YAT® LEARNING CENTERS

وزارة الاتصـــــالات
وتكنولوجيا المعلومات

Software Requirements Specification

For

Fitness Tracker App(FTA)

Fitnessy

Version 5.1

Prepared by:

- **Emad Rabie Hasabelnaby**
- **Yassmina Nader Omar**
- **Jomana Mahmoud Elsaghier**
- **Ahmed Hashim Sayed**

Digital Egypt Pioneers Initiative DEPI

YAT377A_CAI2_SWD4_S5_DEPI2

05-05-2025

Table of Contents

List of Figures 4

1. **Project Planning and Management** 5

   1.1. Project Proposal 5

   1.2. Project Plan 5

   1.3. Task Assignment & Roles 5

   1.4. Risk Assessment & Mitigation Plan 5

   1.5. KPIs (Key Performance Indicators) 5

2. **Literature Review** 5

   2.1. Feedback & Evaluation 5

   2.2. Suggested Improvements 5

   2.3. Final Grading Criteria 6

3. **Requirements Gathering** 7

   3.1. Stakeholder Analysis. 7

   3.2. User Stories & Use Cases 7

   3.3. Functional Requirements 7

   3.4. Non-functional Requirements 7

4. **System Analysis & Design** 7

   4.1. Problem Statement & Objectives 7

      4.1.1 Use Case Diagram & Descriptions 7

      4.1.2 Functional & Non-Functional Requirements 7

      4.1.3 Software Architecture 7

   4.2 Database Design & Data Modeling 7

      4.2.1 ER Diagram 7

      4.2.2 Logical & Physical Schema 7

   4.3 Data Flow & System Behavior 7

      4.3.1 DFD (Data Flow Diagram) 7

      4.3.2 Sequence Diagrams 7

      4.3.3 Activity Diagram 7

      4.3.4 State Diagram 7

      4.3.5 Class Diagram 7

4.4 UI/UX Design & Prototyping 8
4.4.1 Wireframes & Mockups 8
4.4.2 UI/UX Guidelines 8
4.5 System Deployment & Integration 8
4.5.1 Technology Stack 8
4.5.2 Deployment Diagram 8
4.5.3 Component Diagram 8
4.6 Additional Deliverables (if applicable) 8
4.6.1 API Documentation 8
4.6.2 Testing & Validation 8
4.6.3 Deployment Strategy 8

List of Figures

Figure 1: Timeline (Gantt Chart) 6
Figure 2 System Architecture 9
Figure 3: Page Layout 11
Figure 4: Main Page 12
Figure 5: Student Page 12
Figure 6: Search Result Page 13
Figure 7: Add and Edit Form 13

1. Project Planning and Management

1.1 Project Proposal

Overview

The project involves developing a fitness mobile app that allows users to track their daily activities, set fitness goals, and monitor progress. The app will include features such as user authentication, goal setting, activity tracking, fitness exercises, and profile management. The app will be available on both iOS and Android platforms.

Objectives

1. **User Authentication**: Allow users to sign up/login via email/password or social media (Google, Facebook).
2. **User Profile Creation**: Enable users to input personal details (name, DOB, weight, height, gender) and set fitness goals (water intake, calorie intake, target weight).
3. **Activity Tracking Dashboard**: Provide a dashboard to track daily progress (calories, weight, water intake).
4. **Fitness Activities**: Offer guided fitness activities (e.g., muscle building, core exercises).
5. **Profile Management**: Allow users to edit their profile, change settings, and manage privacy/security.
6. **Notifications & Permissions**: Enable users to toggle notifications and location permissions.
7. **Help & Support**: Provide access to help, support, and legal documents (terms and policies).

Scope

- **Platforms**: iOS and Android.
- **Features**: Authentication, profile creation, goal setting, activity tracking, fitness activities, profile management, notifications, privacy settings, and help/support.
- **Deliverables**: Fully functional mobile app, backend server, database, and admin panel.
- **Timeline**: 6 months (adjustable based on team size and resources).

1.2 Project Plan

Timeline (Gantt Chart)

is a high-level timeline for the project

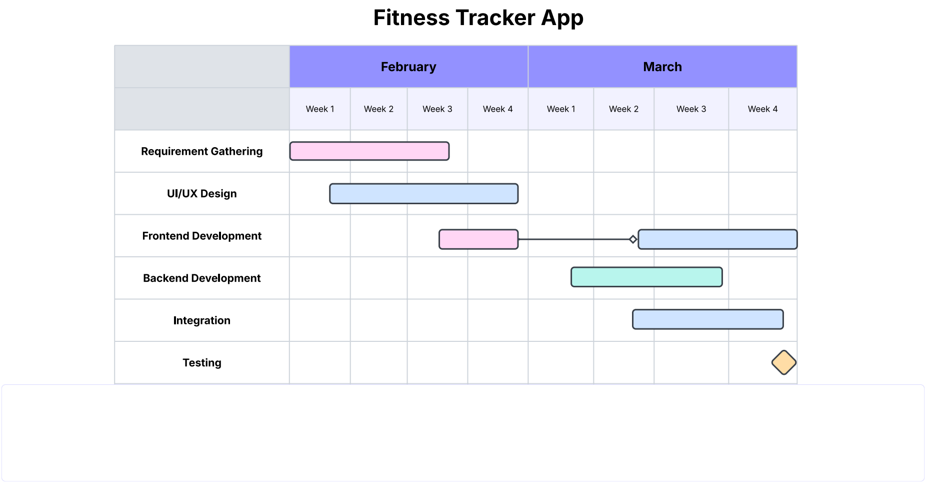


Figure 1: Timeline (Gantt Chart)

Milestones

1. **Requirement Finalization**: Complete by Week 3.
2. **UI/UX Design Approval**: Complete by Week 4.
3. **Frontend Development Completion**: Complete by Week 8.
4. **Backend Development Completion**: Complete by Week 7.
5. **Integration Completion**: Complete by Week 8.
6. **Testing Completion**: Complete by Week 8.
7. **App Launch**: Complete by Week 9.

Resource Allocation

- **UI/UX Designer**: 5 person.
- **Frontend Developers**: 3 people.
- **Backend Developer**: 2 person.
- **QA Tester**: 1 person.
- **Project Manager**: 1 person.

