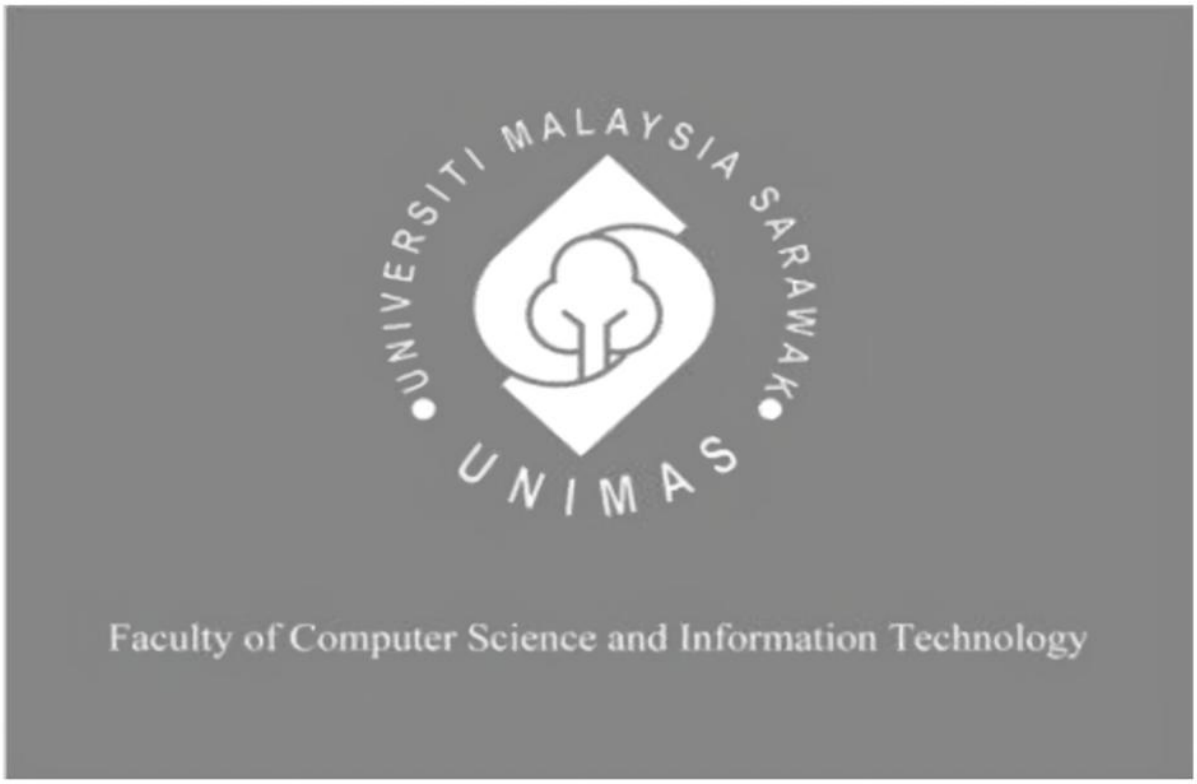


DEVELOPMENT OF A CALORIE TRACKING WEB APPLICATION: MACHINE LEARNING FOR TRACKING PHYSICAL ACTIVITIES OF UNIMAS STUDENTS

NGAWANG DORJE CHONG YIAN HUNG

(80367)

Bachelor of Computer Science with Honours

(Computational Science)

2025

DEVELOPMENT OF A CALORIE TRACKING WEB APPLICATION: MACHINE LEARNING FOR TRACKING PHYSICAL ACTIVITIES OF UNIMAS STUDENTS

NGAWANG DORJE CHONG YIAN HUNG

This project is submitted in partial fulfilment of the requirements for the degree of Bachelor of Computer Science with Honours (Computational Science)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

PEMBANGUNAN APLIKASI WEB PENGESAN KALORI: PEMBELAJARAN MESIN UNTUK MENGESAN AKTIVITI FIZIKAL PELAJAR UNIMAS

NGAWANG DORJE CHONG YIAN HUNG

Projek ini merupakan salah satu keperluan

untuk Ijazah Sarjana Muda Sains Komputer

Fakulti Sains Komputer dan Teknologi Maklumat

UNIVERSITI MALAYSIA SARAWAK

2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE Development of a Calorie Tracking Web Application: Machine Learning for Tracking Physical Activities of UNIMAS Students

ACADEMIC SESSION: 2024/2025

NGAWANG DORJE CHONG YIAN HUNG

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☐	CONFIDENTIAL	(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
☐	RESTRICTED	(Contains restricted information as dictated by the body or organization where the research was conducted)
☑	UNRESTRICTED	

(AUTHOR'S SIGNATURE)

Validated by

(SUPERVISOR'S SIGNATURE)

Permanent Address

Lot 3897, Lorong 1, Taman Rafflesia
Jalan Datuk Mohd Musa
94300 Kota Samarahan, Sarawak

Date: 17 Jul 2025

Date: 27 July 2025

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

DECLARATION

I hereby declare that the project is my original work. I have not copied from any other student's work or from any other sources except due to reference or acknowledgement is not made explicitly in the text, nor has any part had been written for me by another person.

