

HealthLogix – Health Assistance

PROJECT REPORT

(2021 Regulation)

Submitted by

M RAHUL VYAS (RA2211047010096)

B JAYASURYA (RA2211047010123)

II Year/ IIIrd Semester

Academic Year: 2023 -2024

Under the Guidance of

Dr. SUMATHY G

Assistant Professor, Department of Computational Intelligence

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
In
ARTIFICIAL INTELLIGENCE



SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

NOV / DEC 2023



SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203**

BONAFIDE CERTIFICATE

Certified that this Course Project Report titled “**HealthLogix – Health Assistance**” is the bonafide work done by **M Rahul Vyas (RA2211047010096)**, **B Jayasurya (RA2211047010123)** of II Year/ IIIrd Sem B.Tech (AI) who carried out under my supervision for the course **21CSC101T - Object Oriented Design and Programming**. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Faculty In-Charge

Dr. Sumathy G

Assistant Professor

Department of Computational Intelligence

SRM Institute of Science and Technology

Kattankulathur Campus, Chennai

HEAD OF THE DEPARTMENT

Dr. R Annie Uthra

Professor and Head,

Department of Computational Intelligence,

SRM Institute of Science and Technology

Kattankulathur Campus, Chennai

3.3) Sequence Diagram:

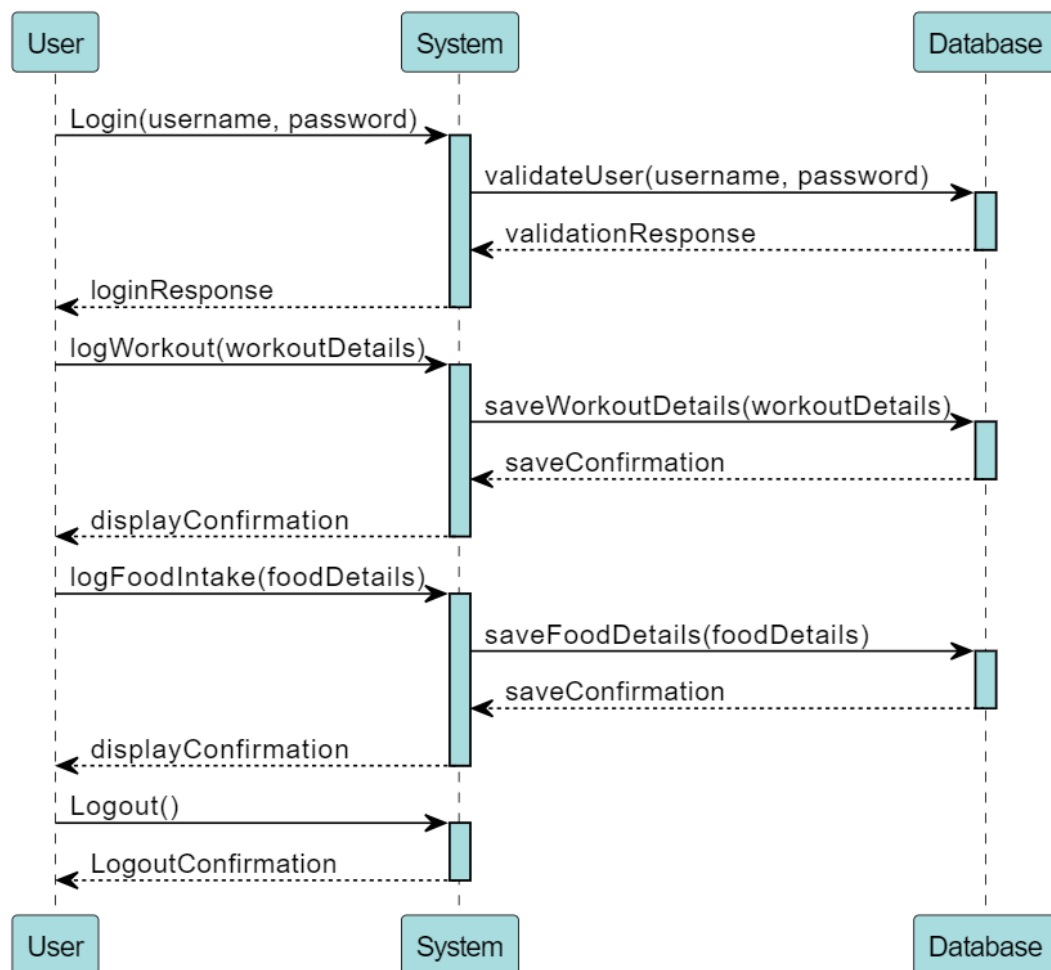


Fig. 3.3

In the sequence diagram:

1. User: Represents the individual using the system.
2. System: Represents the "HealthLogix - Health Assistance" system that provides the necessary functionalities.
3. Database: Represents the backend storage system.