1.3 Task Assignment & Roles

Team Roles & Responsibilities

1. **Project Manager**:
   - Oversee the project timeline and deliverables.
   - Coordinate between team members and stakeholders.
   - Manage risks and ensure KPIs are met.
2. **UI/UX Designer**:
   - Design wireframes and prototypes.
   - Create the app's visual design and user experience.
3. **Frontend Developers**:
   - Develop the app's frontend for iOS and Android.
   - Implement navigation, screens, and user interactions.
4. **Backend Developer**:
   - Develop APIs, database, and authentication system.
   - Implement activity tracking and goal-setting logic.

5. **Tester**:
   - Perform testing (unit, integration, and UAT).
   - Report bugs and ensure the app meets quality standards.

1.4 Risk Assessment & Mitigation Plan

<u>Identified Risks</u>

1. **Delays in Development**:
   - **Cause**: Unclear requirements or technical challenges.
   - **Mitigation**: Regular progress reviews and agile sprints.
2. **Poor User Adoption**:
   - **Cause**: Lack of user-friendly design or features.
   - **Mitigation**: Conduct user testing during the design phase and gather feedback.
3. **Security Vulnerabilities**:
   - **Cause**: Weak authentication or data breaches.
   - **Mitigation**: Implement secure authentication (OAuth, encryption) and regular security audits.
4. **Integration Issues**:
   - **Cause**: Incompatibility between frontend and backend.
   - **Mitigation**: Use standardized APIs and conduct thorough integration testing.
5. **Budget Overruns**:
   - **Cause**: Unplanned expenses or scope creep.
   - **Mitigation**: Define a clear budget and scope, and monitor expenses regularly.

1.5 KPIs (Key Performance Indicators)

Metrics for Project Success

1. **Development KPIs**:
   - **Response Time**: API response time should be under 500ms.
   - **System Uptime**: 99.9% uptime for backend servers.
   - **Bug Rate**: Less than 5 critical bugs at launch.
2. **User Adoption KPIs**:
   - **Sign-Up Rate**: At least 20% of downloads result in sign-ups.
   - **Daily Active Users (DAU)**: 1,000 DAU within the first month of launch.
   - **Retention Rate**: 30% of users return after 7 days.
3. **Engagement KPIs**:
   - **Goal Completion Rate**: 50% of users achieve at least one fitness goal within a month.
   - **Activity Tracking Usage**: 70% of users use the activity tracking feature daily.
4. **Financial KPIs**:
   - **Cost per Download**: Less than $1 per download.
   - **ROI**: Achieve break-even within 6 months of launch.

2. Literature Review

- Feedback & Evaluation

  1. **Top Teams Mashallah,**

- Suggested Improvements

  2. **Model that Suggest near GYM**

- Final Grading Criteria

  3. **Excellent**

3. Requirements Gathering

3.1 Stakeholder Analysis

Stakeholder Analysis
Stakeholders are individuals or groups who have an interest in the project or will be affected by its outcomes. Here are the key stakeholders for fitness app:

Stakeholder	Needs/Expectations
End Users	**- Easy-to-use app with intuitive navigation.** **- Accurate activity tracking and goal monitoring.** **- Personalized fitness recommendations.** **- Secure handling of personal and health data.**
Developers	**- Clear requirements and scope to avoid scope creep.** **- Access to necessary tools and resources.** **- Reasonable deadlines and milestones.**
Project Manager	**- Timely delivery of the project within budget.** **- Effective communication among team members.** **- Risk management and mitigation.**
Investors/Sponsors	**- High user adoption and retention rates.** **- Return on investment (ROI) through app monetization (e.g., ads, subscriptions).**
App Stores	**- Compliance with guidelines (e.g., Google Play Store, Apple App Store).**
Regulatory Bodies	**- Compliance with data protection laws (e.g., GDPR, CCPA).**

3.2 User Stories & Use Cases

User stories describe how different types of users will interact with the app. Use cases provide detailed scenarios for these interactions.

User Stories

1. **As a new user**, I want to sign up using my email or social media account so that I can start using the app quickly.
2. **As a user**, I want to set my fitness goals (water intake, calorie intake, target weight) so that I can track my progress.

3. **As a user**, I want to view my daily activity scores (calories, weight, water) on a dashboard so that I can monitor my progress.
4. **As a user**, I want to perform guided fitness activities (e.g., muscle building, core exercises) so that I can stay fit.
5. **As a user**, I want to edit my profile (name, email, DOB, region) so that my information is up to date.
6. **As a user**, I want to toggle notifications and location permissions so that I can control my privacy.
7. **As a user**, I want to log out of the app so that my account remains secure.

Use Cases

1. **Sign-Up/Login**:
   - User opens the app and selects "Sign Up."
   - User enters email/password or chooses "Continue with Google/Facebook."
   - System verifies credentials and creates an account.
2. **Set Fitness Goals**:
   - User navigates to the "Goals" screen.
   - User inputs target values for water intake, calorie intake, and weight.
   - System saves the goals and displays them on the dashboard.
3. **Track Daily Activity**:
   - User opens the app and views the dashboard.
   - System displays daily scores for calories, weight, and water intake.
   - User can update scores manually or sync with a wearable device.
4. **Perform Fitness Activities**:
   - User navigates to the "Fitness Activities" screen.
   - User selects a workout (e.g., muscle building, core exercises).
   - System provides step-by-step instructions and tracks progress.
5. **Edit Profile**:
   - User navigates to the "Profile" screen and selects "Edit Profile."
   - User updates name, email, DOB, or region.
   - System saves the changes and updates the profile.

3.3 Functional Requirements

These are the specific features and functionalities the app must have.

1. **Authentication**:
   - Sign up using email/password or social media (Google, Facebook).
   - Log in and log out functionality.
2. **User Profile**:
   - Create and edit profile (name, email, DOB, weight, height, gender).
   - Upload profile picture.
3. **Goal Setting**:
   - Set daily goals for water intake, calorie intake, and target weight.
   - Update goals as needed.
4. **Activity Tracking**:

- Track daily calories, weight, and water intake.
- Display progress on a dashboard.