(Ngawang Dorje Chong Yian Hung, 80367) **17th January 2025**

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak

ACKNOWLEDGEMENT

First and foremost, I would like to express my heartfelt gratitude to my academic institution, Universiti Malaysia Sarawak (UNIMAS).

I would also like to extend my deepest appreciation to my supervisor, Dr. Tiong Wei King, for giving me the opportunity to work under his supervision. His invaluable guidance and constructive feedback throughout the project have greatly contributed to the project's direction and completion.

My sincere thanks go to my family and friends for their unwavering support, motivation, and encouragement. Their belief in my abilities always boosted my morale when I feel stressed.

Also, I am truly grateful to all the individuals who participated in the questionnaires for their time and valuable input. Their responses have provided critical insights for this project.

Lastly, I would like to thank Sarawak Information Systems Sdn. Bhd. (SAINS) for granting me the opportunity to intern at their esteemed company. The knowledge, experience, and resources gained during my internship have significantly developed my skills and contributed to this project.

To everyone who has supported me in any way, I extend my deepest gratitude. This project would not have been possible without their contributions and encouragement.

ABSTRACT

This project aims to develop a calorie-tracking web application for students at Universiti Malaysia Sarawak (UNIMAS), utilising machine learning to track physical activities. The application allows users to manually record daily activities such as walking, running, or playing sports, along with details like duration, distance, and intensity. Machine learning algorithms are incorporated to estimate calorie expenditure accurately, providing users with a flexible and intuitive method of tracking their activities. Key features include Body Mass Index (BMI) calculation, weight status classification, and visualizations of calorie consumption trends. Additionally, the platform fosters user engagement through a discussion forum for sharing tips, a leaderboard to encourage healthy competition, and customizable progress tracking. Comprehensive testing confirmed the application's functionality and usability, with all features performing reliably and receiving highly positive feedback from students. Comprehensive testing, including functional and usability tests, was conducted to ensure reliability and ease of use, confirming that all application modules functioned correctly and effectively handled realistic user scenarios. The usability evaluation conducted with students yielded highly positive feedback, highlighting the application's intuitive interface, ease of task completion, and overall user satisfaction. By offering an accessible, cost-effective, and practical solution, this project empowers UNIMAS students to monitor their physical activities, understand their health metrics, and adopt healthier routines. The application demonstrates the integration of machine learning and web technologies to deliver accurate predictions and a user-friendly experience, addressing the limitations of existing fitness tracking solutions. Through this project, UNIMAS students are provided with a tool that not only supports their fitness goals but also encourages long-term wellness and community engagement.

ABSTRAK

Projek ini bertujuan untuk membangunkan sebuah aplikasi web penjejakan kalori untuk pelajar Universiti Malaysia Sarawak (UNIMAS), yang menggunakan pembelajaran mesin bagi menjejaki aktiviti fizikal. Aplikasi ini membolehkan pengguna merekod aktiviti harian secara manual seperti berjalan, berlari, atau bersukan, beserta maklumat seperti tempoh, jarak, dan intensiti. Algoritma pembelajaran mesin digunakan bagi menganggarkan penggunaan kalori dengan tepat, sekali gus menyediakan kaedah yang fleksibel dan intuitif untuk pengguna menjejaki aktiviti mereka. Ciri-ciri utama termasuk pengiraan Indeks Jisim Badan (BMI), klasifikasi status berat badan, dan visualisasi trend penggunaan kalori. Selain itu, platform ini menggalakkan penglibatan pengguna menerusi forum perbincangan untuk berkongsi tip, papan pendahulu untuk mendorong persaingan sihat, serta penjejakan kemajuan yang boleh disesuaikan. Ujian menyeluruh telah mengesahkan fungsi dan kebolehgunaan aplikasi ini, dengan semua ciri berfungsi secara konsisten serta mendapat maklum balas yang amat positif daripada pelajar. Ujian menyeluruh, termasuk ujian fungsi dan kebolehgunaan, telah dijalankan bagi memastikan kebolehpercayaan serta kemudahan penggunaan, mengesahkan bahawa semua modul aplikasi berfungsi dengan baik dan mampu mengendalikan senario pengguna yang realistik secara efektif. Penilaian kebolehgunaan yang dijalankan bersama pelajar telah menerima maklum balas yang sangat positif, menekankan antaramuka intuitif, kemudahan dalam menyelesaikan tugasan, serta kepuasan pengguna secara keseluruhan. Dengan menawarkan penyelesaian yang mudah diakses, kos efektif, dan praktikal, projek ini memperkasa pelajar UNIMAS untuk memantau aktiviti fizikal mereka, memahami metrik kesihatan, serta mengamalkan rutin yang lebih sihat. Aplikasi ini menunjukkan integrasi teknologi pembelajaran mesin dan web bagi memberikan ramalan yang tepat serta pengalaman pengguna yang mesra, sekaligus menangani kelemahan penyelesaian penjejakan kecergasan yang sedia ada. Melalui projek ini, pelajar UNIMAS dibekalkan dengan alat yang bukan sahaja menyokong matlamat kecergasan mereka tetapi turut menggalakkan kesejahteraan jangka panjang dan penglibatan komuniti.

