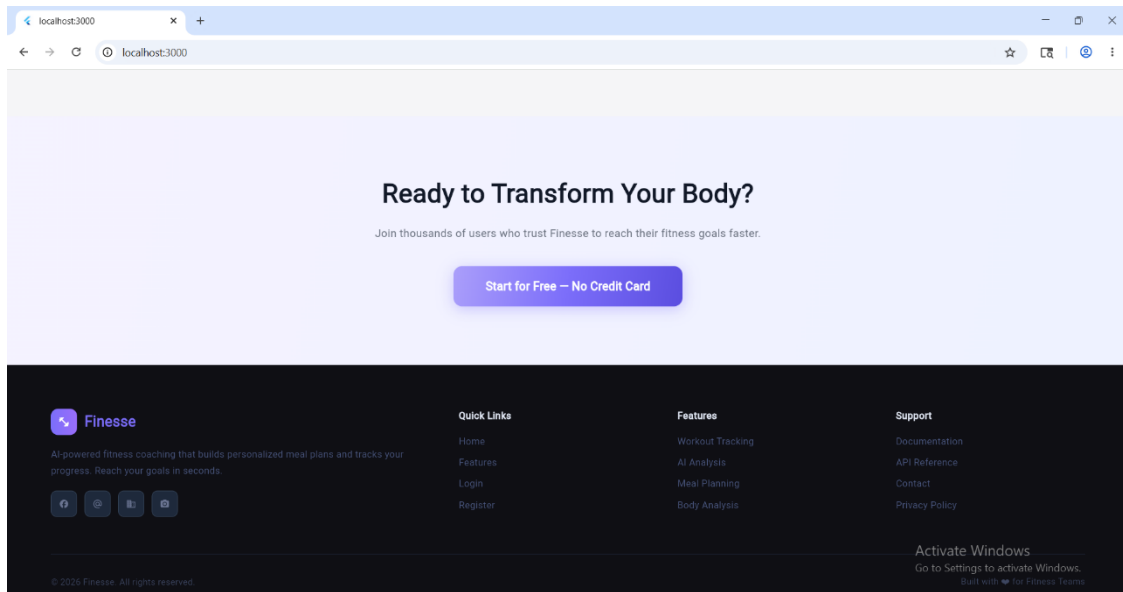


Figure 6: Landing Screens

3.3.4 Authentication and Validation

When users access the Finesse platform, they are presented with a secure authentication interface that allows them to register, log in, and manage their accounts. Authentication mechanisms are implemented to protect user data and ensure that only authorized users can access personalized fitness information and application services.

During registration, users are required to provide their basic information, including their name, email address, and password. Input validation is applied to ensure that the provided email follows a valid format and that passwords satisfy the required security constraints.

The platform supports two authentication methods:

- Traditional email and password authentication.
- Google Sign-In authentication, allowing users to securely access the application using their Google accounts.

For users who forget their passwords, the system provides a password recovery mechanism. A password reset request can be submitted through the "Forgot Password" feature, which sends a verification code or reset token to the user's registered email address. After successful verification, users can create a new password and regain access to their accounts.

After successful authentication, a JSON Web Token (JWT) is generated and used to maintain secure user sessions. The system also supports session persistence through a "Remember Me" option, allowing users to save their login information for future access.

Once authenticated, the platform retrieves the user's profile information. New users who have not completed their profile setup are redirected to the onboarding interface, where personal fitness information such as age, gender, weight, height, and fitness goals can be provided. Users with completed profiles are directed directly to the main dashboard.

These authentication and validation mechanisms ensure secure access control, protect user information, and provide a reliable user experience throughout the Finesse platform.

3.3.5 Sign-Up

To access the Finesse platform, users are required to create an account before using the application's features. The registration process is designed to collect essential user information while ensuring data validity and account security.

During the sign-up process, users are required to provide the following information:

- Full Name
- Email Address
- Password

The system validates all entered information before account creation. The email address must follow a valid email format, while the password must satisfy the required security constraints. The platform also checks for duplicate email addresses to prevent multiple accounts from being registered using the same email.

After successful registration, the user account is created and stored securely in the system database. Users can then proceed to sign in and continue the onboarding process, where additional personal and fitness-related information is collected to generate personalized workout plans, nutrition recommendations, and fitness goals.

This registration mechanism ensures secure access to the platform while providing the necessary foundation for delivering personalized fitness services.

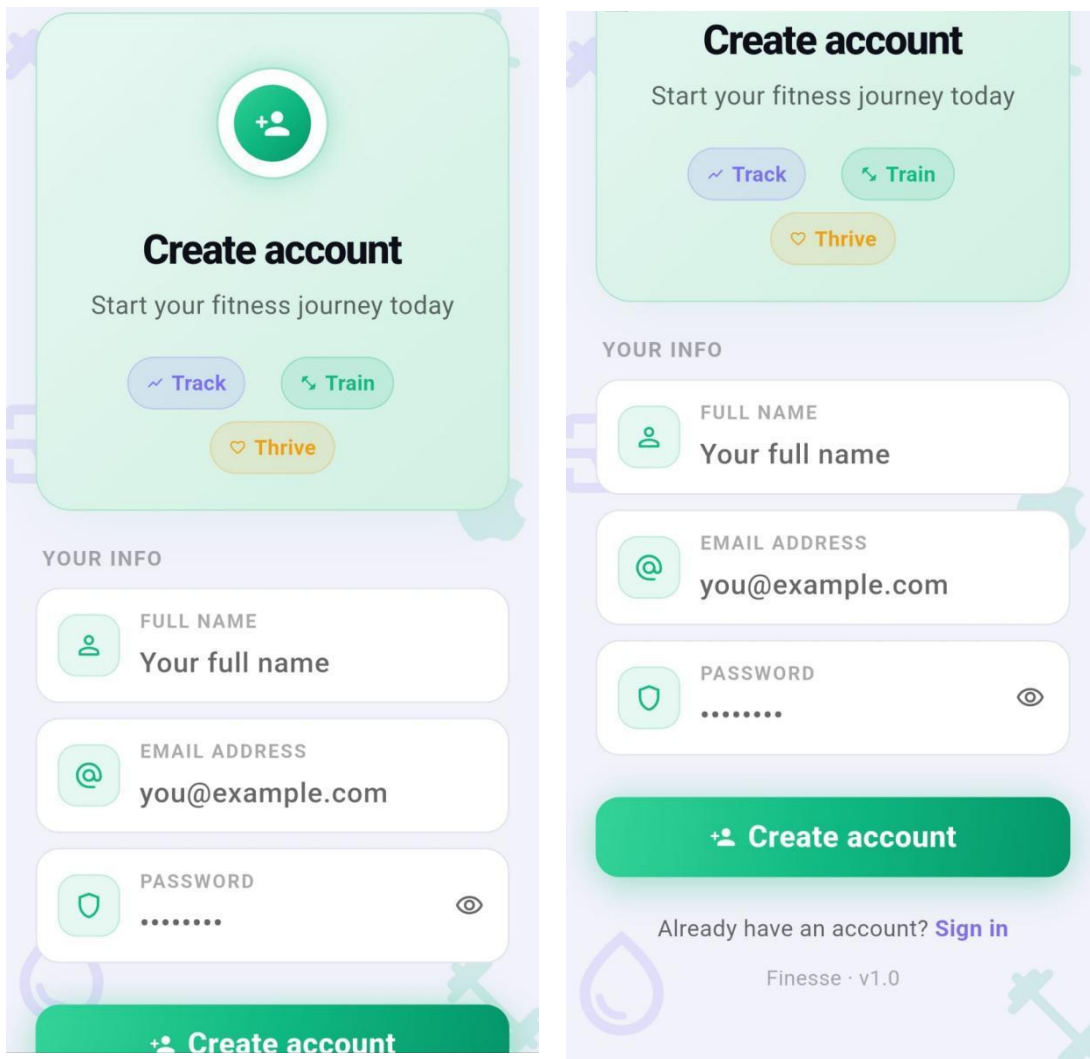


Figure 7: Sign Up Screen

3.3.6 Login

The Login page serves as the main gateway to the Finesse platform, allowing users to securely access their personal fitness accounts. The interface was designed to provide a modern, user-friendly, and seamless authentication experience.

Users can log in by entering their registered email address and password. The page also includes additional features that enhance convenience and

accessibility, such as password visibility control, email autofill support, and session persistence through the Remember Me option.

Upon successful authentication, the system automatically determines whether the user's profile information is complete. Users with completed profiles are redirected to the Home Dashboard, while new users are guided through the onboarding process to provide the required personal information before accessing the platform.

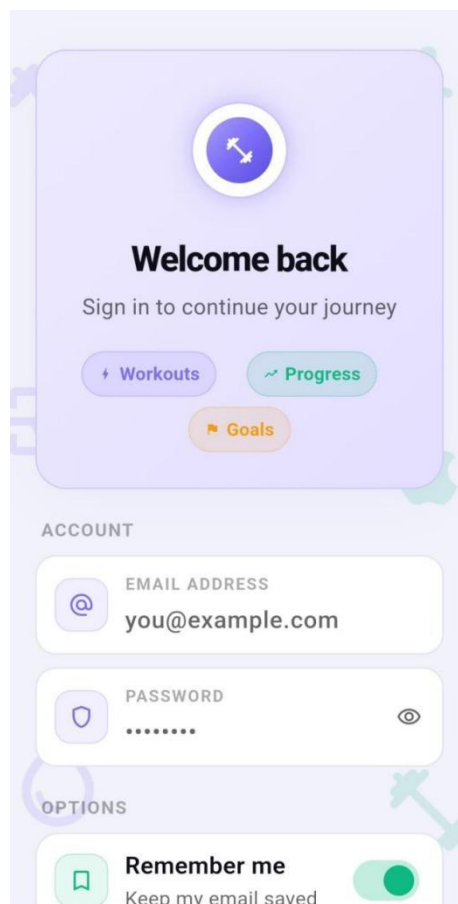


Figure 8: Login Screen

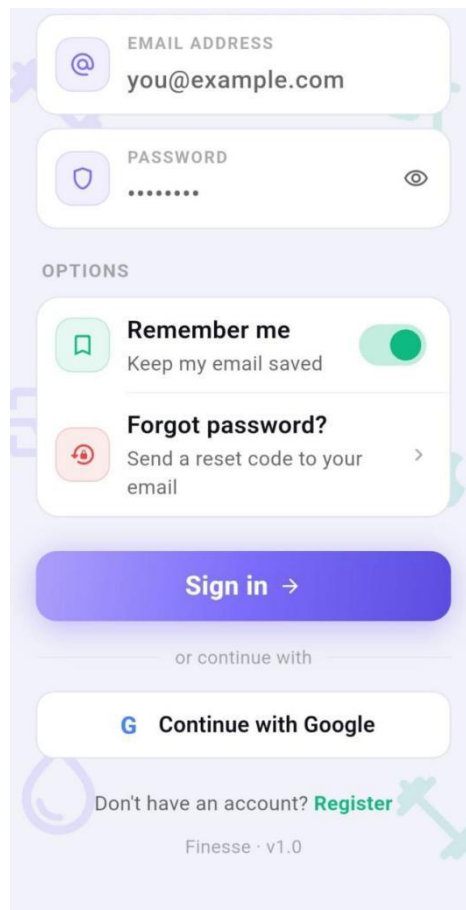


Figure 9: Login Screen

3.3.7 Login Verification and Error Handling

Finesse implements a comprehensive authentication and validation mechanism to ensure secure access and provide immediate feedback to users.

Before sending login credentials to the server, the application performs client-side validation to verify that:

- The email field is not empty.
- The email follows a valid format.
- The password field is not empty.

- The password satisfies the minimum length requirements.

After validation, the entered credentials are securely verified against the database. If authentication is successful, a secure token is generated and stored to maintain the user's session.

In cases where authentication fails, the system displays clear and informative error messages, helping users quickly identify and resolve issues. Examples include invalid email format, missing fields, incorrect passwords, or non-existing accounts.

To further improve user experience, the login page provides:

Remember Me functionality to save the user's email for future logins.

Forgot Password functionality for password recovery through email verification.

Google Sign-In integration, allowing users to authenticate quickly using their Google accounts.

These features enhance both security and usability, providing users with a reliable and efficient authentication experience while maintaining the integrity of their personal data.

3.3.8 Google Sign-In Authentication

The Finesse platform provides Google Sign-In as an alternative authentication method to the traditional email and password login, aiming to simplify the login process and improve user experience.

This feature is implemented using the `google_sign_in` package integrated with Firebase Authentication. When the user selects “Continue with

Google,” the Google authentication flow is initiated and basic authentication data such as `idToken` and `accessToken` is retrieved.

These credentials are then used through Firebase to establish a secure user session. After authentication, the Firebase ID Token is sent to the backend via a POST request to `/api/auth/google-login` for verification. The backend either creates a new user account or links the user to an existing account and returns a JWT token for the application.

The authentication token is stored locally using `SharedPreferences` through the `AuthPrefs` service to maintain the user session.

After successful login, user profile data is retrieved to determine whether the profile is complete. Based on this, the user is redirected either to the Home Dashboard or to the User Information screen to complete onboarding.

The system supports multiple platforms by using a `clientId` for web applications, while mobile platforms rely on Firebase configuration for authentication.

In case of login failure or user cancellation, an error message is displayed using a Snackbar notification.

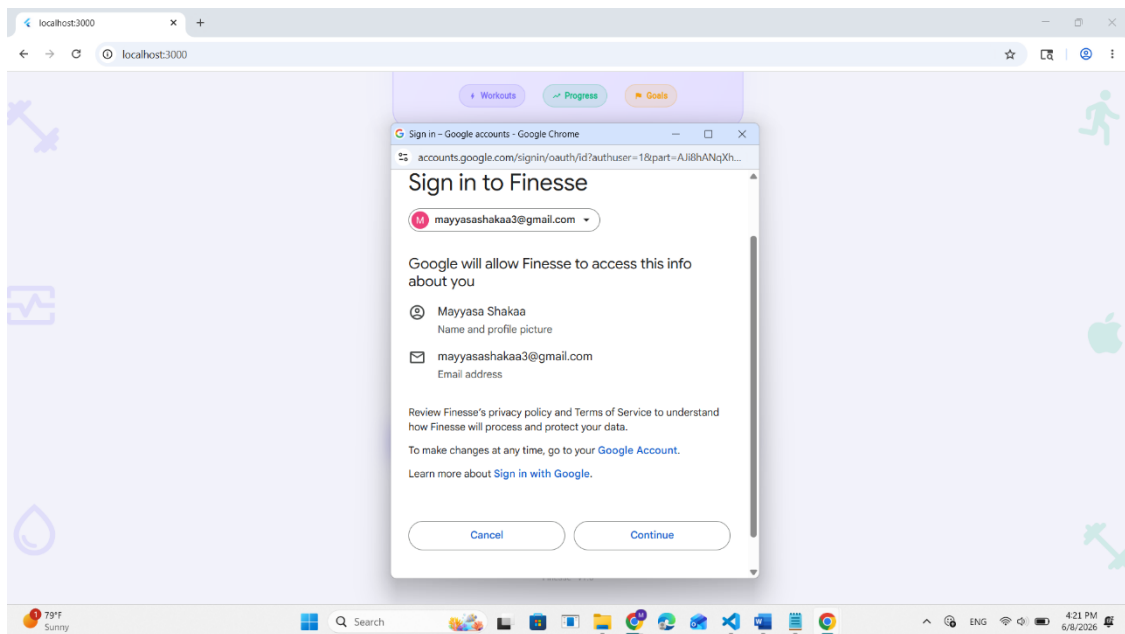
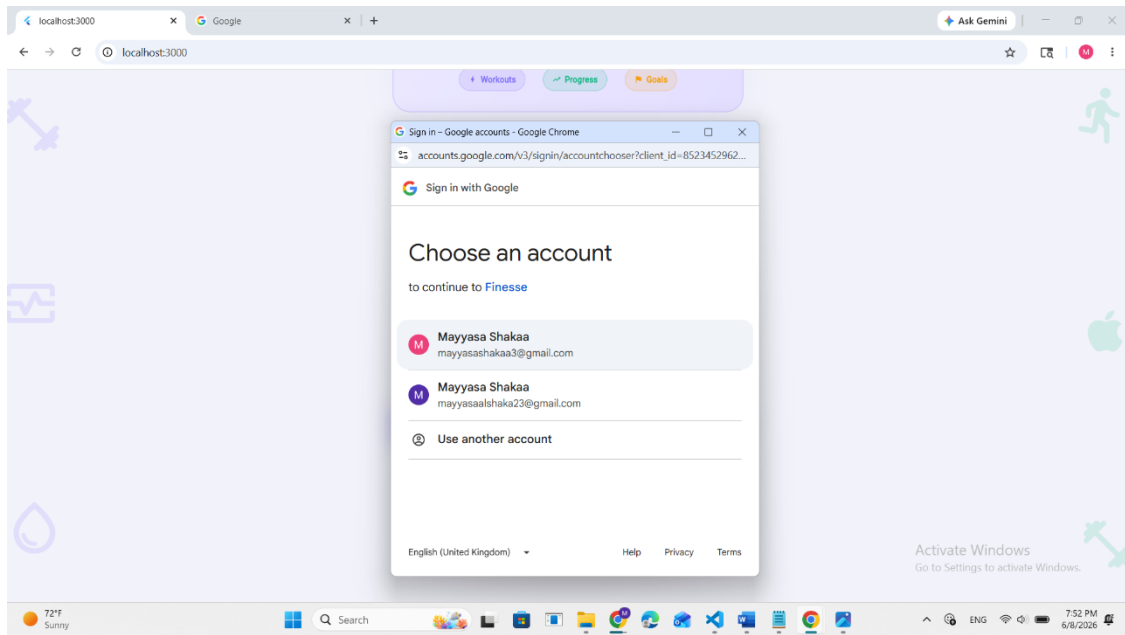


Figure 10: Google Sign-In Screen

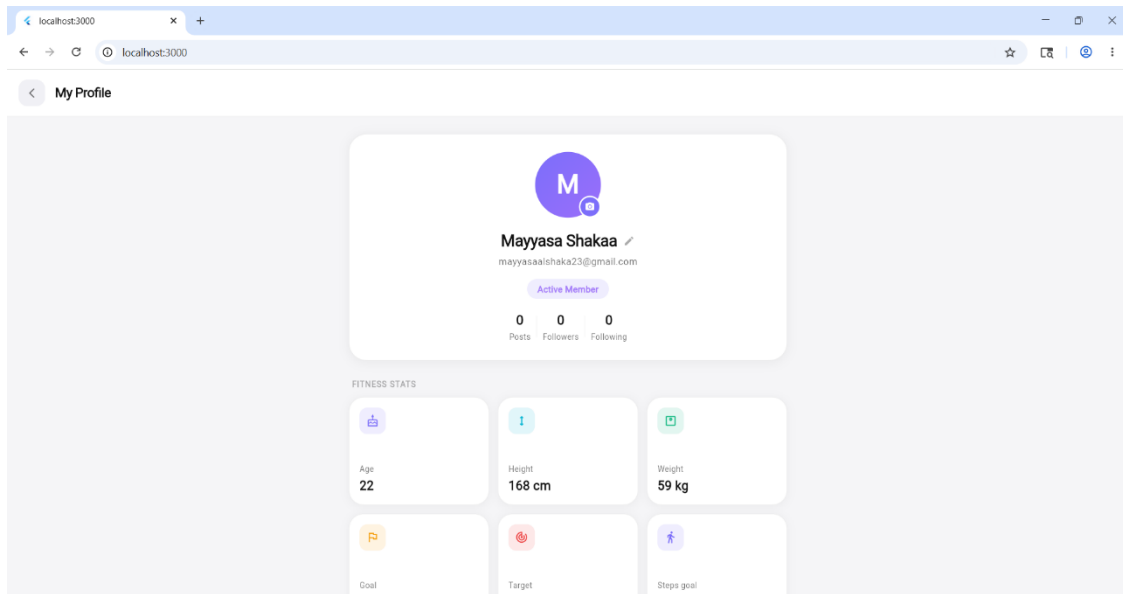


Figure 11: Profile Screen with Google Sign-In

3.3.9 Forgot Password

The Finesse system provides a secure and efficient password recovery mechanism to assist users who are unable to access their accounts. When a user selects the *Forgot Password* option from the login screen, they are prompted to enter the email address associated with their account.

Upon submission, the system sends a password reset request to the backend API endpoint (`/api/auth/request-password-reset`). If the request is successful, the server generates a temporary reset token and sends a 6-digit verification code to the user's email.

The application then navigates the user to the *Reset Password Screen*, where they are required to enter the verification code received via email. This step ensures that only the legitimate account owner can proceed with the password reset process.

For better usability, the system provides real-time feedback using SnackBars to inform the user about missing fields, invalid input, or request status. This improves user experience and reduces errors during the recovery process.

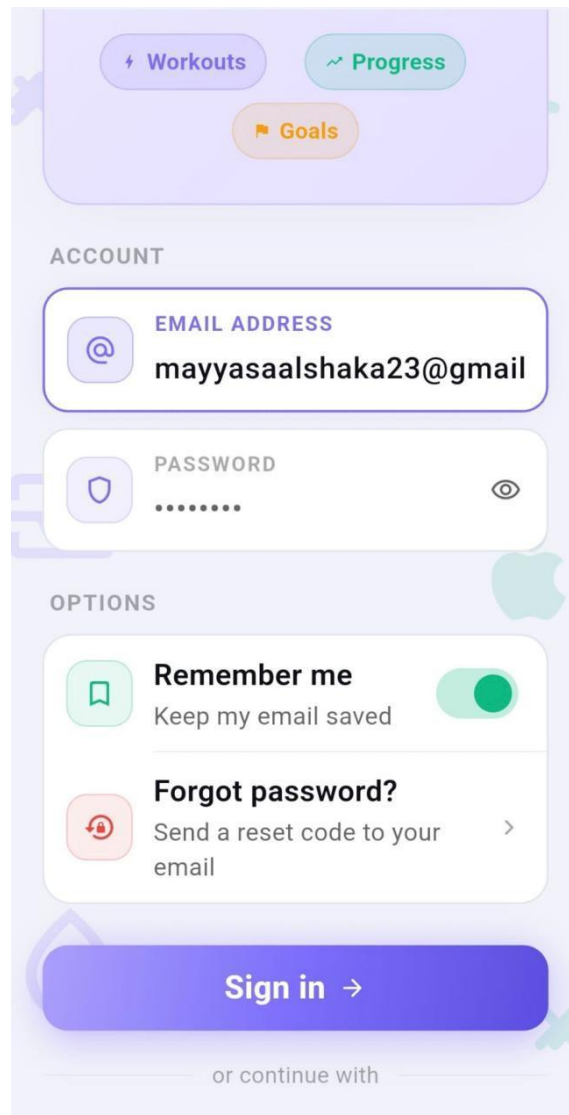


Figure 12: Forgot Password

3.3.10 Reset Password

The Reset Password feature allows users to securely create a new password after verifying their identity. The screen requires the user to enter the verification code sent to their email, along with a new password and confirmation password.

Before submitting the request, the application performs client-side validation to ensure that:

- All fields are filled.
- The new password and confirmation password match.

After validation, the system sends a POST request to the backend endpoint (`/api/auth/reset-password`) including the user's email, reset token, verification code, and the new password.

If the server validates the code and token successfully, the password is updated in the database, and the user is redirected back to the Login screen. A success message is displayed confirming that the password has been changed successfully.

In case of any error (invalid code, expired token, or incorrect input), the system provides clear error messages to guide the user in resolving the issue.

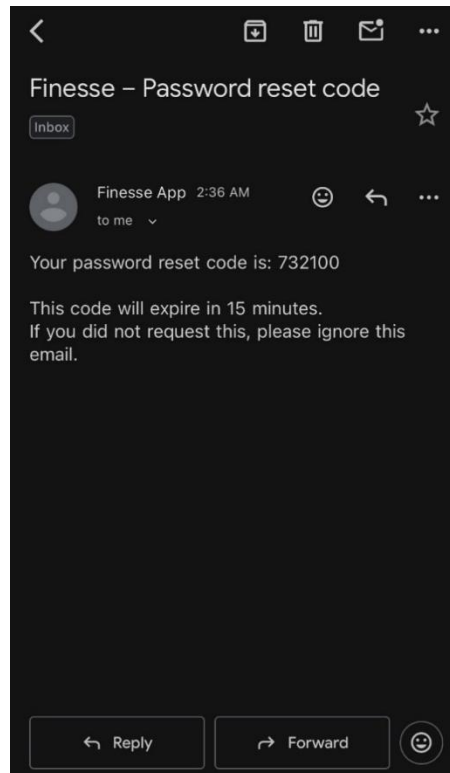


Figure 13: Reset Password Email

<

Reset Password



Reset your password
Enter the 6-digit code sent to mayyasaalshaka23@gmail.com

RESET CODE

#

Verification code
Sent to mayyasaalshaka23@gmail.com

732100

NEW PASSWORD



New password
Min. 8 chars, uppercase, numbers & symbols

123456Mm!



Confirm password

Figure 14: Reset Password Page

< Reset Password

732100

NEW PASSWORD

New password
Min. 8 chars, uppercase, numbers & symbols

123456Mm!

Confirm password
Repeat your new password

123456Mm!

Change password

[Back to login](#)

Finesse - v1.0

Figure 15: Reset Password Page

3.3.11 User Information and Fitness Profile Management

The Finesse platform includes a User Information screen responsible for collecting and managing the user's fitness and health profile. This module is essential for enabling personalization, as all AI recommendations and tracking features rely on accurate user data.

The screen is implemented as a stateful Flutter component and supports both onboarding and profile editing modes. During onboarding, the user completes their profile before accessing the main application, while in edit mode existing data is retrieved from the backend and pre-filled into the form.

User data is fetched via a GET request to `/api/auth/getUserData` and mapped into input fields such as age, height, weight, gender, goal type, and fitness preferences.

The user can update key attributes including age, height, weight, target weight, gender, daily step goal, fitness goal (lose, gain, or maintain weight), and goal pace. Optional inputs such as target weight and goal date are also supported.

The interface uses interactive UI components such as toggle chips for gender and pace selection, selectable goal items with icons, and predefined step goal options for quick input.

Input validation is applied before submission to ensure realistic values for age, height, and weight. Errors are displayed using Snackbar messages.

On submission, the data is sent via a POST request to `/api/auth/updateInfo`, including all profile fields. The request is authenticated using a JWT token in the Authorization header.

The backend stores the data and uses it to calculate derived values such as calorie needs and fitness targets, which are later used in meal planning, workouts, and progress tracking.

After a successful update, the user is redirected to the Home Dashboard (onboarding) or returned to the previous screen (profile editing).

The screen also handles loading states, error handling, and responsive design for web and mobile platforms.

Overall, this module forms the core of personalization in the Finesse system by ensuring all recommendations are based on accurate user data.

3.4 Profile Management

The User Profile screen serves as the central hub for managing personal information, fitness goals, and community engagement within the Finesse platform. It provides users with a comprehensive overview of their profile while allowing them to update personal data and interact with other members of the fitness community.

The screen displays essential profile information, including the profile picture, full name, username, fitness goal, and personal fitness statistics. Users can manage and update important details such as age, height, weight, gender, target weight, goal pace, target date, and daily step goals. These attributes are used throughout the platform to personalize workout plans, meal recommendations, calorie calculations, and overall fitness guidance.

The profile also incorporates social networking features that enhance user engagement. Users can view the number of posts they have shared, as well as their followers and following counts. These statistics provide insight into user activity and participation within the community.

Additionally, users can create and publish fitness-related posts to share their progress, achievements, workout experiences, and healthy lifestyle updates. Published posts appear on the user's profile and can be viewed by other members. The platform also allows users to follow other members, helping them build a supportive fitness network and stay updated with community activities.

Before profile information is saved, the system validates user inputs to ensure data accuracy and consistency. Updated information is securely transmitted to the backend server and stored in the database for future use.

The screen supports both first-time onboarding and profile editing. New users are required to complete their profile before accessing the application, while existing users can revisit the profile at any time to update their information and customize their account settings.

Overall, the User Profile screen combines personalized fitness management with social interaction features, creating a more engaging and motivating experience while enabling the platform to deliver accurate recommendations based on each user's goals and physical characteristics.

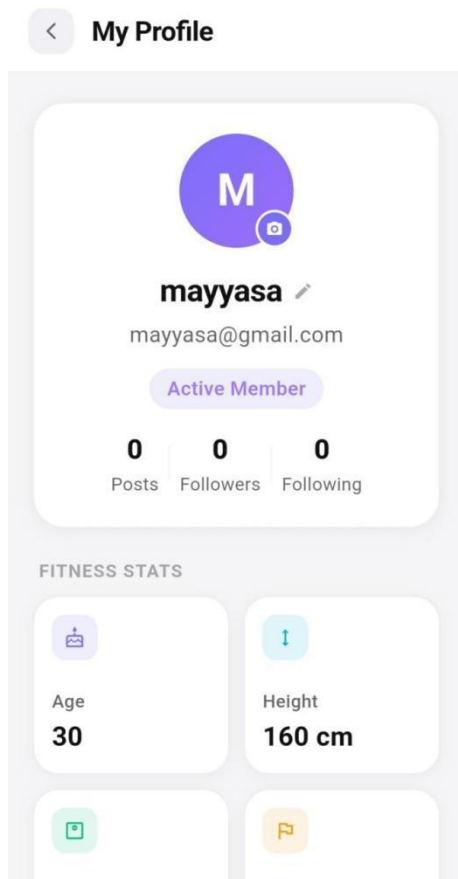


Figure 18: Profile Screen

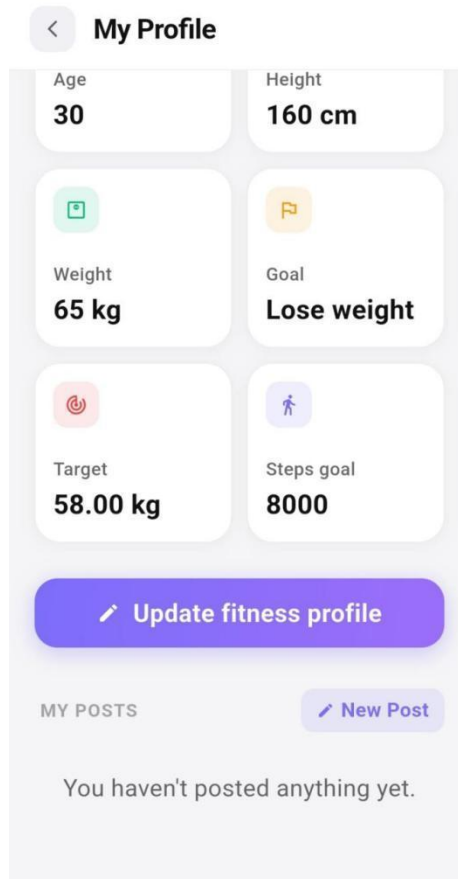


Figure 19: Profile Screen

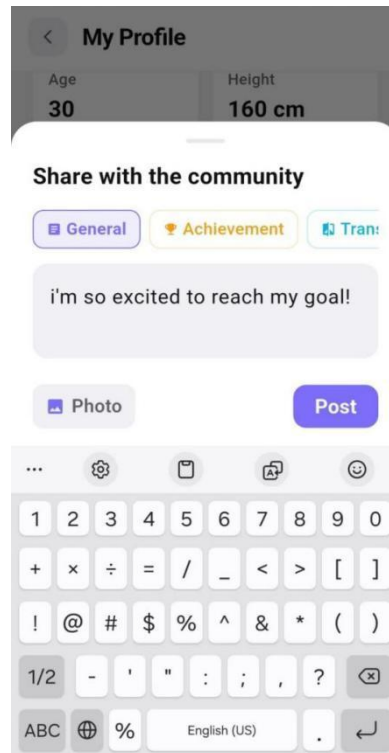


Figure 20: Add Post Screen

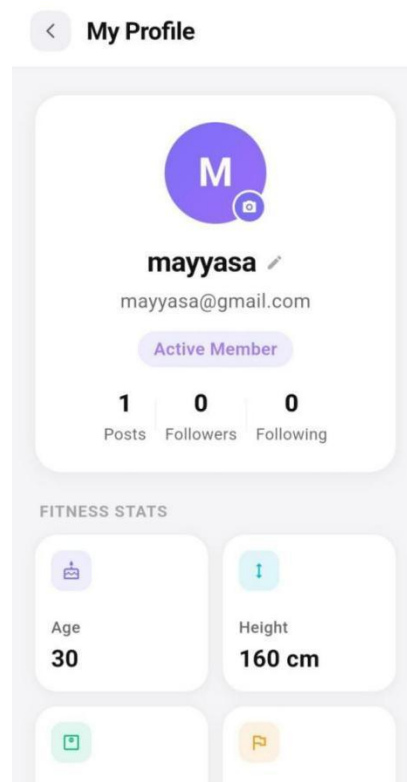


Figure 21: Add Post Screen

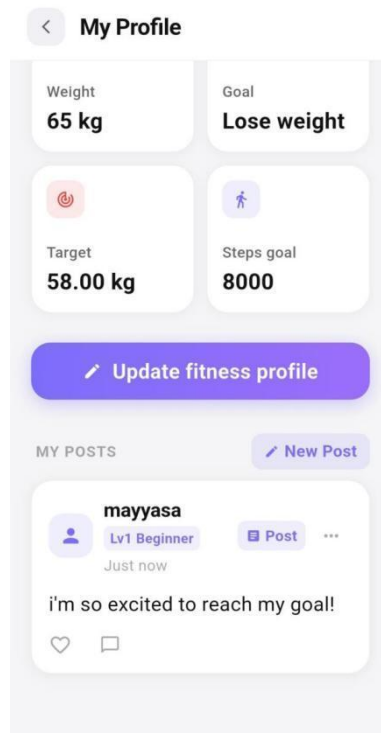


Figure 22: Add Post Screen

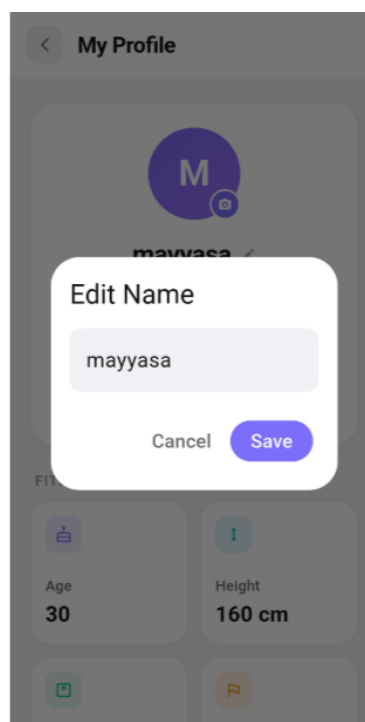


Figure 23: Edit name Screen

3.5 Settings Management

The Settings screen provides users with a centralized interface for managing application preferences, account options, and personalization settings within the Finesse platform.

