



Fitness Tracker App Technical Documentation

Version 1.0.0

November 15, 2023

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Scope	2
1.3	Technologies Used:	2
2	System Architecture	2
2.1	Frontend - React	2
2.2	Backend - Django	2
2.3	Database - MySQL	3
2.4	Real-Time Communication	3
3	Features	3
3.1	Registration and Authentication	3
3.2	Exercise Set Selection	3
3.3	Time Tracking of Exercise	3
3.4	Target/ Goal Setup	3
3.5	Data input and Goal tracking	3
3.6	Exercise History	3
3.7	Get Help Chatbox	4
3.8	Future Plan: Adding Friends and Mutual Tracking	4
4	Implementation Details	4
4.1	React	4
4.2	Django	4
4.3	RDMS	4
5	Testing	4
5.1	Unit Testing	4
5.2	Integration Testing	4
6	Deployment	4
6.1	Frontend Deployment	4
6.2	Backend Deployment	4
6.3	Database Deployment	4
7	Future Updates	5
7.1	Networking	5
7.2	Social Media Integration	5
7.3	Add Payment method	5

1 Introduction

1.1 Purpose

The Fitness Tracker App is designed to help users track their exercise routines, set fitness goals, and connect with friends for mutual tracking. This technical documentation provides an in-depth overview of the app's architecture, features, implementation details, testing procedures, deployment process, security considerations, and future enhancements.

1.2 Scope

The app focuses on providing a user-friendly interface for fitness enthusiasts to monitor their workouts, set achievable goals, and engage in a community for support and motivation.

1.3 Technologies Used:

- Frontend : React
- Backend : Django
- Database : RDMS (MySQL)
- Real-Time Communication : Network Socket

2 System Architecture

2.1 Frontend - React

The frontend is built using React to ensure a dynamic and responsive user interface. It interacts with the backend to fetch and display data, handle user inputs, and manage state.

2.2 Backend - Django

Django is employed as the backend framework for its robustness and scalability. It handles user authentication, and business logic, and communicates with the database.

2.3 Database - MySQL

MySQL is utilized as a relational database to store user data, exercise history, and other relevant information.

2.4 Real-Time Communication

Network sockets are implemented for real-time communication, enabling features like the Get Help Chat Box and future plans for friend tracking.

3 Features

3.1 Registration and Authentication

Users can register for an account and authenticate securely using Django's built-in authentication system.

3.2 Exercise Set Selection

The app provides a predefined set of 10 exercises from which users can choose for their workout routines.

3.3 Time Tracking of Exercise

A stopwatch feature is included to track the time spent on each exercise.

3.4 Target/ Goal Setup

Users can set fitness goals, specifying metrics like duration, intensity, or frequency. It also includes "target-weight in a certain period of time" as well.

3.5 Data input and Goal tracking

Users input relevant data to track progress towards their set goals.

3.6 Exercise History

The app stores one week of exercise history, allowing users to review and analyze their performance day by day.

3.7 Get Help Chatbox

3.8 Future Plan: Adding Friends and Mutual Tracking

4 Implementation Details

4.1 React

4.2 Django

4.3 RDMS

5 Testing

5.1 Unit Testing

Unit tests are conducted to validate the functionality of individual components.

5.2 Integration Testing

Integration tests ensure that different modules of the app work seamlessly together.

6 Deployment

6.1 Frontend Deployment

The React frontend is deployed using [deployment platform] and can be accessed at [frontend URL].

6.2 Backend Deployment

] Django backend is deployed on [deployment platform] and communicates with the frontend through [API URL].

6.3 Database Deployment

MySQL database is hosted on [database hosting service] with secure access controls.

7 Future Updates

7.1 Networking

Add user communication via chat.

7.2 Social Media Integration

Link preferred social media platform and exchange data. [API Integration]

7.3 Add Payment method

Add transactional API and payment method for future premium version upgrades.