TABLE OF CONTENTS

DECLARATION i
ACKNOWLEDGEMENT ii
ABSTRACT iii
ABSTRAK iv
TABLE OF CONTENTS v
LIST OF TABLES vii
LIST OF FIGURES viii
CHAPTER 1: INTRODUCTION 1
1.1 Introduction 1
1.2 Problem Statement 2
1.3 Aims And Objectives 2
1.4 Scope 3
1.5 Brief Methodology 4
1.6 Significance Of Project 6
1.7 Project Schedule 6
1.8 Expected Outcome 8
1.9 Project Report Outline 8
CHAPTER 2: LITERATURE REVIEW 10
2.1 Introduction 10
2.2 Review on Similar Existing Systems 10
2.2.1 MyFitnessPal 10
2.2.2 MyNetDiary 16
2.2.3 ACE Fitness 25
2.3 Comparison of the Existing Systems and the Proposed System 32
2.4 Review on Tools Used to Develop the Proposed System 33
2.4.1 Machine Learning 33
2.4.2 Backend and Frontend Framework 34
2.4.3 Visual Studio Code 35
2.5 Summary 35
CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN 36
3.1 Introduction 36
3.2 Requirement Analysis Phase 36
3.2.1 Data Analysis 36
3.2.2 Functional Requirements 46
3.3 System Planning Phase 47
3.3.1 Logical Design 47

3.3.2 Physical Design 77
CHAPTER 4: IMPLEMENTATION 90
4.1 Introduction 90
4.2 Development Tools 90
4.2.1 Visual Studio Code 91
4.2.2 HeidiSQL 92
4.2.3 Kaggle 93
4.2.4 XGBoost 93
4.2.5 Django 94
4.3 User Interface Design 94
4.3.1 User Login 95
4.3.2 User Registration 96
4.3.3 User Profile 97
4.3.4 Log Activity 100
4.3.5 Calorie Report 106
4.3.6 Discussion Forum 108
4.3.7 Leaderboard 115
4.4 Summary 115
CHAPTER 5: TESTING 116
5.1 Introduction 116
5.2 Functional Testing 116
5.3 Usability Testing 131
5.4 Summary 142
CHAPTER 6: CONCLUSION AND FUTURE WORK 143
6.1 Introduction 143
6.2 Project Achievement 143
6.3 Project Limitation 144
6.4 Future Work 144
REFERENCES… 146
APPENDIX A…. 148
APPENDIX B…. 152

LIST OF TABLES

Table 2.1: Functions in MyFitnessPal. 14
Table 2.2: Functions in MyNetDiary. 22
Table 2.3: Functions in ACE Fitness. 29
Table 2.4: Comparison of the Features of Three Existing Systems and the Proposed System. 32
Table 3.1: Use Case Specification: Login/Register. 49
Table 3.2: Use Case Specification: Log Activity. 50
Table 3.3: Use Case Specification: View Calorie Consumption Trend. 50
Table 3.4: Use Case Specification: View Leaderboard. 51
Table 3.5: Use Case Specification: View Discussion Forum. 51
Table 3.6: Use Case Specification: View Profile. 52
Table 3.7: Use Case Specification: Logout. 53
Table 3.8: Data Dictionary for User Table. 73
Table 3.9: Data Dictionary for Status Table. 74
Table 3.10: Data Dictionary for Activity Table. 75
Table 3.11: Data Dictionary for Activity Log Table. 75
Table 3.12: Data Dictionary for Discussion Table. 76
Table 3.13: Data Dictionary for Reply Table. 76
Table 3.14: Data Dictionary for Leaderboard Table (Derived Data). 77
Table 5.1: Functional Testing on User Registration 117
Table 5.2: Functional Testing on User Login 117
Table 5.3: Functional Testing on Logout 118
Table 5.4: Functional Testing on View User Profile 118
Table 5.5: Functional Testing on Edit Profile 119
Table 5.6: Functional Testing on BMI Calculator 120
Table 5.7: Functional Testing on View Activity Log 120
Table 5.8: Functional Testing on Delete from Activity Log 121
Table 5.9: Functional Testing on View Recent Logs 122
Table 5.10: Functional Testing on Add Activity Log from Recent Logs 122
Table 5.11: Functional Testing on Log New Activity 123
Table 5.12: Functional Testing on View Calorie Report 124
Table 5.13: Functional Testing on Download Calorie Report 124
Table 5.14: Functional Testing on View Discussion & Reply 125
Table 5.15: Functional Testing on Create New Discussion 126
Table 5.16: Functional Testing on Edit Existing Discussion 126
Table 5.17: Functional Testing on Delete Existing Discussion 127
Table 5.18: Functional Testing on Reply to Discussion 128
Table 5.19: Functional Testing on Edit Reply 128
Table 5.20: Functional Testing on Delete Reply 129
Table 5.21: Functional Testing on View Leaderboard 129
Table 5.22: Functional Testing on Toggle Leaderboard Visibility 130
Table 6.1 Summaries of the Objectives and Achievements. 143