The screen displays a profile summary card containing the user's profile picture, name, email address, and account role. Profile information is retrieved from the backend and updated dynamically whenever changes are made to the user's profile.

The settings module allows users to customize the application appearance through theme management options. Users can enable or disable Dark Mode or choose to follow the device's system theme settings. These preferences are stored locally and applied throughout the application.

The platform also supports language preference management, allowing users to select their preferred language. The selected language setting is saved locally for future localization support.

Additional application features are accessible through dedicated settings options, including Notifications and Reminders management, Profile access, Support Chat, and subscription-related services such as Finesse Pro. Administrative users are provided with additional shortcuts to role-specific management dashboards.

The screen includes secure account management functionality through a logout option. Before signing out, the system displays a confirmation dialog to prevent accidental logout. Once confirmed, authentication data is cleared from local storage and the user is redirected to the login screen.

Overall, the Settings screen improves user experience by providing a centralized location for personalization, account management, accessibility settings, and application preferences.

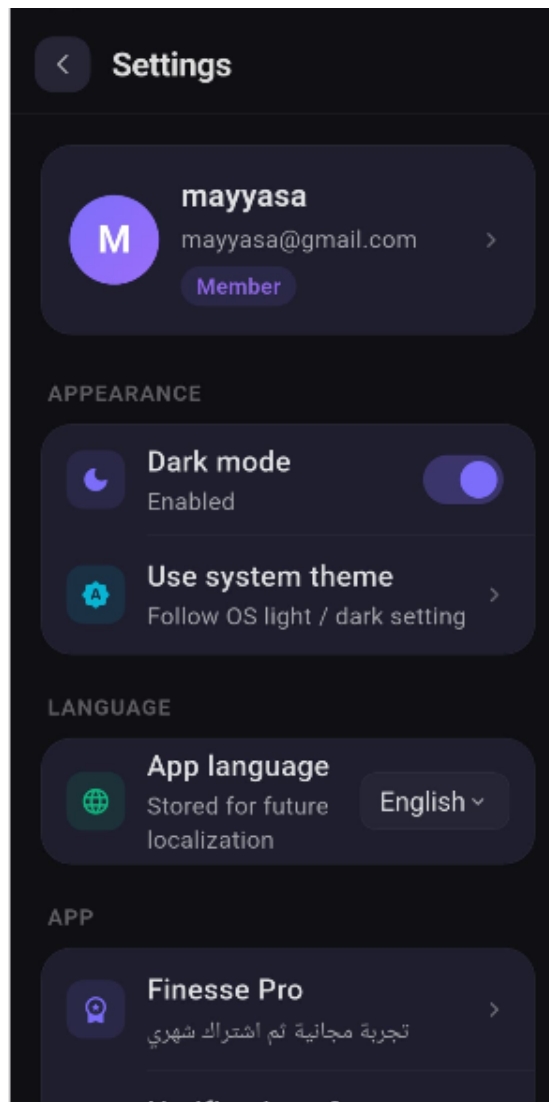


Figure 24: Dark mode Screen

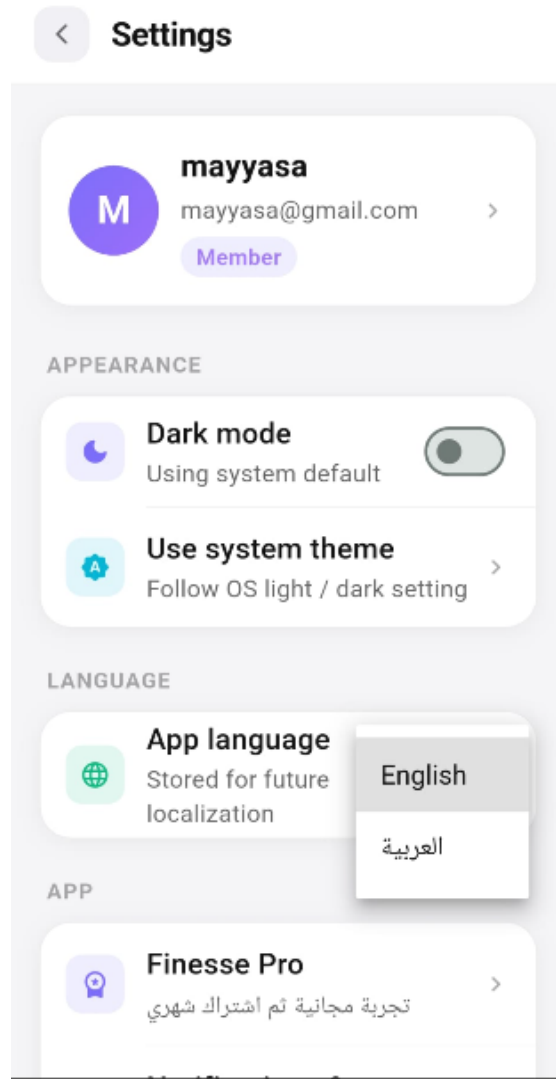


Figure 25: App language Screen

3.5.1 Finesse Pro and Subscription Management

This section keeps the user informed about their account status and handles everything related to the premium Finesse Pro subscription. It manages the onboarding, pricing plans, and checkout operations, ensuring that premium tools like the Smart AI Coach and specialized workout generators are safely unlocked for active subscribers.

To help new users explore the platform, the app automatically gives everyone a 30-day Free Pro Trial upon registration. The screen displays a clear countdown showing exactly how many days are left in the trial, alongside helpful reminders encouraging users to upgrade before their free period runs out.

Once the trial expires, the screen changes to show a clear plan selector where users can easily customize their subscription by choosing between 1, 3, 6, or 12-month packages. The app automatically calculates the total price based on the selected duration so the user knows exactly what they will pay.

The interface features a clean premium design, using prominent badges (like a special gold premium icon) to show active statuses, clear active plan detail cards, and a direct checkout button. There is also a temporary "Enable Trial" button available in developer mode to let the team test the system without using real payment cards.

When a user decides to upgrade and clicks the checkout button, the app securely sends their choice to the backend, which talks directly to the Stripe Payment Gateway. The app then opens a safe, external browser window where the user can enter their card details securely. Once the payment succeeds, the screen instantly updates with a premium badge and unlocks all the premium features across the app without requiring a manual restart.

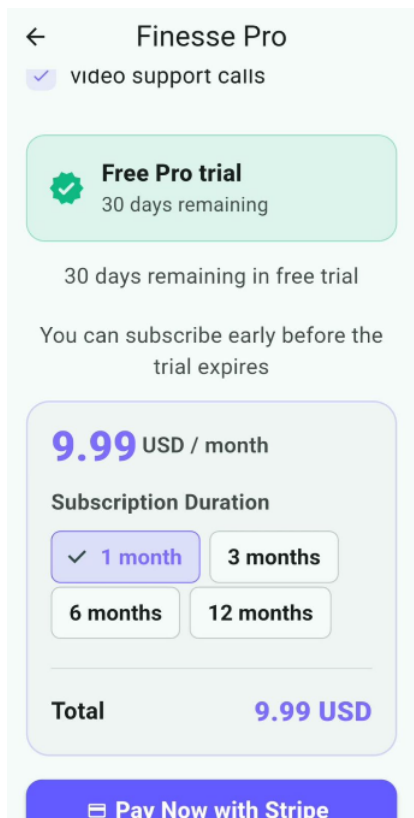


Figure 26: Finesse Pro Screen

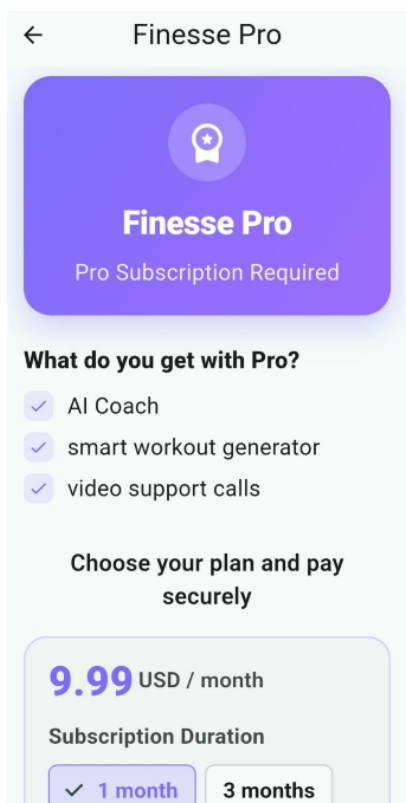


Figure 27: Finesse Pro Screen

checkout.stripe.com

1 USD = 3,0701 IL\$

الدفع باستخدام link

OR

Emailmayyasa@gmail.com

Payment method

Card

Card information

4242 4242 4242 4242

09 / 29123

Cardholder name

MAYYASA SHAKAA

Country or region

Palestinian Territories

☐ Save my information for faster checkout

Pay securely at Finesse sandbox and everywhere Link is accepted.

Pay

Powered by Stripe | Terms | Privacy

Figure 28: Pay for the Pro screen

<

Conversation #11

Status: OPEN

button to join.

Join video call

You

Video call

Join: <https://meet.jit.si/finesse-support-11-03661ce7e6114f2a>

Tap the link or use the video button to join.

Join video call

Coach

keep up with what we have discussed and i will check on you regularly

You

Ok thank you coach

Pro subscription required

Figure 29: End of trial period notification

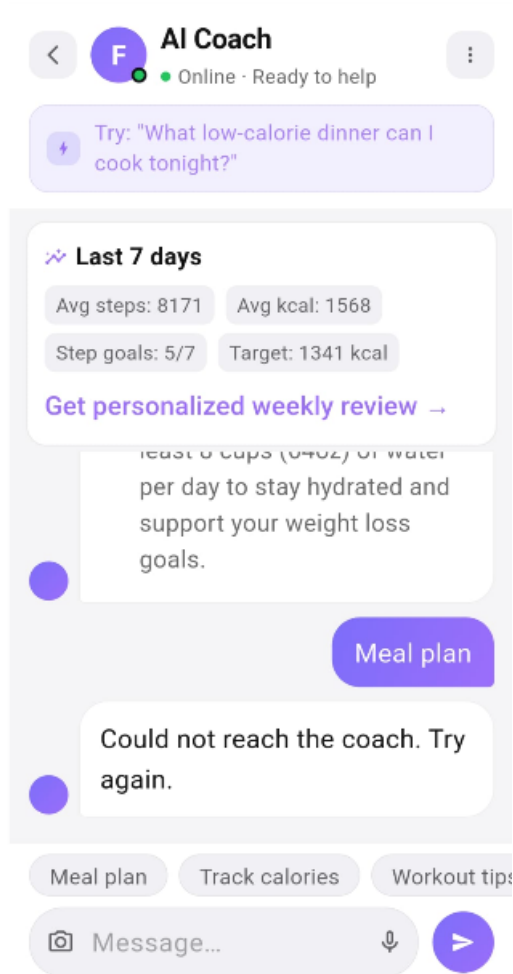


Figure 30: End of trial period notification

3.5.2 Notifications and Reminders Management

This section lets users customize their daily and weekly health reminders to help them stay on track with their fitness goals. Instead of relying on a constant internet connection or heavy server updates, the app sets up local alarms directly on the user's phone, which saves battery and ensures reminders work perfectly even when offline.

To keep the setup simple and easy to manage, the system categorizes reminders into four essential health habits:

1. **Hydration Tracking:** Reminds users to drink water throughout the day, defaulting to helpful reminders at 9:00 AM, 1:00 PM, and 6:00 PM.
2. **Meal Logging:** Encourages users to log their food intake, with a default reminder set at 12:30 PM.
3. **Physical Workout:** Prompts users to start their daily exercise, defaulting to 5:00 PM.
4. **Weight Assessment:** Sets up a weekly check-in reminder every Sunday morning at 10:00 AM to help track weight changes over time.

The screen features an intuitive settings page where users can use simple toggle switches to turn specific reminders on or off and tap to change the exact time they want to be alerted.

When a user saves their changes, the app handles everything smoothly in the background. It first asks the phone for permission to show notifications if it hasn't already. Once allowed, it automatically clears out any old alarms, saves the new times to the phone's built-in memory, and schedules the new reminders. The smart timing system checks the current clock to ensure it never sends an alert for a time that has already passed, automatically moving it to the next day or week. Finally, a small confirmation message pops up at the bottom of the screen to let the user know their new reminder schedule has been successfully set.

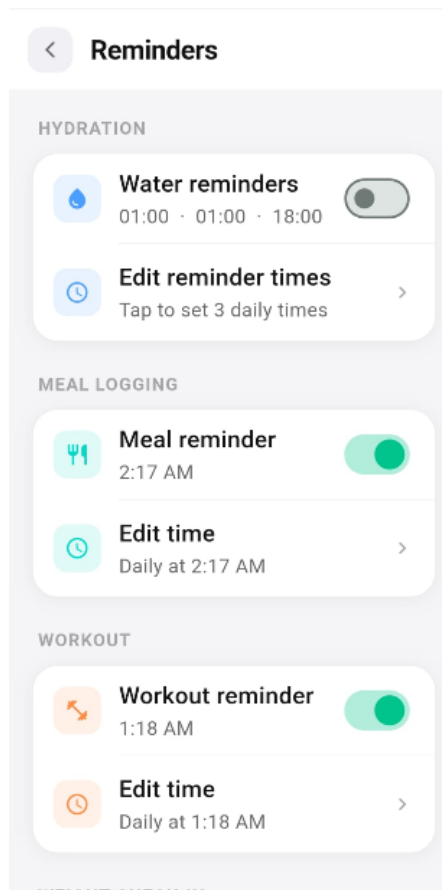


Figure 31: Reminders screen

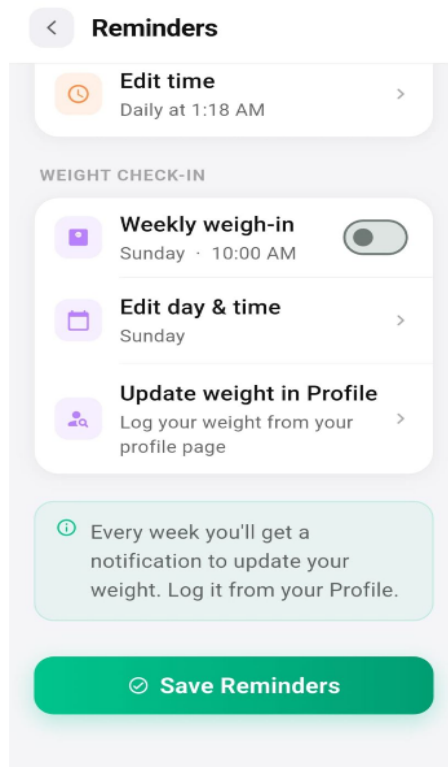


Figure 32: Reminders screen

3.5.3 Support Chat & Multi-Role Communication Management

The Finesse platform incorporates a specialized communication subsystem (SupportChatScreen and SupportChatApi) designed to facilitate multi-role asynchronous data threading and synchronous real-time media streams.

This subsystem acts as a secure client-side gateway, allowing users to address inquiries to specific professional domains within the platform ecosystem.

3.5.3.1 Communication Hierarchy: Target Contexts

When initiating a support thread, users do not broadcast messages to a general queue; instead, the system programmatically routes the data pipeline based on an explicit targetRole parameter. This architecture enables targeted communication with three distinct platform entities:

- **Certified Personal Coaches (coach):** Dedicated to technical athletic coaching, progress monitoring, and customized workout adjustments .
- **Licensed Nutritionists (nutritionist):** Tasked with managing dietary structures, metabolic calculations, and macro-nutrient tracking.
- **Platform Administrators (admin):** Serving as the baseline fallback for payment gateways, account management, subscription overrides, and infrastructure troubleshooting

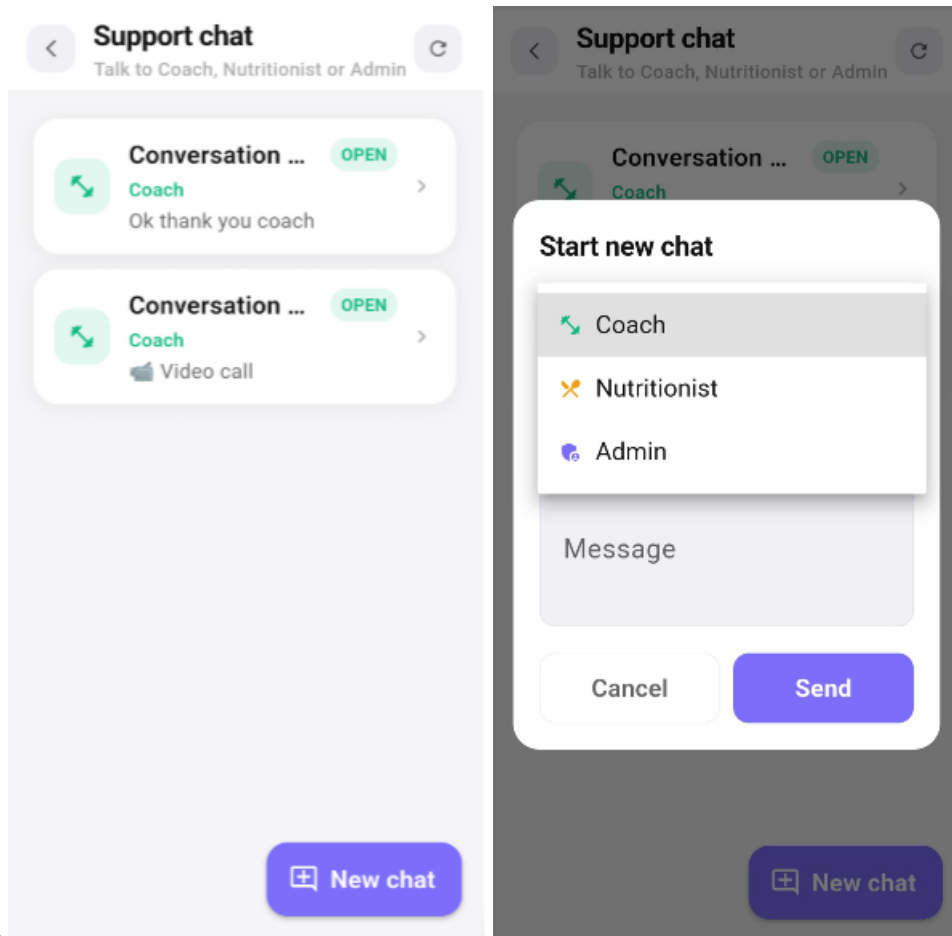


Figure 33: Support chat screen

3.5.3.2 Modalities of Interaction: Dual Asynchronous & Synchronous Channels

The support subsystem supports dual-mode communication, allowing participants to interact via asynchronous text payloads or upgrade the

A. Asynchronous Text-Messaging Stream

Users interact via an input control controller (TextField) linked to the network mutation pipeline `SupportChatApi.sendThreadMessage()`. When

This is transmitted over an HTTP/1.1 POST request to the remote server endpoint `/api/auth/support/threads/$id/messages`. Upon arrival, the data updates chronologically inside the client rendering tree.

B. Synchronous Real-Time Video Escalation

To resolve complex physiological or technical inquiries, an ongoing conversation can be upgraded instantly to a live video call.

1. **Signaling Initialization:** Pressing the action button triggers `SupportChatApi.createVideoCall()`, which establishes a communication bridge hosted on a remote Jitsi Meet infrastructure (`meet.jit.si`).
2. **Dynamic In-Line Component Injection:** The system automatically posts an automated hyperlink string into the chat timeline.
3. **The `SupportVideoJoinChip` Element:** If a URL contains a verified conferencing signature (e.g., `meet.jit.si`), the interface intercepts the plain-text message and renders a dedicated interactive component: **[Join video call]**.

3.5.3.3 Client-Side Media Execution and Hardware Handshake

When a participant interacts with the interactive call interface, the device execution environment executes a series of system actions to establish the communication bridge:

- **Deep-Link Protocol Interception:** The method `launchVideoCallUrl()` dispatches an explicit external application intent (`LaunchMode.externalApplication`). This commands the host operating system to bypass the app's internal sandbox.
- **Native Environment Redirect:** The system routes the URL token directly into the native WebRTC hardware rendering framework (such as an external web browser or a native Jitsi client application).
- **Hardware Ingestion and Room Setup:** the external media link safely sets up camera and microphone input parameters, initializes localized audio/video tracks, and connects the user into an isolated room context

(e.g., "Finesse Support 11...") under an explicit structural alias, such as Coach mia.

3.5.3.4 Thread Constraints and Lifecycle Safeguards

To protect the integrity of historical support records and manage staff availability, the system applies strict structural restrictions on active communication windows:

- **State-Driven Input Filtering:** If a support specialist changes the ticket state from Open to Closed via the top action bar, the client UI catches this state mutation. It immediately tears down the message input controls and blocks video generation buttons.
- **Read-Only Archive State:** This updates the interface into an unalterable, read-only document archive. This safeguard protects historical text transcripts and associated media session histories from being altered after a case is resolved.

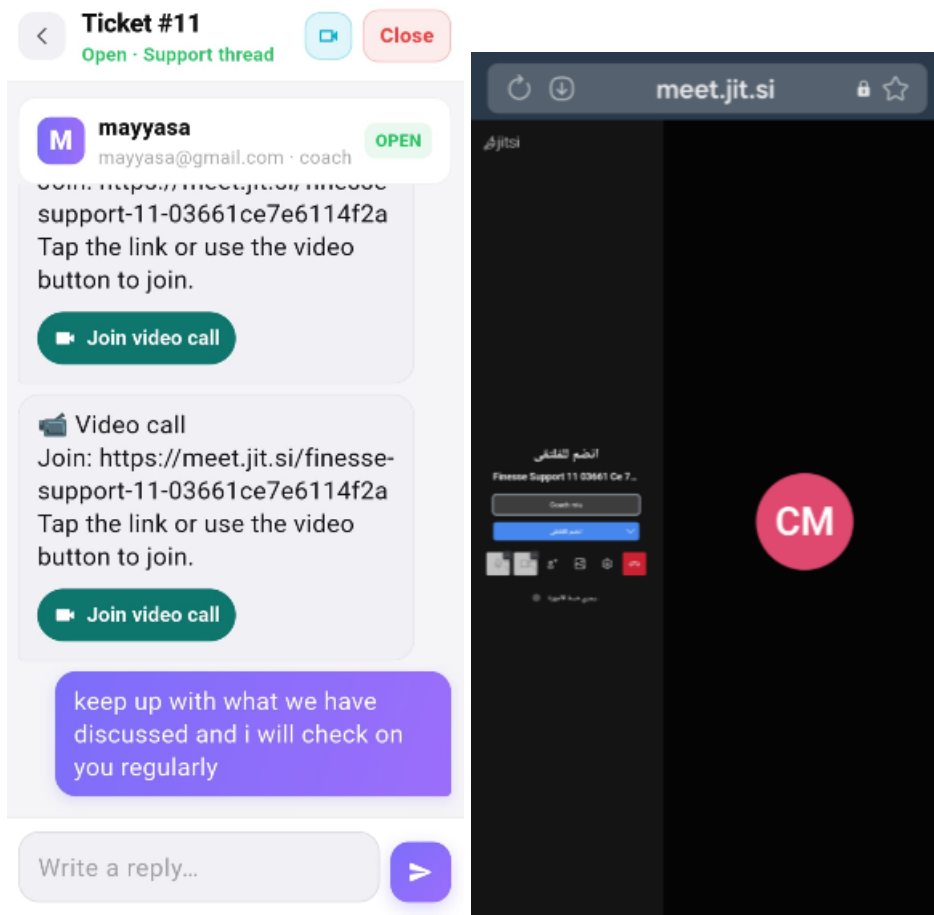


Figure 34: chatting screen

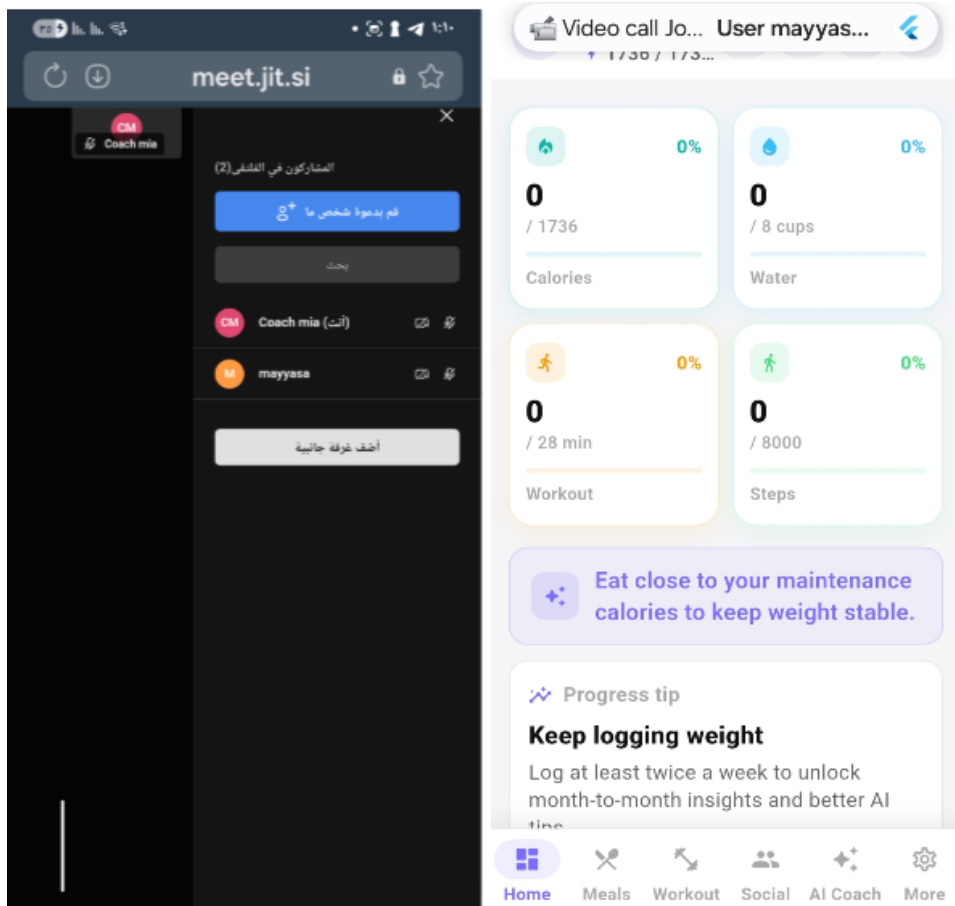


Figure 35: Video call and notification followed screen

3.6 Notifications Feature

This section keeps the user updated with all important activities across the app. It shows real-time notifications categorized by their types, such as support updates, challenge achievements, workout alerts, and meal plan changes.

Whenever a user receives a notification, they can see exactly what type of event occurred through a clean color-coded icon (e.g., green for workouts, orange for challenges, and blue for support chat updates). The system also calculates how long ago the alert happened, displaying relative labels like *"Just now"* or *"2h ago"* to keep the timeline clean and easy to read.

Unread alerts are highlighted with a colored background tint and a small dot on the side to grab the user's attention. Tapping on a notification immediately marks it as read, and if the alert is related to a support ticket, the app automatically navigates the user straight into the Support Chat screen so they can quickly read and reply to their messages. This ensures the user is always informed and can instantly jump to the right place inside the app.

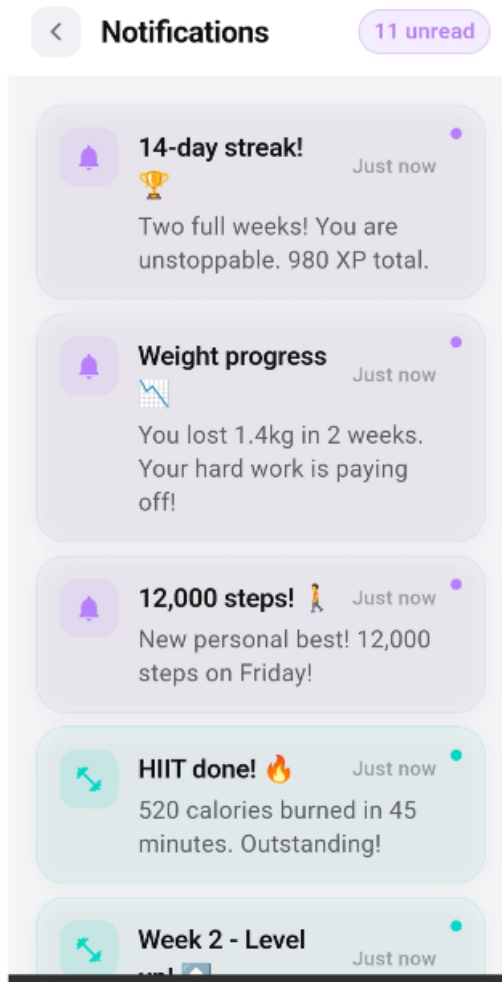


Figure 36: Notification screen

3.7 Home Screen (Dashboard)

The Home Screen serves as the central dashboard for the Finesse application user. It aggregates real-time metrics, behavioral logs, and personalized lifestyle recommendations into a single vertical scroll feed, ensuring the user is always informed about their daily fitness and nutrition lifecycle.

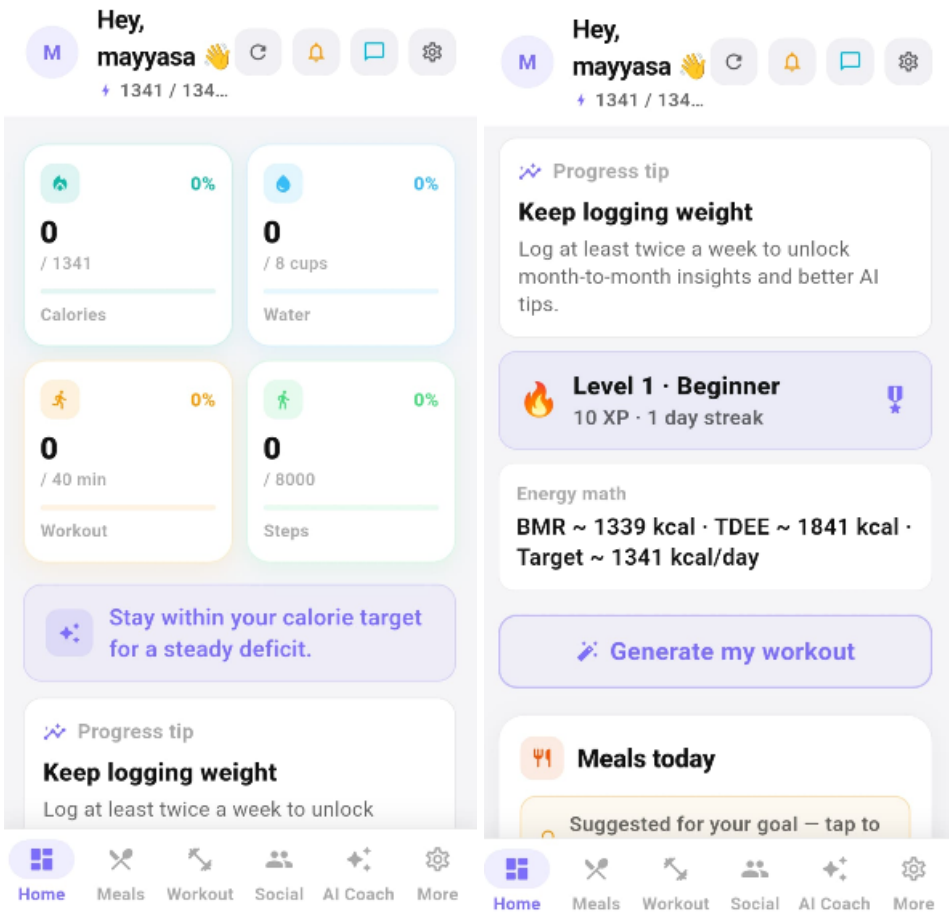


Figure 37: Home screen

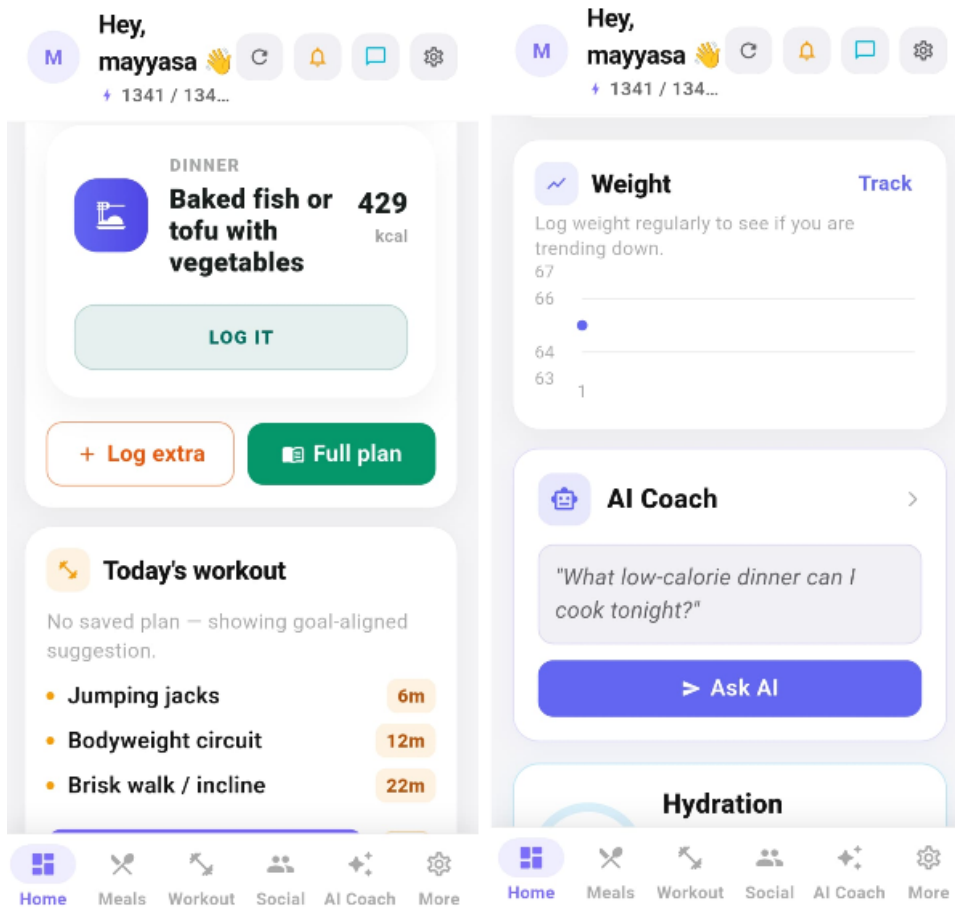


Figure 38: Home screen

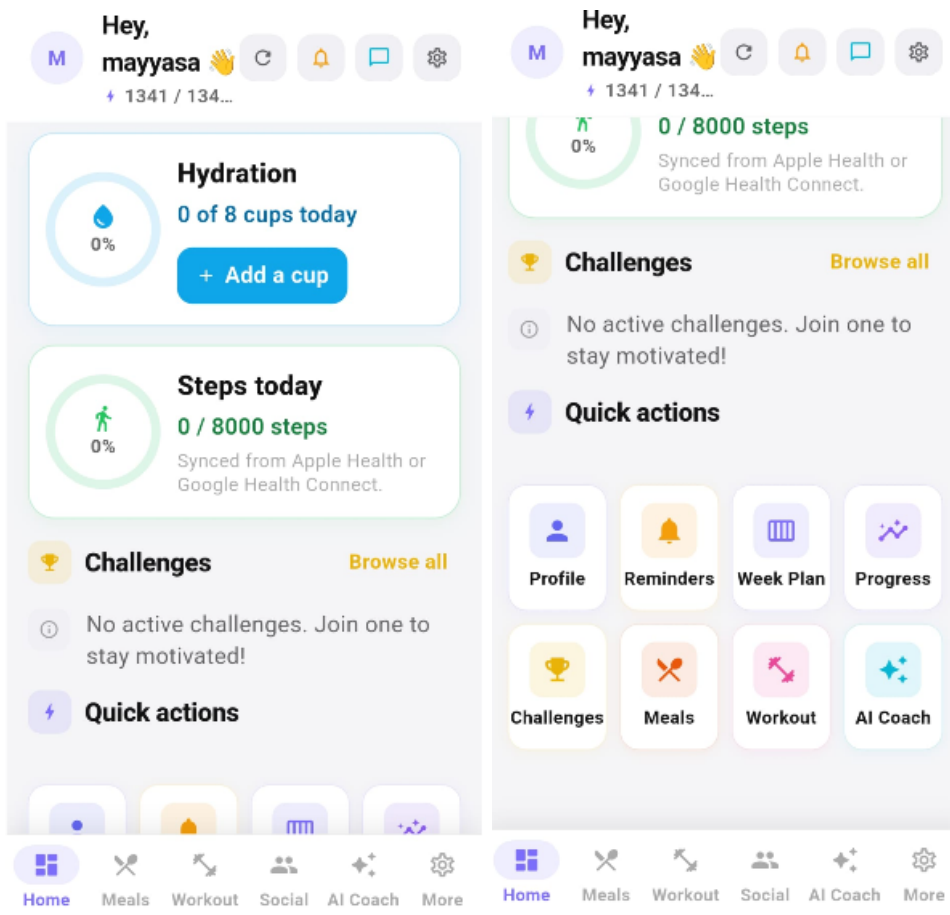


Figure 39: Home screen

1. Header & Progression Status

At the top of the dashboard, the interface displays the user's personal greeting alongside their profile avatar . To drive user engagement and gamify the fitness journey, this section includes an instantaneous view of the user's progression status:

Active Streak Container: Displays the continuous daily habit score (e.g., "1 day streak") paired with a leveling indicator (e.g., "Level 1 • Beginner") and total experience points (10 XP) .