The interactions are as follows:

- The user initiates a login request with the system, which then checks with the database to validate the user's credentials.
- Upon successful validation, the system sends a login response to the user.
- The user then logs a workout. The system processes the workout details and saves them in the database. Once saved, the system sends a confirmation back to the user.
- The user logs their food intake in a similar fashion. The system processes the food details and saves them in the database, following which a confirmation is sent to the user.
- Finally, the user logs out of the system.

3.5) State Chart Diagram:

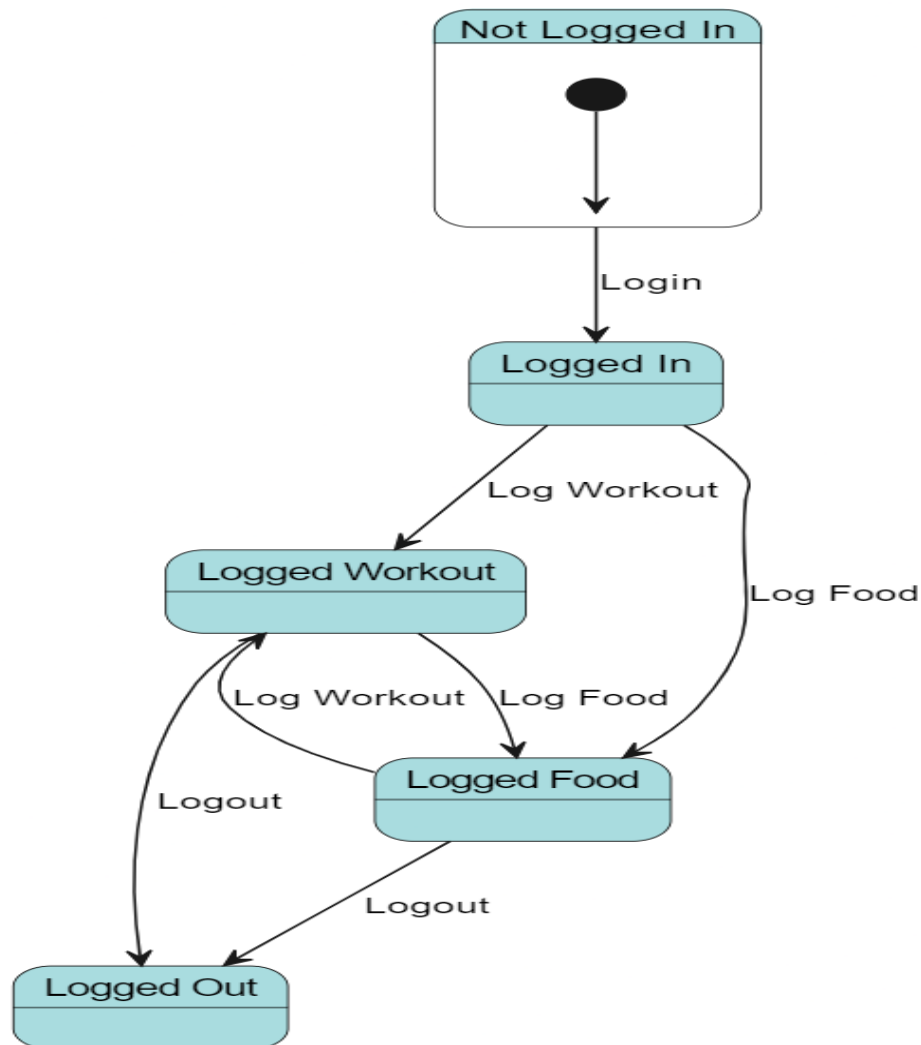


Fig. 3.5

The state chart diagram describes the various states the User can be in while interacting with the system:

1. Not Logged In: The initial state where the user is not authenticated.
2. Logged In: Once the user provides valid credentials, they transition to this state.
3. Logged Workout: After logging into the system, if the user logs a workout, they transition to this state. From here, they can either log food or log out.
4. Logged Food: If the user logs their food intake after logging in, they transition to this state. From here, they can either log a workout or log out.
5. Logged Out: The final state where the user ends their session.