5. **Fitness Activities**:
   - Provide guided workouts (e.g., muscle building, core exercises).
   - Track workout progress and completion.
6. **Notifications**:
   - Send reminders for water intake, workouts, and goal updates.
   - Allow users to toggle notifications on/off.
7. **Privacy & Security**:
   - Change password.
   - Toggle location permissions.
   - Delete account.
8. **Help & Support**:
   - Access to help documents and support.
   - View terms and policies.

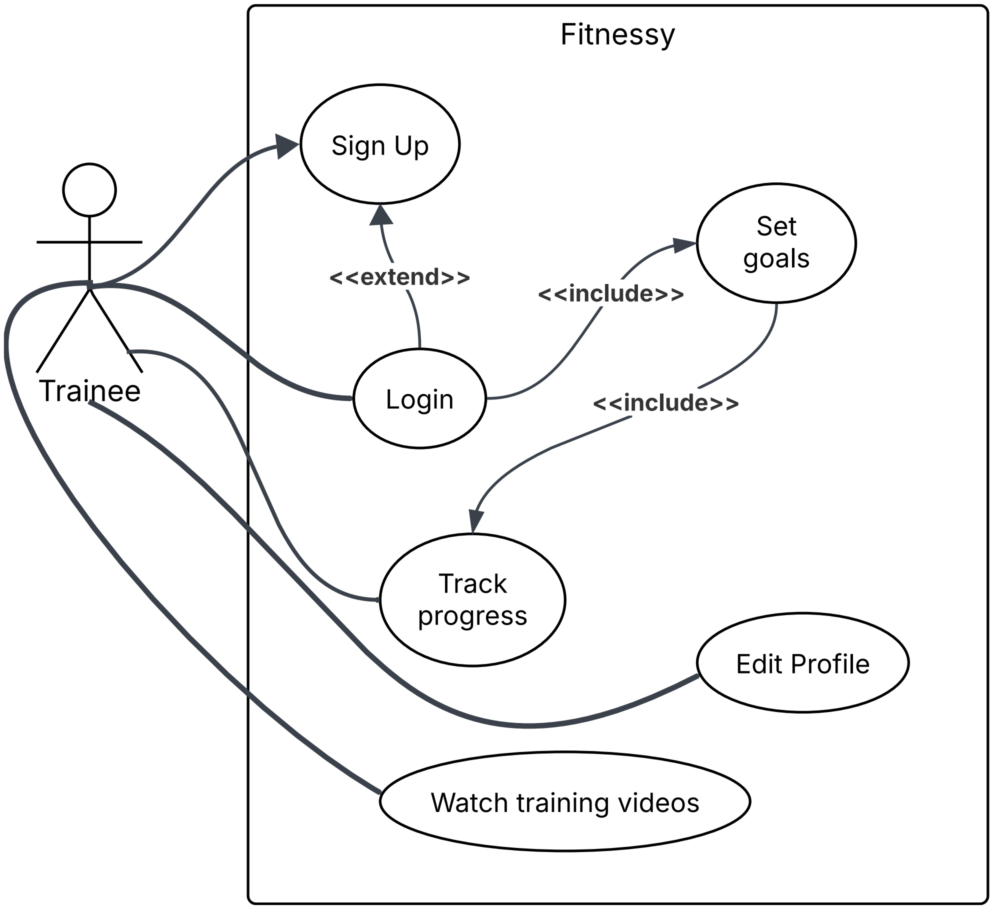
3.4 Non-functional Requirements

These define the quality attributes of the app, such as performance, security, and usability.

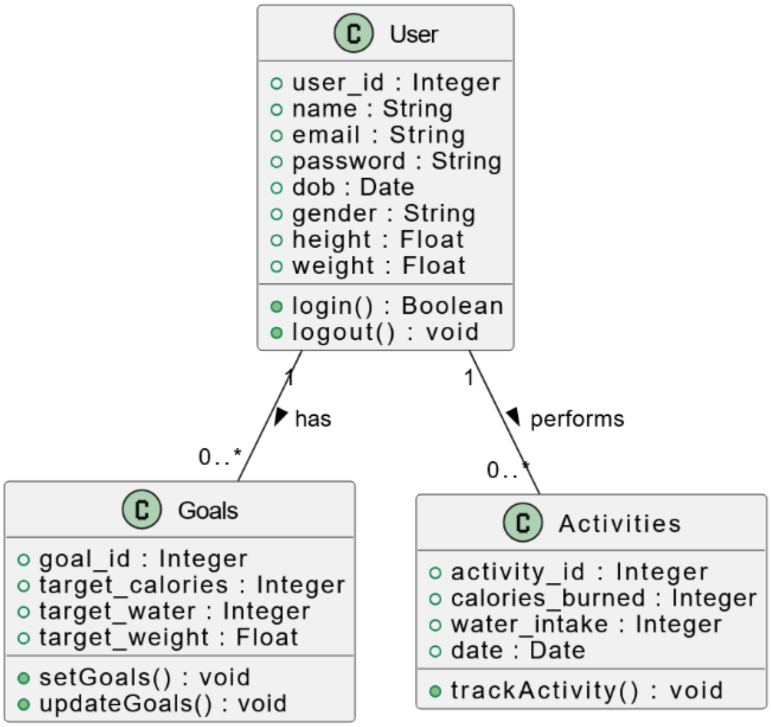
1. **Performance**:
   - The app should load in under 3 seconds.
   - API response time should be under 500ms.
2. **Security**:
   - User data (e.g., passwords, health information) must be encrypted.
   - Implement secure authentication (OAuth for social login, password hashing).
3. **Usability**:
   - The app should have an intuitive and user-friendly interface.
   - Support for multiple languages (if applicable).
4. **Reliability**:
   - The app should have 99.9% uptime.
   - Handle up to 10,000 concurrent users without performance degradation.
5. **Compatibility**:
   - The app should work on iOS (version 14 and above) and Android (version 10 and above).
   - Support for different screen sizes and resolutions.
6. **Scalability**:
   - The backend should scale to support increasing numbers of users.
   - Use cloud-based infrastructure for scalability (e.g., AWS, Firebase).
7. **Maintainability**:
   - Code should be well-documented and modular for easy updates.
   - Use version control (e.g., Git) for collaborative development.

