

Fitness Tracking System

Yeung Lai

System Brief

- A system to monitor daily health habits, track calorie intake, and manage fitness goals effectively.

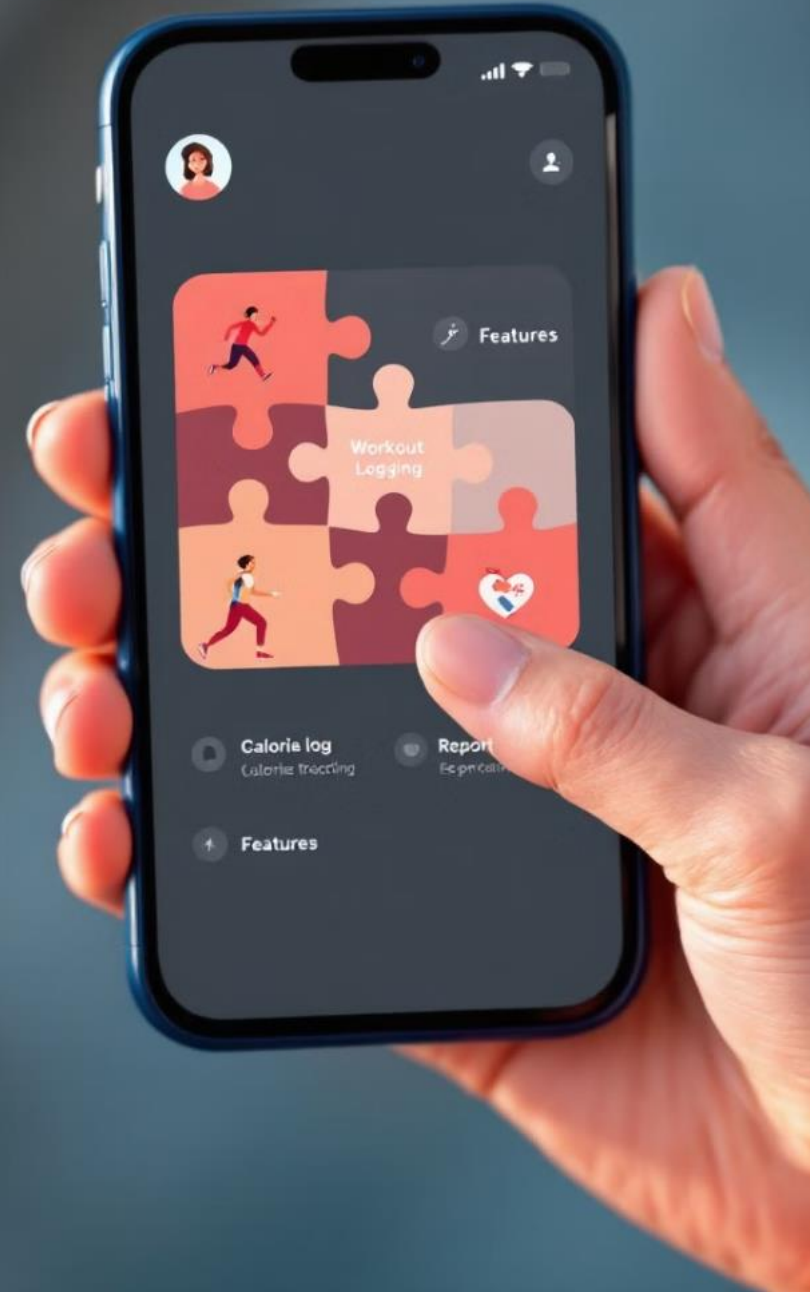
Design Purpose

Motivation

Many people struggle to monitor their daily calorie intake and physical activity. This often leads to inefficiency in managing fitness goals.

Objective

Develop a user-friendly tool that simplifies health management by providing a centralized platform to track and manage fitness goals.



System Role



ADMIN

Administrators manage user profiles, generate reports, and maintain system.



TRAINER

Trainers monitor user activities, generate reports, and provide personalized guidance to users.