Energy Goal Snapshot: Displays a quick numerical ratio of the current target (e.g., 1341 / 1341 kcal) right below the profile name for immediate calorie awareness .

Quick Navigation Utilities: Features top-tier quick icons for refreshing data, accessing the notification center, opening support chat channels, and entering system settings .

2. Core Tracking Grid (The Four Pillars)

The primary layout of the dashboard utilizes a clean, four-quadrant card system to monitor the user's essential daily targets . Each quadrant displays an explicit percentage completion bar, a dedicated metric icon, and structural value boundaries:

Calories Card: Monitors daily food energy intake by showing consumed calories against the absolute target limit (e.g., 0 / 1341 Calories) .

Water Card: Tracks hydration progress against standard fluid intake benchmarks (e.g., 0 / 8 cups).

Workout Card: Displays active exercise commitment by tracking completed training time against the daily plan (e.g., 0 / 40 min) .

Steps Card: Records daily physical movement by tracking step count against the standard health target (e.g., 0 / 8000 Steps) .

3. Advanced Conversational AI Coach Chatbot & Energy Mathematics

The dashboard embeds an intelligent, conversational guidance system that bridges real-time biological mathematics with a dedicated generative coaching chatbot directly on the user's feed:

Conversational AI Coach Interface: Operates as a specialized, interactive fitness chatbot. Users can launch this system to ask direct lifestyle questions, receive tailored meal modifications, or get immediate feedback on their training form .

AI Behavioral Insight: Generates personalized, automated motivation banners above the chatbot module to guide daily actions, such as advising the user to *"Stay within your calorie target for a steady deficit."* .

Energy Math Panel: Displays calculated metabolic targets derived from real-time user attributes, explicitly listing Basal Metabolic Rate (BMR ~ 1339 kcal), Total Daily Energy Expenditure (TDEE ~ 1841 kcal), and the final calculated daily target (Target ~ 1341 kcal/day) for structural deficit planning .

Instant Workout Generation: Includes a dedicated "Generate my workout" action button, allowing the user to bypass standard schedules

and command the AI Coach Chatbot to dynamically assemble a goal-aligned exercise circuit on the fly .

4. Automated Progress Tips & Analytics

To prevent drop-outs and improve historical data consistency, a dedicated Progress Tip component is heavily prioritized in the middle of the screen layout:

Dynamic Reminders: Prompts the user with data-driven notifications, such as *"Keep logging weight,"* alerting them to log their metrics at least twice a week .

Locked Value Preview: Explicitly informs the user that consistent logging unlocks hidden dashboard benefits, including month-to-month structural trends, deep analytical insights, and highly optimized conversational tips from the AI Coach Chatbot .

5. Integrated Meal Plan & Dietary Tracking

The dashboard incorporates a direct entry point into the user's customized diet plan for the current period . It showcases the specific meal type (e.g., "DINNER"), the exact recipe name (e.g., "Baked fish or tofu with vegetables"), and its total caloric value (429 kcal) . Users can interact with this panel using three direct transactional controls:

Log It Button: Automatically commits the recommended meal and its full caloric weight to the daily tracker database with a single tap .

Log Extra: Allows users to manually add custom food items or snacks consumed outside the template layout .

Full Plan Button: Redirects the user directly to the complete weekly nutritional directory for full recipe breakdowns .

6. Today's Workout Circuit

For physical training adherence, the dashboard renders a streamlined workout block if no specific custom plan is active .

It displays a goal-aligned suggestion list detailing explicit exercise sets and target durations, such as:

Jumping jacks (6m) .

Bodyweight circuit (12m) .

Brisk walk / incline (22m) .

7. Hydration & Step Expansion Containers

Scrolling further down populates expanded logging modules for granular data entry :

Hydration Core: Renders a large circular progress gauge with an interactive "+ Add a cup" utility button for fast intake updates .

Steps Core: Displays a physical activity visual ring, explicitly detailing that step data is synced smoothly in the background from device hardware bridges like Apple Health or Google Health Connect .

Challenges Framework: Shows the active community engagement queue, detailing ongoing milestone statuses or displaying empty-state motivation text when no challenges are active .

3.7.1 Quick Actions Section

The Quick Actions section is positioned towards the lower boundary of the dashboard view to ensure smooth, unhindered navigation across the application's native modules . Rather than cluttering the interface with dense menus, the platform organizes its core components into an easily scannable grid layout consisting of eight interactive shortcut buttons (. Each action item features a unique color-tinted background tile and a clear vector graphic:

Profile: Provides immediate access to personal goal configuration settings, biometric data adjustments, and user account parameters .

Reminders: Directly routes the user to the local notification and alarm manager to configure habit alerts for water, meals, and check-ins .

Week Plan: Opens the comprehensive weekly calendar view to review upcoming workout splits and scheduled nutrition menus .

Progress: Launches the analytics suite to view historical weight charts, trend statistics, and monthly performance reports .

Challenges: Directs users to the community engagement hub to browse, select, and participate in active gamified challenges .

Meals: Opens the dedicated meal planner and dietary history logs for recipe management.

Workout: Instantly opens the training library, routine builders, and exercise execution views .

AI Coach Chatbot: Launches the full-screen interactive messaging portal, allowing the user to converse directly with the specialized virtual fitness coach for automated, on-the-spot consultation and program tuning .

3.7.2 Dynamic Workout Generation Flow (AI Coach Engine)

When the user triggers the standalone "**Generate my workout**" action button on the main dashboard, the platform bypasses static, pre-scheduled templates. It initiates an on-demand generation pipeline driven by the virtual AI coach to build a highly targeted, goal-aligned exercise circuit.

1. Interactive Experience Level Selection (Bottom Sheet)

Upon tapping the generation button, a native context-aware Bottom Sheet transitions into view from the lower boundary of the screen . This overlay temporarily blocks underlying activity to capture the user's immediate fitness state through three distinct, descriptive experience tier triggers:

Beginner • easier moves: Configures the generator to prioritize low-difficulty, shorter-duration movements to prevent physical overexertion .

Intermediate • balanced: Instructs the algorithm to pull a randomized, structural mix of exercises tailored for standard stamina levels .

Advanced • more volume: Prompts the engine to scale up constraints, allowing higher set counts, longer sustained exercise blocks, and intense movement profiles.

2. Curation Logic & Process Operations

Once the user registers their experience choice, the application personalized data processing rules to assemble the workout sheet based on their unique fitness settings.

A. Personalized Time Calculation

The system checks the user's profile to extract their primary fitness objective (such as losing or gaining weight) and their desired pacing style.

The system evaluates these variables to determine an optimized total workout duration target tailored precisely to the user's current physical needs.

B. Filtering and Goal Alignment Rules

The system automatically scans the available exercise directory to filter movements that match the user's specific settings, applying four essential rules:

Goal Alignment Filtering: If the user's profile goal is to lose weight, the system isolates cardio-focused movements. If the goal is to gain weight, it filters for strength-building and resistance exercises.

Ordering and Shuffling: For beginners, the exercises are arranged in a structured, progressive manner based on low difficulty scores. For intermediate and advanced tiers, the list is completely randomized using a continuous shuffling method to keep the routine fresh and unpredictable.

Dynamic Capacity Control: The system caps the total number of exercises to control physical fatigue, allowing a maximum of **4 exercises** for beginners, **5 exercises** for intermediate users, and **7 exercises** for advanced athletes.

Intelligent Duration Normalization: Individual exercise times are strictly bounded to prevent unrealistic durations. If an exercise doesn't have a specified time, the system automatically estimates its length based on its estimated calorie burn rate or uses pre-defined constants depending on its type.

3. Generated Circuit Delivery View

After completing all background metrics validation, the interface instantly updates to display the customized "**Today's workout**" feed view

Granular Exercise Cards: Displays the picked exercises sequentially as clear interactive rows showing the exact name and required duration (e.g., *Stair Climbing - 40 min, Walking Lunges - 33 min*) alongside helpful instruction cues .

Individual Logging Buttons: Each card features its own dedicated "**Log +X min**" button, allowing the user to log and check off individual movements to their profile as soon as they complete them .

Aggregated Session Commits: A prominent, full-width sticky button labeled "**Log remaining (+X min)**" rests at the bottom margin of the screen . This utility automatically sums up the times of all remaining exercises and logs the entire left-over session to the daily tracking history with a single tap.

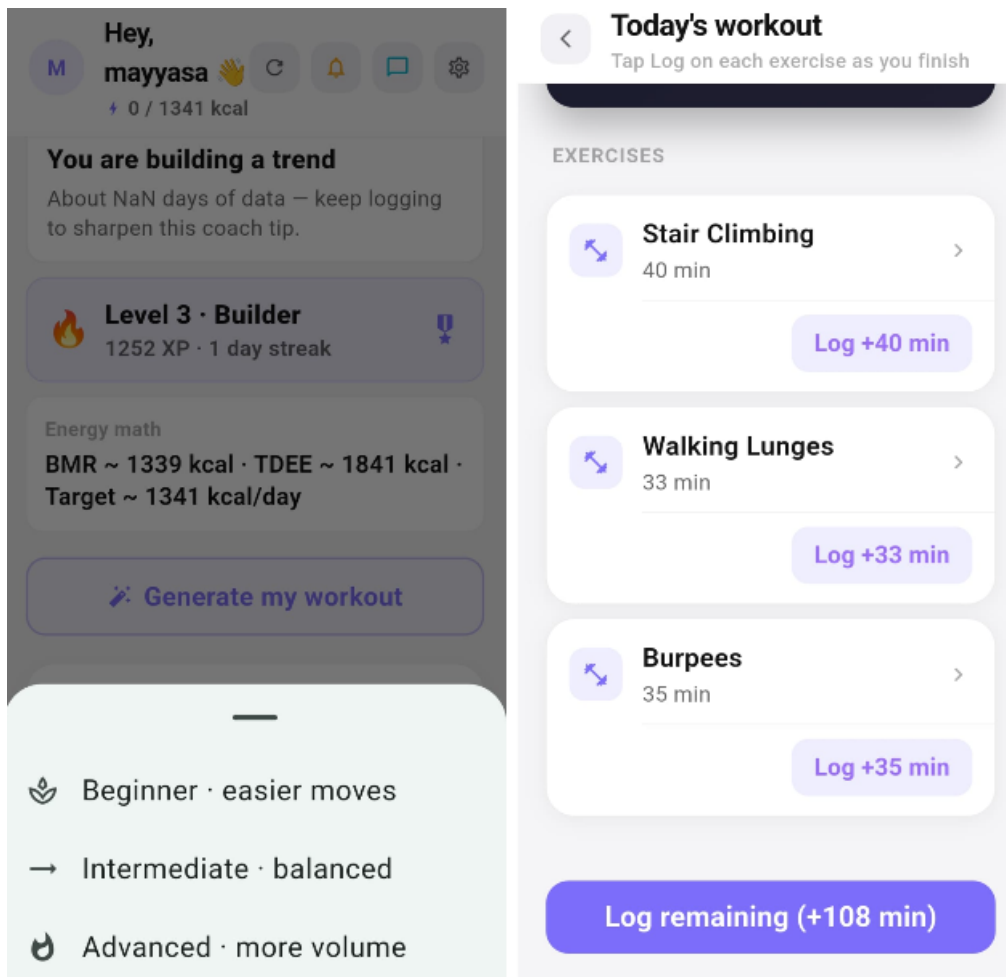


Figure 40: Generate workout screen

3.7.3 Workout Plan and Execution Directory

The platform provides a unified navigation interface for physical activity tracking. A user can seamlessly access their customized daily training execution view through three primary navigation entry points built across the main dashboard:

1. **AI Generation Path:** Tapping the "Generate my workout" action button on the dashboard.
2. **Quick Actions Shortcut:** Tapping the dedicated "**Workout**" shortcut tile within the lower grid navigation component.
3. **Dietary Integration Control:** Tapping the structural "**Full plan**" action button located directly beneath the daily meal suggestion block.

Once any of these paths are triggered, the user interface routes the user smoothly to the live training terminal, presenting the exact set of exercises scheduled for that day along with their calculated execution times.

1. Interactive Exercise Exploration & Media Support

Within the active workout list, each individual exercise container features an interactive exploration path designed to guide the user's form:

- **Exercise Detail Navigation:** Tapping the right-facing directional arrow (Chevron) on any exercise card opens a dedicated **Exercise Info** sub-view.
- **Performance Insights:** This sub-view provides structured breakdown points, including the targeted muscle groups (e.g., *Legs & Cardio*), a

clear difficulty rating (e.g., *beginner*), recommended sets or repetitions, and the estimated duration.

- **Equipment & Video Cues:** It highlights the exact equipment needed (e.g., *Stairs/Machine*) and includes a prominent, full-width "**Watch video**" action button that streams a visual tutorial to ensure correct movement execution.

2. Scheduled Weekly Calendar View

Positioned adjacent to the standard "Full plan" entry point on the dashboard, a dedicated secondary calendar button allows users to transition into a broader structural planning view:

- **Weekly Agenda Organization:** This panel maps out the user's full training schedule by day of the week (e.g., *Monday, Tuesday*) under a clear calendar week heading.
- **Muscle Group Focus:** Each day displays the specific muscle focus breakdown (e.g., *Chest & Triceps, Full Body*) and lists the total number of exercises curated for that focus block.
- **Integrated Shortcuts:** Every listed exercise item shows its expected time allocation and includes a nested navigational arrow, allowing users to pre-screen video tutorials and textual descriptions for any day of the upcoming week.

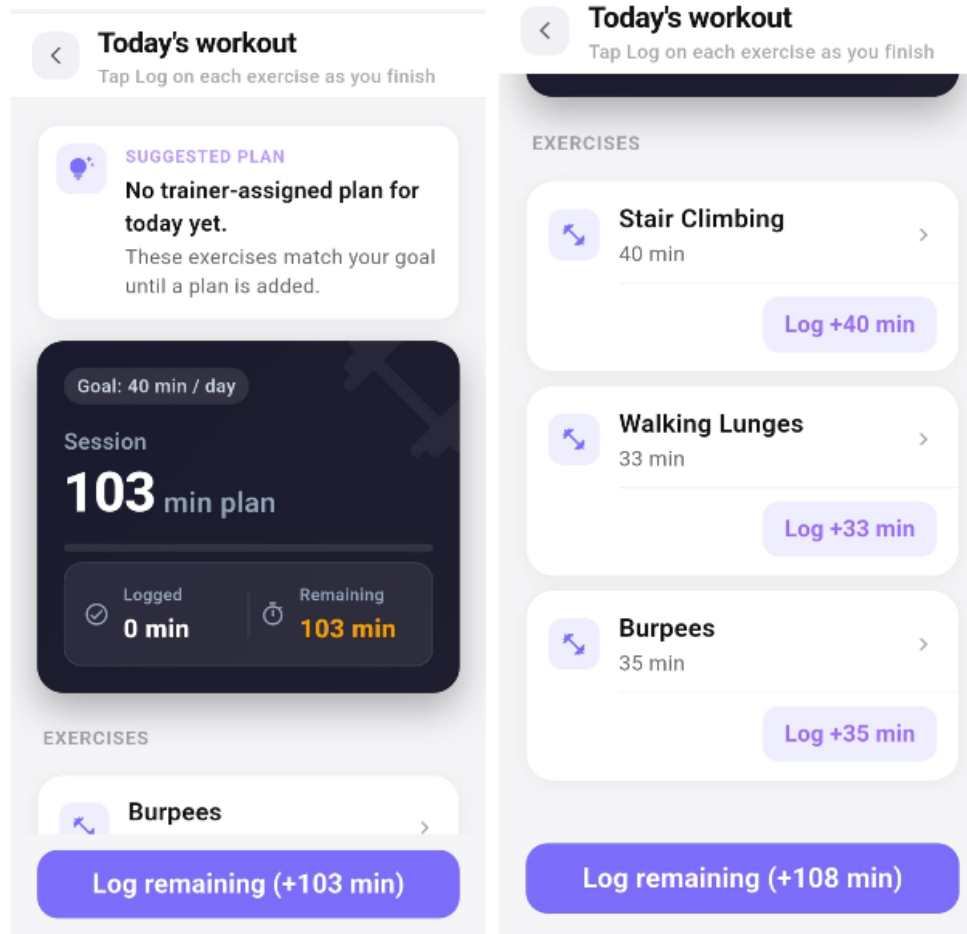


Figure 41:Workout screen

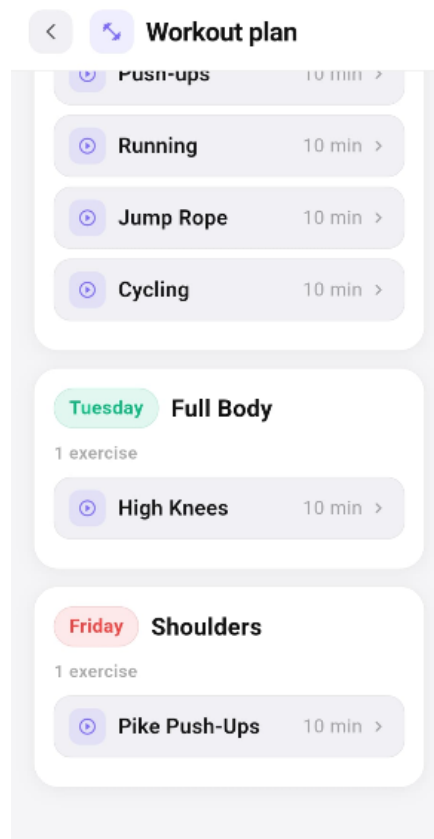
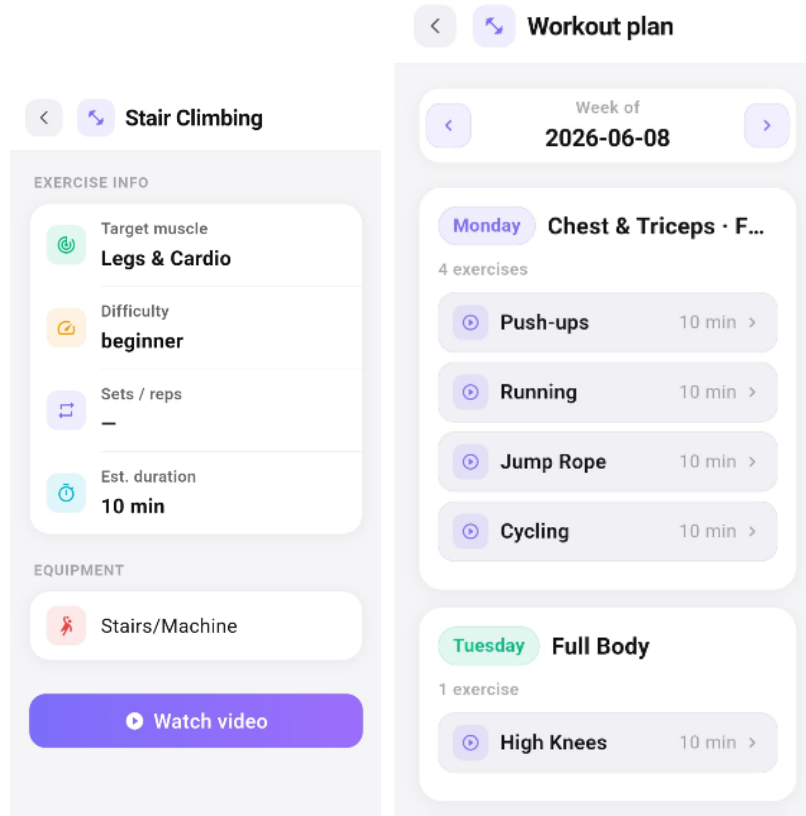


Figure 42:Workout screen

3.7.4 Goal-Aligned Meal Planning and Calorie Tracking

The platform incorporates an intelligent nutrition tracking framework designed to align the user's daily dietary intake with their macro objectives. This module dynamically evaluates nutritional needs and provides real-time feedback based on consumption behaviors.

1. *Target-Based Calorie Allocation*

The meal management dashboard partitions the user's daily caloric budget into distinct, goal-calculated slots:

Custom Meal Profiles: Standard eating windows are populated with personalized suggestions, such as *Breakfast* (e.g., 295 kcal), *Lunch* (e.g., 456 kcal), *Dinner* (e.g., 429 kcal), and structured *Snacks* (e.g., 161 kcal).

One-Tap Logging: Each pre-calculated item includes a dedicated "LOG IT" action shortcut, letting the user instantaneously commit that exact caloric weight to their daily progress history.

2. *Custom Intake Expansion (Log Extra)*

To accommodate items outside the standard automated recommendations, the system provides flexible log expansion options:

Quick Dashboard Integration: Users can tap the "+ Log extra" button directly on the main summary view to append custom entries.

Dedicated Snack & Food Log: Inside the full nutrition sub-view, a prominent bottom action button labeled "+ Log snack or other food" allows the manual insertion of unlisted meals, snacks, or outside ingredients, ensuring comprehensive daily tracking.

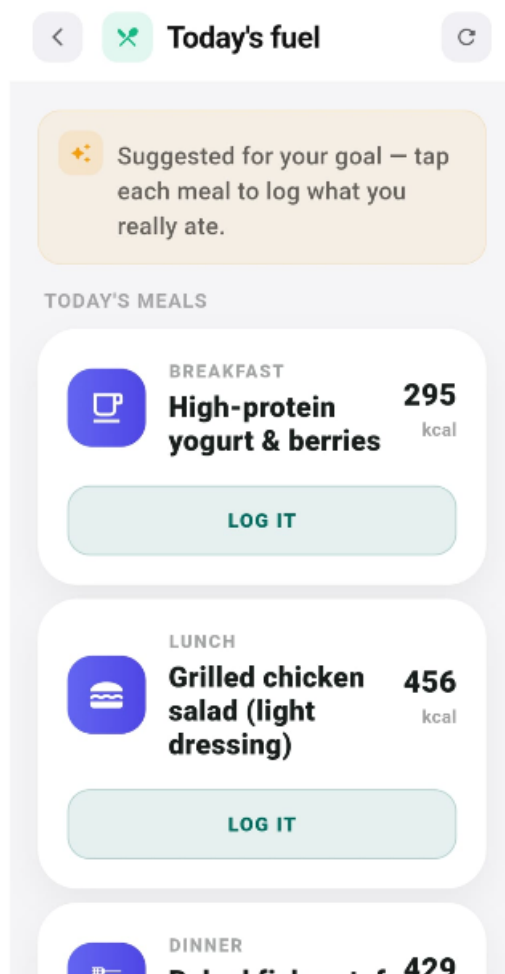
3. *Intelligent Caloric Threshold Warnings*

The framework continuously balances consumed macros against the user's target threshold to enforce fat loss or weight management goals:

Dynamic Header Balance: The top application profile bar continuously tracks active consumption values against the total daily allowance (e.g., 1341 / 1341 kcal).

Exceeded Capacity Warning Banner: If a user logs custom entries or extra items that push the total daily intake past their assigned metric, the dashboard instantly displays a soft-red warning container titled "Daily calorie limit exceeded!".

Behavioral Guidance: This alert explicitly informs the user of their caloric surplus (e.g., *"You ate extra kcal over your daily goal  — try to eat less next meal to reach your weight loss goal."*) to assist them in adjusting their remaining choices for the day.



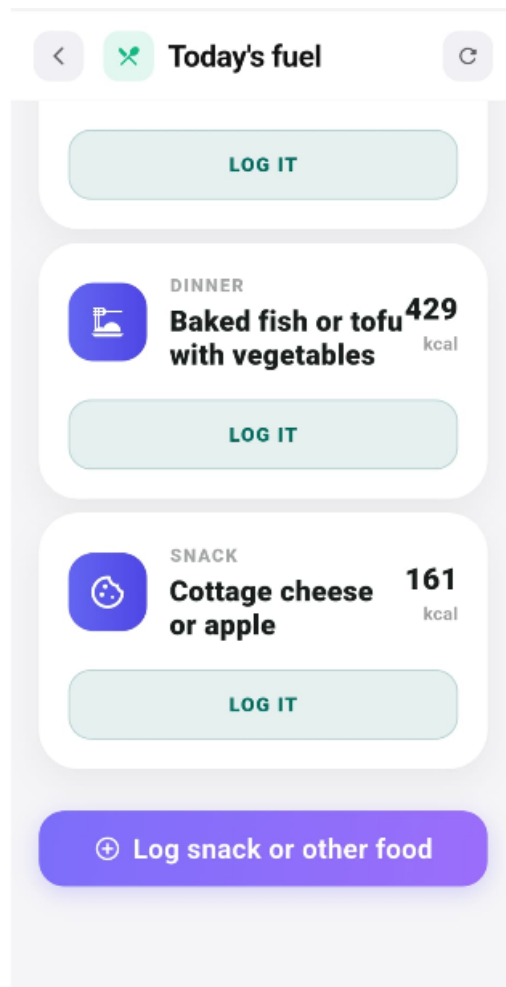


Figure 43:Meal plan screen

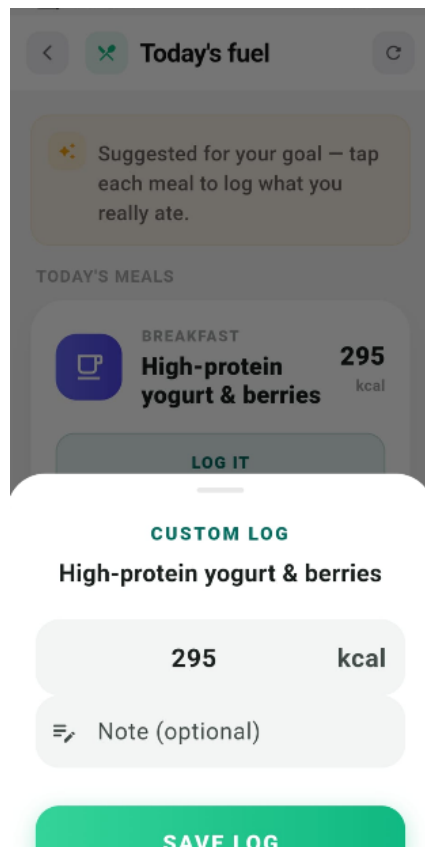


Figure 44:Log meal screen

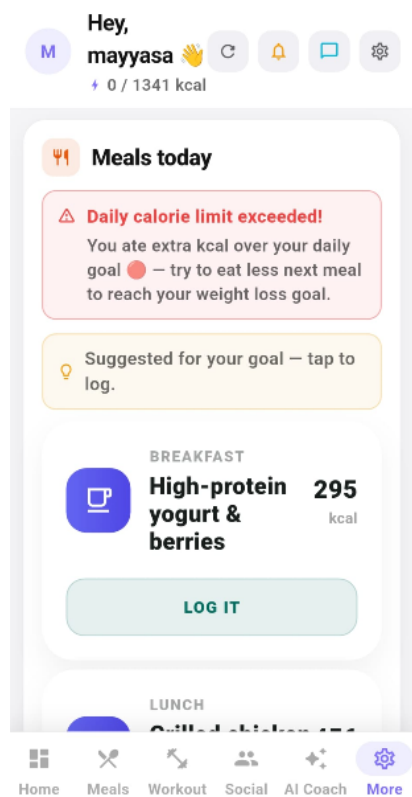


Figure 45:Warning notice screen

3.7.5 Gamified Challenges System

The platform features an integrated engagement framework that allows users to discover, join, and track multi-day fitness and lifestyle challenges designed to enhance continuous motivation.

1. Discovery and Entry Points

Users can access the centralized training challenges directory through two straightforward navigation entry points built on the home screen:

Dashboard Summary Feed: A dedicated "**Challenges**" section sits on the main dashboard. If no challenges are currently running, it displays an encouragement message (*"No active challenges. Join one to stay motivated!"*) alongside a "**Browse all**" shortcut.

Quick Actions Shortcut: Users can tap the dedicated "**Challenges**" trophy icon within the lower quick navigation grid to launch the main directory instantly.

2. The Challenge Exploration Hub (Discover Tab)

Upon entering the terminal, users are presented with a primary "**Discover**" view featuring curated health and activity programs:

Diversified Categories: Programs are structured with clear duration tags and concrete goals, such as the *Sugar Free Week (7-Day Challenge)*, *Drink Water Challenge (14-Day Challenge)*, *Weight Loss Challenge (30-Day Challenge)*, and *10K Steps Daily (30-Day Challenge)*.

Seamless Commitment: Each card features a clear description of the requirement (e.g., *"Lose 3 kg in a month"*) and a prominent "+ **Join**

challenge" action button. Once clicked, the button transitions immediately into a green "**✓ Joined**" state.

3. Active Engagement Tracking & Logging (My Challenges Tab)

Once a user commits to one or more programs, their ongoing journeys are organized inside a dedicated secondary tab named "**My Challenges**":

Automated Progress Syncing: Progress increments seamlessly and automatically whenever the application detects new entries in the user's daily logs, such as completed workouts, tracked meals, water intake, or steps synchronous with device health features.

Manual Log Backup: In addition to the automated syncing, users have the flexibility to manually record completion for the current day by tapping the explicit "**✓ Log today**" button on the challenge card.

Visual Status & Opt-Out Control: Every enrolled card displays an active progress percentage bar (e.g., 47%) alongside an interactive "**Leave**" text link, allowing the user to instantly exit the program if their personal goals change.