LIST OF FIGURES

Figure 1.1 Agile Software Development (GeeksforGeeks, 2025) 4
Figure 1.2 The project schedule of FYP 1. 7
Figure 1.3: The project schedule of FYP 2. 7
Figure 1.4: The project schedule of FYP 1. 7
Figure 1.5 The project schedule of FYP 2. 8
Figure 2.1: Dashboard of MyFitnessPal Before Sign In or Register as a Member. 11
Figure 2.2: Input Personal Metrics and Goal Weight. 11
Figure 2.3: Interface of Physical Activity Log for a day. 12
Figure 2.4: Quick Log Options of Recent Entries. 12
Figure 2.5: Add Exercise from Search into the Log. 12
Figure 2.6: Chart Displaying Calorie Burned for the Last 7 Days. 13
Figure 2.7: Discussion Community on MyFitnessPal. 13
Figure 2.8: Profile of User. 14
Figure 2.9: Dashboard of MyNetDiary Before Sign In or Register as a Member. 17
Figure 2.10: Dashboard of MyNetDiary After Sign In. 18
Figure 2.11: Input Personal Metrics and Fitness Goal. 18
Figure 2.12: Interface of Physical Activity Log for a day. 19
Figure 2.13: Quick Log Options of Recent Entries. 19
Figure 2.14: Add Exercise from Search into the Log. 20
Figure 2.15: Chart Displaying Net Calorie for the Last 7 Days. 20
Figure 2.16: Discussion Community on MyNetDiary. 21
Figure 2.17: User Health Profile. 22
Figure 2.18: Dashboard of MyNetDiary Before Sign In or Register as a Member. 26
Figure 2.19: Interface of Calorie Estimation Based on Physical Activity. 27
Figure 2.20: Select Physical Activity from Given Options. 28
Figure 2.21: Compute body mass index (BMI) and classify weight status categories. 29
Figure 3.1: Agile Model. 36
Figure 3.2: Age Analysis of Respondents. 37
Figure 3.3: Gender Analysis of Respondents. 38
Figure 3.4: How do you currently track your calorie consumption or physical activities. 39
Figure 3.5: How often do you currently track your calorie consumption or physical activities. 40
Figure 3.6: I find it difficult achieving my health goals without activity tracking. 40
Figure 3.7: Existing calorie tracking applications are easy to use for beginners. 41
Figure 3.8: Which of the following features users find most helpful in calorie tracking system. 42
Figure 3.9: I prefer calorie estimation based on machine learning predictions. 43
Figure 3.10: I prefer applications with a straightforward and intuitive interface. 43
Figure 3.11: I would like to track the intensity of my activities. 44
Figure 3.12: Discussion forums or community support would make the application more engaging. 45
Figure 3.13: Having a leaderboard to compare my performance would be motivating. 45

Figure 3.14: Do you have any concerns about using a calorie tracking web-based system. ...46
Figure 3.15: Overall System Architecture of the Proposed System.....48
Figure 3.16: Use Case Diagram of Proposed System. 49
Figure 3.17: Activity Diagram: Login. 54
Figure 3.18: Activity Diagram: Registration. 55
Figure 3.19: Activity Diagram: Logout. 56
Figure 3.20: Activity Diagram: Log Activity/Calorie. 57
Figure 3.21: Activity Diagram: View Logged Activity List. 58
Figure 3.22: Activity Diagram: Remove Logged Activity from List. 59
Figure 3.23: Activity Diagram: View Calorie Consumption Trend. 60
Figure 3.24: Activity Diagram: View Discussions and Replies. 61
Figure 3.25: Activity Diagram: Create Discussion Post. 62
Figure 3.26: Activity Diagram: Create Reply. 63
Figure 3.27: Activity Diagram: Edit Discussion Post. 64
Figure 3.28: Activity Diagram: Edit Reply. 65
Figure 3.29: Activity Diagram: Delete Discussion Post. 66
Figure 3.30: Activity Diagram: Delete Reply. 67
Figure 3.31: Activity Diagram: View Leaderboard. 68
Figure 3.32: Activity Diagram: View Profile. 69
Figure 3.33: Activity Diagram: Update Profile.....70
Figure 3.34: Activity Diagram: Calculate BMI. 71
Figure 3.35: ERD of Proposed System. 72
Figure 3.36: Login Page.....78
Figure 3.37: Registration Page.....79
Figure 3.38: Log Activity and Calorie Estimation Page. 80
Figure 3.39: Calorie Report Page.....81
Figure 3.40: Discussion Forum Page. 82
Figure 3.41: Create Post.....83
Figure 3.42: View Post and Reply.....84
Figure 3.43: View Your Post and Reply Page. 85
Figure 3.44: Edit Post. 86
Figure 3.45: Edit Reply. 87
Figure 3.46: Leaderboard Page. 88
Figure 3.47: User Profile Page. 89
Figure 4.1: Visual Studio Code 91
Figure 4.2: HeidiSQL.....92
Figure 4.3: Kaggle 93
Figure 4.4: Login Page.....95
Figure 4.5: Registration Page.....96
Figure 4.6: User Profile Page.....97
Figure 4.7: BMI Calculator (Initial View) 98
Figure 4.8: BMI Calculator (With Result) 99
Figure 4.9: Edit Profile Page.....99