4. System Analysis & Design

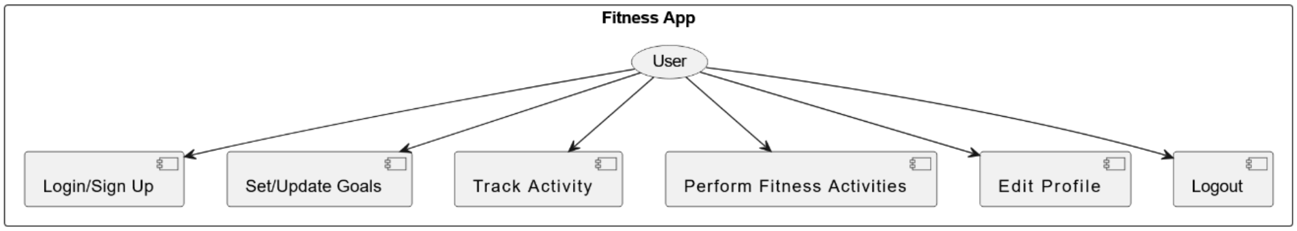
4.1 Use Case Diagram



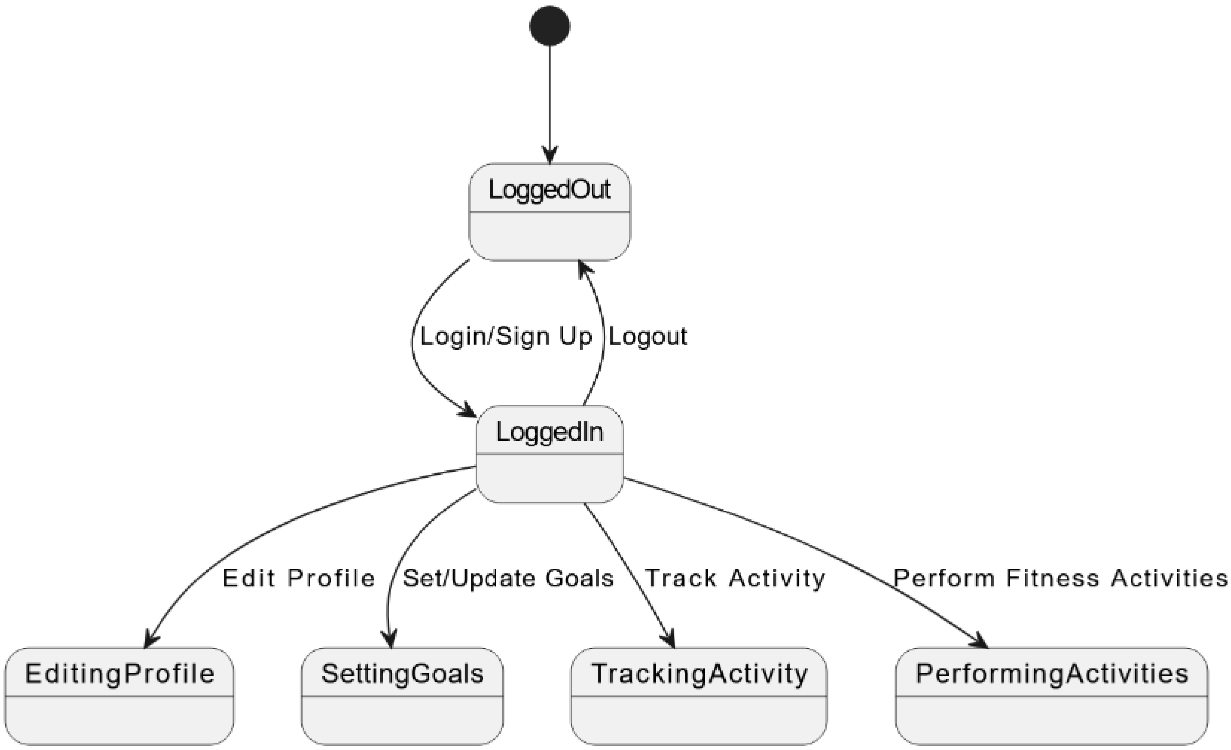
4.2 Class Diagram



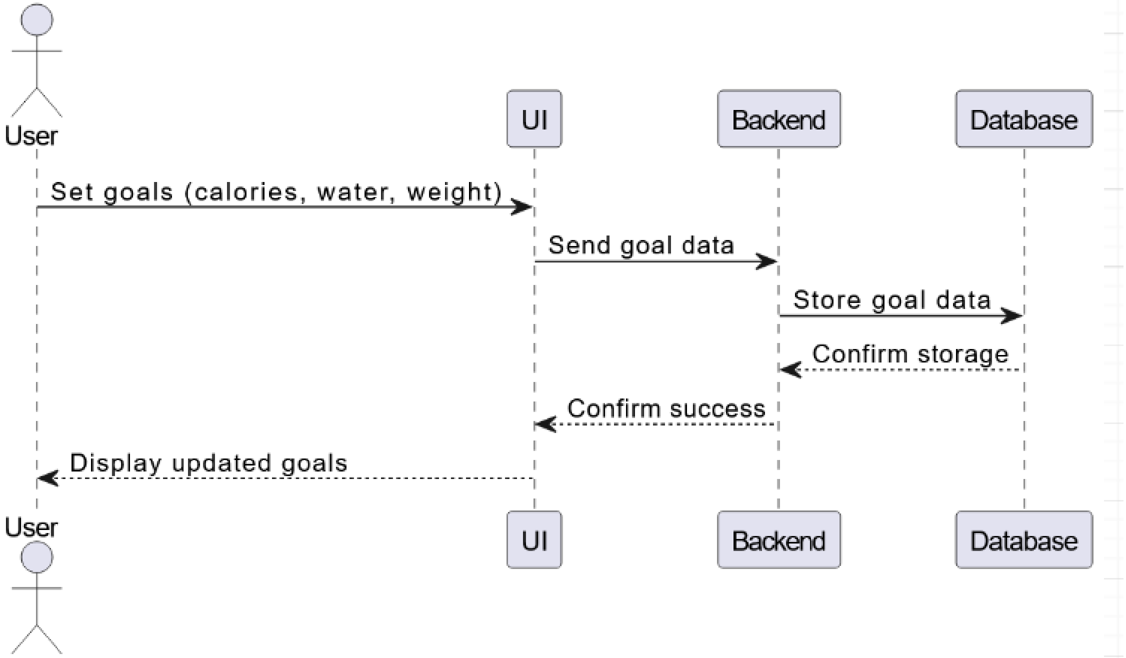
4.3 Package Diagram