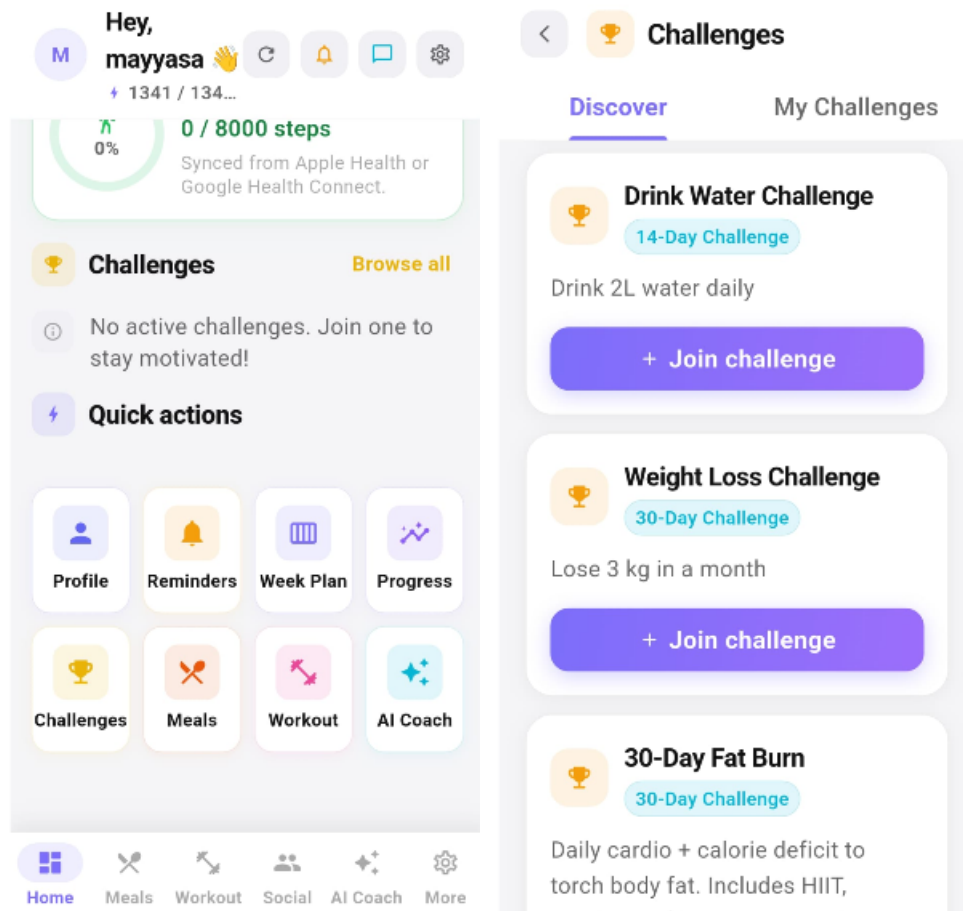


Figure 46:Challenges screen

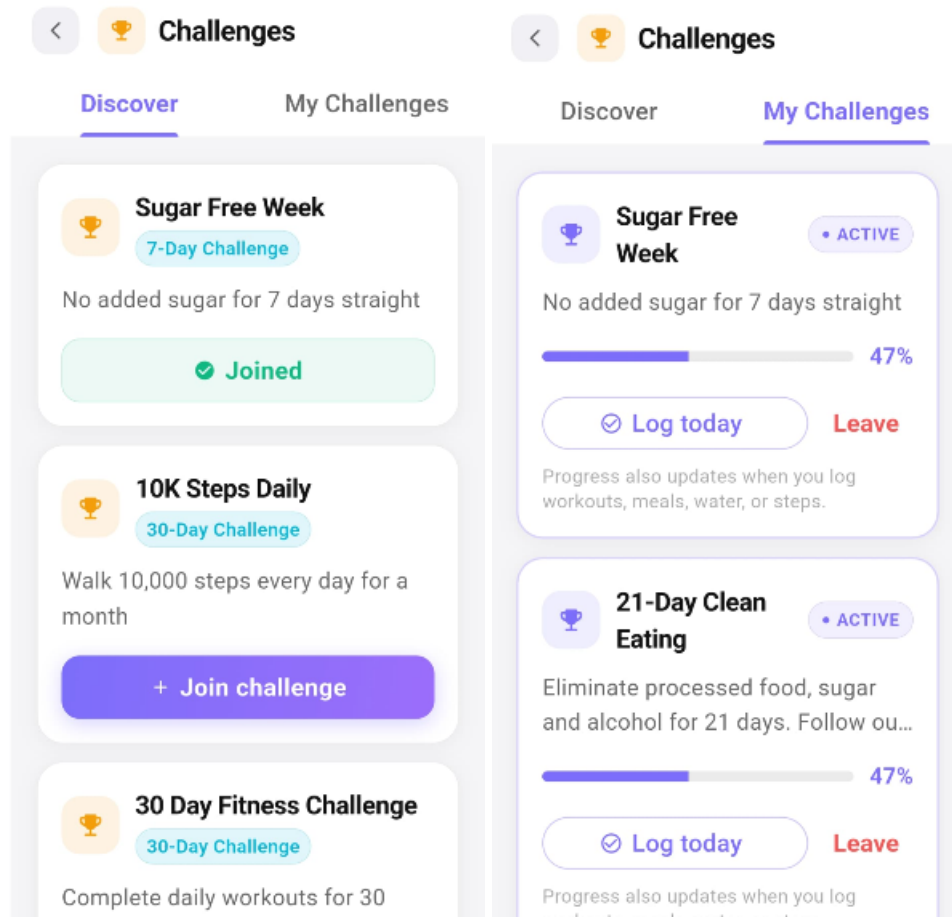


Figure 47:My challenges screen

3.7.6 Hydration Tracking System

The platform features an intuitive hydration monitoring utility that assists users in maintaining optimal fluid intake throughout the day by translating milliliter metrics into easily actionable consumption units.

1. Dashboard Widget & Quick Logging

Users can view and update their hydration status directly from the primary application feed:

Visual Progress Circle: A dedicated "**Hydration**" widget displays a circular percentage ring detailing completion against the daily target (e.g., *0%* of the goal).

Quick Addition Action: Tapping the primary "+ **Add a cup**" action button directly within the widget commits a standard metric unit immediately to the daily log without forcing the user to leave the main dashboard.

2. Comprehensive Hydration Sub-View (Stay Refreshed)

Tapping on the hydration container block routes the user to a detailed sub-view titled "**Stay Refreshed**" for granular consumption analytics:

Volumetric Progress Metrics: The sub-view balances numerical metrics side-by-side, displaying consumption in both standard cups and exact volumes against the configured daily target (e.g., *0 / 8 cups* and *0 ml of 2000 ml goal*).

Granular Breakdown Ledger: A structured **Stats** table lists real-time updates for key telemetry fields:

Consumed: Total volume successfully logged for the current day (e.g., *0 ml*).

Daily Goal: The structured operational target set for the active profile (e.g., *2000 ml*).

Cups Today: A dynamic count showing completed slots relative to total estimated cup targets (e.g., *0 / 8*).

Remaining: Automatically calculates the deficit required to fulfill the goal balance (e.g., *2000 ml*).

3. Standardized Unit Inputs & Reminders

To encourage consistent user engagement, the logging system incorporates standard portioning controls:

Fixed Volumetric Commits: At the bottom margin of the expanded interface, a full-width sticky button labeled **" + Add 250 ml "** allows users to record fluid intake in precise, regular-sized cup increments with a single tap.

Motivational Micro-Copy: A persistent informational tip box reinforces positive tracking behaviors by displaying healthy habit reminders, noting that *"Every sip counts toward clearer skin, better energy, and sharper focus."*

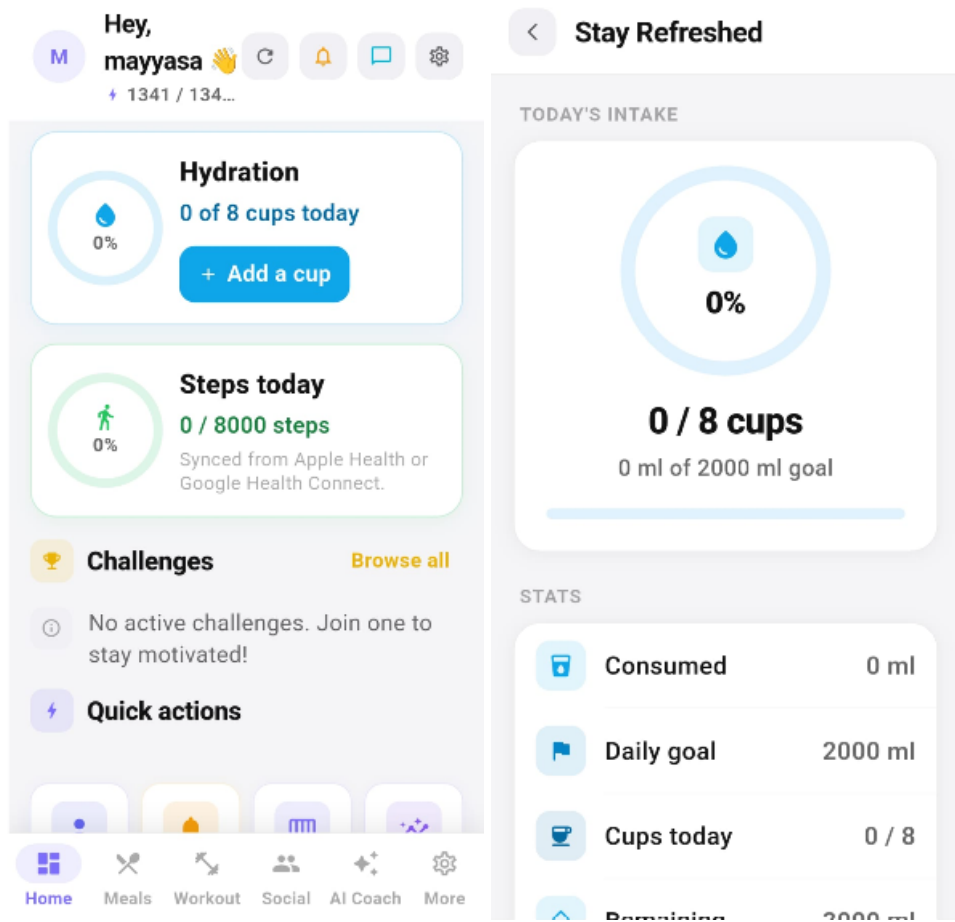


Figure 48:Hydration screen

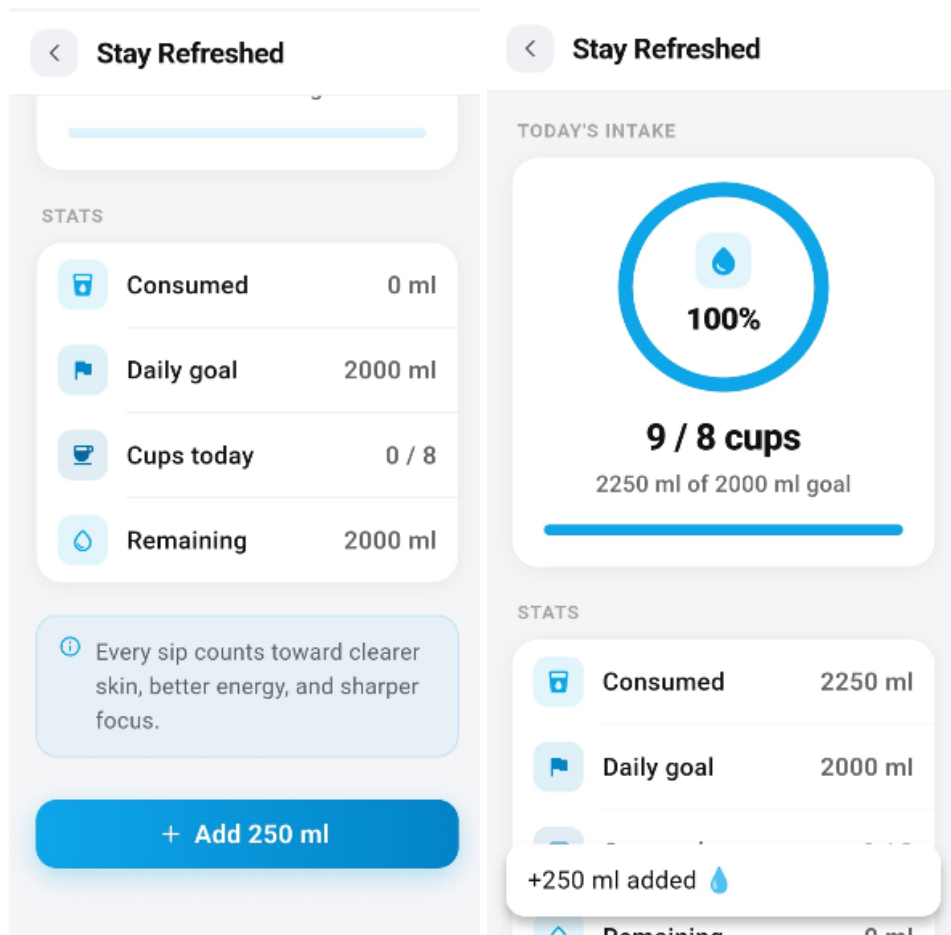


Figure 49:Hydration screen

3.7.7 Weight & Progress Tracking System

The platform includes a dedicated tracking module that enables users to record mass measurements, configure personal target weights, and evaluate long-term trends against their specified fitness plans.

1. Dashboard Summary Card

A structured preview widget on the home dashboard provides an immediate summary of body mass tracking status:

Trend Indicator Line: Displays a simplified, compact visualization graph showing recent data points.

Navigation Shortcut: A prominent "**Track**" text button sits in the upper right corner of the container, routing users directly to the comprehensive tracking dashboard.

2. Metric Overview & Analytics (Weight & Progress View)

Upon entering the main terminal, users are presented with a primary card that summarizes their current progress based on their chosen plan (e.g., *Plan: Lose weight*):

Latest Weight Telemetry: Displays the most recent logged value in large typography (e.g., *63.5 kg*).

Relative Variance: Computes and textually details changes relative to the user's initial baseline (e.g., *-1.5 kg vs start — moving toward your loss goal*).

Baseline Comparisons: A split lower banner explicitly locks in secondary parameters for quick evaluation:

Started at: The initial recorded body mass value (e.g., *65.0 kg*).

Change: The absolute net change in value since beginning the plan (e.g., *-1.5 kg*).

Automated Coaching Insights: An integrated **Insight** container populates data-driven feedback underneath, encouraging users to preserve tracking continuity (*"You are building a trend. Keep logging to sharpen this coach tip"*).

3. Target Weight Configuration

A modular **Target Weight** terminal allows users to manually specify and adjust their primary target metric:

Interactive Target Field: Features an editable numeric input displaying the targeted metric (e.g., *58.0 kg*) alongside a completion progress circle (e.g., *21%*).

Linear Distance Bar: A horizontal bar graphically indicates how far the user has progressed from their starting point toward the target goal.

Commit Control: A prominent **"Save target"** button locks in any modified targets.

4. *Weight Trend Visualization*

The interface integrates an analytical tracking line chart labeled "**Weight Trend**" for visual progress assessment:

Dual-Axis Mapping: Automatically plots the user's logged historical weight entries (purple data line) side-by-side with their configured goal threshold (dotted orange line marked *Goal 58.0 kg*).

Chronological Axis: Maps entries across a clear calendar timeline along the horizontal axis (e.g., *05-31* through *06-12*) to help users identify longitudinal patterns.

5. *Manual Weigh-In Logger*

Positioned at the bottom margin of the view is the "**Add a Weigh-In**" component, allowing users to append new telemetry blocks to their historical logs through structured fields:

Weight Input: A dedicated input box to key in the current mass measurement in kilograms (Weight (kg)).

Body Fat Percentage: An optional field to track body composition changes over time (Body fat %).

Date Selection: A date-picker line default-populated with the current tracking date (e.g., *2026-06-08*), allowing users to backdate historical entries easily.

Submission Action: A full-width teal button labeled "**✓ Save weigh-in**" commits the entered metrics to the database and refreshes all related analytical trends instantly.

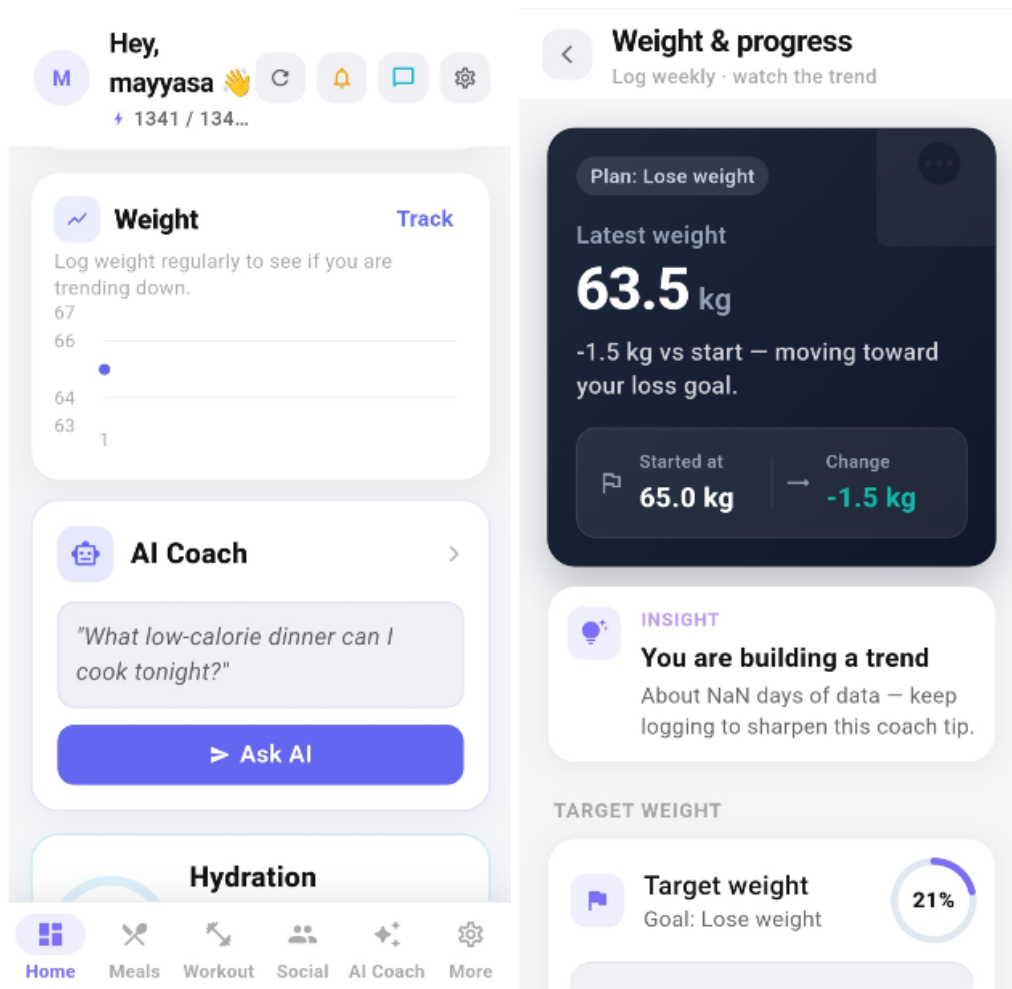


Figure 50:weight and track weight screen

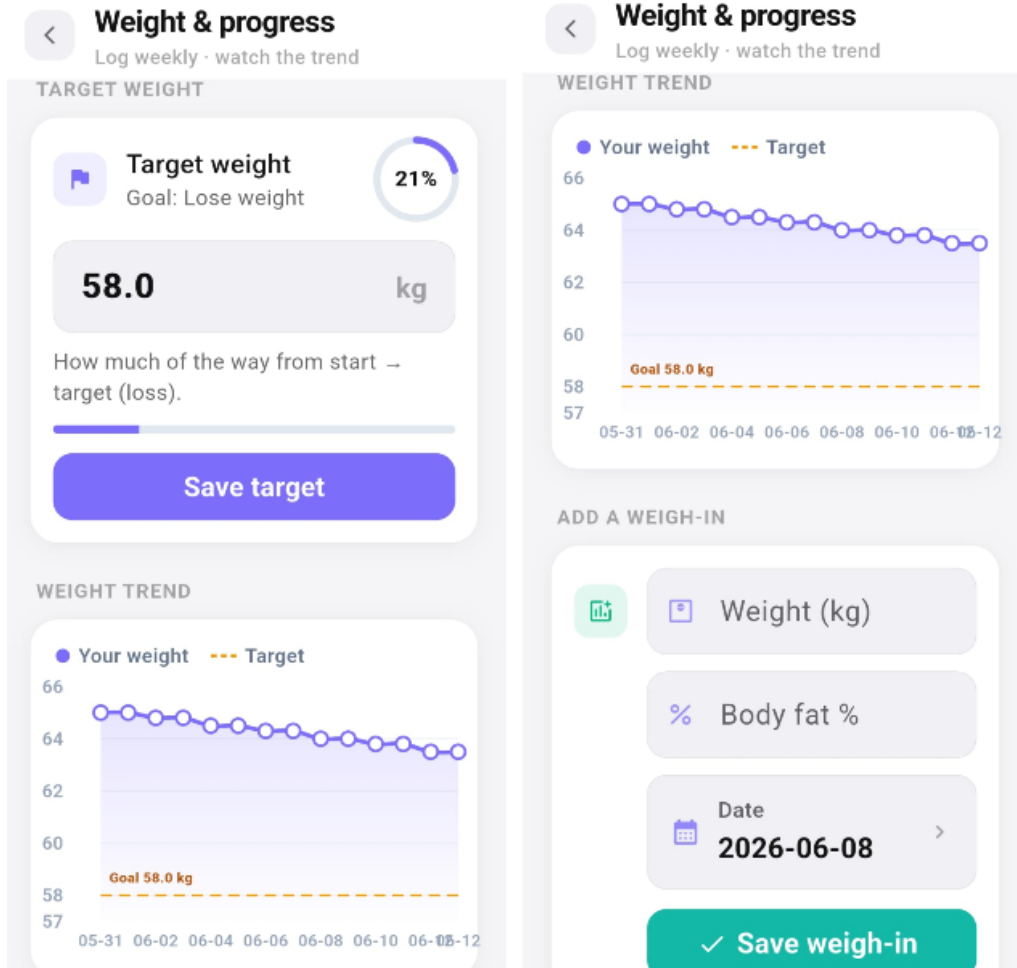


Figure 51:weight and track weight screen

3.7.8 Progress Analytics & PDF Report Export

The platform integrates a centralized analytics suite that aggregates logged biometrics, macro consumption, and activity levels to generate comprehensive historical reports.

1. Analytics Access Entry

Users can access the analytical control dashboard directly from the primary workspace:

Quick Grid Path: Tapping the "**Progress**" tile inside the dashboard's lower quick-navigation grid routes the user to the unified tracking view.

Dashboard Context: The sub-header outlines that the module actively computes parameters like *Weight*, *Body fat*, and *Weekly averages*.

2. Statistical Analysis Terminal

The **Progress tracking** terminal displays a data-driven view of recent health developments:

Weekly Average Tracking: Two dedicated summary metric blocks compare historical weight averages side-by-side (e.g., "**This week (avg)**" and "**Last week (avg): 63.8 kg**") to give users a high-level view of their trend direction.

Weight Trend Plotting: Includes a continuous multi-point line graph that visualizes daily mass data fluctuations across a defined

scale (e.g., from 62 kg up to 67 kg) to highlight clear progressive trends.

AI Coach Insights: Incorporates an analytical container labeled **"COACH INSIGHT"** that automatically interprets metric trends, offering reassurance such as *"You are building a trend. About NaN days of data — keep logging to sharpen this coach tip."*

3. Document Generation (Export Progress Report)

To support offline tracking or external coaching reviews, the application includes a built-in document generation mechanism:

Export Action Button: Positioned prominently at the top of the tracking terminal is a full-width action button labeled **"Export Progress Report"** accompanied by a document icon.

PDF Report Composition: Tapping this action generates an official document titled **"Finesse – Progress Report"**. The generated PDF structures data over a designated historical window (e.g., *Period: 2026-06-02 to 2026-06-08 (last 7 days)*).

Target Configurations Header: The report document locks in the user's explicit preset targets for reference, including tags like *Calorie target: 1341 kcal, Steps goal: 8000, and Water goal: 2500 ml.*

Aggregated Historical Ledger Table: The core of the PDF features an exhaustive structured matrix compiling five key telemetry fields by calendar date:

Date: Sequential daily ledger dates (e.g., 2026-06-02 through 2026-06-08).

Weight (kg): Precise body mass entries recorded for those tracking intervals (e.g., 64.8 kg, 64.3 kg, 64.0 kg).

Steps: Daily recorded movement activity captured by device sensors (e.g., 8700, 10200, 11500).

Water (ml): Fluid hydration quantities completed on those specific days (e.g., 2500 ml, 4750 ml).

Calories: The sum total of caloric food intake logged over the interval (e.g., 870 kcal, 3082 kcal).

Explanatory Footnotes: The document concludes with a clarification note emphasizing data sources, stating: *"Calories are based on your logged intake. Steps/water depend on what you synced or logged in the app."*

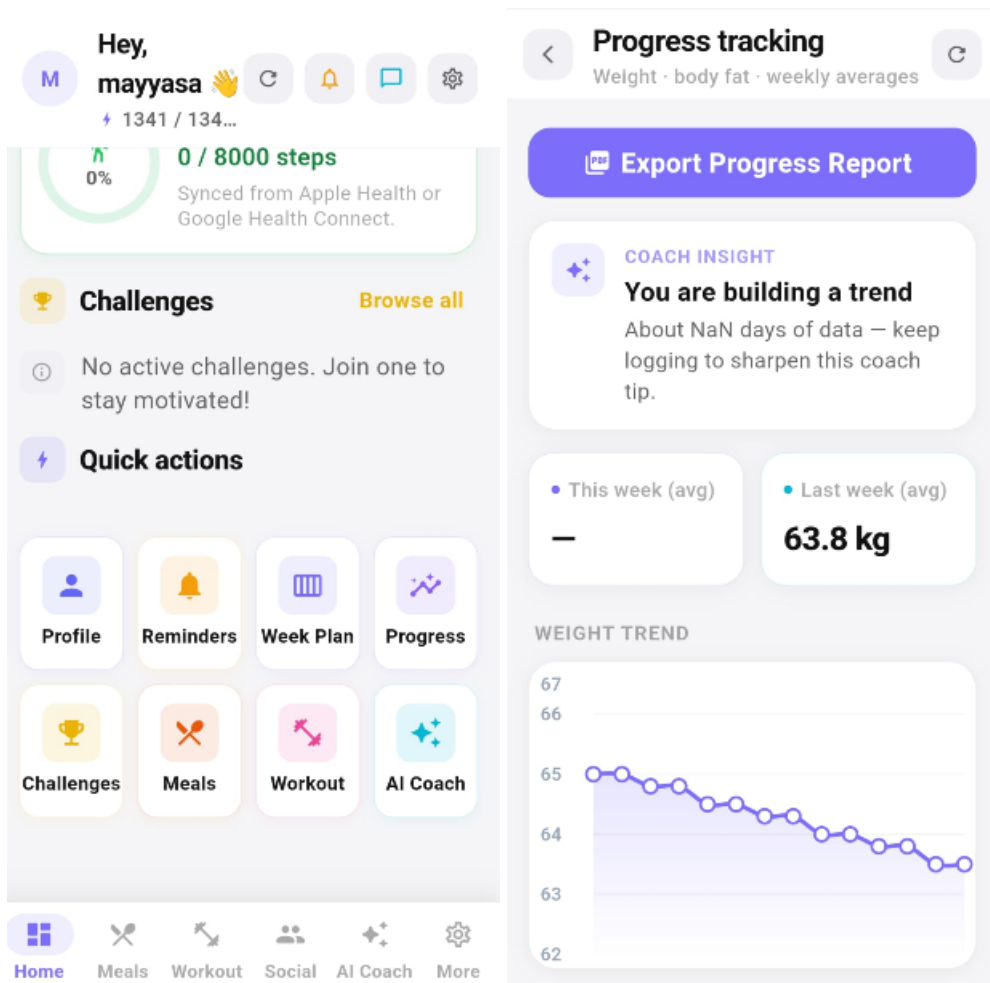


Figure 52:Weight progress screen

Finesse – Progress Report

Period: 2026-06-02 to 2026-06-08 (last 7 days)

Calorie target: 1341 kcal

Steps goal: 8000

Water goal: 2500 ml

Date	Weight (kg)	Steps	Water (ml)	Calories
2026-06-02	64.8	8700	2500	870
2026-06-03	64.8	10200	2250	950
2026-06-04	64.5	7800	2500	1090
2026-06-05	64.5	9100	2250	1120
2026-06-06	64.3	11500	4750	3082
2026-06-07	64.3	100	2250	1621
2026-06-08	64.0	9800	2750	2240

Notes

Calories are based on your logged intake. Steps/water depend on what you synced or logged in the app.

Figure 53:Weight progress PDF file screen

3.7.9 AI Coach Integration (Powered by Groq)

The platform features a highly specialized, context-aware AI Coach built on top of the **Groq API** (supporting backend models like llama-3.1-8b-instant and vision-capable layers like llama-3.2-11b-vision-preview). The assistant operates under a strict specialized mode tailored for sports nutrition, exercise science, and behavior modification, directly aligned with the user's live health metrics.

1. Contextual Performance Injector (Last 7 Days Banner)

When entering the conversation view, the AI interface automatically captures and mounts a data summary module tracking real-time context:

Synchronized Header: Displays calculated parameters directly derived from backend data rows (e.g., Avg steps: 8171, Avg kcal: 1568, and the user's computed dynamic target Target: 1341 kcal).

Automated Data Matching: System instructions ensure that when historical blocks are provided, the agent avoids data fabrication and crafts personalized recommendations explicitly matching the numbers shown (e.g., evaluating consistency in calorie limits or step goals).

2. AI Vision-Based Calorie Estimation (analyzeFoodImage)

The system incorporates an intelligent food logging framework via image recognition using Groq's vision architecture:

Visual Portions Analyzer: Through the chat field's camera icon, users can submit or capture actual food photography. The system converts the file into a sanitized Base64 sequence.

Reference-Scale Processing: The backend instructs the vision engine to judge portion volumes using a standard **26cm dinner plate** baseline reference.

Structured Response Return: The model enforces a rigid JSON response template containing the mapped food attributes (meal_name, calories, and an itemized breakdown including weight_g).

Interactive Logger Widget: Upon successful evaluation, the interface intercepts the JSON output and builds a custom reactive widget inside the chat window (e.g., displaying "**Banana — 105 kcal — Estimated from image only**"). It gives the user a quick-action button labeled "+ **Add to today's log**" to inject the estimated calorie metrics directly into their active diet database table.

3. Core Operational Rules & Behavior Standards

The Groq backend execution is governed by a strict set of programmatic guardrails defined in the system prompt:

Linguistic Mirroring: The agent reads user inputs natively and switches linguistic output modes dynamically, responding in the exact same language used by the user (Arabic ->Arabic, English -> English).

Comparative Performance Analysis: When receiving the last 7 days of performance data, the coach is restricted to evaluating real trends (better or worse than their own average) and must deliver between **2 to 4 personalized recommendations** tied strictly to their core goal (lose, gain, or maintain).

Contextual Recipe Selection: When users query available ingredients, the coach must output a singular, clear recipe-style format. If the internal [Finesse meal bank] catalog is injected into the prompt context, the engine must prioritize selecting the closest matching row using its exact name and calorie count.

Scope of Practice & Medical Safeguards: The system prompt explicitly enforces that the coach is not a physician. For instances involving injuries, chronic diseases, pregnancy, or eating disorders, the agent must immediately output a disclaimer advising consultation with a healthcare professional. It is programmatically blocked from recommending extreme diets or unsafe supplement stacks.

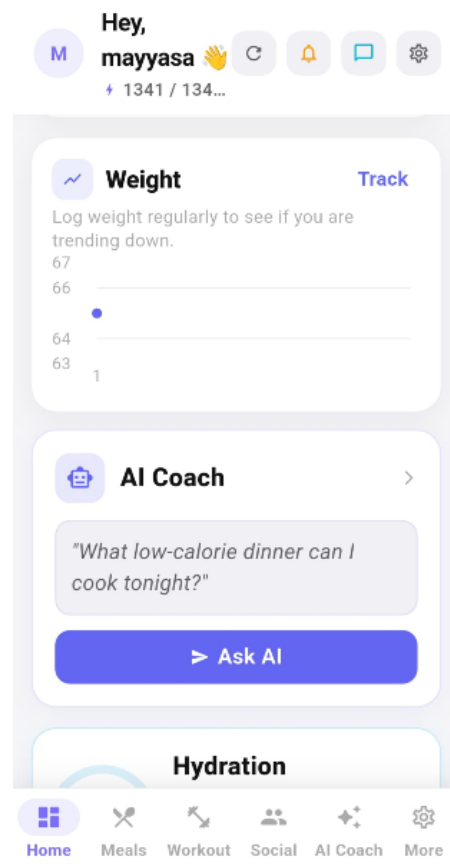


Figure 54: Ai coach screen

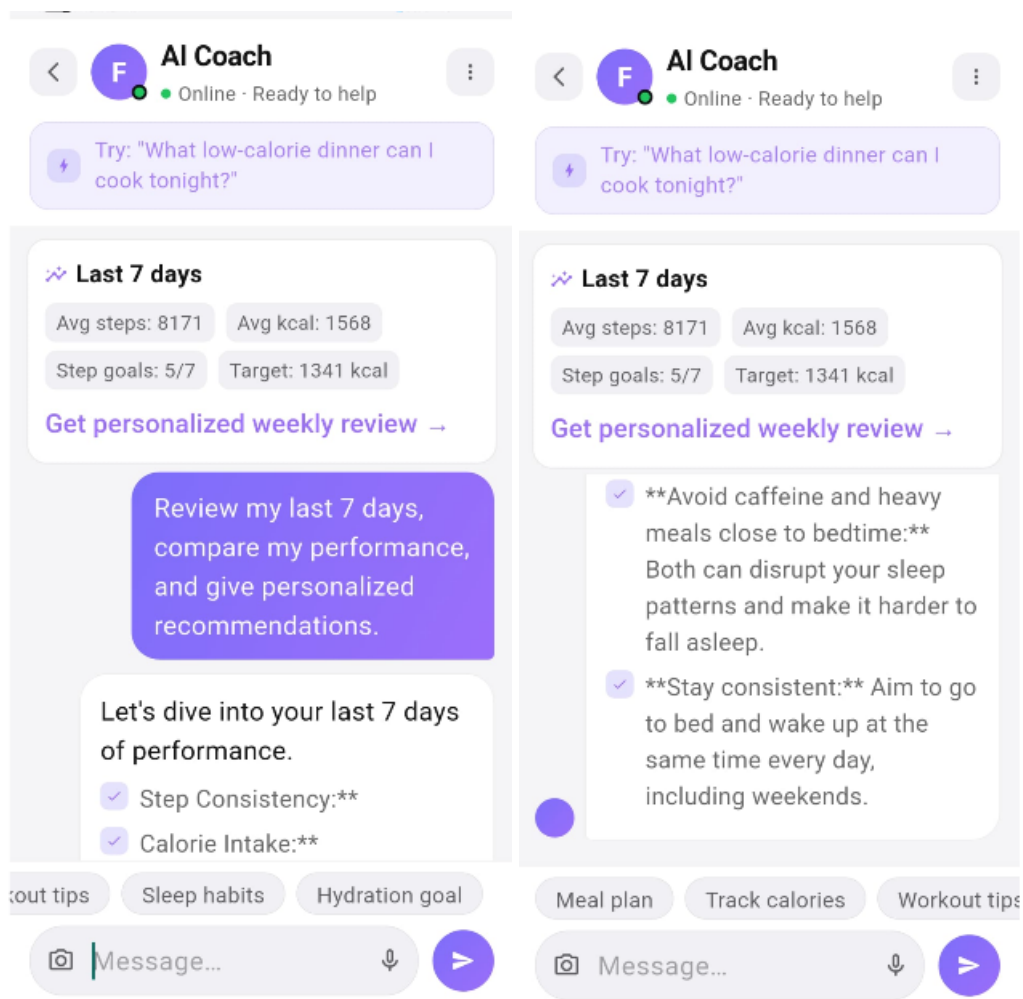


Figure 55:AI coach screen

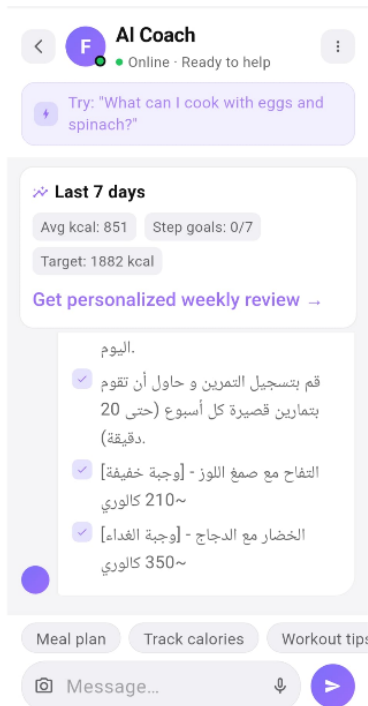


Figure 56:AI coach screen

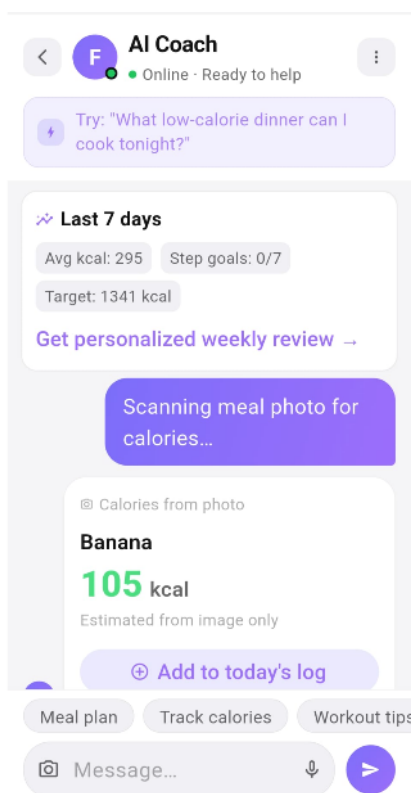


Figure 57:Scanning calories screen

3.7.10 Social Community & Interactive Engagement Hub

The platform incorporates a centralized social micro-network designed to foster collaborative motivation among users.

