FitConnect: A Comprehensive Fitness Tracking Application

Author: Yousaf Khawar Raja

Course: Object-Oriented Programming, Spring 2025

University: FAST-NUCES Lahore Campus

Roll Number: 24L-0552

Instructor: Usman Anwer

Date: May 13, 2025

Abstract

FitConnect is a comprehensive fitness tracking software designed to help users achieve their health and fitness goals. This application provides workout tracking, nutrition monitoring, social interaction, community engagement features, and user management. FitConnect also prioritizes user experience with robust user management, detailed workout tracking, social interaction features, and comprehensive progress analysis. It utilizes OOP principles such as Inheritance, Polymorphism, Composition, and Aggregation to structure and implement features efficiently, ensuring scalability and maintainability of the code.

Table of Contents

Introduction

Problem Statement

With the rising focus on health and fitness, individuals require a comprehensive solution to manage and monitor their fitness journeys. FitConnect addresses this need by providing

users with detailed workout tracking, nutrition logging, social interactions, and community support, all while ensuring secure and easy-to-navigate user management.

Objectives

- To design a scalable and maintainable fitness tracking system.
- To integrate OOP concepts to enhance the system's modularity and functionality.
- To provide users with tools for tracking workouts, nutrition, and progress while promoting community interaction.
- To ensure data security and easy user management.

Motivations

As the fitness tracking market grows, there is a need for more user-centric platforms that are not only easy to use but also offer advanced features to improve fitness outcomes. The motivation behind FitConnect is to create an application that blends the best of tracking tools and community support while using modern software engineering techniques like OOP.

OOP Concepts Used

This section will discuss how Inheritance, Polymorphism, Composition, and Aggregation were applied in the development of FitConnect.

- Inheritance: FitConnect uses inheritance to extend the `User` class into different roles like `RegularUser`, `Trainer`, and `Admin`, each with specialized functionality.

```
class Admin : public User {
private:
    int MAX_USERS = 100;  // Maximum number of users
    int MAX_TRAINERS = 50;  // Maximum number of trainers
    char adminLevel[30];
```

- Polymorphism: Polymorphism is employed in the social interaction module (User) interact with the system in a manner specific to their role.

```
virtual ~User() {}

virtual void login() {
    std::cout << name << " logged in successfully." << std::endl;
}

 virtual void logout() {
     std::cout << name << " logged out." << std::endl;
 }

virtual void updateProfile() = 0; // Pure virtual function makes User an abstract class
```

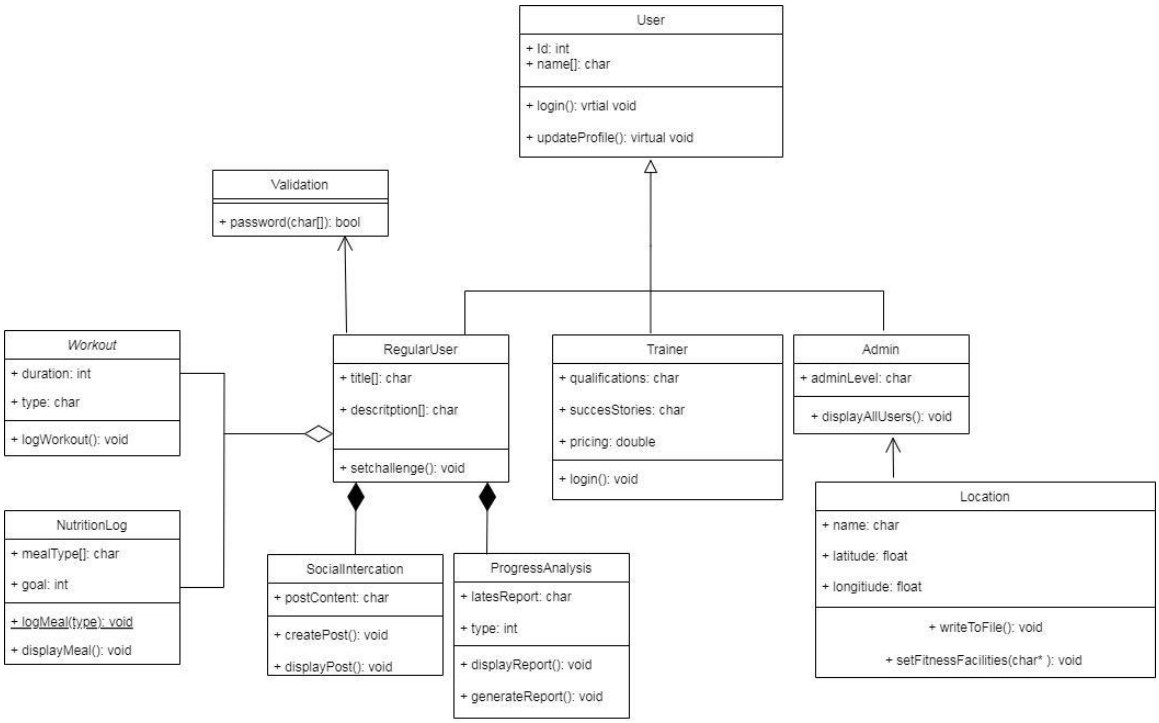
- Composition: The `Social Interaction`, `Progress Analysis`, classes are composed within the `Regular User` class to encapsulate various aspects of a user's fitness journey.

```
SocialInteraction socialInteraction; // Composition
ProgressAnalysis p; // Composition
char facilities[30];
```