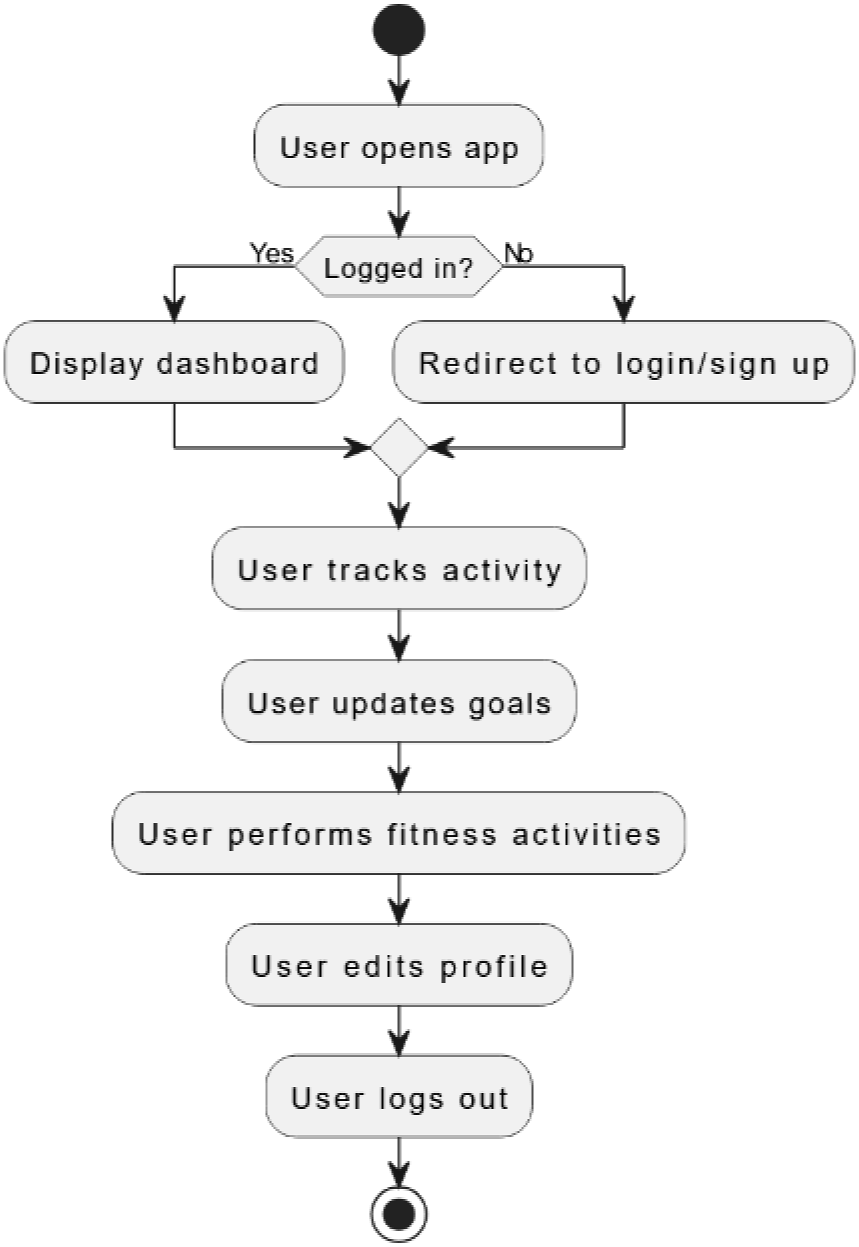
4.4 State Diagram



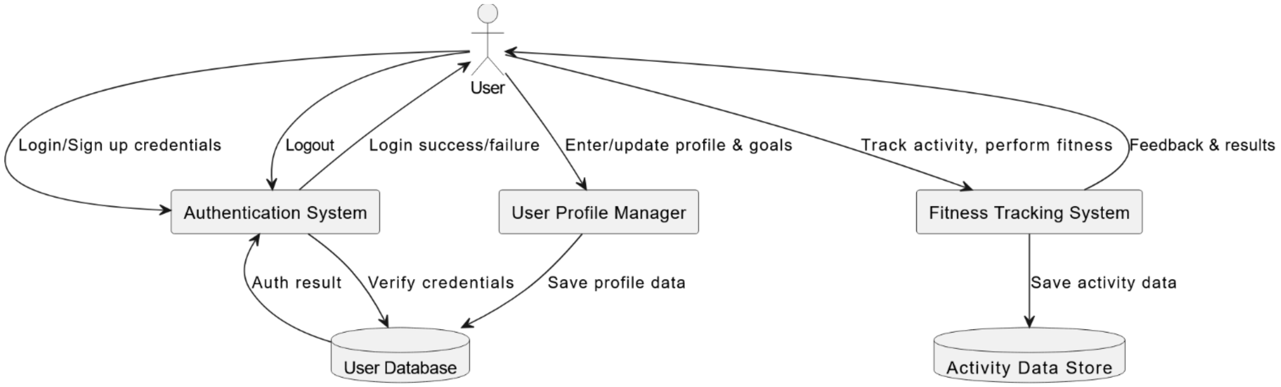
4.5 Sequence Diagram



4.6 Activity Diagram



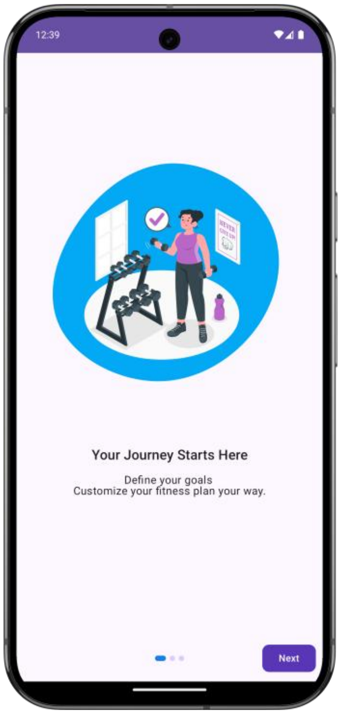
4.7 Data Flow Diagram (DFD)