Transitions between states are marked by actions, such as "Login", "Log Workout", and "Log Food".

3.7) Package Diagram:

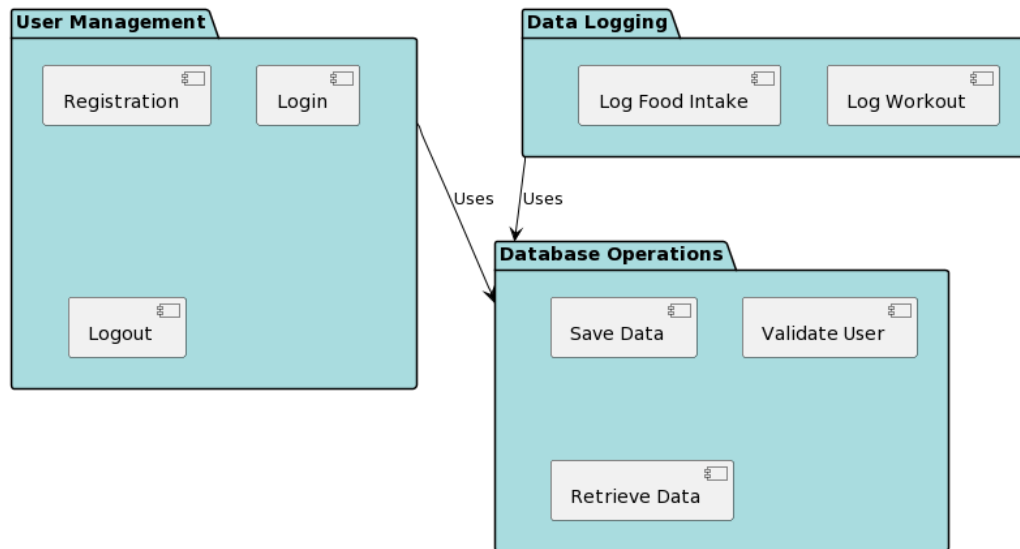


Fig. 3.7

The package diagram organizes functionalities into three conceptual packages:

1. User Management: Contains functionalities related to user interactions such as Registration, Login, and Logout.
2. Data Logging: Encompasses functionalities for logging activities, specifically Log Workout and Log Food Intake.
3. Database Operations: This package handles the underlying database interactions, which include Save Data, Validate User, and Retrieve Data.

There are also relationships (dependencies) between the packages:

- User Management uses Database Operations for tasks like validating user credentials during login.
- Data Logging uses Database Operations to save workout and food intake details..

3.8) Component Diagram:

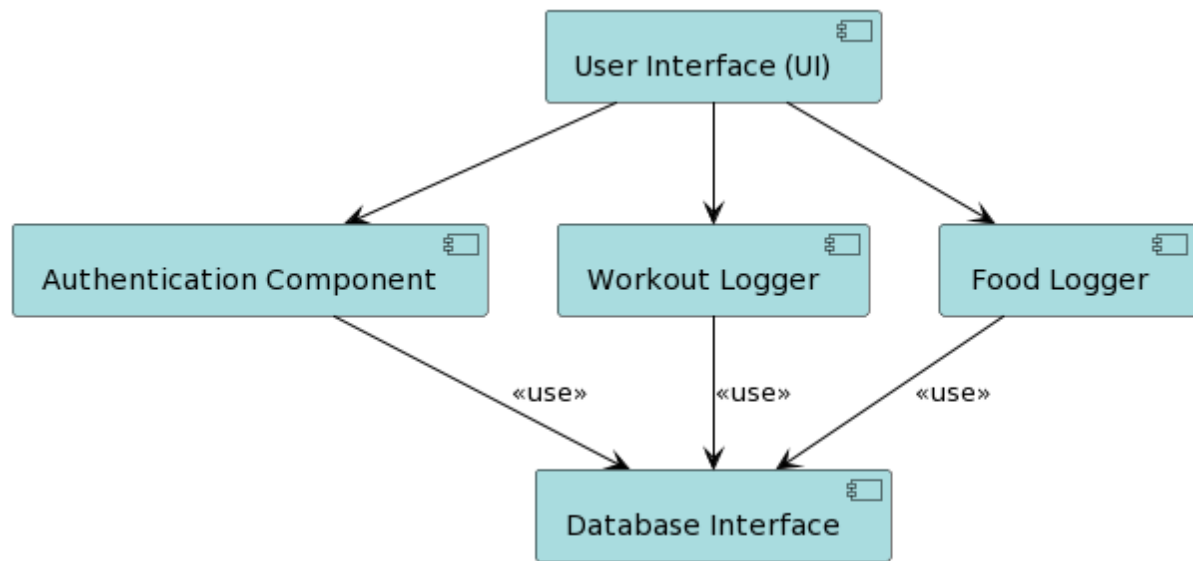


Fig. 3.8

The component diagram organizes the system into various components:

1. User Interface (UI): Represents the front-end of the system where users perform various activities.
2. Authentication Component: Handles user registration, login, and logout processes.
3. Workout Logger: Manages the logging and retrieval of workout data.
4. Food Logger: Handles the logging and fetching of food intake details.
5. Database Interface: Provides a consistent interface for other components to interact with the database.

In terms of interactions:

- The User Interface (UI) interacts with the Authentication Component, Workout Logger, and Food Logger.
- Both the Authentication Component, Workout Logger, and Food Logger use the Database Interface to interact with the underlying database.

3.9) Deployment Diagram:

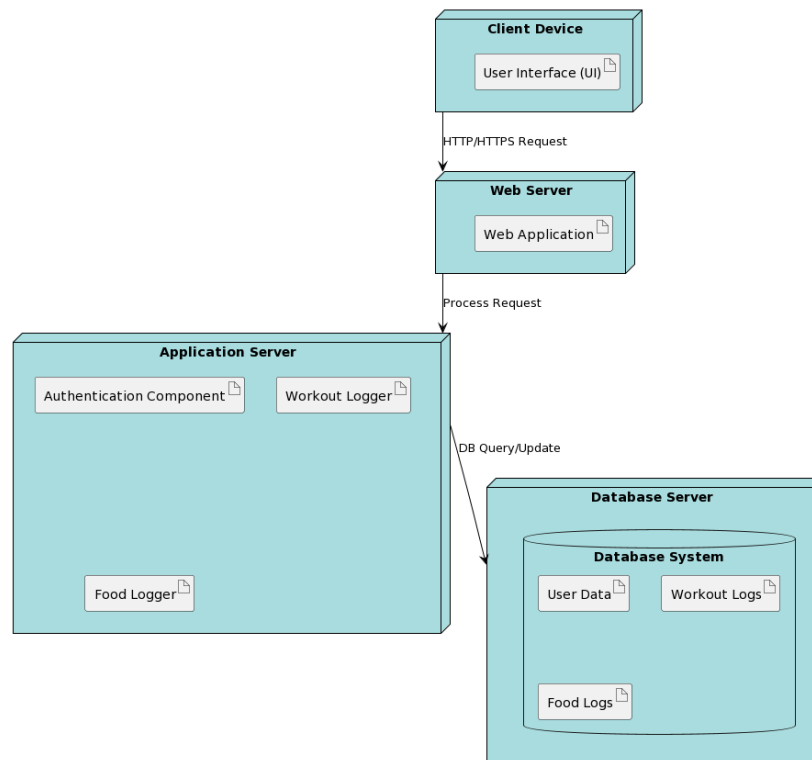


Fig. 3.9

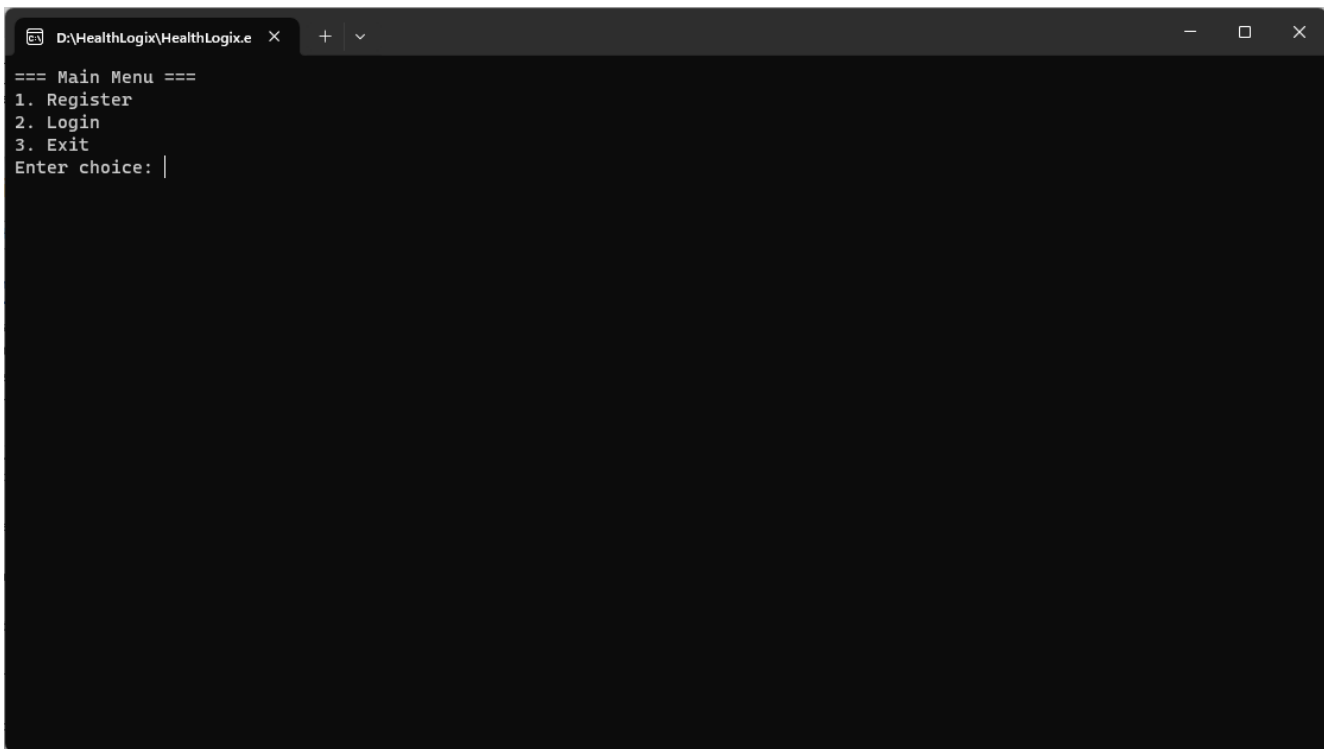
The deployment diagram visualizes the physical architecture of the system:

1. **Client Device:** Represents the user's device. The User Interface (UI) artifact runs here, allowing users to interact with the system.
2. **Web Server:** Hosts the Web Application artifact, which receives and responds to requests from the client device.
3. **Application Server:** Contains the core logic components of the system, including the Authentication Component, Workout Logger, and Food Logger.
4. **Database Server:** Hosts the Database System that stores various artifacts like User Data, Workout Logs, and Food Logs.