1. Community Feed & Content Interactions

The primary **Community** terminal serves as an interactive timeline where users share and engage with content:

Post Publishing: Users can tap the dedicated action button to draft and publish custom status updates and text descriptions directly to the public feed.

Engagement Metrics: Every published card tracks live user interactions, displaying distinct quantitative indicators for **Likes** and **Comments**.

Discussion Threads: Tapping into a post opens a dedicated comment view where users can type, submit, and browse sequential comment strings tailored to that thread.

Feed Sorting Utility: Positioned at the top margin of the feed is a structured drop-down sorting selector. This allows users to filter the content layout dynamically between **Newest**, **Oldest**, **Most Liked**, and **Most Commented** views.

2. Quick Navigation Shortcuts

To enhance user experience and provide frictionless access to core feature modules, the interface features a dedicated horizontal shortcut row positioned directly above the user timeline :

Pro Button: Tapping the "**Pro**" action button instantly redirects the user to the Pro Version Registration screen, allowing them to subscribe to premium features.

Challenges Button: Tapping the "**Challenges**" button routes the user directly to the dedicated Challenges Hub page to browse and join active fitness competitions.

Water Button: Tapping the "**Water**" button dynamically navigates the user straight to the Water Tracking module to log and review daily hydration metrics.

3. Gamified Leaderboard & Rank Rankings

The platform integrates a global competition ranking framework accessible via the secondary **Leaderboard** tab at the top of the interface to reward active community performance:

Podium Visualization Layout: The top area features a distinct visual tier structure showcasing the leading three community members with their respective scores:

Rank #1: Anchored inside a premium gold-accented podium block displaying the user's name (e.g., *mayyasa*) and total experience weight (e.g., *1417 XP*).

Rank #2 & Rank #3: Modeled symmetrically on the sides using silver and bronze indicators for runner-up profiles.

Persistent Personal Rank Banner: Positioned prominently below the podium is a dedicated card titled "**Your Rank**". This module locks the current user's personal context, displaying their exact numerical standing, experience total, active leveling badge (e.g., *Lv3 Builder*), and daily streak indicator.

All Rankings Ledger Table: Below the individual status card is an exhaustive list array sorting remaining platform users chronologically by XP totals, mapping account nicknames, current level markers (e.g., *Lv2*), progress completion bars, and current standings sequentially.

4. Member Discovery & Connections

To build a highly active network, the platform includes a direct system for social discovery:

Adding Users: Users can browse the community timeline or directory and instantly follow or add other people to their active social circle using the dedicated interaction buttons.

5. Contextual Environmental Synchronization

To assist users in planning outdoor training blocks or regulating daily activities, the main community feed integrates a location-based context module:

Live Weather Display: The system dynamically streams localized real-time updates directly to a top-anchored banner, displaying the city, current temperature, and atmospheric status (e.g., "**Nablus · 19°C · Clear sky**").

6. App Referral & Invite Utilities

The application includes built-in social utility shortcuts to simplify sharing the platform with outside networks:

Referral Link Sharing: Embedded directly within the timeline view is a dedicated **"Invite a Friend"** card displaying the user's explicit alphanumeric referral code.

One-Tap Share Button: Tapping the full-width **"Share with a friend"** action button leverages the native device layout to broadcast the app's installation link seamlessly to friends.

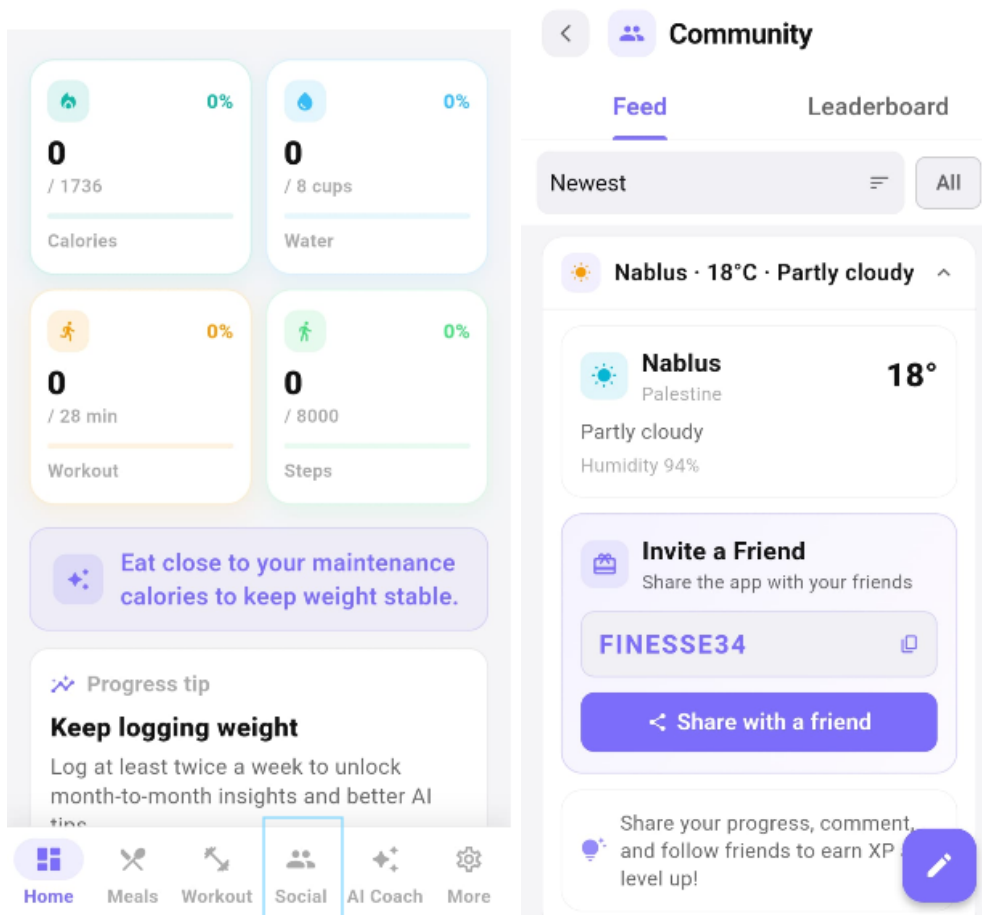


Figure 58:Community screen

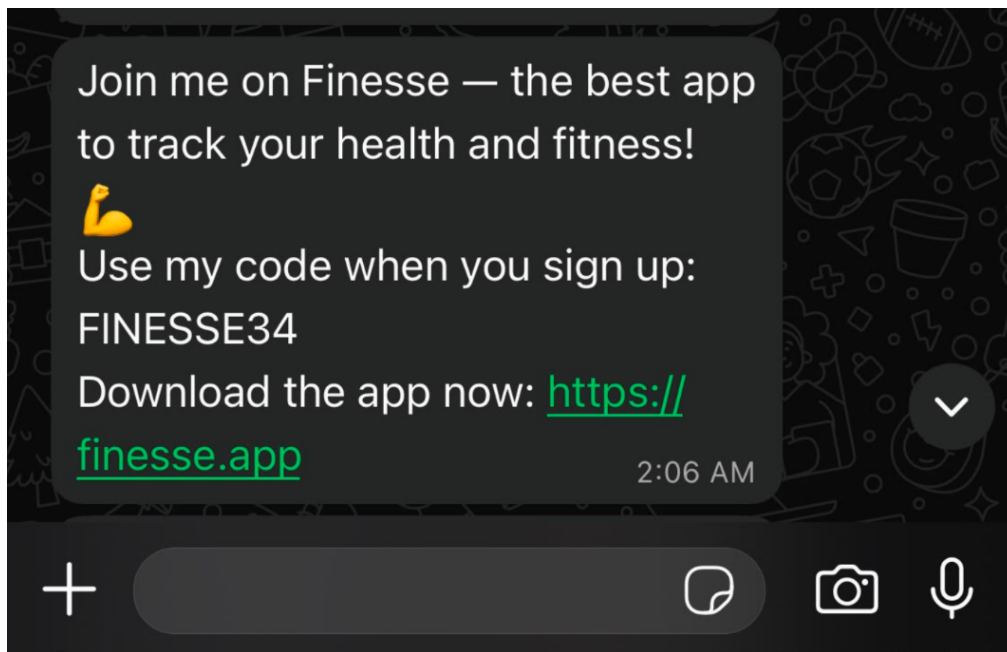


Figure 59:Invittation screen

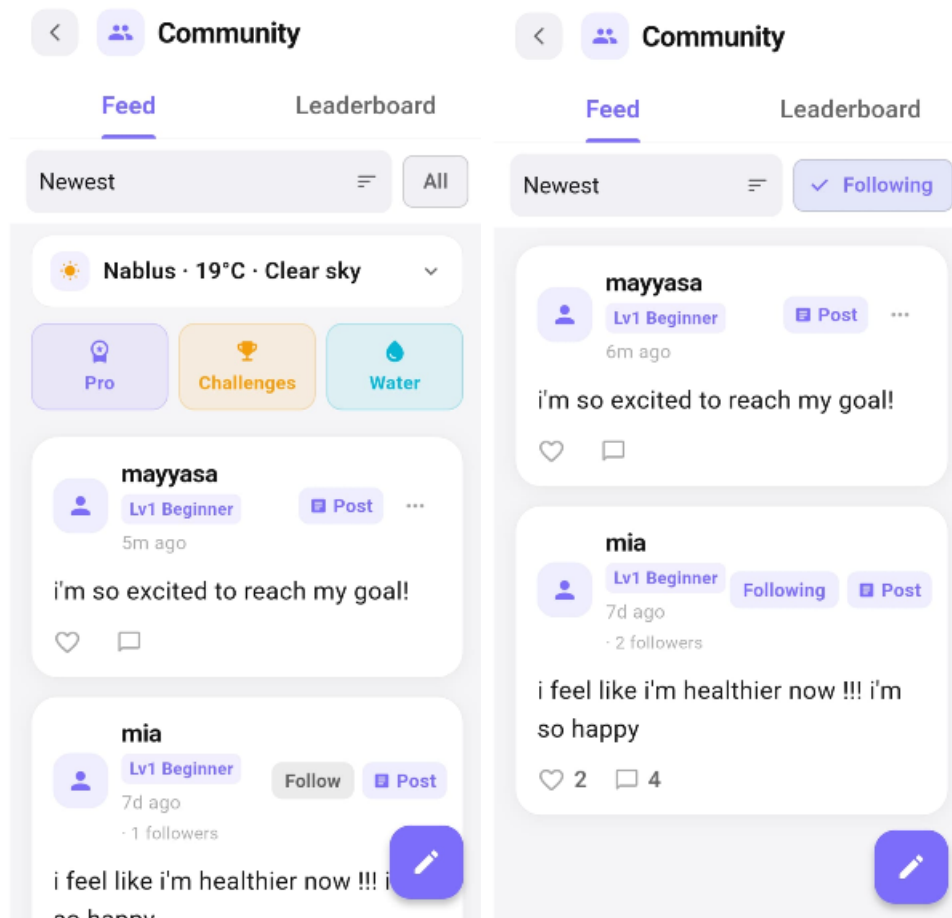


Figure 60:Community screen

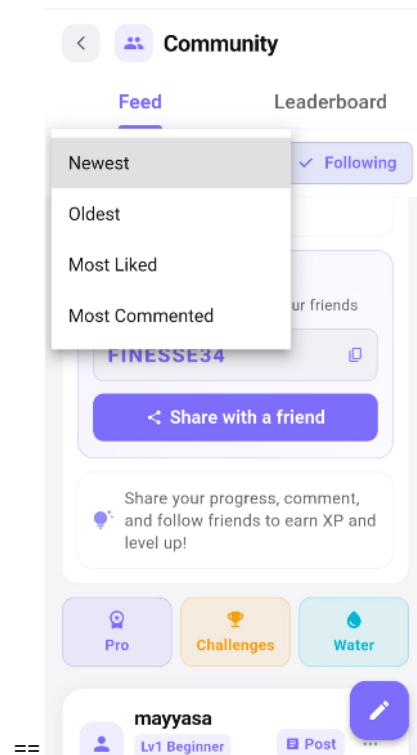


Figure 61:Post order screen

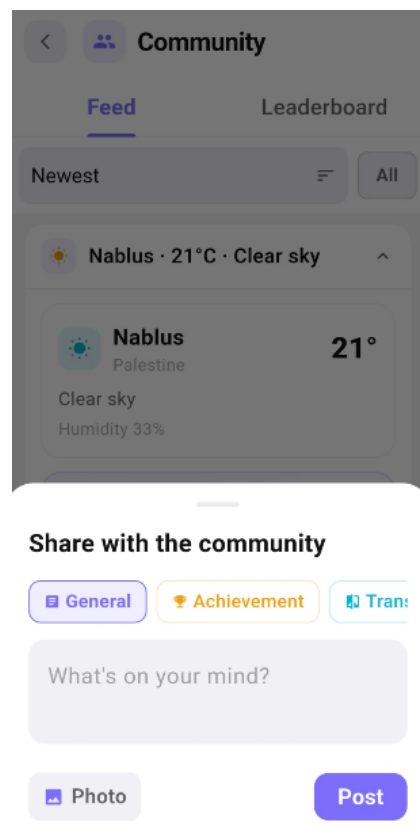


Figure 62:Post writing screen

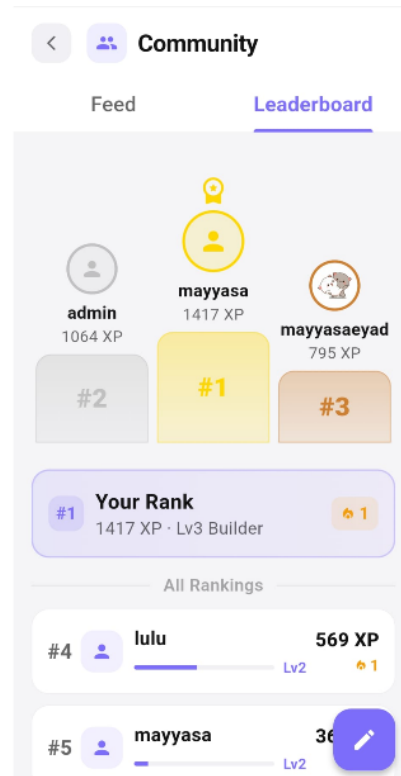


Figure 63:Leaderboard screen

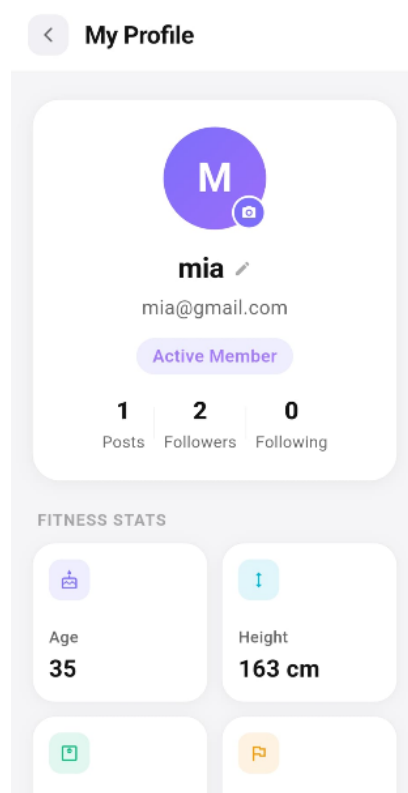


Figure 64:Profile screen

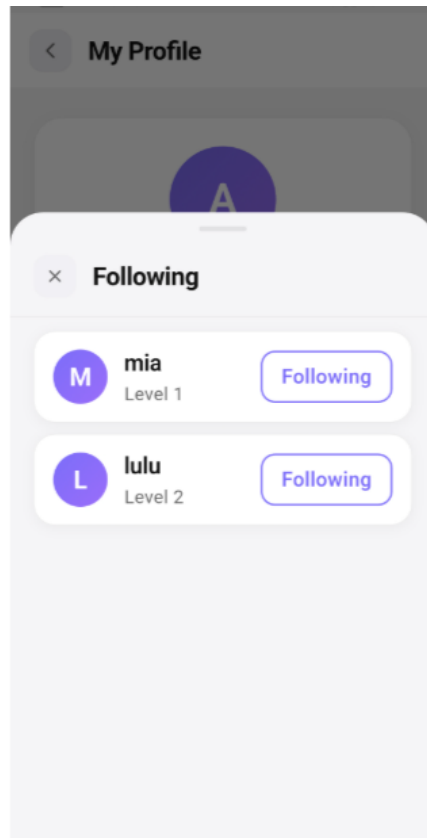


Figure 65:Follwore screen



Figure 66:Profile screen

3.8 Admin Screens & Management Infrastructure

3.8.1 Admin Panel Overview & System Metrics

The platform features a secure, centralized **Admin Panel** web terminal designed exclusively for account management, platform health monitoring, and system-wide telemetry tracking. Authorized accounts can swap their navigation scope from standard user view tracking directly into the backend management console.

1. Portal Accessibility & Security Entry

Admin Navigation Link: Users assigned with administrative privileges gain access to a dedicated "**Admin Panel**" route located at the bottom section of the primary navigation sidebar layout .

Session Verification: Tapping this module safely pivots the view interface from user statistics to the global monitoring layout, instantly loading live telemetry data blocks straight from the database cluster.

2. Key Metrics Telemetry Grid

The screen mounts a multi-card data visualization layout displaying aggregated historical and live operation stats grouped as "**Key metrics**" :

Registered Users Card: Summarizes total database account rows across all registration brackets (e.g., displaying **14 Registered users**).

Admin Seats Card: Isolates and tracks active system configuration roles containing master structural privileges (e.g., displaying **1 Admin seats**).

Meal Logs Monitor: Pulls dynamic database additions over a trailing window to monitor active lifestyle logging (e.g., displaying **60 Meal logs / Last 7 days**).

Steps Rows Monitor: Tracks aggregated physical integration streams to watch background processing synchronization (e.g., displaying **13 Steps rows / Last 7 days**).

3. Premium Subscriptions & Business Health Monitors

Positioned within the active tracking tier are dedicated commercial tracking parameters to evaluate application health and subscription funnels over time:

Pro Trials Active Card: Evaluates conversion funnels by displaying the absolute count of active user pools currently exploring unbilled premium features (e.g., **4 Pro trials active**).

Paid Pro Active Card: Tracks total active paying customers or manually administrative-granted elite tier profiles currently tied to monetization features (e.g., **2 Paid Pro active**).

4. Real-Time Table Inventory Diagnostics

To guarantee structural database health and flag systemic table faults immediately, the overview terminal integrates a technical inspection module :

Table Inventory Ledger: Displays an exhaustive micro-summary mapping active tables inside the database cluster (e.g., highlighting **20 tables** verified on the server).

Fallback Status Indicators: Subtext parameters are configured to display a clean em dash placeholder (—) if an explicit data entity array or specific table model is missing from database configurations, ensuring rapid diagnostics.

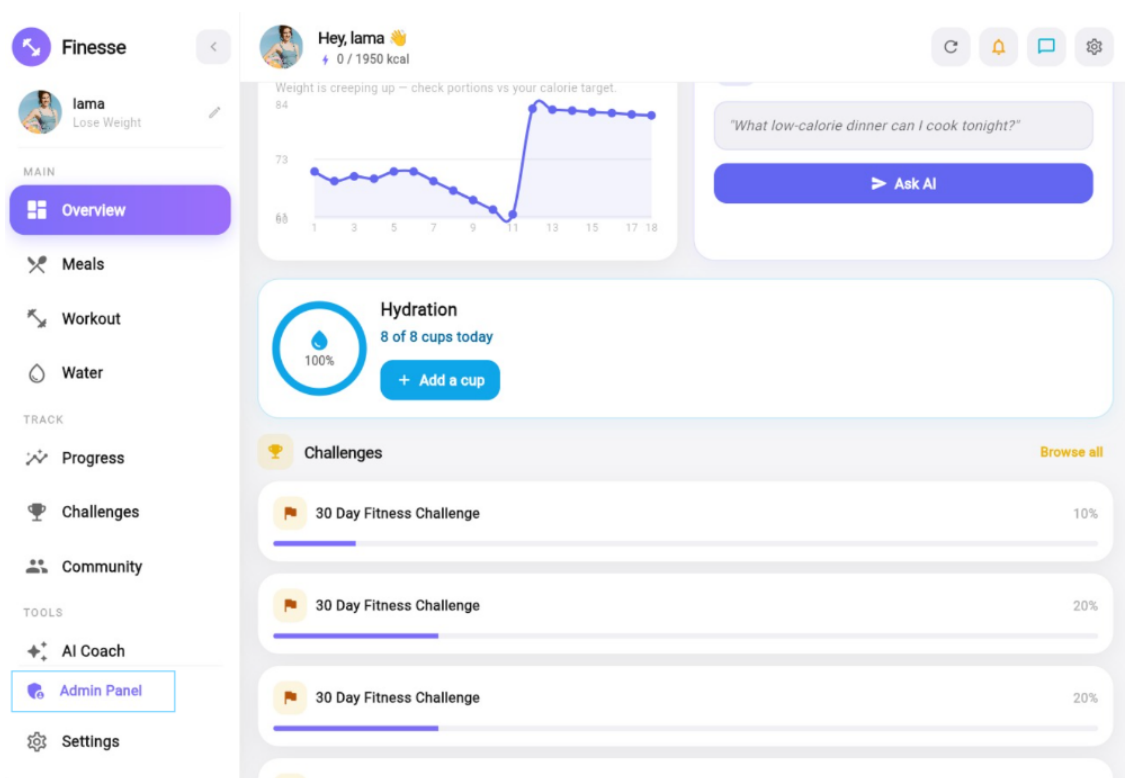


Figure 67:Home screen

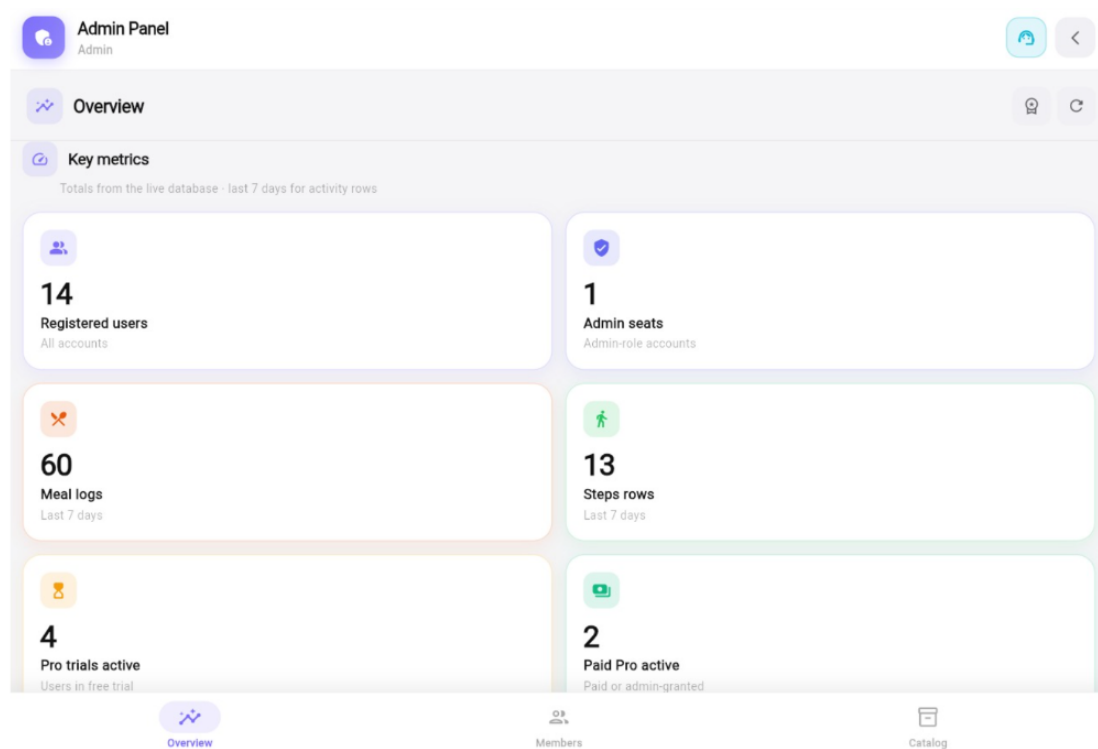


Figure 68:Admin panel screen

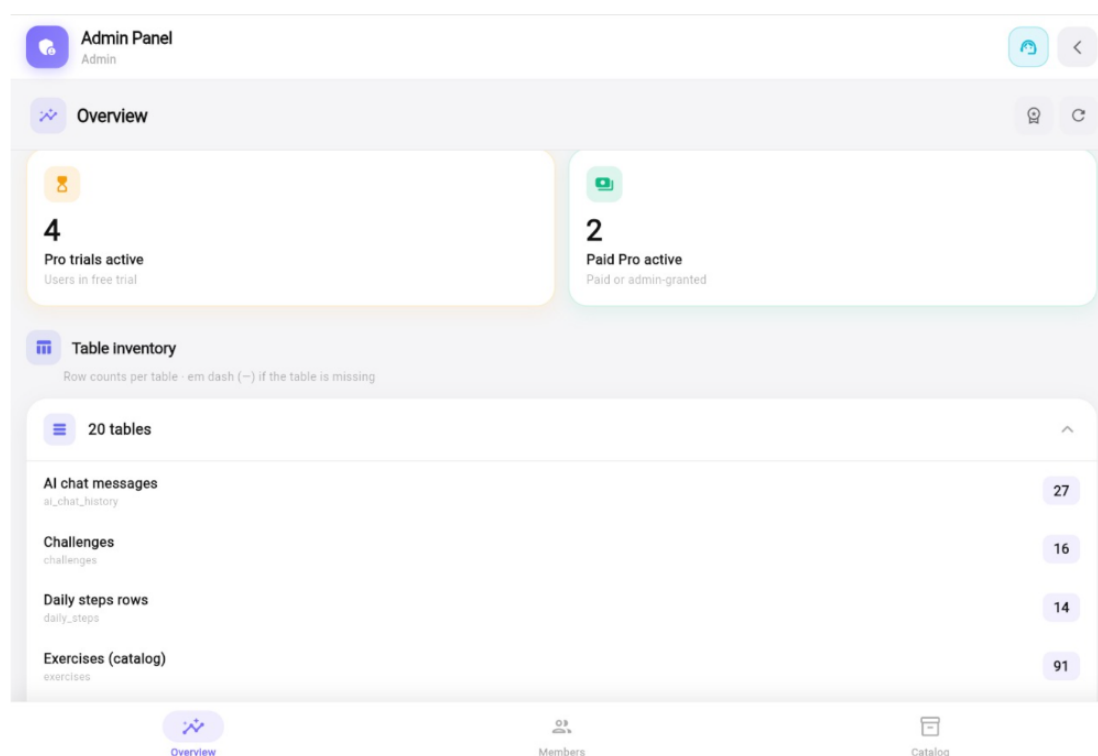


Figure 69:Admin panel screen

3.8.2 Pro Version Settings & Configuration

The **Pro Settings** sub-module provides administrators with an interface to manage subscription tiers, modify configurations, adjust pricing models, and customize the feature descriptions displayed to end-users.

1. Real-Time Subscription Telemetry Banner

At the top of the configuration portal, a dedicated telemetry strip provides an instant operational overview of the premium tier's volume :

Trial Active Indicator: Displays the real-time count of users currently inside the free evaluation period (e.g., **4 Trial active**).

Paid Pro Indicator: Monitors actively billing memberships or admin-approved accounts (e.g., **2 Paid Pro**).

Expired Indicator: Tracks accounts that have transitioned out of active billing cycles without renewal (e.g., **0 Expired**).

2. Trial & Pricing Configuration

Administrators can fine-tune the financial and timing mechanics of the premium conversion funnel using direct input fields :

Trial Duration (days): An input field to set the exact lifespan of the free evaluation tier (e.g., set to **30** days).

Monthly Price: Controls the standard baseline recurring billing rate for premium entry (e.g., set to **9.99**).

Currency: A dedicated text field to define the global transaction denomination framework (e.g., set to **USD**).

3. Pro Features Labeling & Instruction Controls

This section lets administrators customize the promotional marketing copy and onboarding guidance visible to regular users when they encounter the premium paywall :

Pro Features Label Input: A wide text container used to define and showcase the core value proposition of the premium upgrade (e.g., "**AI Coach, smart workout generator, video support calls**"). This text directly populates the user-facing upgrade prompt screens.

Payment Instructions (user): An editable text space allowing administrators to outline explicit manual checkout steps or customer support directions for finalizing a purchase (e.g., "**Contact support to activate Pro after your free trial.**").

4. Sandbox Deployment Controls

Demo Payment Mode Toggle: Positioned above the primary submission bar is an isolated system switch labeled "**Demo payment mode**". When toggled active, it enables an *Instant Pro without real gateway* simulation environment, allowing internal testing teams to bypass external payment processor checks.

Persistent Configuration Save: A full-width "**Save Settings**" action button locks all input changes, synchronizing updates to the global database cluster immediately .

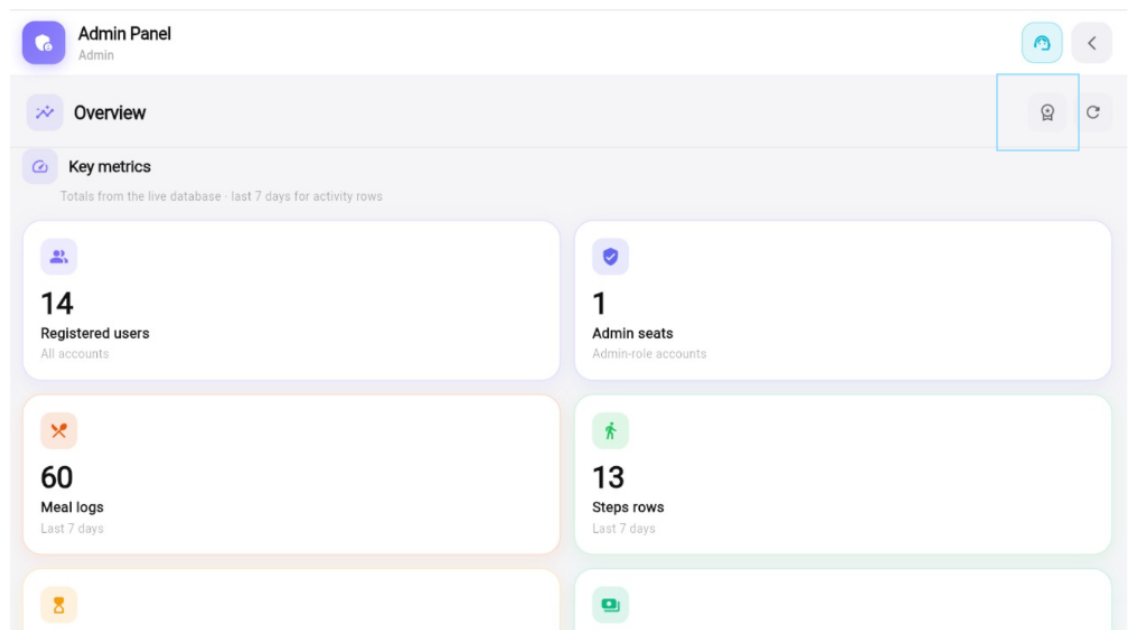


Figure 70:Admin panel screen

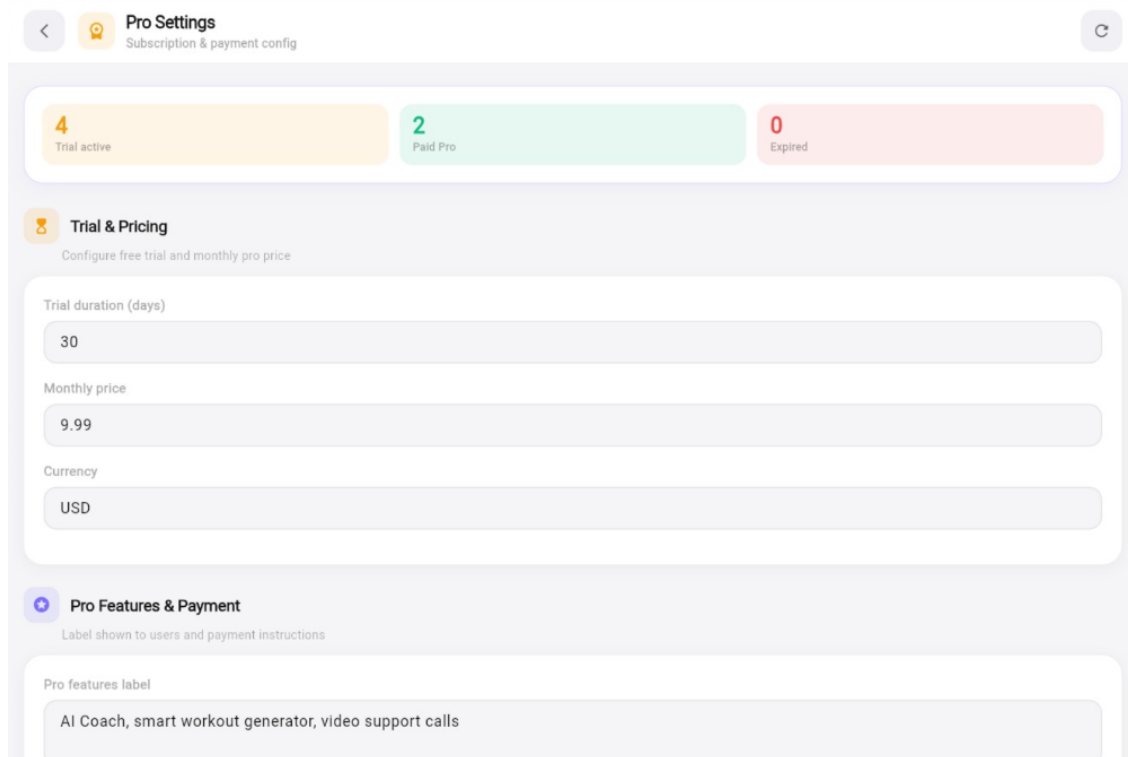


Figure 71:Pro setting screen

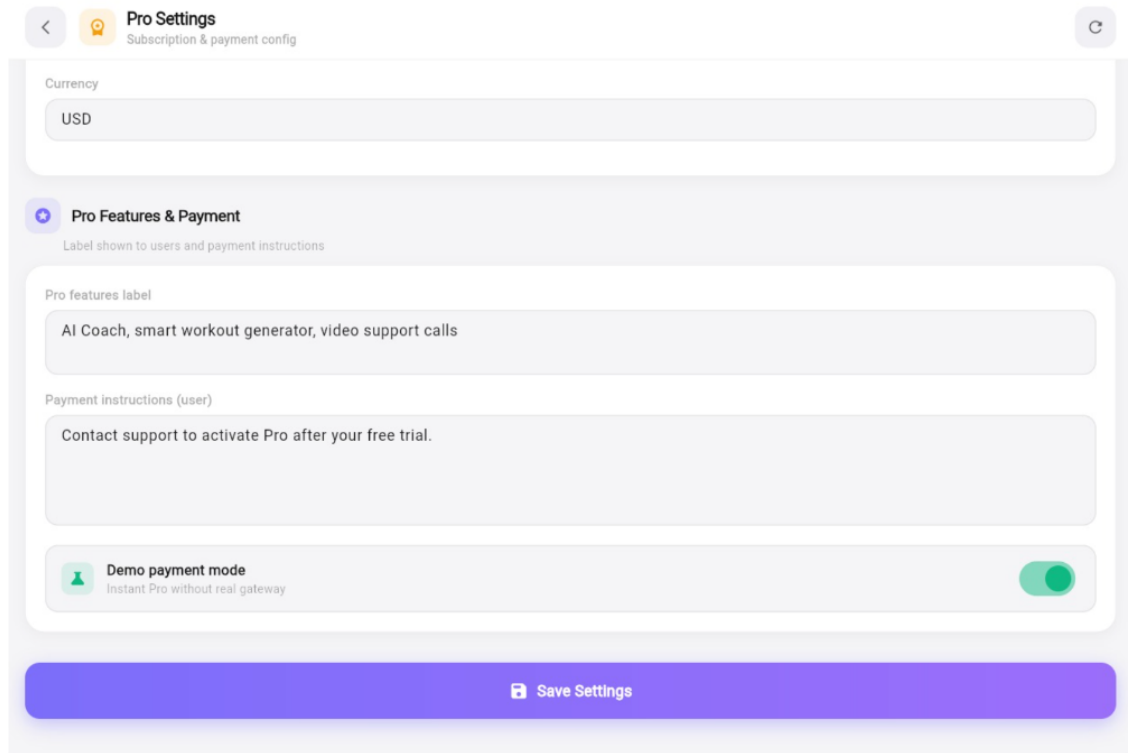


Figure 72:Pro setting screen

3.8.3 Support Inbox & Ticket Management

The platform incorporates a unified communication auditing center named the **Support Inbox** . This sub-module acts as a centralized dashboard allowing system administrators to monitor, review, and oversee multi-role ticket threads and chat channels throughout the system ecosystem.