Figure 4.10: Log Activity Page (Empty State)....100
Figure 4.11: Log Activity Page (Activity Recorded)....101
Figure 4.12: Log Activity Page (Log Activity Option)....102
Figure 4.13: Log Activity Page (Log Activity)....103
Figure 4.14: Log Activity Page (Recent Activities)....104
Figure 4.15: Activity Log (Past Date)....105
Figure 4.16: Log Activity Page (Log Activity for Past Date)....105
Figure 4.17: Calorie Report (Last 7 Days)....106
Figure 4.18: Calorie Report (Last 30 Days)....107
Figure 4.19: Create Discussion Modal....108
Figure 4.20: Discussion Post Created in Explore Tab....109
Figure 4.21: Discussion Post Appearing in 'Your Posts' Tab....110
Figure 4.22: Post Seen by Another User in Explore Tab....111
Figure 4.23: Viewing into a Discussion Post....112
Figure 4.24: Comment Submitted on a Discussion Post....113
Figure 4.25: Updated 'Your Posts' View with a Comment....113
Figure 4.26: Editing a Discussion Post....114
Figure 4.27: Post After Being Edited....114
Figure 4.28: Leaderboard Page....115
Figure 5.1: Your Age Group....131
Figure 5.2: Your Gender....132
Figure 5.3: Are you able to successfully register an account for the calorie tracking system without confusion?....132
Figure 5.4: Are you able to login into the calorie tracking system?....133
Figure 5.5: Is your profile information (name, gender, height, weight) easy to view and update?....133
Figure 5.6: Is the BMI Calculator easy to use and understand?....134
Figure 5.7: Are you able to log a new activity without difficulty?....134
Figure 5.8: If yes, how easy was it to add a new activity for today or a past date?....135
Figure 5.9: Did the system clearly show total minutes and calories burned after logging an activity?....135
Figure 5.10: Are you able to reuse recent activities from the Recent Logs section?....136
Figure 5.11: Did the calorie report chart clearly visualize your calories burned for the selected date range (7, 30, or 60 days)?....136
Figure 5.12: Are you able to download the calorie report as PDF successfully?....137
Figure 5.13: How useful did you find the calorie summary and chart?....137
Figure 5.14: Is the process of create a discussion post clear and straightforward?....138
Figure 5.15: Are you able to edit or delete a discussion post easily?....138
Figure 5.16: Is the process of replying to a discussion clear and straightforward?....139
Figure 5.17: Are you able to edit or delete a reply easily?....139
Figure 5.18: How helpful do you find the discussion forum feature overall?....140
Figure 5.19: Are you able to view the leaderboard easily?....141
Figure 5.20: Calorie Tracking System Satisfaction....141

Figure A.1: Survey Form Description. 148
Figure A.2: Survey Section I: Demographic. 149
Figure A.3: Survey Section II: Current Fitness Habit & Tools. 150
Figure A.4: Survey Section III: Desired Features & Feedback. 151
Figure B.1: Survey Form Description. 152
Figure B.2: Survey Section I: Demographic. 153
Figure B.3: Survey Section II: Feature-specific Feedback. 154
Figure B.4: Survey Section II: Feature-specific Feedback. 155

CHAPTER 1: INTRODUCTION

1.1 Introduction

The emphasis on maintaining healthy lifestyles has been increasing recently, there is a growing demand among individuals, including students at UNIMAS, for accessible and reliable methods to track their calorie expenditure from physical activities (Bhole et al., 2024). Existing fitness applications, which often rely on wearable devices for real-time tracking, may lack flexibility for situations like tracking activities like swimming or playing sports, which carrying a phone or other devices is not practical. Recognizing this need for accessibility, flexibility, and affordable solutions, this project aims to develop a web-based calorie-tracking application tailored for UNIMAS students.

This application enables users to manually log their daily activities, such as walking, running, or playing sports, along with relevant details like duration, distance, and intensity. Powered by a machine learning estimation mechanism, the system provides accurate calorie burn predictions based on user inputs, offering a more intuitive and flexible approach to activity tracking. Additionally, the system provides users' Body Mass Index (BMI) calculation and categorizes their respective weight status into predefined categories such as underweight, normal, overweight, or obesity, giving users a clearer understanding of their health metrics.