USER

Users track their habits, log meals and activities, generate reports, and receive reminders.

Key Function -ADMIN

- User profile management
- Create a new user , enter personal fitness details.
- Generate report
 - Specific user
 - All user had registered in the system
 - Summaries saved as reports



Key Function -TRAINER

- Monitor user activities
- Generate report
 - Specific user
 - All user had registered in the system
 - Daily and Summaries saved as reports



Key Function -USER

- Keep track their own hobbit and record their daily healthy
- Log meal
 - with calories and timestamps.
 - Log activity
 - With message on reminder “You have consumed a total of xxx kcal today.”; “You have hit your daily intake goal.”...
- Log activity
 - With activity type with duration and calculate calories burned.
- Generate report
 - Generate their own report



System Workflow

Welcome page and Login with username and password.

```
Enter username: test
Enter password: test
Logged in successfully!
```

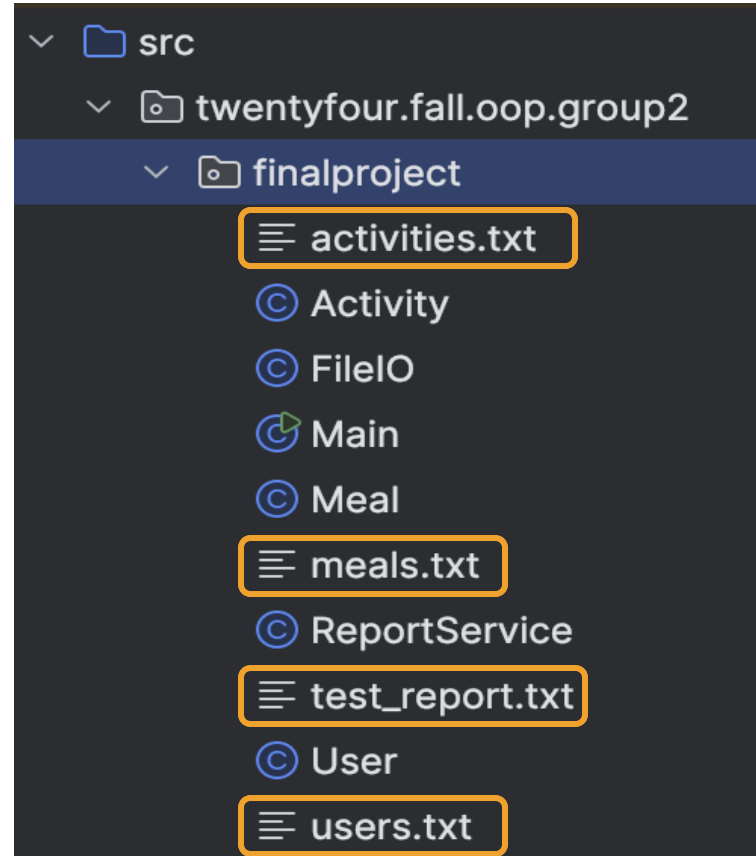
```
===== Welcome to Fitness Tracking System =====
1. Login
2. Exit
Enter your choice:
```

Role: User

```
===== User =====
Logged user: test
1. Log Meal
2. Log Activity
3. Generate Report
4. Save and Exit
Enter your choice:
```

User can base on their request to log meals and activities or generate their own report on their progress with calories consumed and burned.

Data Persistence



All logs are stored in files (meals.txt, activities.txt).

Create new user are stored in users.txt.

Reports are saved in text files with user name.