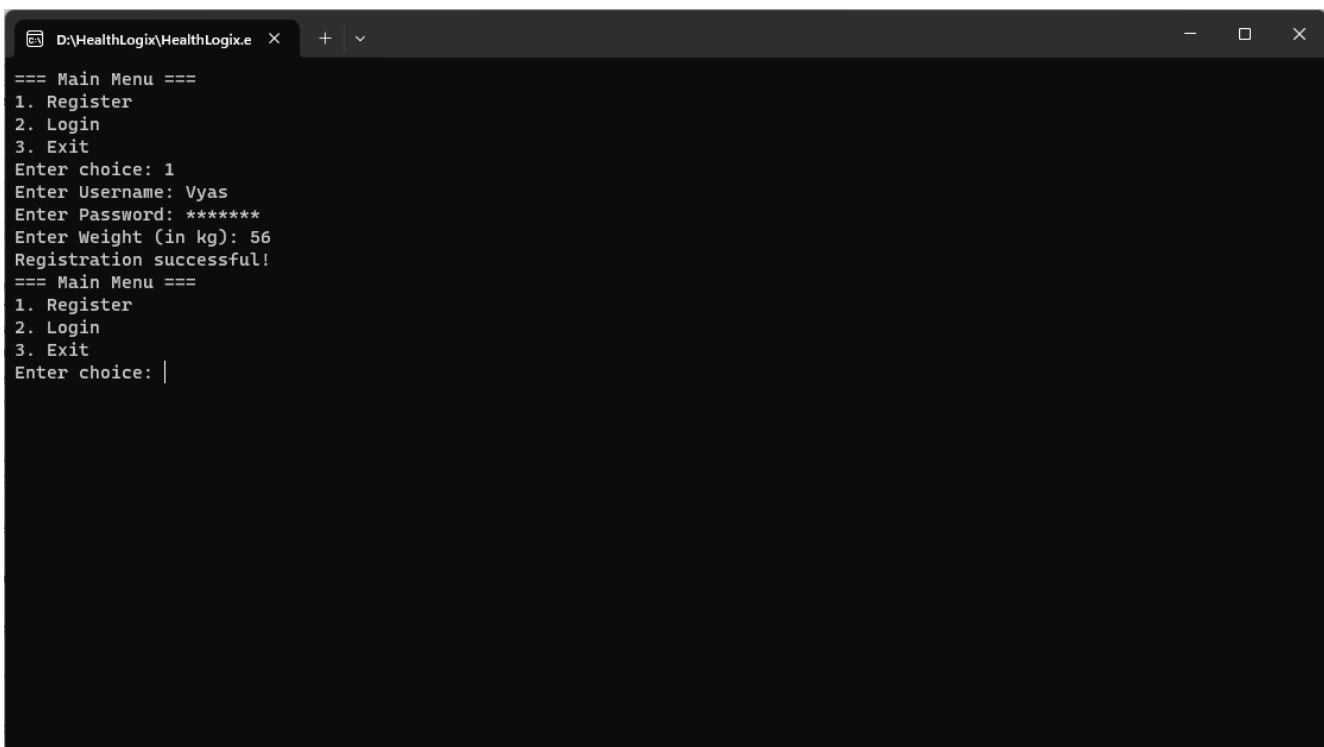
In terms of interactions:

- The Client Device communicates with the Web Server via HTTP/HTTPS requests.
- The Web Server sends processing requests to the Application Server.
- The Application Server interacts with the Database Server for database-related operations.

