Fitness App Documentation

Introduction

In today's health-conscious environment, easy access to effective workout information is essential for individuals aiming to enhance their fitness routines. The Fitness Workout App is designed to provide users with a curated list of workouts for each muscle group, ensuring they have the resources they need to achieve their fitness goals.

This cross-platform mobile application features a straightforward interface that allows users to browse workouts categorized by muscle groups, making it simple to find exercises tailored to their needs. Developed using Flutter, the app guarantees a smooth and engaging user experience.

Key features include user authentication through Firebase, enabling secure login and registration processes. Users can create and edit their profiles, allowing for a personalized experience as they track their fitness journey. The app focuses on providing clear instructions and visuals for each exercise, making it accessible for users of all backgrounds.

Designed for simplicity, reliability, and quick access to information, the Fitness Workout App serves as a valuable tool for anyone looking to improve their workout routine. By delivering an intuitive and responsive platform, the app is a must-have companion for fitness enthusiasts.

This document outlines the requirements, specifications, and system design of the Fitness Workout App, providing a comprehensive roadmap for its development and implementation.

Functional Requirements

1. User Authentication

- Users must be able to register for a new account using email and password.
- Users must be able to log in using their registered email and password.
- Users must be able to log out from their account.

2. Profile Management

- Users must be able to edit their profile information (e.g., name, email).
- Users must be able to view their profile details.

3. Workout Catalog

- The app must display a list of workouts categorized by muscle groups (e.g., arms, legs, back, chest).
- Each workout entry must include exercise name, description, and visual demonstration (images or videos).

4. User-Friendly Navigation

- The app must provide an intuitive navigation system to access workouts, profile, and settings.

6. Responsive Design

- The app must be optimized for both mobile phones and tablets, ensuring a consistent user experience across devices.

7. Error Handling

- The app must provide clear error messages for failed login attempts, registration issues, and data retrieval problems.

8. Performance

- The app must load workout information within a few seconds to ensure a seamless user experience.

9. User Feedback

- Users must be able to provide feedback on workouts, which can be stored for future reference.

Non-Functional Requirements

1. **Performance**
    - The app must respond to user actions within 2 seconds to ensure a smooth experience.
    - The app must efficiently handle multiple concurrent users without performance degradation.
2. **Security**
    - User data, including passwords and personal information, must be securely encrypted during transmission and storage.
    - The app must comply with relevant data protection regulations (e.g., GDPR).
3. **Usability**
    - The app must have a clean and intuitive interface, ensuring ease of navigation for users of all skill levels.
    - User onboarding should be simple, with clear instructions for registration and profile setup.
4. **Compatibility**
    - The app must be compatible with the latest versions of both iOS and Android operating systems.
    - The app should function effectively on a range of device sizes and resolutions.
5. **Reliability**
    - The app must maintain 99% uptime, ensuring availability for users.
    - The app should gracefully handle errors and provide meaningful feedback to users.

System Design Architecture and Components

1. Client-Side (Mobile App)

- **User Interface (UI)**
  - Built using Flutter to provide a responsive and attractive design.
  - Includes screens for user authentication (login, registration), profile management, and workout browsing.
- **State Management**
  - Utilizes state management solutions (e.g., Provider) to manage app state efficiently.

2. Backend (Firebase)

- **Authentication**
  - Firebase Authentication to manage user sign-up, login, and session management securely.
- **Hosting**
  - Firebase Hosting for any static assets, if required.

Components Overview

- **User Authentication Module**
  - Handles user registration, login, and profile management.
- **Workout Module**
  - Displays a list of workouts, categorized by muscle groups.
- **Profile Management Module**
  - Enables users to view and edit their profile information.
- **Feedback System**
  - Collects user feedback on workouts for future enhancements.

Data Flow Diagram (DFD)

1. External Entities

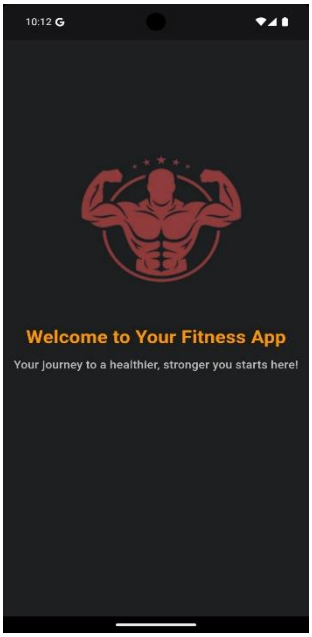
- **User**: The individual interacting with the app for various functionalities.