Source Code

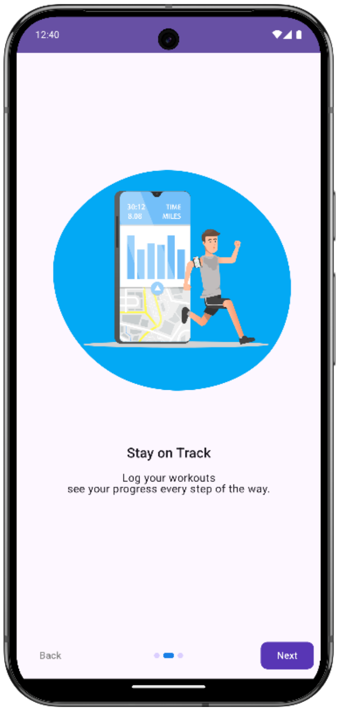
App Shots

Onboarding | Login | SignUp



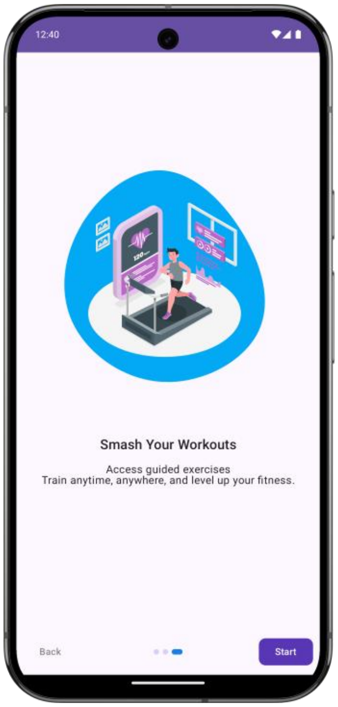
Your Journey Starts Here

Define your goals
Customize your fitness plan your way.



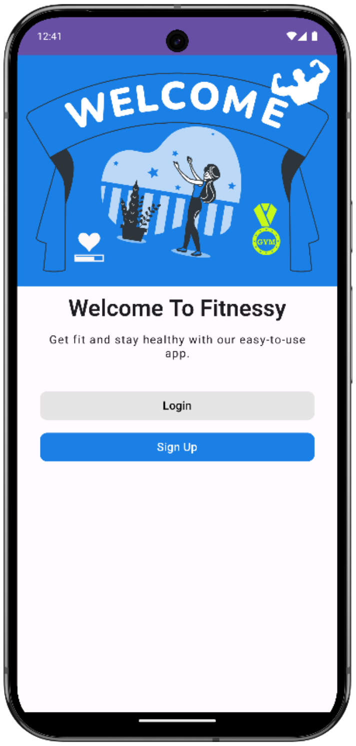
Stay on Track

Log your workouts
see your progress every step of the way.



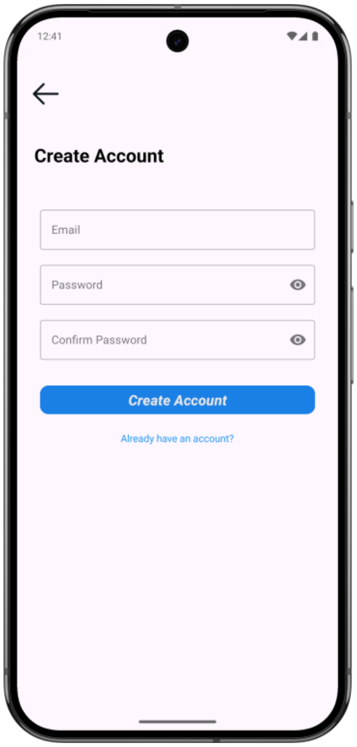
Smash Your Workouts

Access guided exercises
Train anytime, anywhere, and level up your fitness.



Welcome To Fitnessy

Get fit and stay healthy with our easy-to-use app.

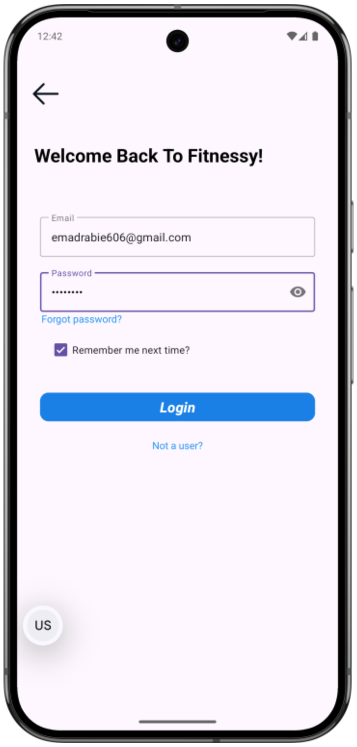


Create Account

Email

Password

Confirm Password



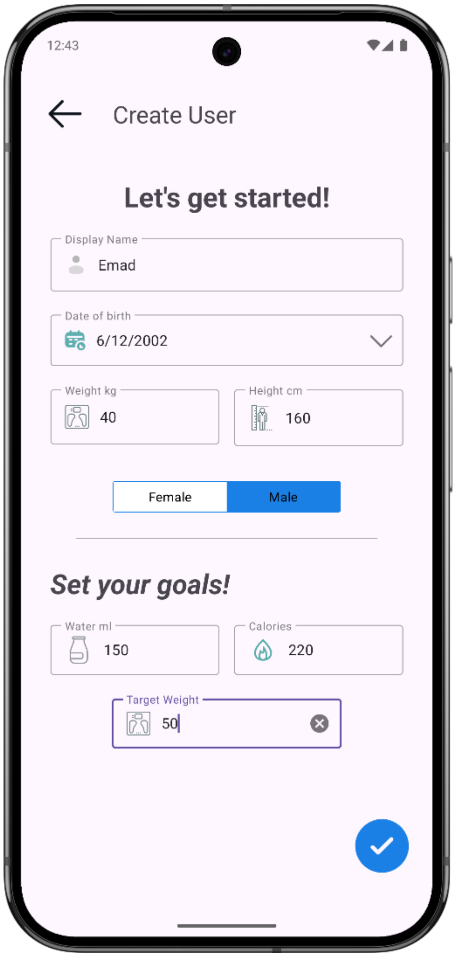
Welcome Back To Fitnessy!