OUTPUT:



```
D:\HealthLogix\HealthLogix.e  X + v
=== Main Menu ===
1. Register
2. Login
3. Exit
Enter choice: |
```



```
D:\HealthLogix\HealthLogix.e  X + v
=== Main Menu ===
1. Register
2. Login
3. Exit
Enter choice: 1
Enter Username: Vyas
Enter Password: *****
Enter Weight (in kg): 56
Registration successful!
=== Main Menu ===
1. Register
2. Login
3. Exit
Enter choice: |
```



```
D:\HealthLogix\HealthLogix.e X + v
=== Main Menu ===
1. Register
2. Login
3. Exit
Enter choice: 2
Enter Username: Vyas
Enter Password: *****|
```

```
D:\HealthLogix\HealthLogix.e X + v
=== Main Menu ===
1. Register
2. Login
3. Exit
Enter choice: 2
Enter Username: Vyas
Enter Password: *****
Login successful!
=== Vyas's Dashboard ===
1. Log Workout
2. Log Food
3. Update Weight
4. Show Recent Workout and Food Details
5. Export Details
6. Logout
Enter choice: 1
Enter Activity: Swimming
Enter Duration (in minutes): 60
Enter the Calories burned: 210|
```

```
D:\HealthLogix\HealthLogix.e X + v
Enter Activity: Swimming
Enter Duration (in minutes): 60
Enter the Calories burned: 210
Workout logged successfully!
=== Vyas's Dashboard ===
1. Log Workout
2. Log Food
3. Update Weight
4. Show Recent Workout and Food Details
5. Export Details
6. Logout
Enter choice: 4

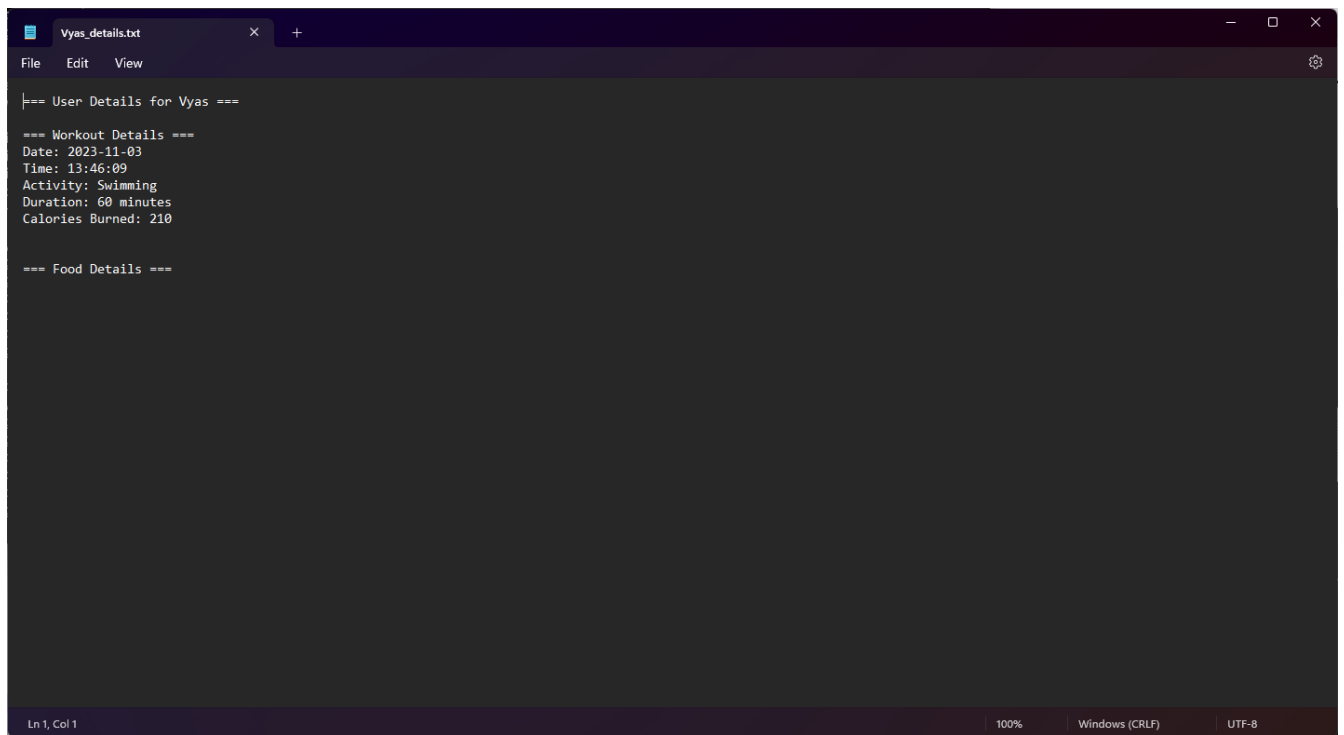
--- Recent Details ---
Last Workout:
Date: 2023-11-03
Activity: Swimming
Duration: 60 minutes
Calories Burned: 210
No foods logged yet.
-----

=== Vyas's Dashboard ===
1. Log Workout
2. Log Food
3. Update Weight
4. Show Recent Workout and Food Details
5. Export Details
6. Logout
Enter choice: |
```

```
D:\HealthLogix\HealthLogix.e X + v
5. Export Details
6. Logout
Enter choice: 4

--- Recent Details ---
Last Workout:
Date: 2023-11-03
Activity: Swimming
Duration: 60 minutes
Calories Burned: 210
No foods logged yet.
-----

=== Vyas's Dashboard ===
1. Log Workout
2. Log Food
3. Update Weight
4. Show Recent Workout and Food Details
5. Export Details
6. Logout
Enter choice: 5
User details exported successfully to Vyas_details.txt
=== Vyas's Dashboard ===
1. Log Workout
2. Log Food
3. Update Weight
4. Show Recent Workout and Food Details
5. Export Details
6. Logout
Enter choice: |
```



```
{}
{
  "name": "Vyas",
  "age": 30,
  "gender": "Male",
  "height": 180,
  "weight": 75,
  "email": "vyas.v@example.com",
  "phone": "9876543210",
  "address": {
    "street": "123 Main St",
    "city": "New York",
    "state": "NY",
    "zip": "10001"
  },
  "workout": {
    "date": "2023-11-03",
    "time": "13:46:09",
    "activity": "Swimming",
    "duration": 60,
    "calories_burned": 210
  },
  "food": {
    "meal": "Lunch",
    "items": [
      {
        "name": "Salmon",
        "quantity": 150,
        "calories": 300
      },
      {
        "name": "Rice",
        "quantity": 100,
        "calories": 200
      },
      {
        "name": "Broccoli",
        "quantity": 50,
        "calories": 50
      }
    ]
  }
}
```

Ln 1, Col 1

100% Windows (CRLF) UTF-8