2. Processes

- **User Authentication**
  - Handles user registration and login.
- **Profile Management**
  - Allows users to view and edit their profile information.
- **Workout Browsing**
  - Fetches and displays workout data based on user requests.
- **Feedback Submission**
  - Collects user feedback on workouts.

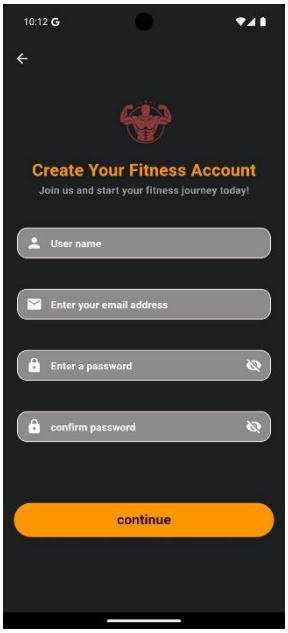
3. Data Flow

- **User Registration/Login**
  - User inputs credentials → User Authentication process validates credentials and provides access.
- **Profile Management**
  - User requests to view/edit profile → Profile Management process retrieves or updates user information.
- **Workout Browsing**
  - User requests workout information → Workout Browsing process fetches workout data and displays it to the user.
- **Feedback Submission**
  - User submits feedback → Feedback Submission process stores the feedback for future analysis.