Email: emadrabie606@gmail.com

Password: ••••••••

☑ Remember me next time?

Create User | Dashboard



Create User

Let's get started!

Display Name: Emad

Date of birth: 6/12/2002

Weight kg: 40

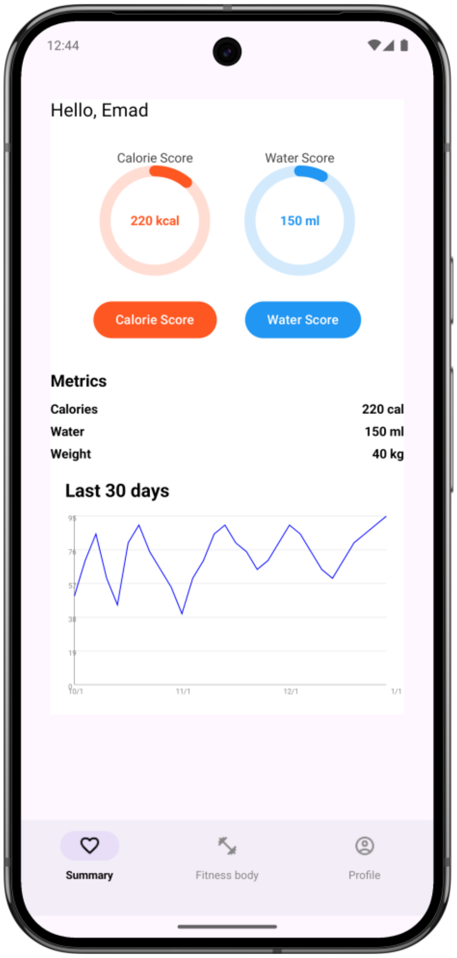
Height cm: 160

Set your goals!

Water ml: 150

Calories: 220

Target Weight: 50



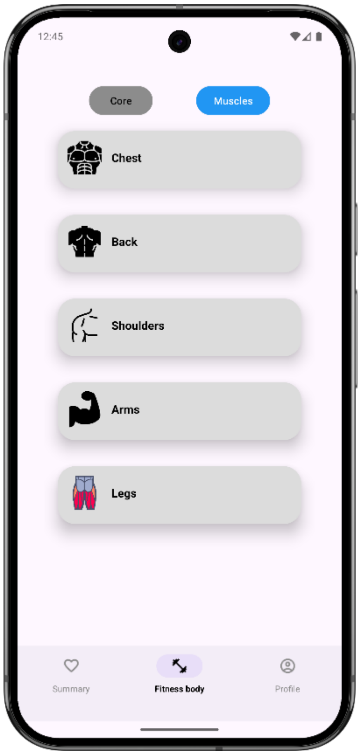
Hello, Emad

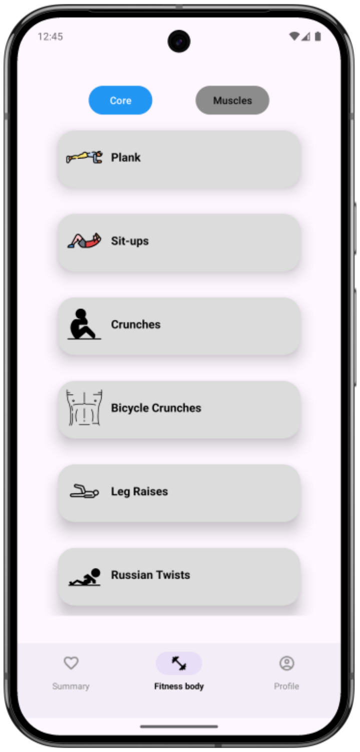
Metrics

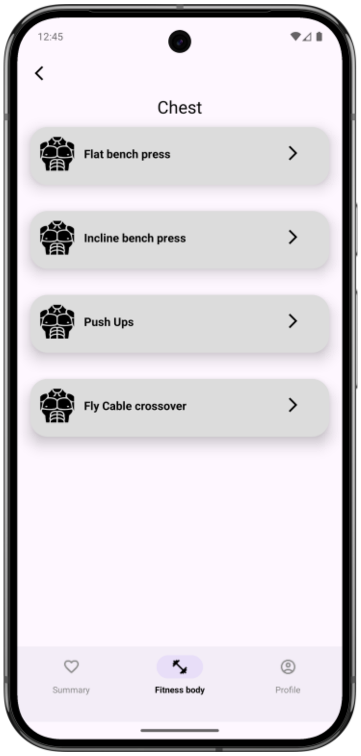
Calories	220 cal
Water	150 ml
Weight	40 kg