1. Inbox Entry Trigger

- **Header Support Control:** Administrators can access this center by tapping the dedicated customer-service head-set icon located at the top right margin of the global Admin Panel view header .

Global Access Matrix: Tapping this icon queries and compiles all active communications into a structured, real-time ticket ledger view.

2. Cross-Role Conversation Auditing

The inbox aggregates conversation streams from different departments into an itemized chronological ticket matrix :

User-to-Coach Streams: Isolates tickets originating from or directed to fitness trainers (e.g., tagged with an explicit green **coach** role badge like *reem* under **Ticket #4**), highlighting workout schedules or video support call appointments.

User-to-Nutritionist Streams: Displays direct consult history with clinical dietary specialists (e.g., tagged with an orange **nutritionist** role badge under tickets like **Ticket #3**, **Ticket #2**, and **Ticket #1**).

Admin Support Inquiries: Serves as the primary landing terminal for incoming user help desk messages, technical errors, and direct requests targeted at admin assistance (e.g., specific user requests like *"i need get 10 kilo in 2 months"* or *"hi can give me a daily meal"*).

3. Structured Ticket Metadata Interface

Every indexed conversation row presents immediate operational parameters to help administrators judge and filter tickets quickly at a glance :

Ticket Indexing: Each block maps a unique identifier header (e.g., **Ticket #1** through **Ticket #4**) to preserve communication logs.

Context Micro-Badges: Identifies the explicit user account nickname paired dynamically with its respective color-coded role category badge (coach vs nutritionist).

Message Snapshot Snippets: Previews either the literal first text string submitted by the user or an active service category type descriptive label (e.g., "📺 **Video call**").

Deep-Dive Navigation Indicators: Each card features a rightward-facing navigation arrow indicator (>), letting administrators tap into any row to unlock full contextual message histories and audit the corresponding communication log.

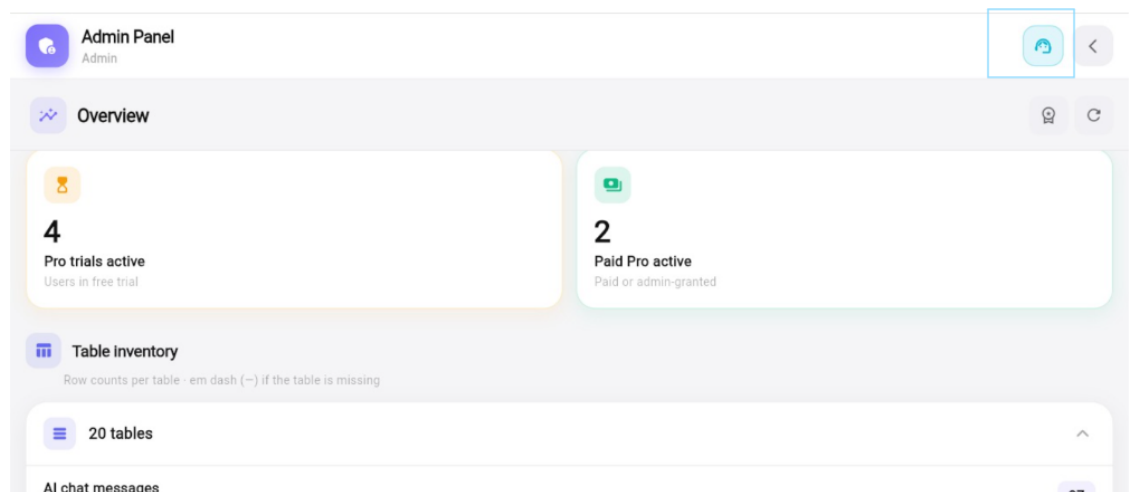


Figure 73:Admin panel(support inpx button) screen

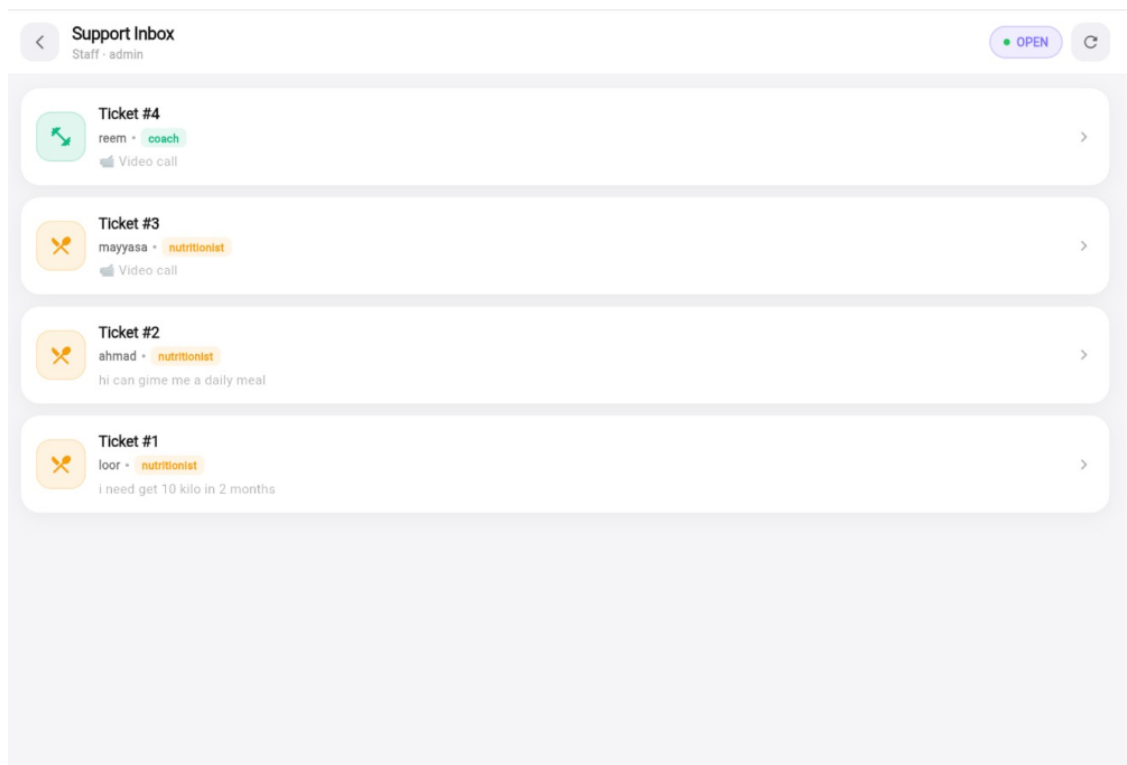


Figure 74:Support innox screen

3.8.4 Members Directory & User Profile Management

The **Members** tab establishes a comprehensive administration management ledger allowing system operators to search, audit, configure, and maintain individual user accounts registered across the database ecosystem.

1. Members Directory Layout & Indexing

The primary landing layout under the members route focuses on high-capacity account discovery :

Global Search Utility: Features an omni-search input bar labeled **"Directory"** supporting advanced inline parameters.

Administrators can search dynamically by account name, unique ID, email string, wellness goal, or explicit account roles .

Account Registry Cards: Displays structured user rows containing the user's name, email address, unique indexing identifier (e.g., ID 13), and a primary role tag (e.g., User) to allow immediate scannability .

2. Detailed User Management Panel (User Deep-Dive)

Tapping any account row from the directory layout shifts the administrator's terminal into an itemized profile control deck (e.g., entering **User #13** profile view). This sub-interface breaks down into five key operation layers:

A. Data Footprint Telemetry Matrix

An automated tracking grid that displays exactly how much personal metadata the user has synchronized with the cloud tables :

Logged meal intake: Displays absolute meal database row increments (e.g., 4).

Workout minutes log: Monitors synchronized fitness routine records (e.g., 2).

Water entries & Calorie snapshots: Reflects vital lifestyle logs (e.g., 5 water logs, 1 calorie snapshot).

AI chat messages: Tracks interactions generated inside the premium automated guidance engine (e.g., 4 chat logs).



B. Pro Subscription Controls & Trials Overrides

A dedicated billing and monetization management deck configured to adjust premium membership variables manually without leaving the user's view:

Status Readout: Shows the active subscription tier and exact time thresholds remaining (e.g., **Active • trial / 20 days remaining**).

+30 trial days Button: Appends a 30-day evaluation extension increment directly to the user's active billing profile row.

Grant Pro 30d Button: Manually forces an administrative, complimentary premium tier override for 30 full days.

Restart trial Button: Wipes active subscription dates to reset the standard registration evaluation lifespan back to day one.

Revoke Pro Action: An instant administrative override that terminates active premium status, reverting the member back to the standard free tier.

C. Access Level & Multi-Role Provisioning

An identity management dropdown allowing administrators to elevate or demote system-wide authority metrics on the fly:

App role Dropdown Menu: Clicking this selector triggers an overlay containing four system roles: **user**, **coach**, **nutritionist**, and **admin**.

Safety Guardrail: The interface includes built-in security logic preventing active administrators from accidentally demoting their

own admin security tier from this route (*"you cannot demote your own admin from here"*).

D. Profile Metadata Editor

An editable text segment mapping core body composition attributes and account basics directly from the cloud cluster:

Fields Captured: Provides manual overwrites for Name, Email, Age, Height, Weight, and specific destination targets (e.g., **Goal key: lose_weight**).

Commit Actions: Tapping the green full-width **"Save profile"** bar commits edited values to the specific database rows instantly.

E. Data Maintenance & Danger Zone Infrastructure

Positioned at the baseline of the deep-dive controller are destructive structural commands reserved for specialized support scenarios :

Clear logs & tracking Button: Flushes temporary data arrays, custom tracking metrics, and historical logs without destroying the core account profile or credentials.

Full reset (+ meal/workout plans) Button: Performs a deeper wipe, clearing all generated digital routines, nutrition matrices, and customized assets.

Delete user permanently Button: Located within the distinct **Danger zone** card block, triggering a cascade delete function that permanently drops all database rows tied to the user's unique identification key across all tables.

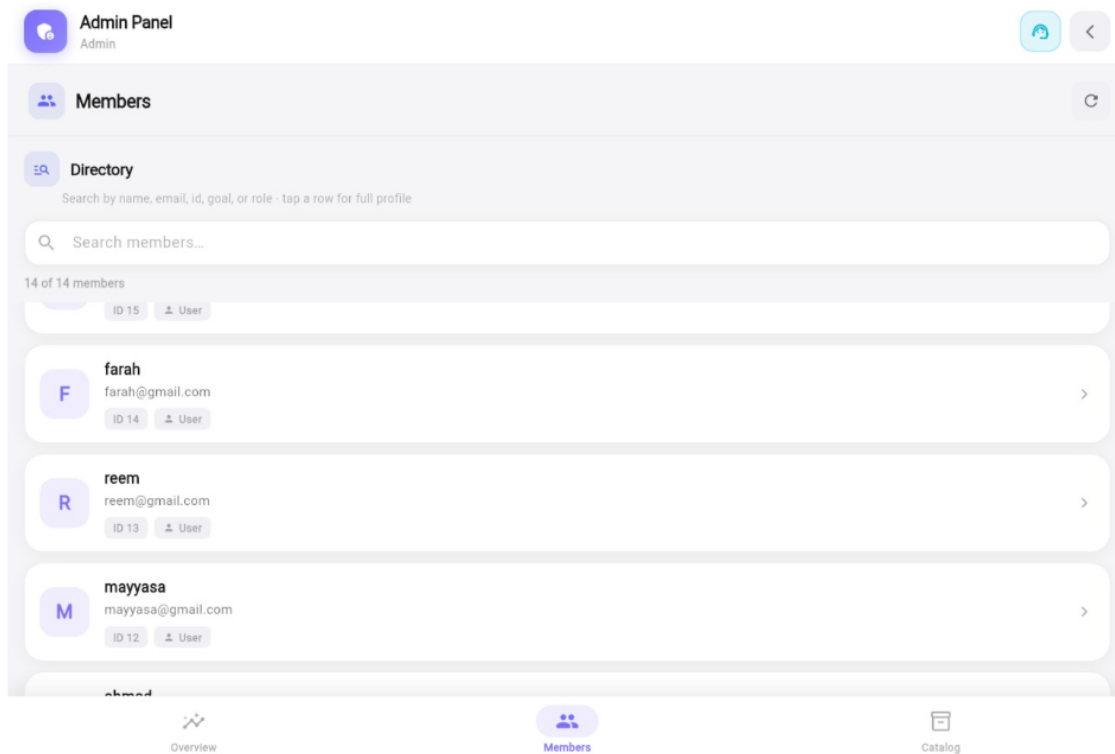


Figure 75:Members screen

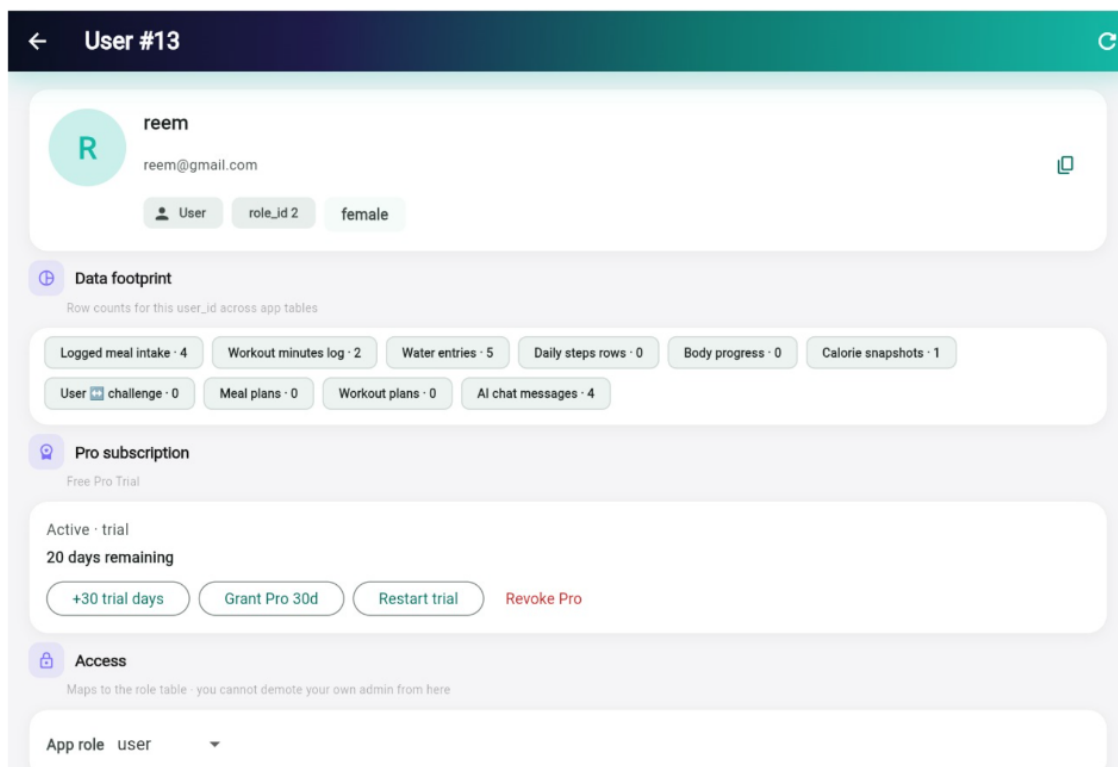


Figure 76:Member screen

←

User #13

Pro subscription

Free Pro Trial

Active · trial

20 days remaining

+30 trial days

Grant Pro 30d

Restart trial

Revoke Pro

Access

Maps to the role table - you cannot demote your own admin from here

App role

user

▼

Profile

Saved to users row - invalid numbers are skipped on save

Name

reem

Email

reem@gmail.com

Age

29

Height

164

Weight

89

Figure 77:Member screen

←

User #13

Email

reem@gmail.com

Age

29

Height

164

Weight

89

Goal key (e.g. lose_weight)

lose_weight

Save profile

Data maintenance

Clears logs and generated plans for support – confirm dialogs list scopes

Clear logs & tracking

Full reset (+ meal/workout plans)

Danger zone

Permanent delete cascades related rows

Delete user permanently

Figure 78:Member screen

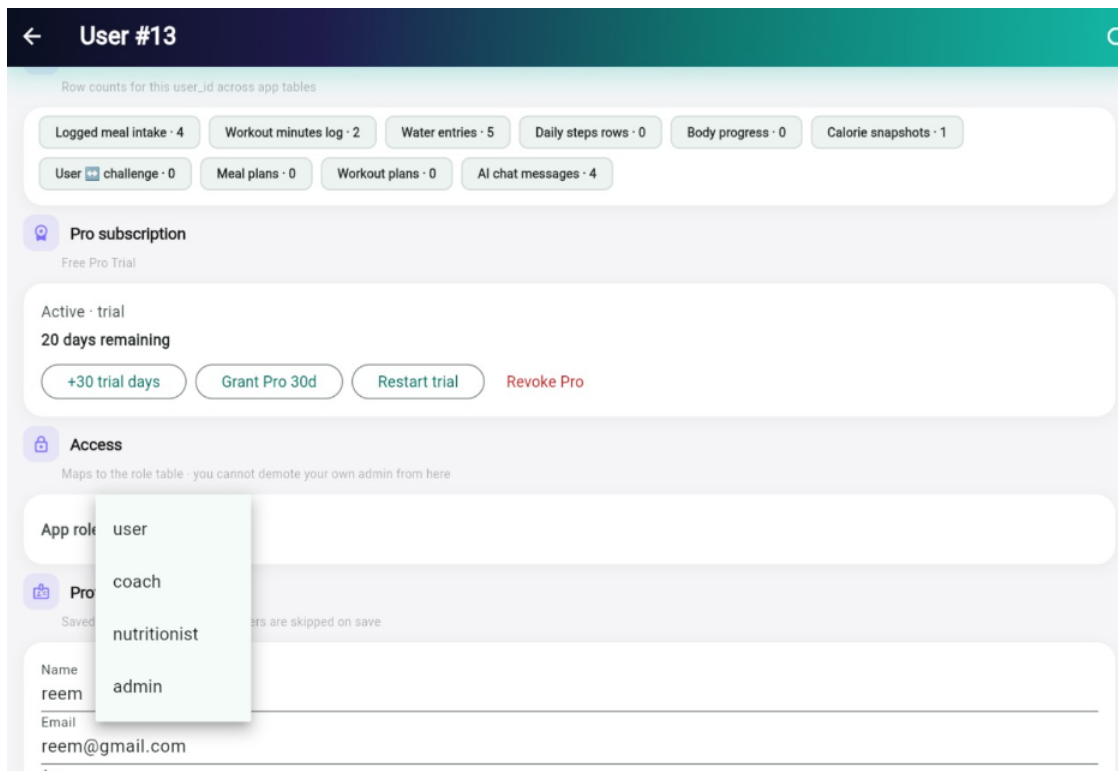


Figure 79:Member screen(role)

3.8.5 Content Library & Catalog Management

The **Catalog** tab navigates administrators directly into the system's global **Content Library** portal . This centralized configuration module gives operators full CRUD (Create, Read, Update, Delete) capability over core data items, lifestyle resources, macro configurations, and active fitness challenges deployed across the application infrastructure.

The Content Library is systematically cataloged into five functional sub-tabs:

1. Meals Management Inventory (Meals)

This interface indexes the baseline nutritional assets accessible to specialists and users for automated meal provisioning :

Itemized Nutrition Logs: Every entry displays its global string name (e.g., *Oatmeal*, *Edamame*, *Boiled Eggs*), unique index key (e.g., id 72), and calculated caloric payload value (e.g., 250 kcal).

Creation & Removal Actions: Operators can instantly append new culinary options to the central asset array via the floating "+ **Add meal**" action command, or drop stale options globally via individual red trash bin action icons.

2. Exercises Database (Exercises)

A master list containing the global fitness and movement library framework utilized by coaches or generative AI routines :

Movement Logs: Lists specific movement protocols alongside distinct database tags (e.g., *Hip Flexor Stretch* id 91, *Yoga Flow - Beginner* id 89, *Box Jumps* id 86).

Database Maintenance: Incorporates a prominent orange "+ **Add exercise**" button to scale the exercise library dynamically, backed by isolated cascading delete buttons on each row.

3. Challenges Repository (Challenges)

Manages gamified system-wide retention activities and performance objectives designed to drive user lifestyle engagement :

Engagement Indicators: Rows hold preset target headers such as *7 Day Water Challenge* (id 1), *30 Day Fitness Challenge* (id 2), and *No Sugar Week* (id 6).

Expansion Node: Features a yellow full-width corner trigger labeled "+ Add challenge" to push fresh structured criteria and milestones to active tiers.

4. Macro Food Categories Hierarchy (Food cats)

Establishes the macro-group taxonomies that govern how recipes, logging systems, and dietary search parameters behave :

Taxonomy Matrix: Lists structural categories including *Proteins* (id 1), *Vegetables* (id 2), *Fruits* (id 3), *Grains & Carbs* (id 4), *Dairy* (id 5), and *Nuts & Seeds* (id 6) and else.

Structural Modifications: Integrates a green "+ Add category" control node to safely scale parent classification strings, coupled with row-level deletions to clean out obsolete database references.

5. User Challenge Assignments Ledger (Assign)

The operational bridging hub where administrators audit or manually forge direct relationships between specific users and system-wide challenges :

Relational Mapping Rows: Explicitly displays transactional linkages mapping a user identifier to a specific challenge entry (e.g., **User 1 → 30 Day Fitness Challenge**).

State Monitors: Color-coded structural pills verify the real-time interaction progress of each instance—marking them dynamically as green **active** or green **completed** arrays.

Link Interruption & Provisioning Controls:

Manual Binding Tool: Features a prominent "🔗 **Link challenge**" launcher, enabling manual forced assignment of challenge schemas onto user profiles immediately.