The application also includes features to enhance user engagement and motivation, such as a discussion forum for sharing tips, a leaderboard to encourage healthy competition, and calorie consumption graphs to help users monitor progress and adjust their routines effectively. By providing an accessible platform that allows students to track their physical activities, understand their health metrics, and connect with others, this application seeks to encourage healthier routines, promote better lifestyles, and support long-term wellness goals.

1.2 Problem Statement

University students face difficulties when trying to keep up with a healthier lifestyle due to often packed schedules, academic pressures, and limited access to convenient fitness tracking tools. The lack of affordable, flexible solutions for monitoring daily calorie expenditure often contributes to unhealthy lifestyle patterns, including increased sedentary behaviour and poor physical health (Carballo-Fazanes et al., 2020). While many existing fitness solutions emphasize real-time tracking through wearable devices, often don't serve as a practical or affordable option for students. Moreover, activities such as swimming or sports where carrying a phone or device is not practical cannot be easily tracked through these solutions. As a result, users may miss out the opportunities to properly monitor their calorie use, creating lower awareness about how much calories they burn each day and about their health in general.

To deal with these barriers, this project offers a web-based calorie tracking system tailored for university students, which allow students to record physical activities manually and receive accurate calorie burn predictions through machine learning. By providing a more adaptable and accessible solution, this system allows students to make informed decisions about their health, supporting a more active and balanced lifestyle that aligns with their academic responsibilities.

1.3 Aims And Objectives

The objectives of this project are to:

1. Design a web-based calorie tracking application tailored to the needs of UNIMAS students, which allows them to manually log their physical activities and receive calorie burn predictions.
2. Develop a web-based calorie tracking application with a comprehensive interface that enables UNIMAS students to log various physical activities and visualize their calorie consumption trends through intuitive graphs.

3. Evaluate the usability and functionality of the proposed application.

1.4 Scope

This project is dedicated to developing a calorie tracking web application specifically tailored for UNIMAS students, addressing their unique physical activity tracking needs. Unlike most existing fitness applications fitness trackers that rely on wearable devices, this web application will allow users to log activities manually, making it accessible and convenient. The scope includes:

- Calorie Tracking and Prediction: Users can log physical activities such as competitive badminton and slow-paced walking, specifying details like activity type, duration, and distance (if applicable). The system will calculate the estimated calories burned using a machine learning model trained on relevant datasets.
- BMI Calculation: The system calculates a user's Body Mass Index (BMI) based on their height and weight. It dynamically categorizes the BMI into predefined weight status categories, such as underweight, normal, overweight, or obesity, to provide users with an understanding of their health status.
- Calorie Consumption Graph: The system provides users with visual representations of their calorie consumption trends over customizable timeframes, such as weekly or monthly. This feature helps users track their progress and make informed decisions about their physical activities.
- Discussion Forum: Users can engage in a discussion forum where they can post and reply to topics related to health, fitness, and lifestyle. This feature encourages community engagement and provides a platform for sharing advice and experiences.
- Leaderboard: A leaderboard ranks users based on their total calories burned over specified periods, such as daily, weekly, or monthly. Users can choose to withdraw from the leaderboard if they prefer not to participate in rankings.

- Limitations: The project will focus on calorie estimations only, and it will not provide real-time tracking or recommendations for activity optimization. Also, the system only provides estimation of calorie consumption, estimation of calorie intake is not provided.

1.5 Brief Methodology

Agile methodology was adopted for the development of the web-based calorie tracking application to ensure flexibility, iterative progress, and continuous alignment with user needs. Although Agile is commonly used in team-based environments, its principles were adapted effectively for individual development by structuring the workflow into manageable iterations, regular reviews, and progressive refinement.