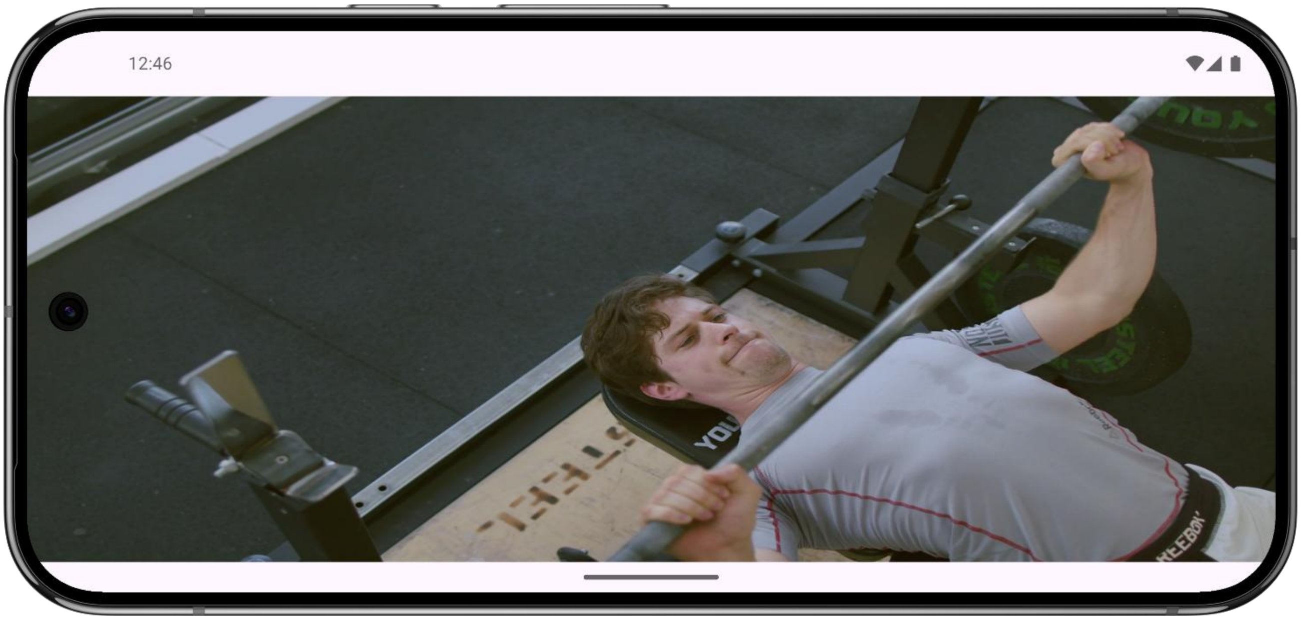
Last 30 days

Muscles, Coer, and video trainings

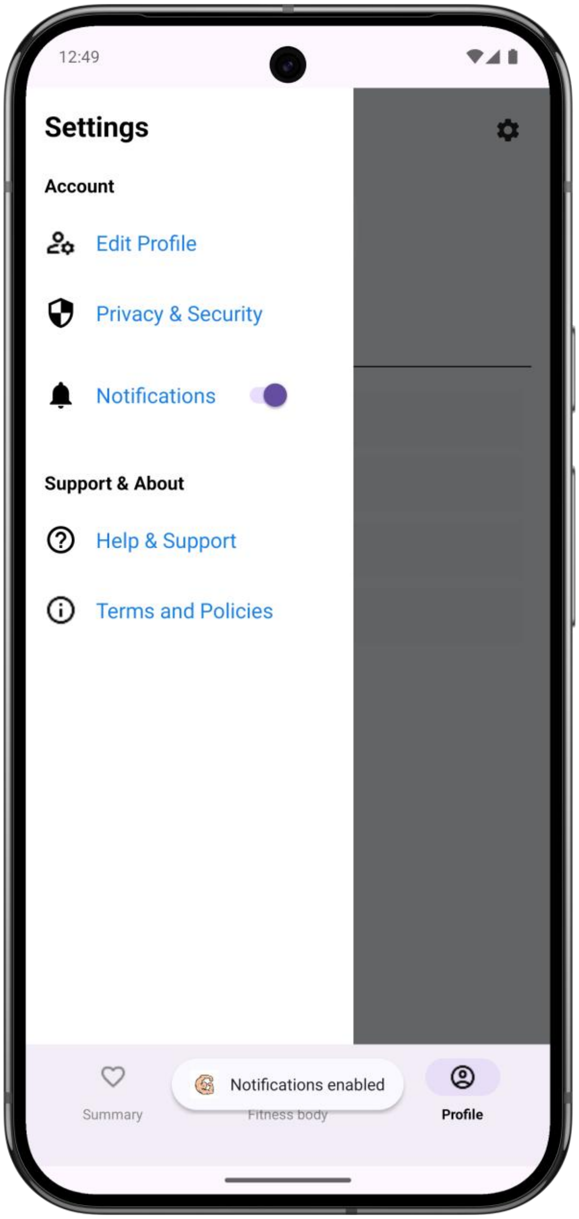




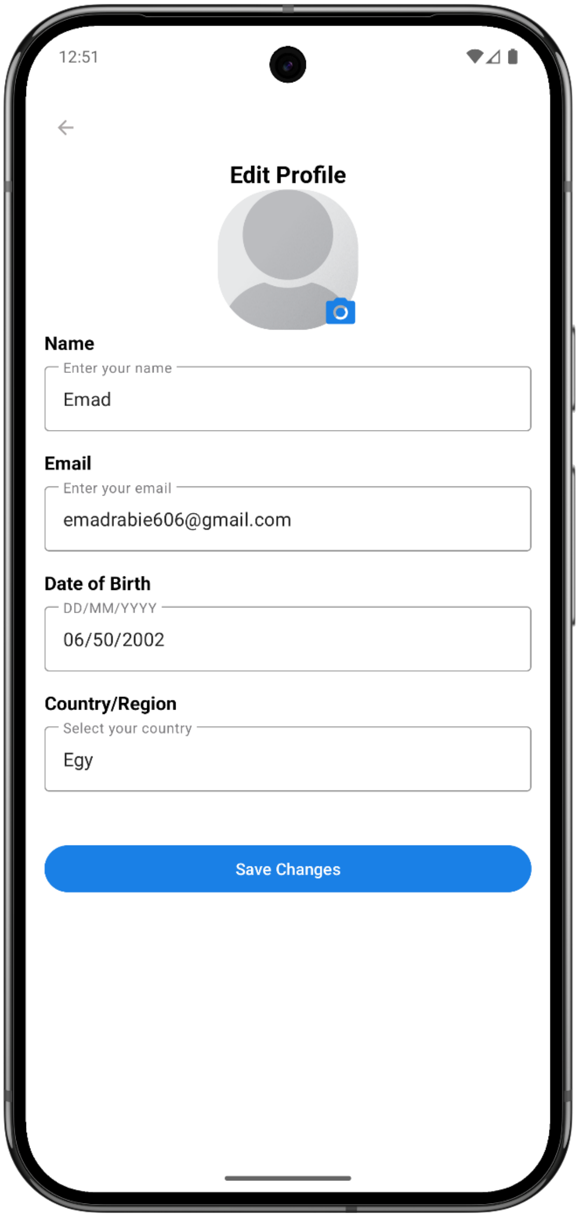




User Profile and Settings



Notifications enabled



Name
Enter your name
Emad

Email
Enter your email
emadrabie606@gmail.com

Date of Birth
DD/MM/YYYY
06/50/2002

Country/Region
Select your country
Egy