- Aggregation: FitConnect uses aggregation to manage data such as workout logs and social interactions, where users can own multiple workouts and interactions.

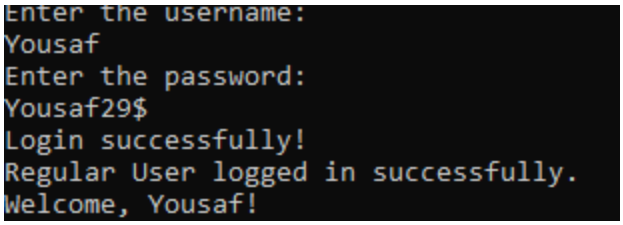
```
Workout workouts[MAX_WORKOUTS]; // Aggregation
int workoutCount;
NutritionLog nutritionLogs[MAX_MEALS]; // Aggregation
int mealCount;
Challenge challenges;
//int challengeCount;
```

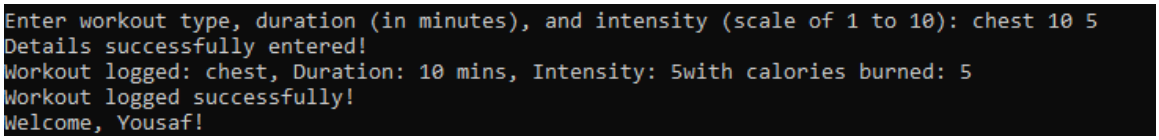
Class Diagrams

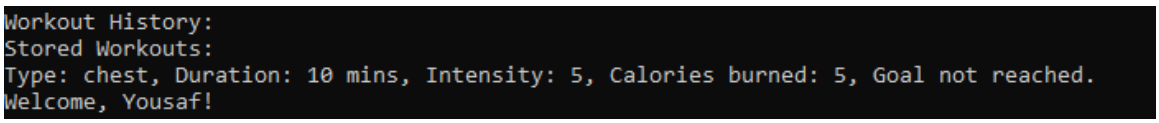


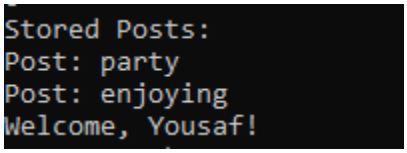
Test Cases

This section will list a set of inputs used to test each feature of the application, along with expected outputs. Test cases should cover User Authentication, Workout Tracking, Nutrition Tracking, Progress Analysis, and Social Interaction.









Future Directions

To enhance the scope of FitConnect, the following features could be implemented in future versions:

- Integration with Wearables: Sync the app with fitness devices like Fitbits or Apple Watches for real-time activity tracking.
- Gamification: Implement achievements, badges, and leaderboards to boost user motivation.
- AI Recommendations: Use machine learning to provide personalized workout and nutrition plans based on user data and goals.
- Advanced Analytics: Incorporate machine learning models to predict user progress and recommend specific improvements.