Example of Output (USER)

```
===== User =====
Logged user: test
1. Log Meal
2. Log Activity
3. Generate Report
4. Save and Exit
Enter your choice: 1
Enter meal name: lunch (egg and chicken)
Enter calories: 389
Enter date and time (YYYY-MM-DD HH:mm): 2025-01-08
Meal logged successfully!
You have consumed a total of 389 kcal today.
You have 1111 kcal left to reach your daily intake goal.
```

Log Meal

```
===== User =====
Logged user: test
1. Log Meal
2. Log Activity
3. Generate Report
4. Save and Exit
Enter your choice: 2
Available activities: Running, Swimming, Walking, Cycling
Enter activity type: Running
Enter duration (minutes): 30
Enter date and time (YYYY-MM-DD HH:mm): 2025-01-08
Activity logged successfully!
Calories burned: 300.0 kcal
You have exceeded your daily burn goal by 416.0 kcal!
```

Log Activity

```
===== Generate Report for test =====
Total Calories Consumed: 3189 kcal
Total Calories Burned: 536.0 kcal

Meals:
- ham and cheese bread, 300 kcal, logged at 2025-01-04
- breakfast, 300 kcal, logged at 2025-01-03
- lunch, 500 kcal, logged at 2025-01-03
- dinner, 300 kcal, logged at 2025-01-03
- breakfast , 300 kcal, logged at 2025-01-05
- lunch, 400 kcal, logged at 2025-01-05
- dinner, 400 kcal, logged at 2025-01-05
- dinner, 300 kcal, logged at 2025-01-06
- lunch (egg and chicken), 389 kcal, logged at 2025-01-08

Activities:
- walking, 60.0 kcal burned, logged at 2025-01-04 13:00
- swimming, 176.0 kcal burned, logged at 2025-01-05
- Running, 300.0 kcal burned, logged at 2025-01-08

Daily Goal Summary:
Daily Intake Goal: 1500 kcal
Daily Burn Goal: 120 kcal

Report successfully saved as: src/twentyfour/fall/oop/group2/finalproject/test_report.txt

===== End of Report for test =====
```

Generate Report

Example of Output (ADMIN)

```
===== Admin =====
Logged user: admin
1. Create new user
2. Generate Report
3. Save and Exit
Enter your choice: 1

===== Create new user =====
Create username:
yvonne
Create password:
password
Enter role (USER, TRAINER, ADMIN)
User
Enter age: 28
Enter sex (Male/Female): female
Enter weight (kg): 55
Enter height (cm): 165
Enter goal for daily calorie intake: 1000
Enter goal for daily calorie burning: 300
New user created successfully!
```

Create new user

```
===== All Users Report =====

===== Generate Report for yvonne =====
Total Calories Consumed: 120 kcal
Total Calories Burned: 0.0 kcal

Meals:
- aa, 90 kcal, logged at aa
- meal, 30 kcal, logged at 2024-12-29 12:30

Activities:
No activities logged.

Daily Goal Summary:
Daily Intake Goal: 0 kcal
Daily Burn Goal: 0 kcal

Report successfully saved as: src/twentyfour/fall/oop/group2/finalproject/yvonne_report.txt

===== End of Report for yvonne =====
-----

===== Generate Report for user =====
Total Calories Consumed: 0 kcal
Total Calories Burned: 0.0 kcal
```