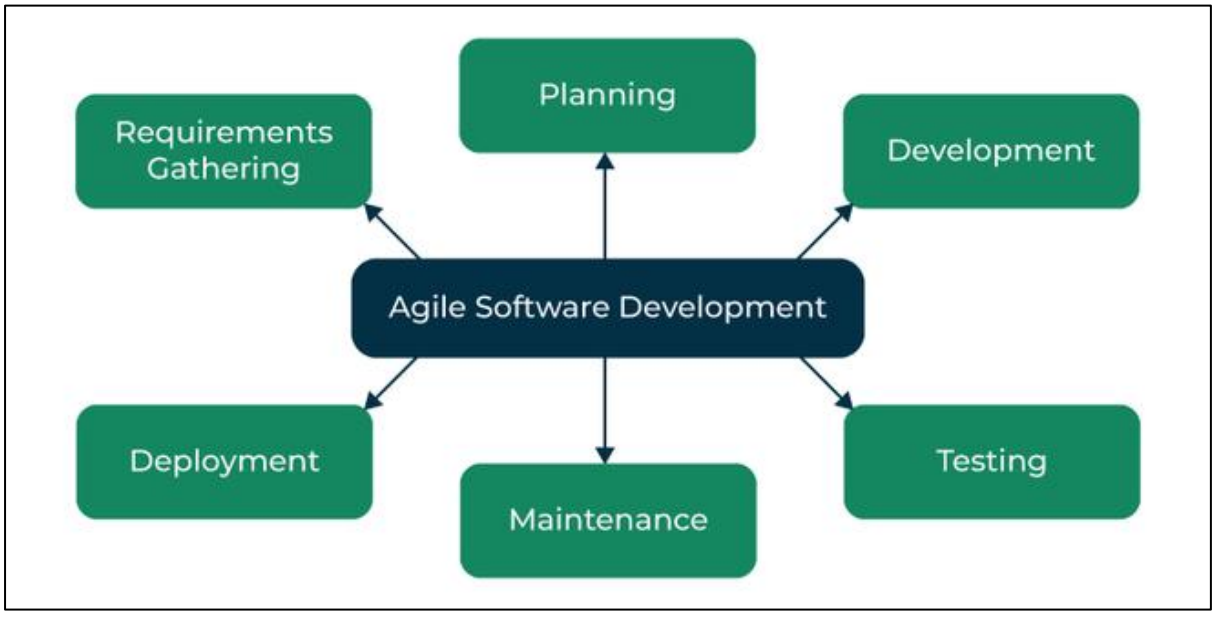


Figure 1.1 Agile Software Development (GeeksforGeeks, 2025)

The project began with a requirement-gathering phase. A Google Forms survey was distributed to UNIMAS students to identify their needs and expectations for tracking physical activity and calories burned. The survey results were analysed to extract core requirements such as user login, activity logging, calorie estimation, progress tracking, visual reporting,

discussion forum and leaderboard. These formed the foundation of the product's functional scope and guided the rest of the development process.

Once the requirements were gathered, the project tasks were broken down into manageable components. A feature backlog was created and prioritized. Each development cycle focused on implementing a small set of related features. Planning also included determining the best technology stack which is Python for backend development and machine learning logic, along with HTML, CSS, and JavaScript for building a user-friendly frontend.

Development was carried out in iterative cycles, focusing on individual modules. The frontend interface was designed to allow students to easily log their physical activities and view their progress. The backend managed user data and activity logs and integrated a machine learning model for calorie prediction based on user input like activity type and duration. Each component was implemented based on the current sprint's priorities.

Testing was integrated into every development cycle. After completing each module, testing was conducted to verify its functionality. Bugs were fixed immediately after detection, which ensured each part of the application worked correctly before moving to the next phase.

Once all modules were complete and tested, the application was prepared for deployment. A final system integration test was conducted to ensure all components functioned seamlessly together. After confirming system stability and performance, the application was deployed for use. The deployment process included ensuring that the user interface was responsive and that the database operations were stable.

Even after deployment, the application continued to be monitored and improved. Feedback from test users and the project supervisor was collected to guide further enhancements. Issues such as UI adjustments or algorithm improvements were documented and updated accordingly. The Agile approach allowed for continuous improvements to keep the application relevant and functional for its intended users.

1.6 Significance Of Project

The development of this web-based calorie tracking application offers meaningful benefits to individual users, particularly UNIMAS students, by aiding them to address the challenge of monitoring calorie expenditure without the need for wearable devices. By logging and reviewing their activities, users can gain insights into how their exercises and routines contribute to their calorie burn over time. This helps users make informed choices that align with their fitness goals, leading to a more active and balanced lifestyle. The ability to track progress also encourages motivation and consistency in maintaining a physically active routine, which is essential for well-being and stress management during university life. Through features like BMI calculation and weight status categorization, the system provides users with clear indicators of their health status. By understanding whether they fall into categories such as "underweight," "normal," or "obesity," users can set realistic goals and make well-informed decisions to improve their well-being.

In addition to personal health management, the system emphasizes community engagement. The discussion forum creates a space for users to share experiences, exchange fitness tips, and find support from like-minded individuals. This social aspect enhances the user experience by fostering motivation and encouraging consistent participation. Furthermore, the leaderboard introduces a gamified element to the system, promoting friendly competition among users based on their calorie-burning activities. By recognizing and ranking achievements, the leaderboard inspires users to stay active and engaged.

1.7 Project Schedule

The project schedule outlines a detailed timeline for completing various phases and milestones of the Final Year Project (FYP). It is divided into several components as described below:

Project Phase & Activity	Start	Finished	Duration (days)
Final Year Project:	**07/10/2024**	**25/01/2025**	**111**
1. Final Year Project 1	07/10/2024	25/01/2025	111
1.1 FYP Brief Proposal	07/10/2024	19/10/2024	13
Milestone 1	19/10/2024	19/10/2024	1
1.2 FYP Feedback and Comment from Examiner	20/10/2024	28/10/2024	9
Milestone 2	28/10/2024	28/10/2024	1
1.3 FYP Full Proposal	29/10/2024	14/11/2024	17
Milestone 3	14/11/2024	14/11/2024	1
1.4 FYP Chapter 1	15/11/2024	21/11/2024	7
Milestone 4	21/11/2024	21/11/2024	1
1.5 FYP Chapter 2	22/11/2024	13/12/2024	22
Milestone 5	13/12/2024	13/12/2024	1
1.6 FYP Chapter 3	14/12/2024	05/01/2025	23
Milestone 6	05/01/2025	05/01/2025	1
1.7 Submission of FYP 1 Final report & Paper for assessment	06/01/2025	17/01/2025	12
Milestone 7	17/01/2025	17/01/2025	1
1.8 FYP Presentation	24/01/2025	25/01/2025	2
Milestone 8	25/01/2025	25/01/2025	1

Figure 1.2 The project schedule of FYP 1.