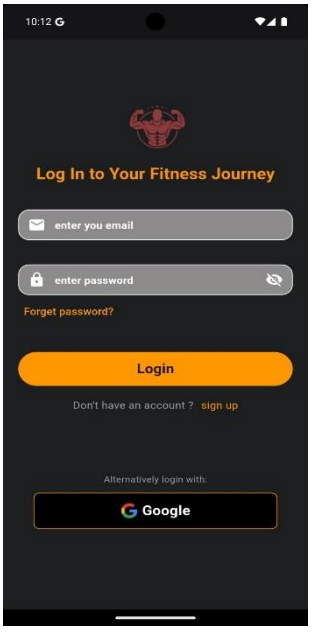


Splash screen

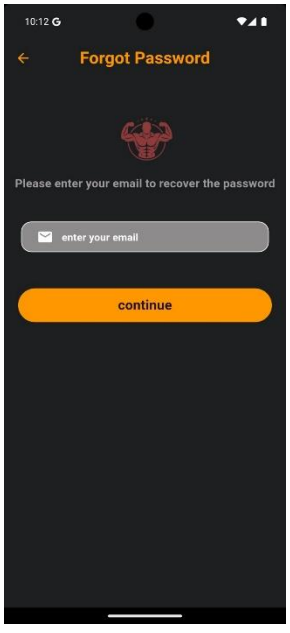
onboarding screen



Register screen

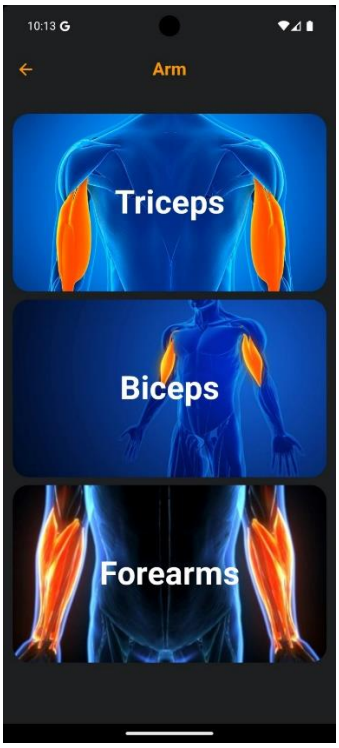


Login screen

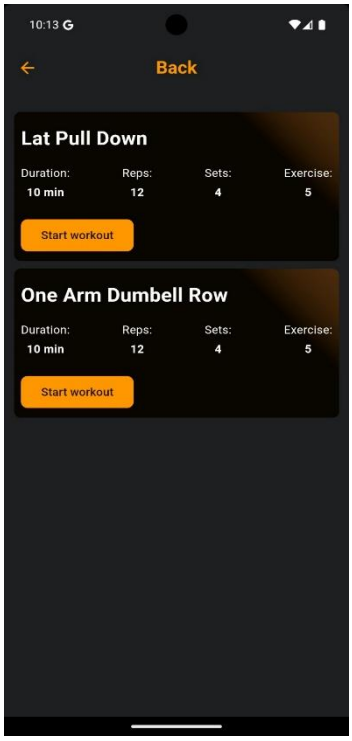


Forgot password screen

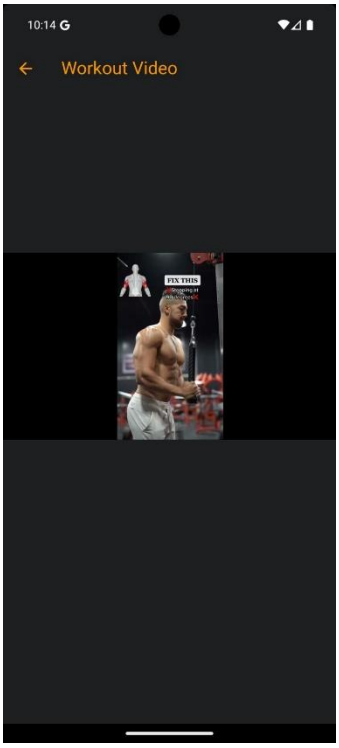
Home screen



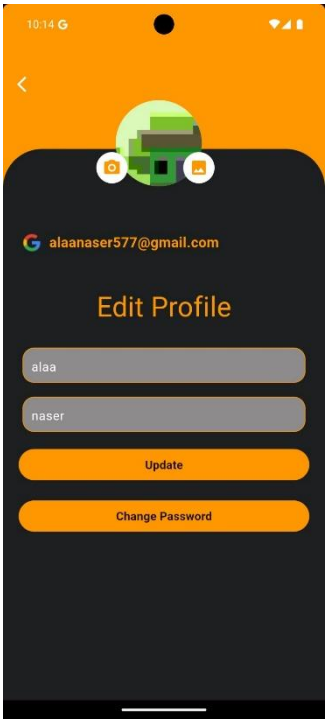
Sub muscle screen

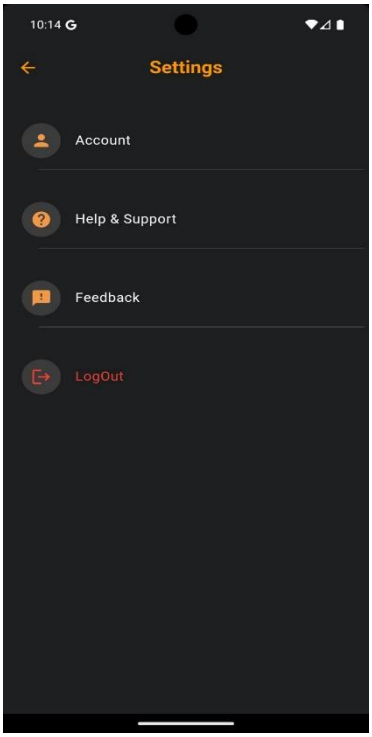


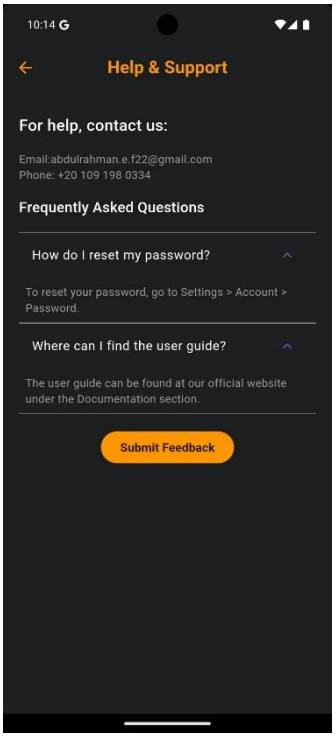