Generate report for all user

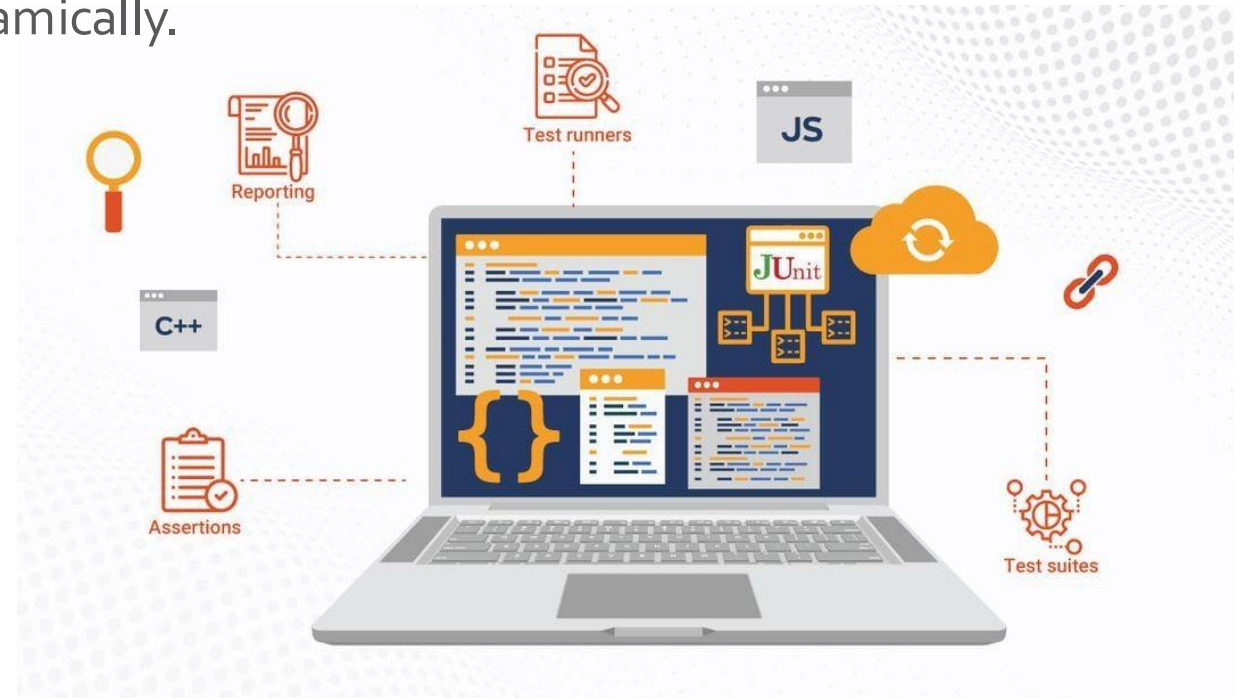
Project time log

Section	Hours
Design menu	4
Activity + Meal	3
Report service	4.5
FileIO	3
Message reminder	3
Error: When created new user didn't save in "users.txt"	2
Error: Activity calculate calories burned method didn't work	1
Problem found: USER should only can generate their own report, ADMIN // TRINER able to filer the user or generate all user.	1.5
Problem Found: The report should not only be displayed in the output but also saved as a file, with the user's report name as the file name.	1.5
Organize and clean the code (navigate, getUserChoose)	1.5
Final testing	1
Total	26

Spent around 5 days and approximately 26 hours in total.

Core concepts

- Encapsulation: Protect sensitive user data (e.g., passwords).
- Polymorphism: Flexible handling of data (e.g., meals and activities).
- File I/O with Exception Handling:
- Writing reports to files and handling IOException gracefully. Java Collections API: Manage meal/activity logs dynamically.



Challenge and Improvement

- **Managing File Read/Write Operations**
 - Ensuring data consistency across files (users.txt, meals.txt, activities.txt) was challenging.
 - Added **exception handling** to ensure the program doesn't crash when files are missing or errors occur during file operations.
- **Accurate Daily Calculations**

Used the `startsWith(currentDate)` method to filter meals and activities logged on the same day, ensuring calculations for daily calorie totals were precise.
- **Dynamic Filtering for Users**
 - Implemented `.stream().filter()` to dynamically filter meals and activities based on the logged-in user.
 - Allowed trainers and admins to access specific user data or generate reports for all users.
- **Adjustments to Visual Reports**

Initial proposal included **graph-based progress reports**, but since the app is console-based, this was replaced with summaries of total calories consumed and total calories burned.

Improvement

- **Weekly Workout Plans**

Trainers can assign a **weekly workout plan** to users.

Users will be able to track progress and receive weekly summaries of completed workouts.

- **Flexible Reporting Options**

Add the ability to generate reports for specific **dates**, **weeks**, or **years**, improving user insights over time.

- **Additional Health Features**

Introduce new metrics for a more comprehensive fitness tracking experience, including:

- **BMI**
- **Heart Rate Tracking**
- **Body Composition Metrics** (fat and muscle percentage).

END