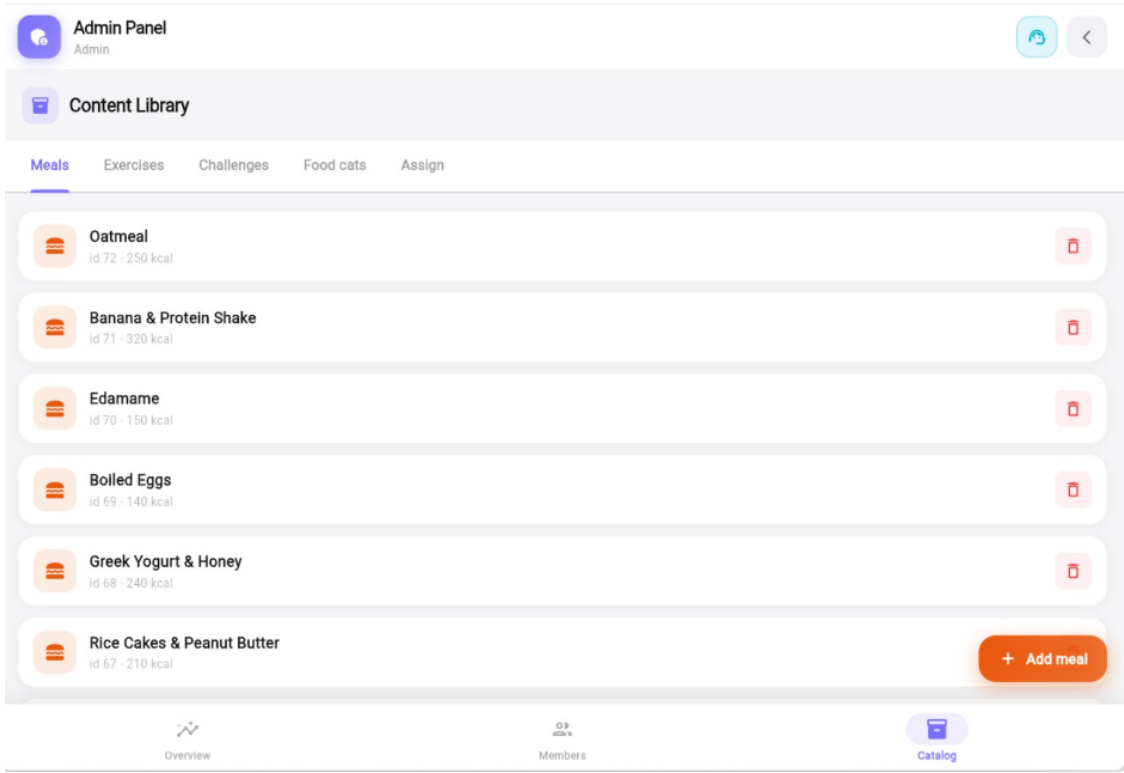


Figure 80:Meals screen

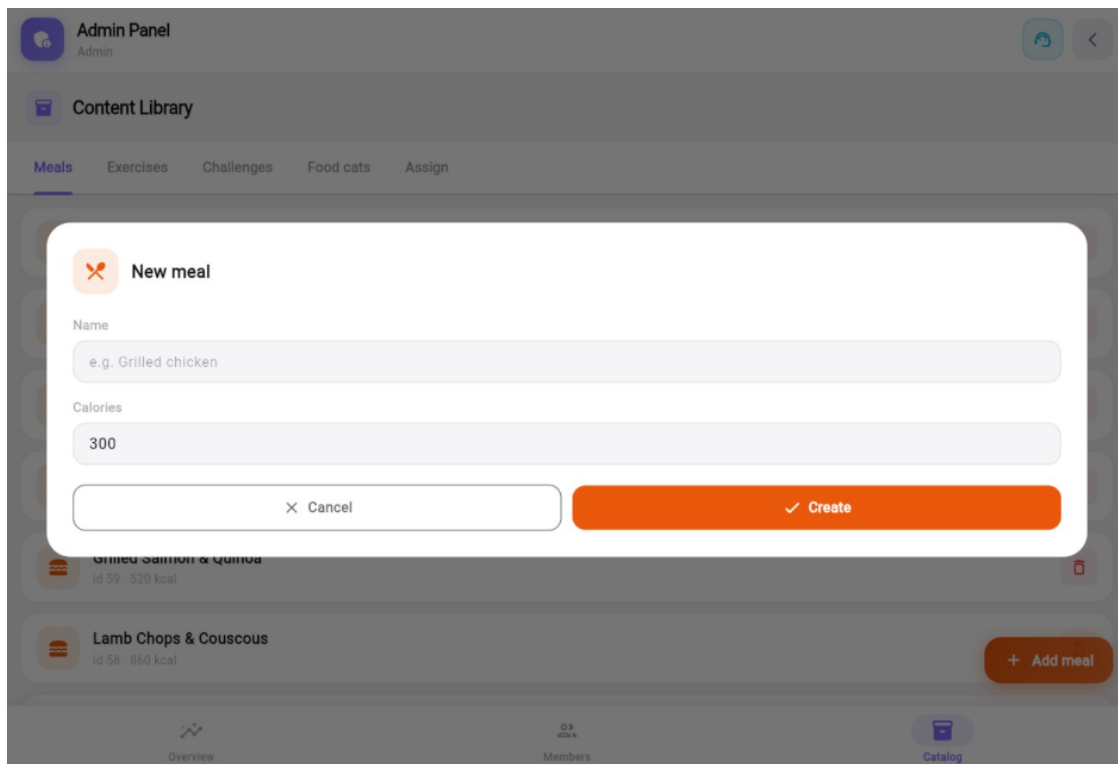


Figure 81:Meals screen

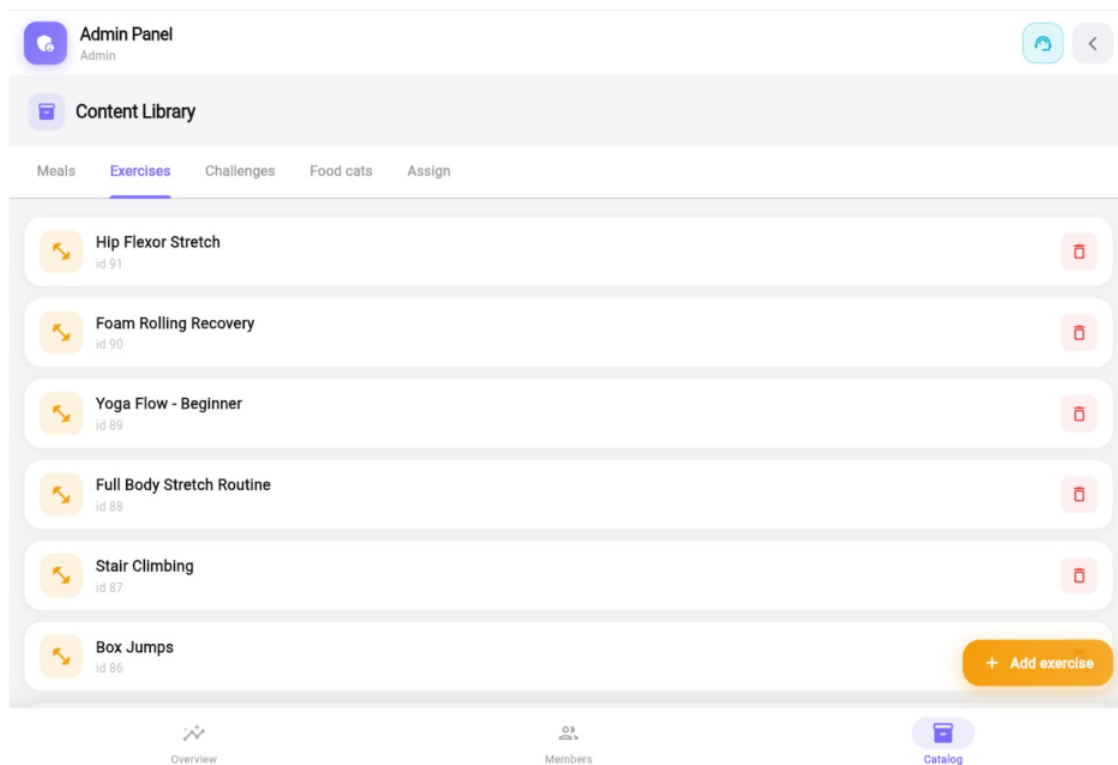


Figure 82:Exercises screen

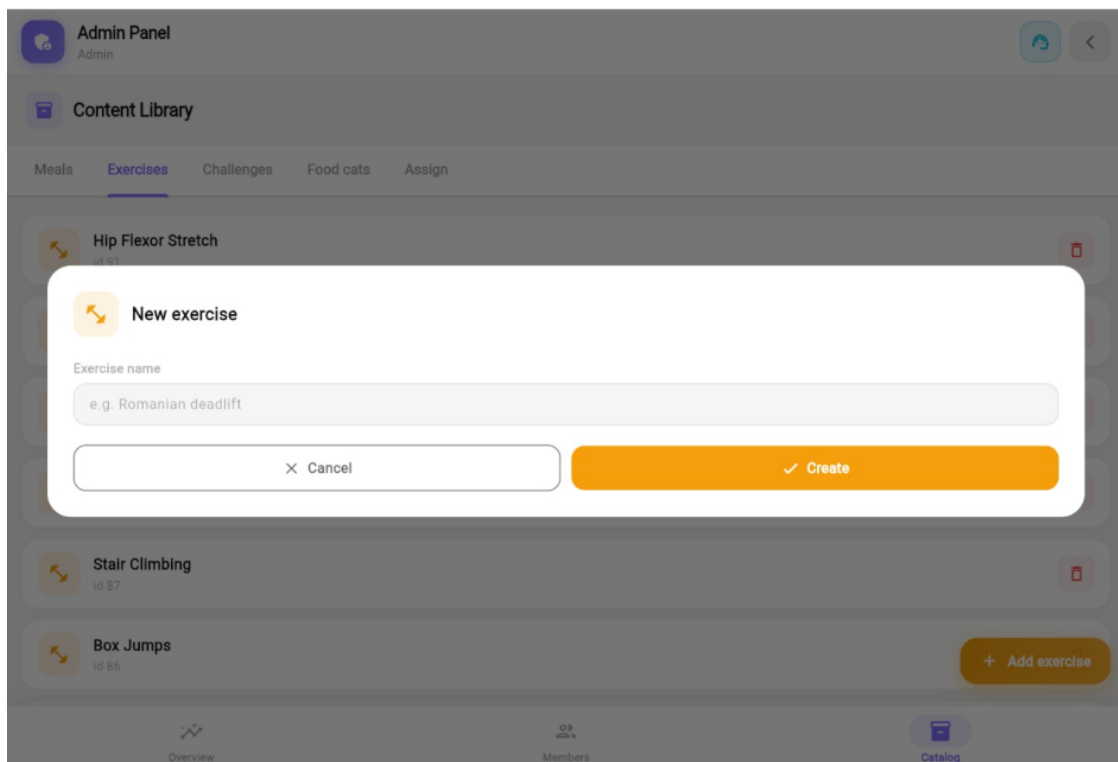


Figure 83: Add exercises screen

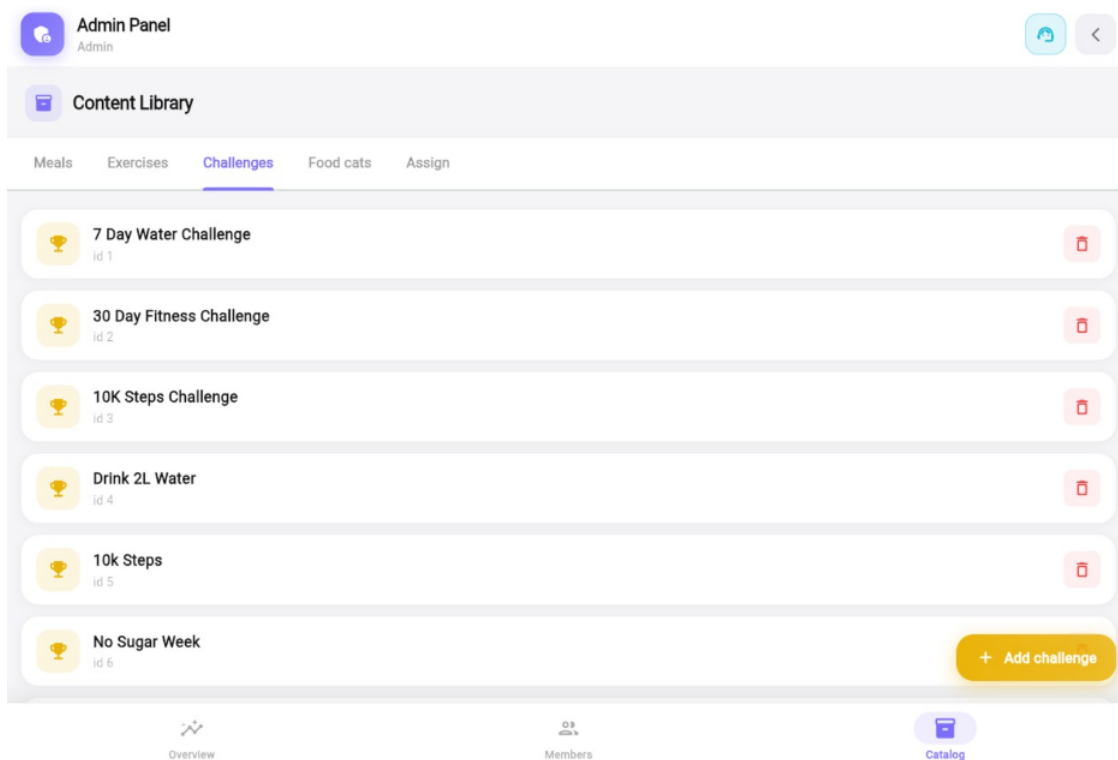


Figure 84: Challenges screen

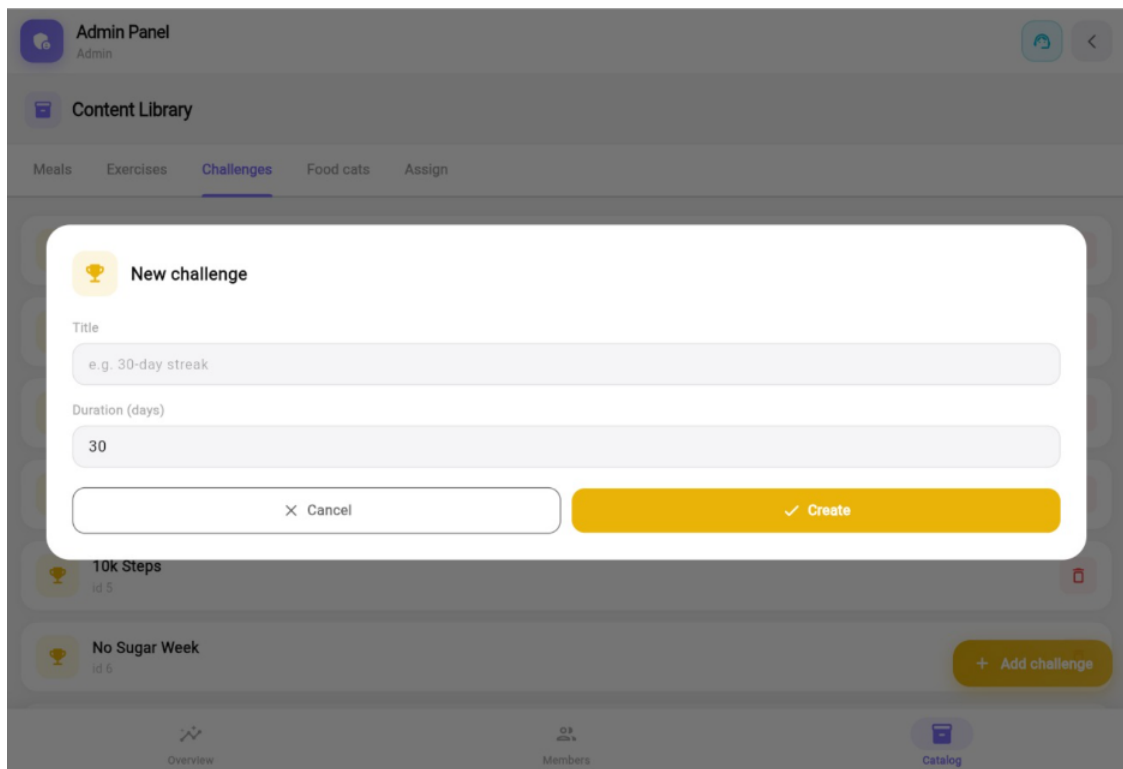


Figure 85: Add challenges screen

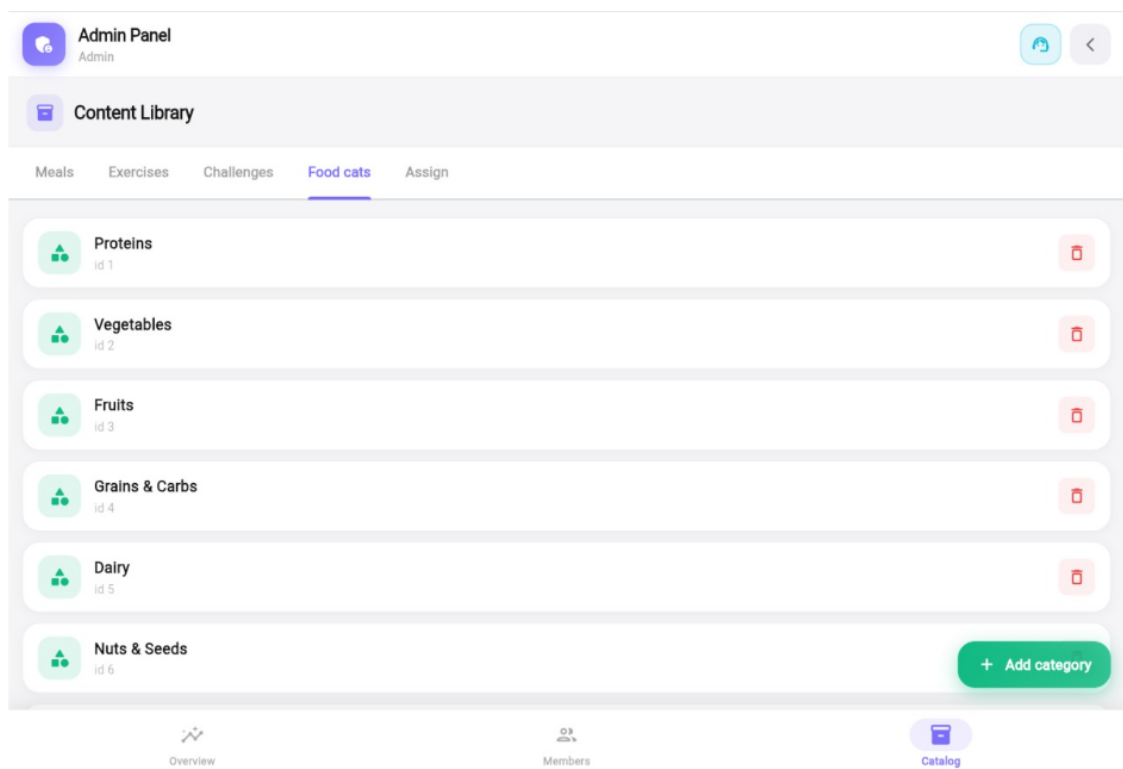


Figure 86: Food category screen

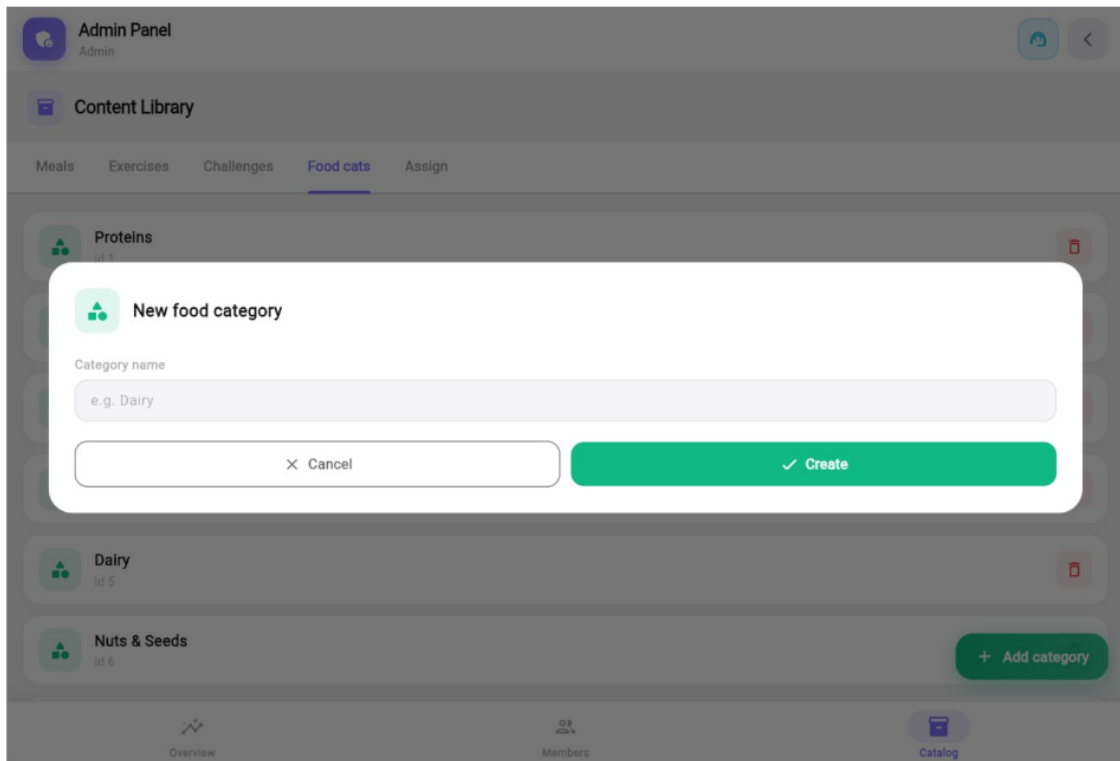


Figure 871:Add food cats screen

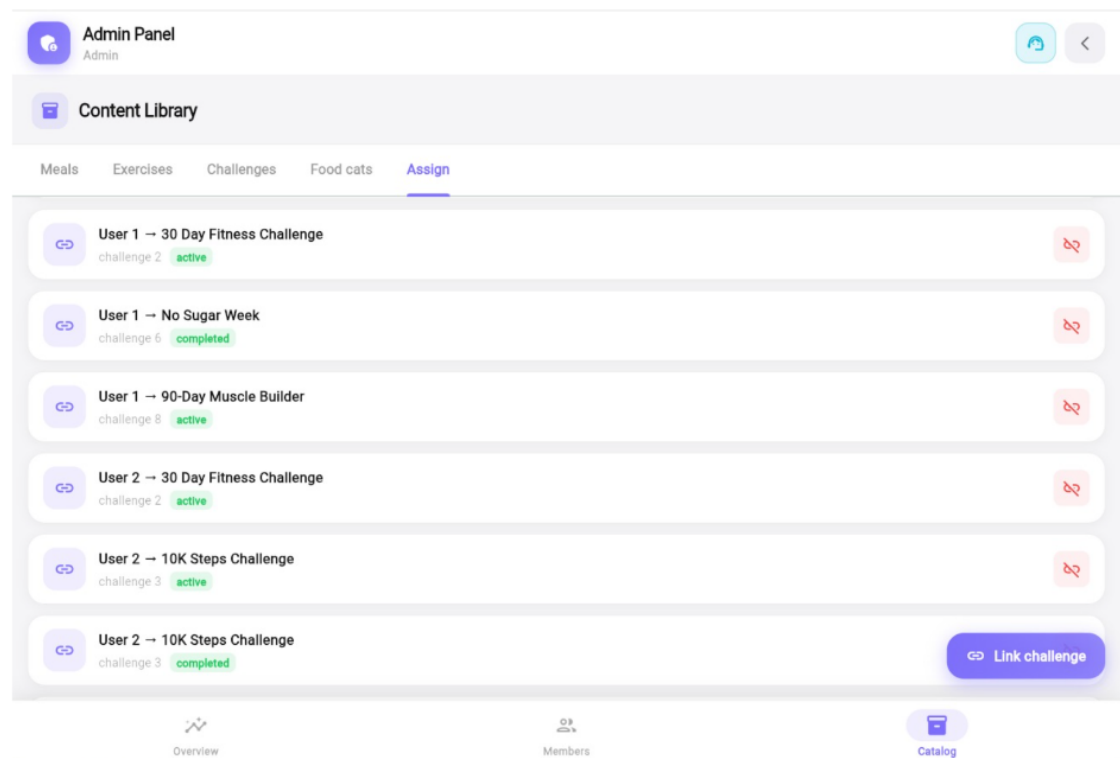


Figure 88:Add food cats screen

3.9 Nutritionist Portal (Nutrition Hub)

The **Nutrition Hub** provides licensed nutritionists with a specialized dashboard tailored for dietary oversight, patient tracking, and nutritional optimization without exposing structural infrastructure controls .

3.9.1 Members Directory (Nutritionist View)

The primary landing sub-module maps directly onto the client registry layout :

Targeted Directory Search: Includes an optimized search bar to locate specific clients by name, email, unique system ID, or health goal .

Client Ledger Cards: Displays clear rows indicating active users (e.g., *farah*, *reem*, *mayyasa*, *ahmad*) along with their global database index keys .

Global Support Shortcut: Retains the specialized customer-service headset icon in the upper margin to jump directly into designated chat ticket channels with clients .

3.9.2 Client Profile Deep-Dive & Read-Only

Governance

When a nutritionist taps into an itemized user file (e.g., entering **User #10** profile view), the system renders a specialized monitoring layout .

Unlike the Admin view, this sub-interface enforces a secure **Read-Only** access structure to protect core identity rows while providing clinical visibility :

1. Dietary & Lifestyle Telemetry Block

Provides the nutritionist with an immediate diagnostic snapshot of the user's operational lifestyle footprint across the application tables :

Logged meal intake: Monitors the volume of meals tracked by the user (e.g., 0).

Calorie snapshots: Checks user-submitted metabolic metrics or calorie capture logs (e.g., 1).

Activity Logs: Tracks synced variables such as *Workout minutes log*, *Water entries*, *Daily steps rows*, and *Body progress* counters .

AI chat messages: Tracks client reliance on automated assistant nodes (e.g., 1).

2. Read-Only Personal Profile Matrix

Displays the client's explicit bio-composition metrics compiled directly from their health record :

Data Fields Rendered: Shows the user's Name (*loor*), Email (*loor@gmail.com*), Age (*20*), Height (*160*), Weight (*50*), and active target objective (**Goal key: maintain**).

Security Access Constraint: Accompanied by a clear inline operational disclaimer: "**Read-only for coaches & nutritionists**" (image_0d7ed9.png). This safety mechanism blocks the nutritionist from modifying the client's baseline profile parameters or deleting historical rows, limiting their focus entirely to nutritional guidance and custom meal mapping under the secondary **Nutrition** tab.

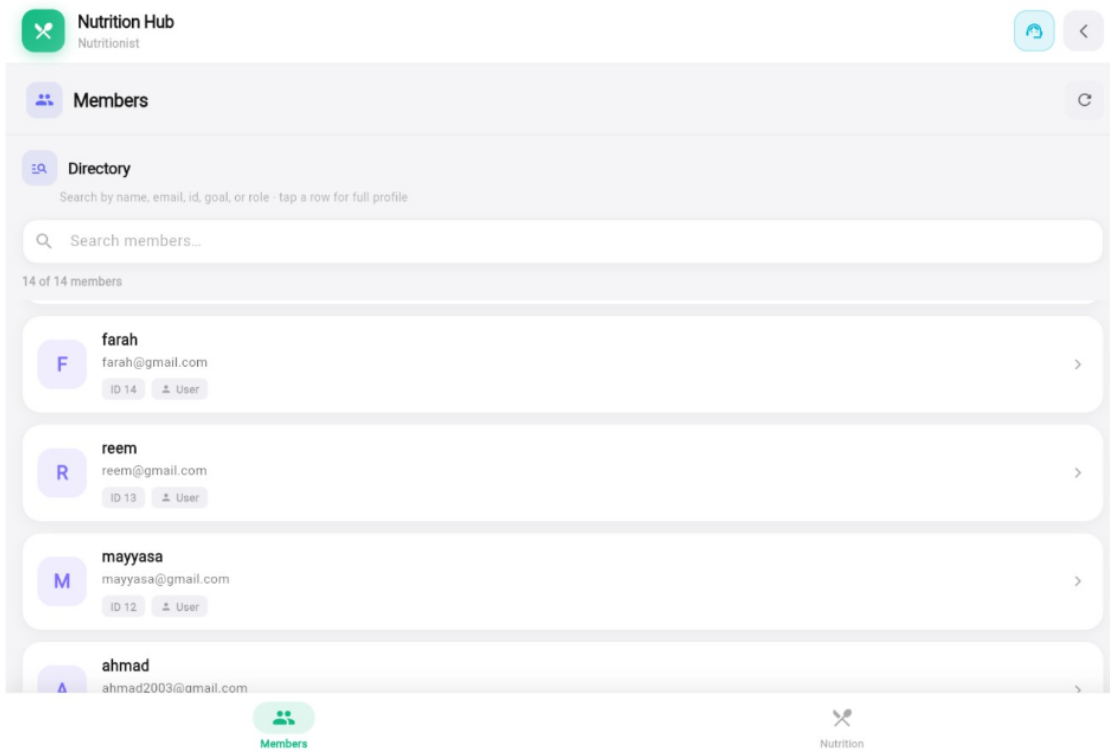


Figure 89:Members screen

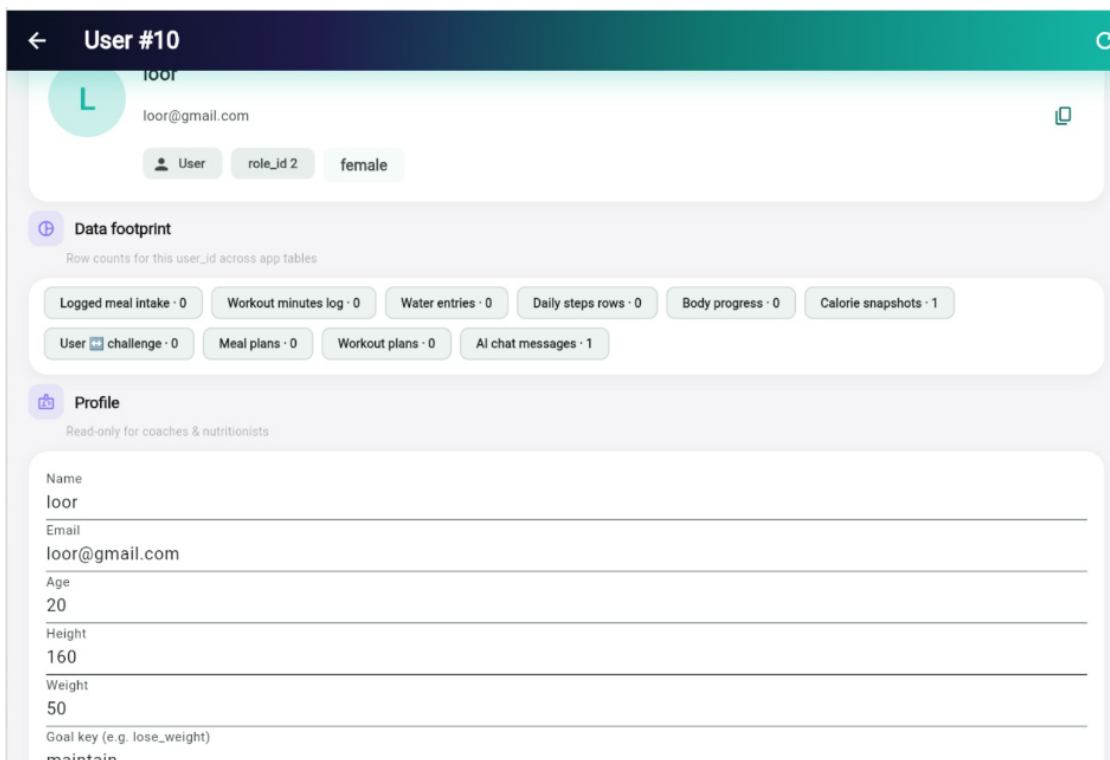


Figure 90:Member screen

3.9.3 Nutritionist Content Library & Isolated Support

When swapping from the clients ledger to the **Nutrition** module via the global bottom navigation layout, the platform renders the **Nutrition Library**. This interface isolates dietary management functions, allowing the specialist to update the platform's core dietary assets independently of global system configurations.

1. Meals Management Panel (Meals)

The primary sub-tab acts as a localized portal for managing macro and micro dietary options inside the cloud food cluster:

Dietary Inventory Tracking: Lists items dynamically with their system identifier strings, database index keys, and calculated energy payload metrics (e.g., *Oatmeal* id 72 · 250 kcal, *Banana & Protein Shake* id 71 · 320 kcal, *Edamame* id 70 · 150 kcal).

Resource Scaling Utilities: Specialists can seamlessly introduce newly configured recipes into the system using the orange "+ **Add meal**" trigger button. Conversely, obsolete meal items can be dropped from the active options matrix using row-level deletion triggers.

2. Food Categories Architecture (Food cats)

The secondary sub-tab exposes the classification taxonomies that govern database sorting, recipe generation, and menu mapping:

Taxonomy Strings: Features localized macro classifications such as *Beverages* (id 9), *Snacks* (id 10), *Soups & Stews* (id 11), *Breakfast Foods* (id 12), *Desserts* (id 13), and *Oils & Fats* (id 14).

Dynamic Extension Node: Includes a green contextual action bar labeled "+ Add category" to extend database indexing structures on the fly, supported by inline removal bins to delete outdated categories.

3. Isolated Staff-Level Support Inbox

By accessing the top-margin support trigger, the nutritionist enters a secure, role-filtered **Support Inbox** tailored explicitly to their credential scope (Staff · nutritionist):

Role-Isolated Thread Filtering: The interface automatically filters out general system faults or trainer-focused workout streams. It strictly exposes threads flagged with the orange **nutritionist** assignment badge.

Direct Consult Management: Directly links the specialist to incoming patient inquiries or service requests assigned to their queue (e.g., tracking active consultations like *mayyasa* under **Ticket #3** for a video call, *ahmad* under **Ticket #2** requesting a daily meal, or *loor* under **Ticket #1** seeking a 10kg weight gain plan over 2 months).

Operational Status Toggles: Integrates a prominent status micro-pill displaying • **OPEN** to confirm that the messaging queue is live, enabling real-time responses to critical patient inquiries.

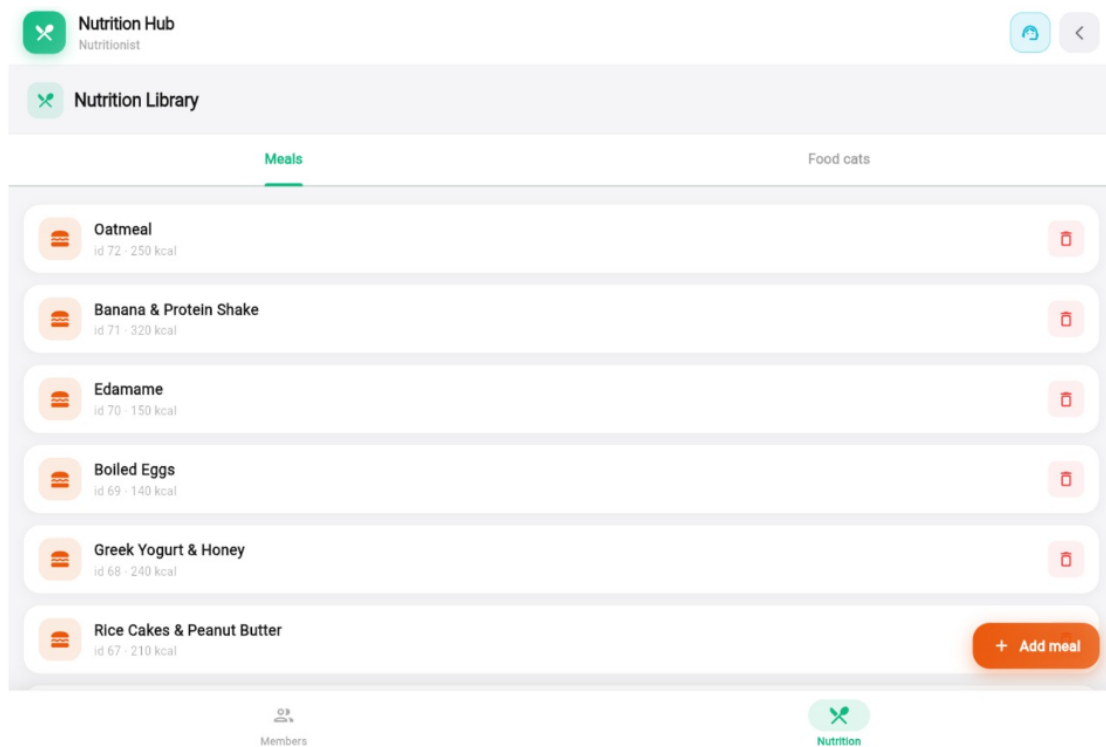


Figure 91:Meals screen

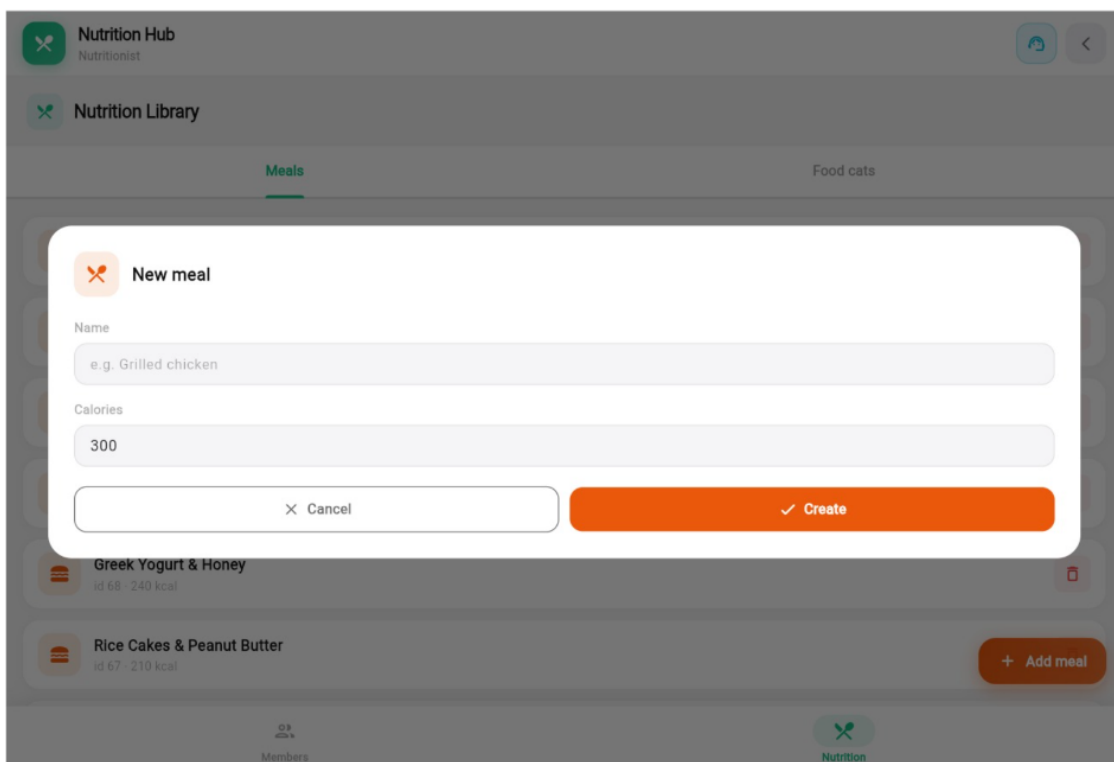


Figure 92:Add meal screen

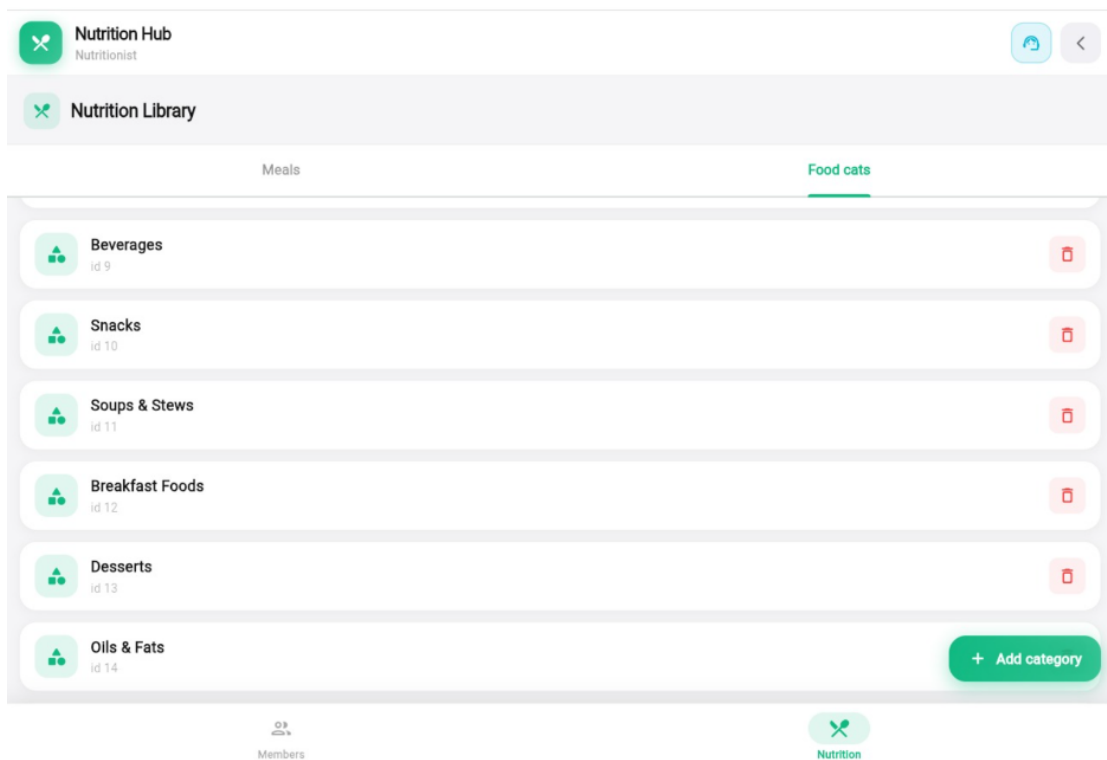


Figure 87;Food cats screen

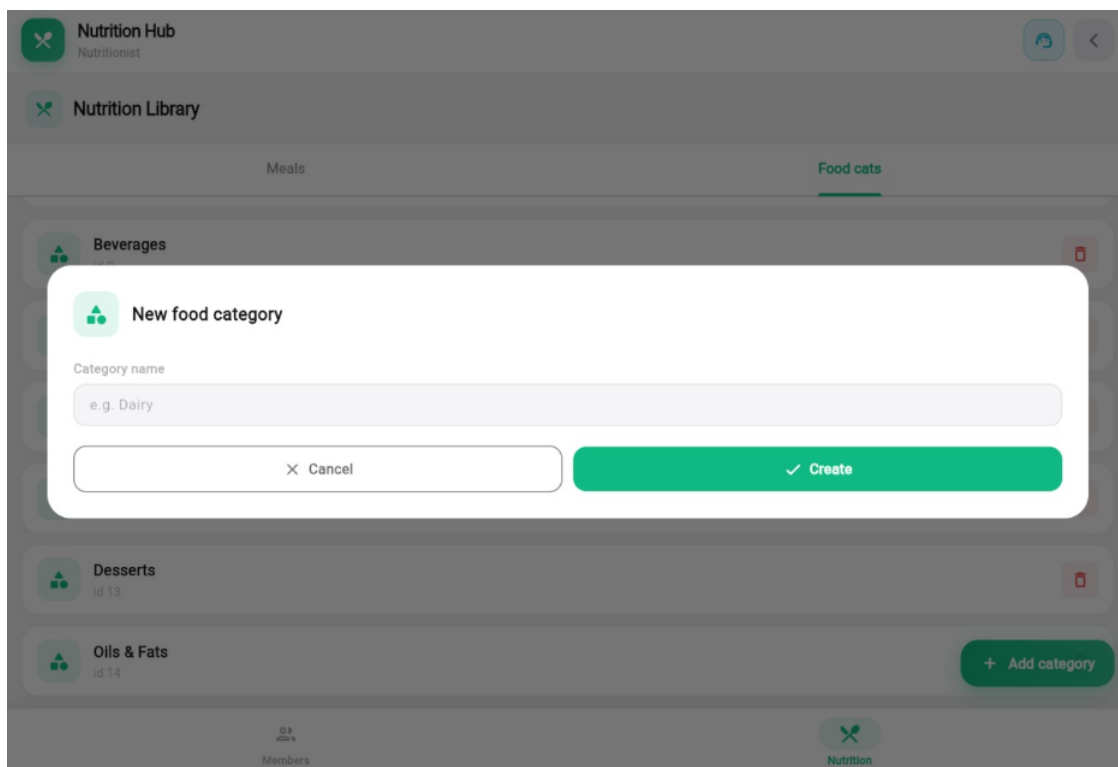


Figure 88:Add food cat screen

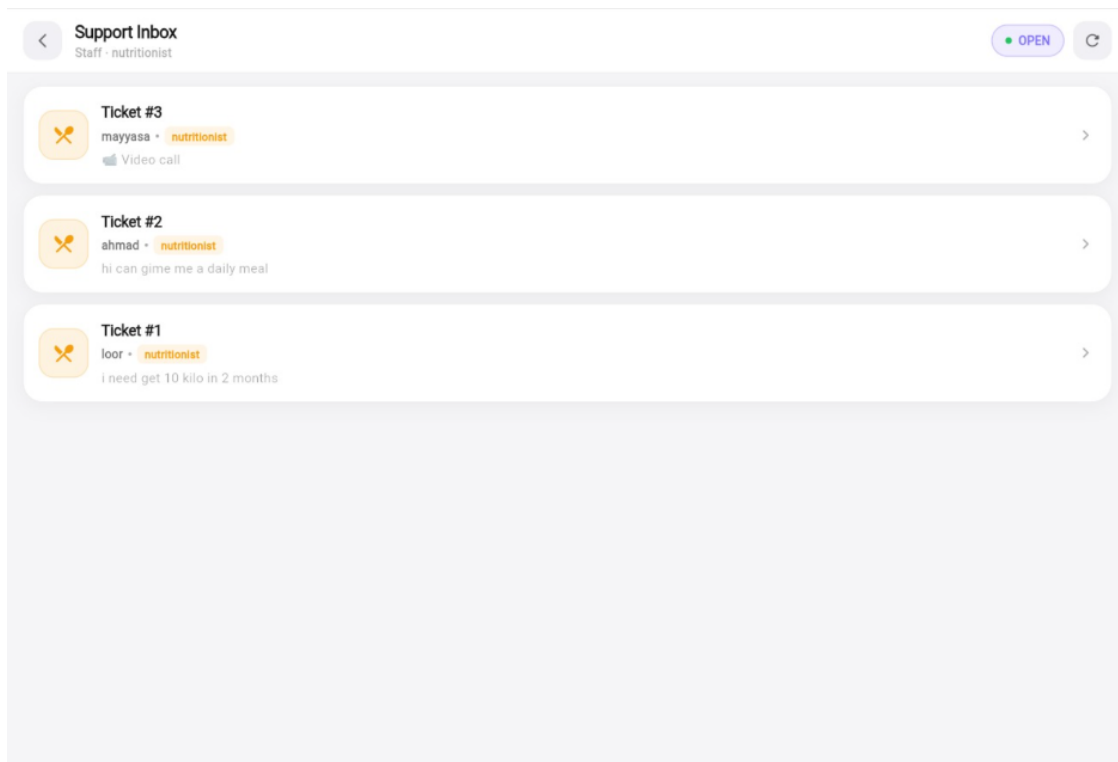


Figure 89:Support inbox screen

3.10 Coach Portal (Coach Hub)

The **Coach Hub** provides fitness coaches and personal trainers with a dedicated interface focused on physical assessment, movement progression tracking, and isolated training program support. This module maintains an orange visual theme across its core action grids.