Workouts screen



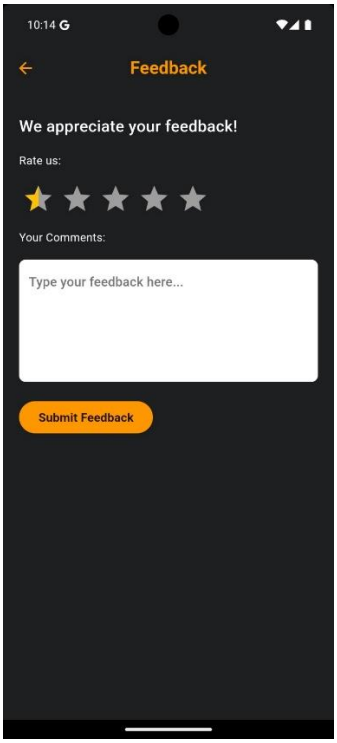
Workout video screen



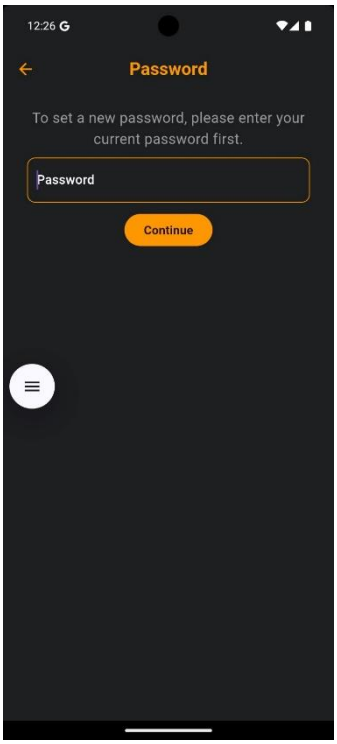




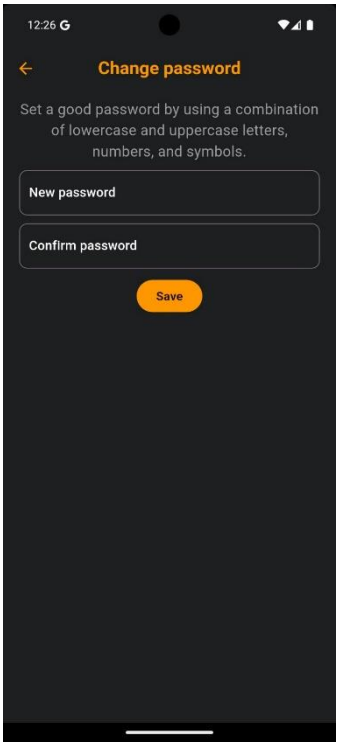
Profile screen



Setting screen



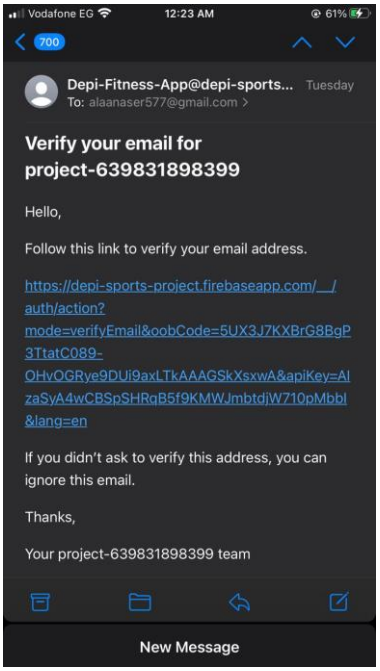
Help screen

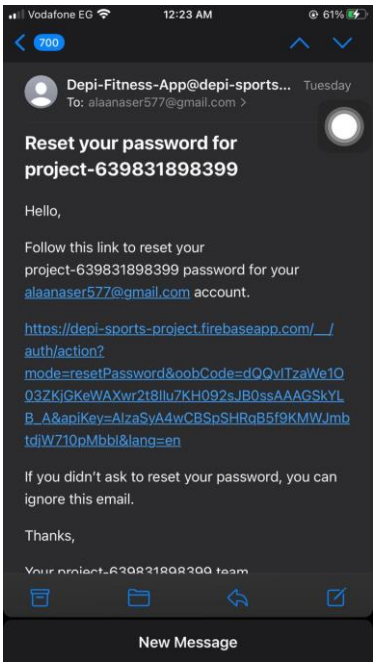


Feedback screen

Verify password screen

Change password screen





Activation account message

Reset password message