3.10.1 Members Directory & Assessment Registry

The primary navigation node maps onto the member directory framework, filtered for coaching workflows:

Global Member Indexes: Displays active clients registered across the tenant schema (e.g., *ahmad* id 11, *loor* id 10). It also highlights internal cross-role profiles, such as *hala* who carries a green **Nutritionist** badge.

Read-Only Telemetry Inspection: Tapping a client profile (e.g., **User #14 - farah**) surfaces their real-time application footprint and physical biometrics:

Activity Footprint: Audits tracked habits including *Workout minutes log* (0), *Daily steps rows* (0), and *Calorie snapshots* (1).

Biometric Profile Readout: Enforces an explicit **"Read-only for coaches & nutritionists"** policy. It reveals core data items like Age (30), Height (162), Weight (77), and their self-stated target objective (**Goal key: gain_weight**).

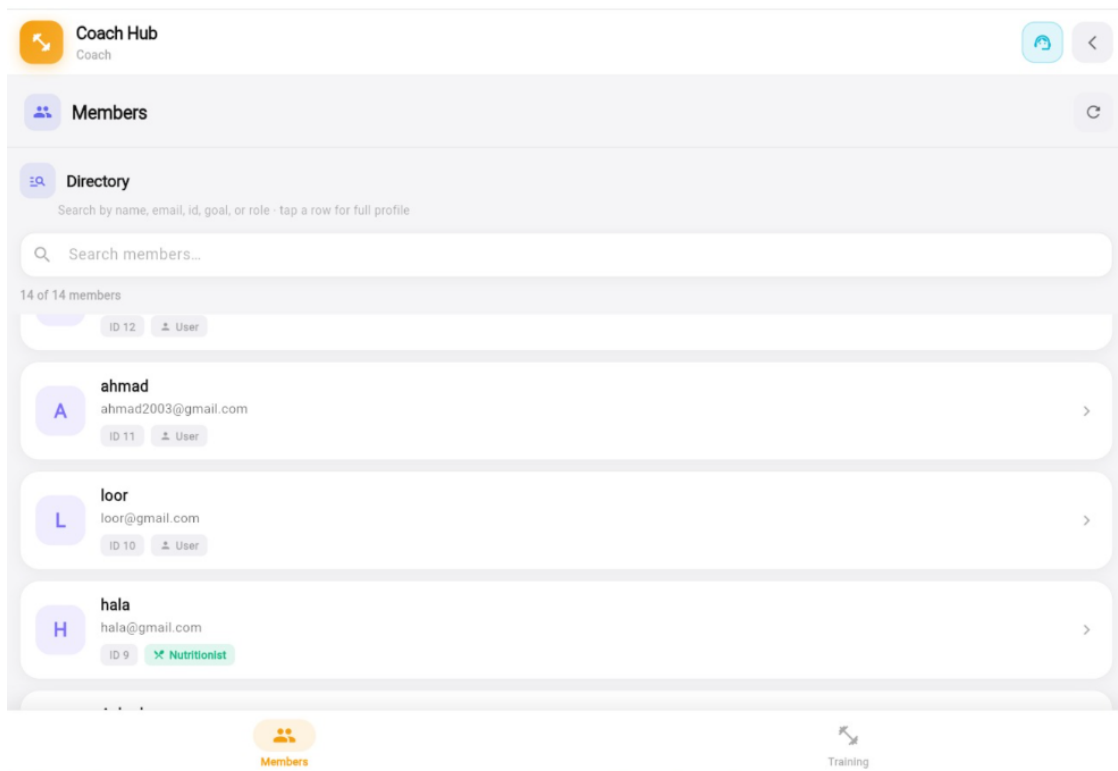


Figure 93:Members screen

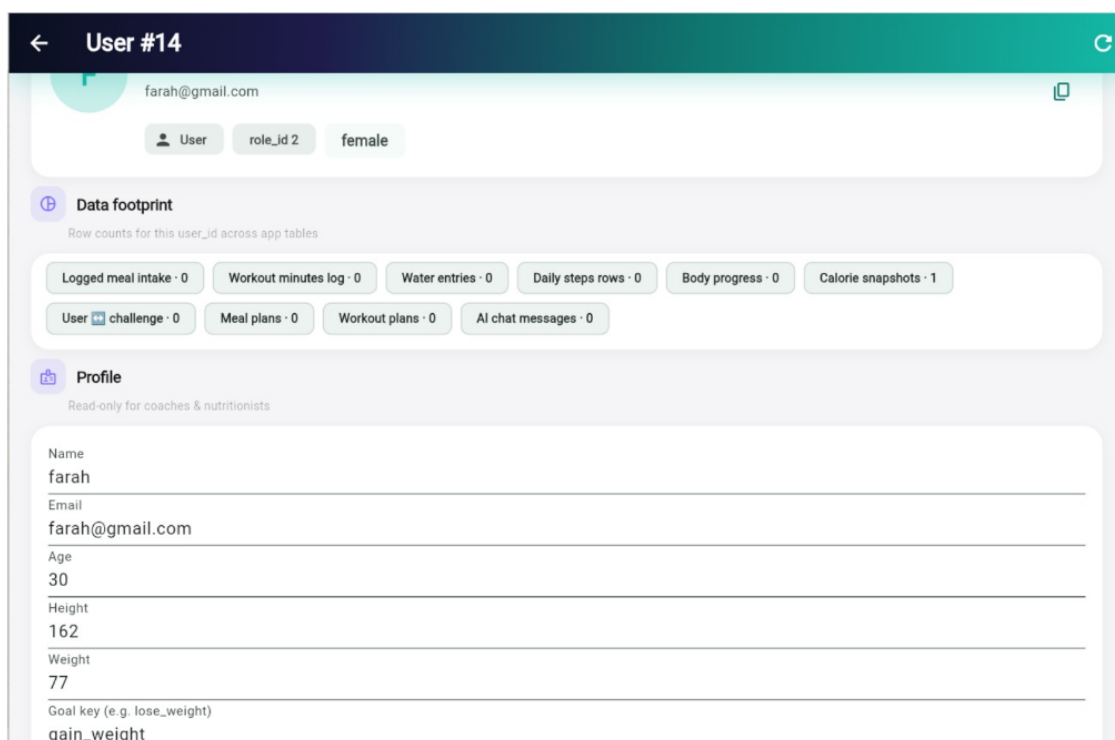


Figure 94:Member screen

3.10.2 Exercises Database & Training Library (Training)

Swapping to the bottom **Training** tab directs coaches into the unified exercise inventory portal:

Structural Movement Inventory: Lists all core exercises available for workout prescription along with their persistent database keys (e.g., *Hip Flexor Stretch* id 91, *Foam Rolling Recovery* id 90, *Yoga Flow - Beginner* id 89, *Box Jumps* id 86, *Swimming* id 85).

Global Deletion Action: Features dedicated red trash bin items on each row to let certified staff drop obsolete or unsafe training schemas instantly from the centralized data array.

Exercise Provisioning Workflow (Modal Pop-up)

Tapping the prominent orange "+ **Add exercise**" button launches an overlay form window designed for database scaling:

Dynamic Form Fields: Prompts the operator with an Exercise name text input field featuring contextual placeholders (e.g., *e.g. Romanian deadlift*).

Transaction Controls: Integrates a soft gray "× **Cancel**" option to safely abort the thread, and a high-contrast orange "✓ **Create**" action node to execute the insert query and push the new string directly to the production cluster.

3.10.3 Isolated Coach-Level Support Inbox

By accessing the top-margin customer-service headset icon, the coach navigates to a role-isolated messaging channel designed specifically for training consultations:

Staff-Role Filter: The header locks the visibility scope strictly to the active staff assignment tier (Staff · coach).

Targeted Ticket Queue: Displays inbound support threads explicitly flagged with a green **coach** routing badge. For example, **Ticket #4** maps a direct consultation request from *reem* seeking a live **Video call** session regarding her routine.

Status Monitors: Embeds a top-right • **OPEN** pill indicating a live connection queue, ensuring coaches can respond to client fitness blocks promptly.

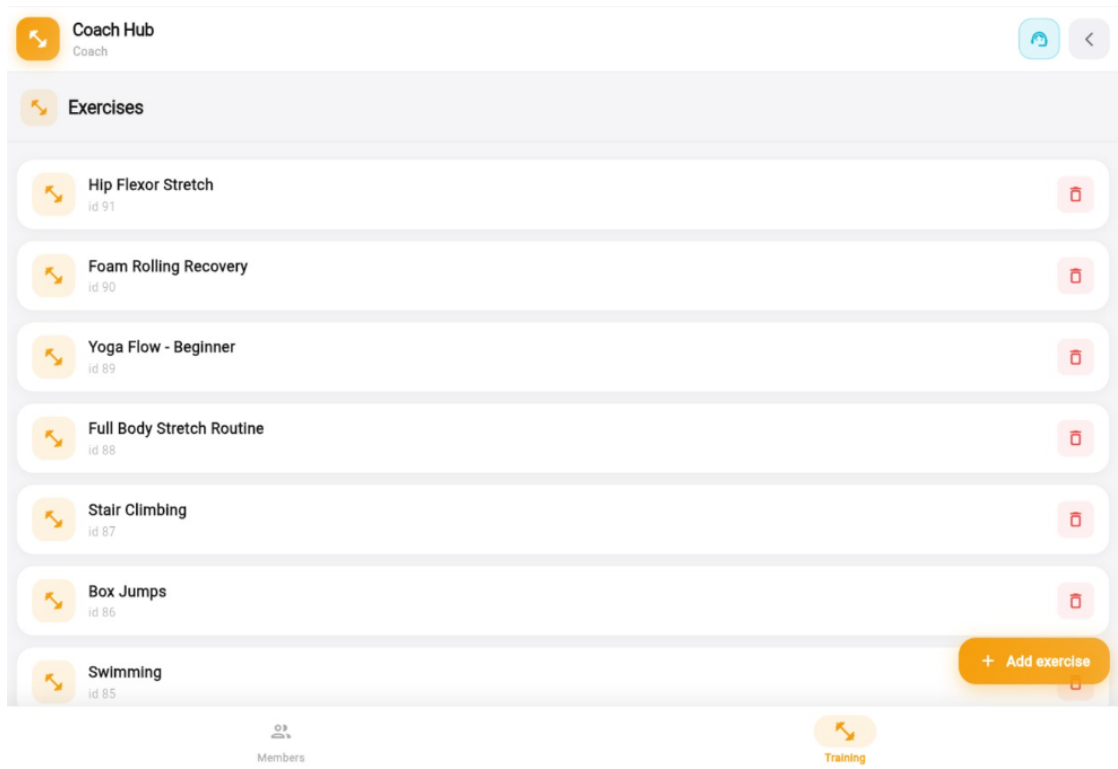


Figure 95:Exercises screen

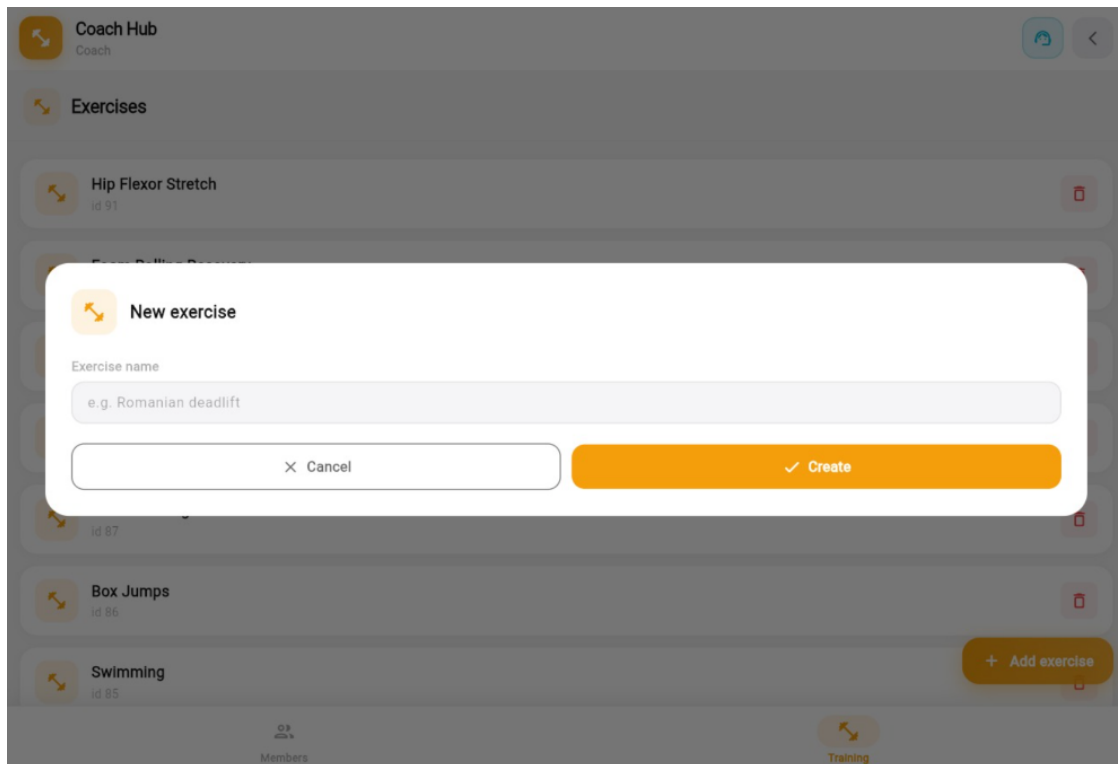


Figure 96:Add exercises screen

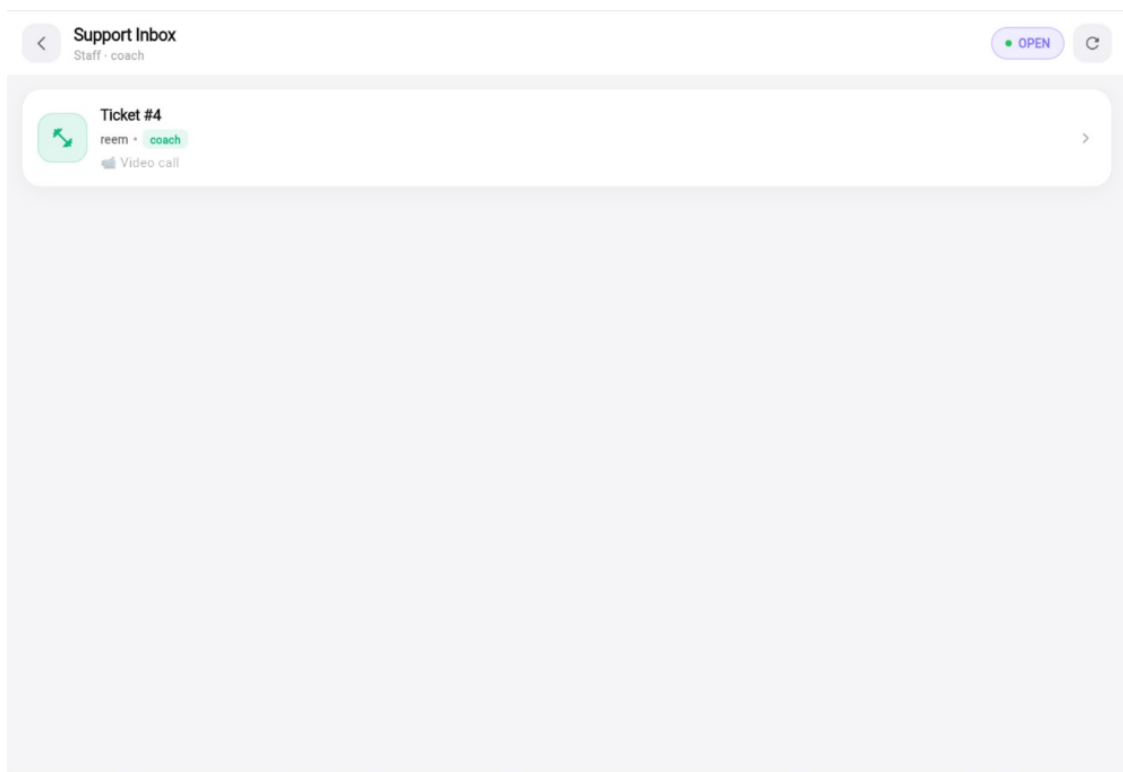


Figure 97:Support inbox screen

3.11 Smartwatch Integration

As an additional enhancement to the Finesse platform, smartwatch integration was implemented to provide automatic health and activity tracking. Through health synchronization services, the application can retrieve step counts, workout records, calories burned, and other activity-related metrics directly from wearable devices. This integration reduces manual data entry, improves tracking accuracy, and enables the AI recommendation system to generate more personalized fitness insights. The functionality was successfully tested on Android Emulator environments and demonstrated reliable synchronization of health-related data between the wearable device and the application.

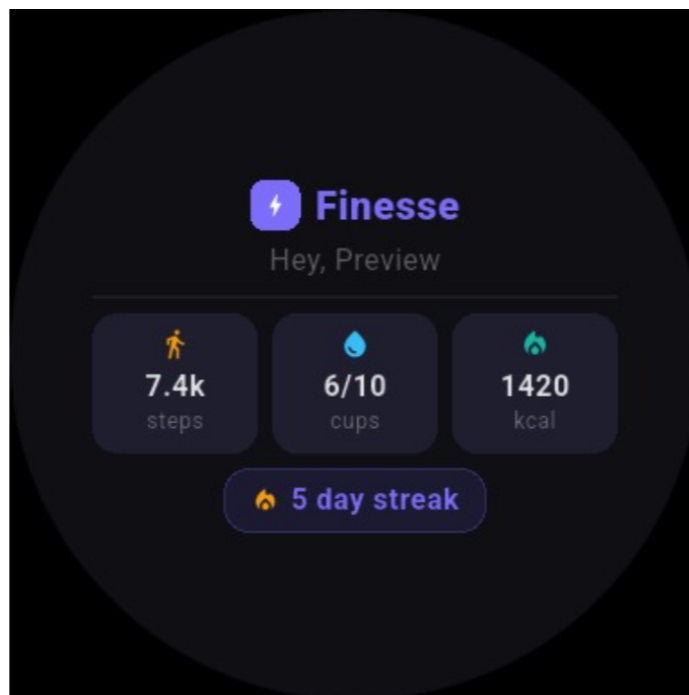


Figure 98:Smart watch screen

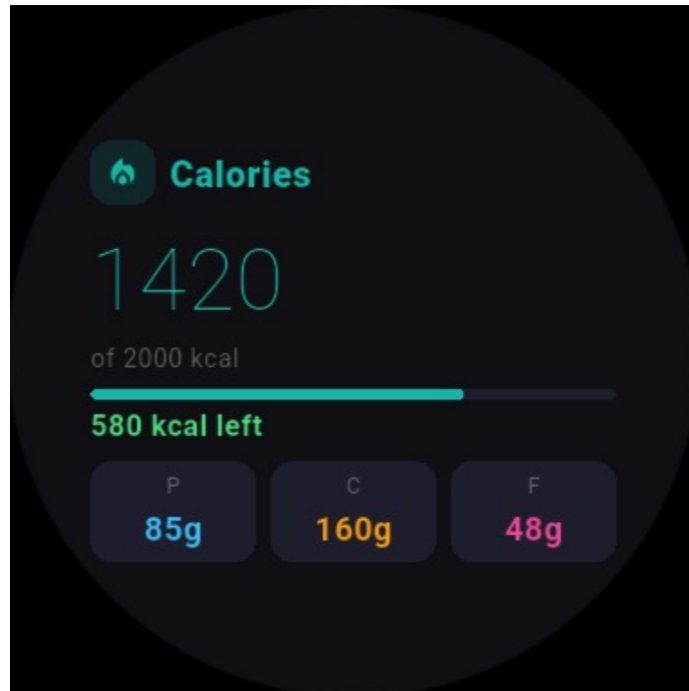


Figure 99:Smart watch(Calories) screen

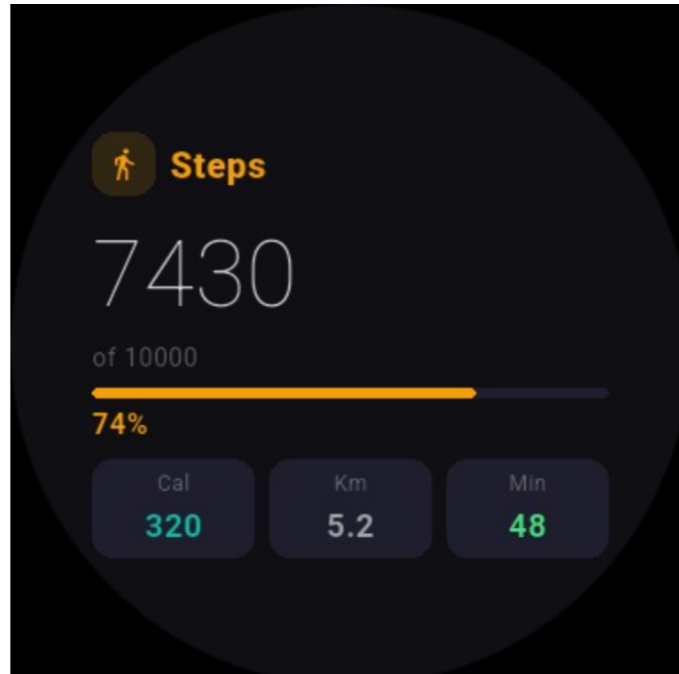


Figure 100:Smart watch(Steps) screen



Figure 101:Smart watch (Water)screen

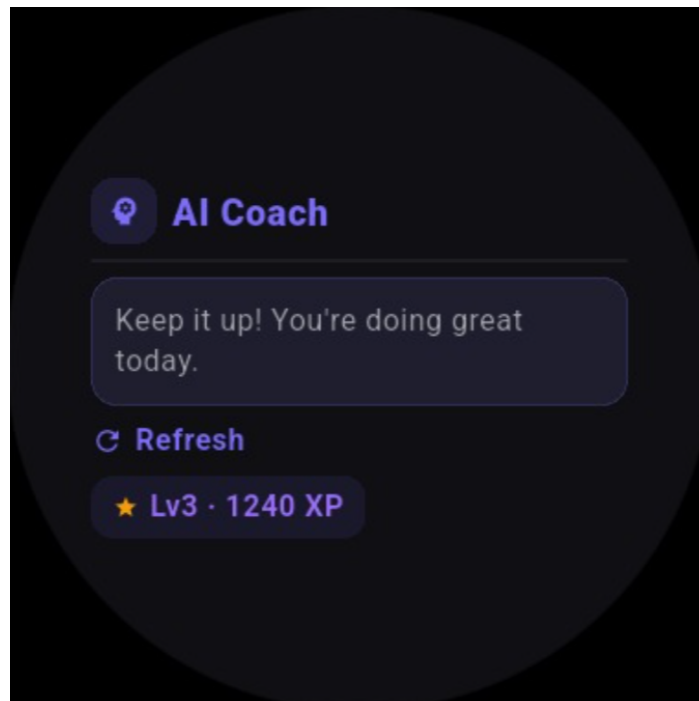


Figure 102:Smart watch(Ai Coach) screen



Figure 103:Smart watch(generate byAi) screen

Chapter 4

Results and Analysis

This chapter summarizes the results obtained from implementing and testing the Finesse fitness and wellness platform. The evaluation focused on the core functionalities of the system, including personalized workout generation, meal planning, AI coaching, smartwatch integration, progress tracking, hydration monitoring, challenges, leaderboard rankings, and communication features. The testing process was conducted through multiple scenarios simulating real user interactions across different modules of the application.

The results showed that the system successfully supported users in managing their fitness journey by allowing them to create personalized profiles containing information such as age, gender, weight, height, fitness goals, and activity preferences. Based on this information, the platform generated customized workout plans, nutritional recommendations, and daily fitness targets.

The AI Coach module provided personalized fitness guidance and workout recommendations according to user goals and activity levels. The generated recommendations were consistent and relevant across different test scenarios, demonstrating the effectiveness of the recommendation logic and personalization mechanisms implemented within the platform.

An important achievement of the project was the successful integration with smartwatch and health-tracking services. The system was able to synchronize fitness data such as daily step counts, workout activities, and calories burned. The synchronized data was automatically reflected within the dashboard and utilized by the recommendation engine to provide more accurate fitness insights without requiring manual user input.

The meal planning and calorie tracking modules successfully generated daily nutritional recommendations aligned with user objectives. Users were able to log meals, monitor calorie intake, and receive alerts when daily calorie limits were exceeded. The hydration tracking feature also enabled users to monitor daily water consumption and receive reminder notifications throughout the day.

The challenge management system and community features enhanced user engagement by allowing participation in fitness challenges, progress monitoring, and interaction with other users. In addition, the platform successfully implemented a leaderboard system that encouraged healthy competition among users. The leaderboard rankings were updated according to user achievements, challenge participation, accumulated points, and activity records, helping motivate users to remain active and achieve their fitness goals.

The support chat module enabled effective communication between users, coaches, nutritionists, and administrators. Messages were exchanged successfully, providing users with additional guidance and support throughout their fitness journey.

Potential measurement errors were mainly associated with user-provided information and manually entered fitness data. However, smartwatch

synchronization helped reduce data entry inaccuracies and improved overall tracking reliability. System response times remained within acceptable limits throughout testing and did not negatively affect user experience.

Overall, the results confirm that Finesse successfully achieved its objectives by providing an intelligent, user-friendly fitness platform that combines personalized recommendations, health tracking, wearable device integration, gamification features, and community engagement in a single environment.

Chapter 5

Discussion

This project successfully addressed several challenges associated with fitness management by providing an integrated platform that combines health tracking, personalized recommendations, and user engagement features. The results demonstrate that the use of artificial intelligence, wearable device integration, and adaptive fitness planning can significantly improve the user experience and simplify health management activities.

The main contribution of this project is the development of an intelligent fitness ecosystem that automatically adapts recommendations based on user characteristics, goals, and activity data. The integration of smartwatch

synchronization further enhanced the accuracy of collected fitness information while reducing the need for manual data entry.

The AI Coach feature also contributed to improving user engagement by providing personalized guidance and instant feedback. Furthermore, the combination of workout planning, meal tracking, hydration monitoring, challenge participation, smartwatch synchronization, and leaderboard rankings created a comprehensive wellness environment capable of supporting users throughout their fitness journey. The leaderboard feature increased motivation by encouraging healthy competition among users and rewarding continuous engagement.

A limitation of the system is its dependency on the availability and accuracy of health data provided by users or external wearable devices. In addition, some advanced health metrics such as sleep quality, heart rate variability, and stress analysis were not included in the current version.

Overall, the results indicate that intelligent fitness platforms supported by artificial intelligence and wearable technologies can significantly improve health monitoring, user motivation, and fitness goal achievement. Future improvements may focus on expanding health data integration, enhancing recommendation algorithms, and incorporating more advanced predictive analytics.

Chapter 6

Conclusion and Recommendation

6.1 Summary

This project presented Finesse, an intelligent fitness and wellness platform designed to help users manage their health and fitness activities through a personalized and integrated environment. The platform combines fitness tracking, workout planning, nutrition management, hydration monitoring, community engagement, leaderboard rankings, smartwatch integration, and AI-powered coaching within a single application.

The system successfully achieved its primary objectives by providing personalized workout recommendations, meal plans, progress tracking tools, and health monitoring features. In addition, the integration with smartwatch and health synchronization services enabled automatic collection of fitness-related data, improving tracking accuracy and reducing manual input requirements.

The platform also incorporated a leaderboard system that enhanced user engagement, increased motivation, and encouraged users to maintain healthy habits through gamification techniques and friendly competition.

The implementation of AI-assisted recommendations enhanced the overall user experience by providing customized fitness guidance based on individual goals, preferences, and activity levels. Furthermore, the communication and support modules facilitated interaction between users and fitness professionals, creating a more engaging and supportive fitness ecosystem.

Overall, the project demonstrated the effectiveness of combining artificial intelligence, wearable technology, gamification concepts, and modern mobile application development techniques to create a comprehensive fitness management solution.

6.2 Recommendations

To further improve the platform, several enhancements can be considered in future releases. It is recommended to improve the accuracy of recommendation algorithms by incorporating additional health and behavioral data. Expanding the existing smartwatch integration to support a wider range of wearable devices and advanced health sensors would also increase accessibility and data accuracy.

The platform could benefit from enhanced security mechanisms, advanced health analytics, and more detailed progress visualization tools. Additionally, integrating more health-related metrics such as heart rate monitoring, sleep analysis, and stress tracking would provide a more complete picture of user wellness.

Implementing these improvements would further increase system effectiveness while maintaining scalability and usability.

6.3 What We Have Learned

Through the development of Finesse, we gained valuable technical and practical experience in various areas of software engineering and application development. The project provided hands-on experience in mobile application development using Flutter, backend development using Node.js and Express.js, database management using MySQL, and secure authentication mechanisms using Firebase and JWT.

We also learned how to integrate third-party services such as payment gateways, health synchronization platforms, smartwatch technologies, and artificial intelligence APIs. In addition, the project strengthened our understanding of software architecture, API development, user interface design, testing methodologies, and system integration.

The implementation of gamification features such as challenges, achievements, and leaderboard rankings provided valuable experience in designing systems that improve user motivation and engagement. Beyond technical skills, the project enhanced our teamwork, communication, project planning, problem-solving, and time management abilities.

Working on a real-world application helped us better understand the challenges involved in designing and developing scalable software solutions that address actual user needs.

6.4 Future Work

Future development of Finesse may focus on transforming the platform into a more advanced intelligent health assistant capable of providing deeper personalization and predictive health insights.

Potential future enhancements include integrating advanced health monitoring features such as heart rate tracking, sleep quality analysis, stress detection, and blood oxygen monitoring. Machine learning models can also be incorporated to provide more accurate recommendations based on user behavior patterns and historical progress data.

Additional future work may include developing dedicated smartwatch applications, implementing voice-based AI coaching, expanding multilingual support, enhancing the existing leaderboard system through seasonal competitions, team-based challenges, achievement badges, and reward mechanisms, and integrating telehealth services that connect users directly with healthcare professionals.

Furthermore, advanced analytics dashboards could be added to provide detailed reports on user performance, long-term health trends, and fitness achievements. These enhancements would further improve user engagement, recommendation accuracy, and the overall effectiveness of the platform as a comprehensive digital health solution.

References

- [1] Google, “Flutter Documentation,” *Flutter*, 2026. [Online]. Available: <https://docs.flutter.dev/>
- [2] Google, “Firebase Documentation,” *Firebase*, 2026. [Online]. Available: <https://firebase.google.com/docs>
- [3] OpenJS Foundation, “Node.js Documentation,” *Node.js*, 2026. [Online]. Available: <https://nodejs.org/docs/latest/api/>
- [4] Oracle Corporation, “MySQL Documentation,” *MySQL*, 2026. [Online]. Available: <https://dev.mysql.com/doc/>
- [5] Postman, “Postman Documentation,” *Postman*, 2026. [Online]. Available: <https://learning.postman.com/docs/>
- [6] Groq, “Groq API Reference,” *Groq Documentation*, 2026. [Online]. Available: <https://console.groq.com/docs/api-reference>
- [7] MyFitnessPal, “Calorie Counter and Nutrition Tracking,” *MyFitnessPal*, 2026. [Online]. Available: [MyFitnessPal](#)
- [8] Jitsi, “Jitsi Meet Documentation,” *Jitsi*, 2026. [Online]. Available: [Jitsi Meet](#)