Project Phase & Activity	Start	Finished	Duration (days)
Final Year Project:	**17/03/2025**	**02/07/2025**	**108**
2. Final Year Project 2	17/03/2025	02/07/2025	108
2.1 Proposed/Revised Structure of FYP Report	17/03/2025	01/04/2025	16
Milestone 1	01/04/2025	01/04/2025	1
2.2 Development of the Proposed System	02/04/2025	04/05/2025	33
Milestone 2	04/05/2025	04/05/2025	1
2.3 FYP Chapter 4	05/05/2025	15/05/2025	11
Milestone 3	15/05/2025	15/05/2025	1
2.4 Testing & Evaluation of the Proposed System	16/05/2025	27/05/2025	12
Milestone 4	27/05/2025	27/05/2025	1
2.5 FYP Chapter 5 & Chapter 6	28/05/2025	31/05/2025	4
Milestone 5	31/05/2025	31/05/2025	1
2.6 Submission of the FYP 2 Report and Paper Draft	01/06/2025	10/06/2025	10
Milestone 6	10/06/2025	10/06/2025	1
2.7 Submission of the FYP 2 Report, Paper, Source Code, and User Manual Completed	11/06/2025	23/06/2025	13
Milestone 7	23/06/2025	23/06/2025	1
2.8 FYP 2 Presentation	24/06/2025	02/07/2025	9
Milestone 8	02/07/2025	02/07/2025	1

Figure 1.3: The project schedule of FYP 2.

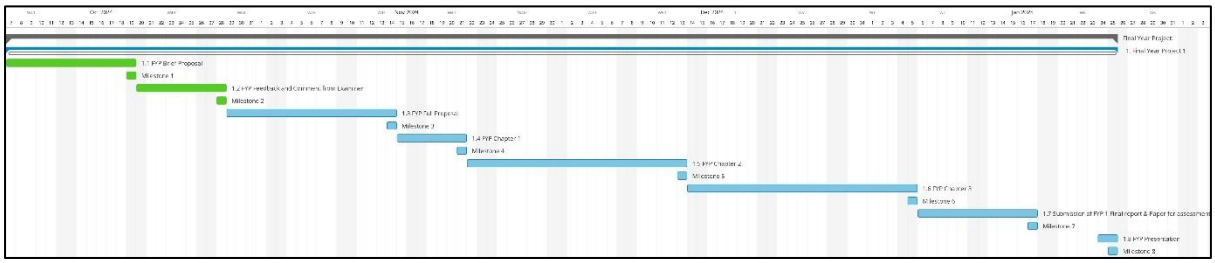


Figure 1.4: The project schedule of FYP 1.

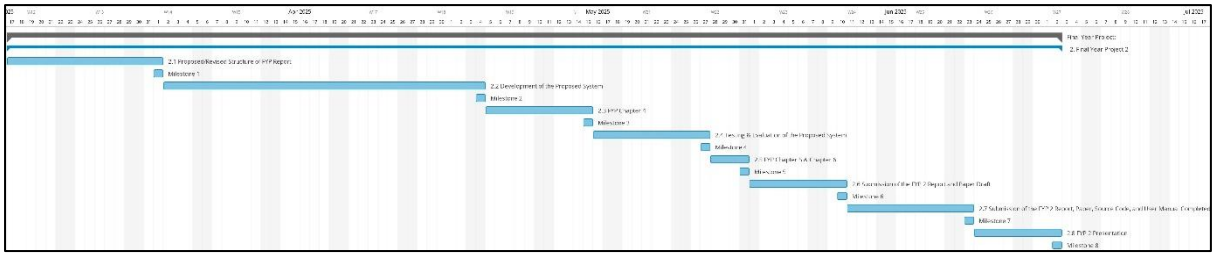


Figure 1.5 The project schedule of FYP 2.

The Gantt chart visually represents this timeline, with tasks and milestones aligned chronologically to aid in tracking progress and ensuring timely completion.

1.8 Expected Outcome

The expected outcome of this project is a fully functional web application that allows UNIMAS students to log their physical activities and receive predictions of calorie expenditure based on machine learning algorithms. Also, users will gain access to visual trends of their calorie consumption over weekly or monthly. The system will calculate users' BMI and categorize their weight status into predefined ranges such as underweight, normal, overweight, and obesity to provide better health awareness. The system will include a discussion forum and a leaderboard to enhance user engagement.

1.9 Project Report Outline

The project report for the Final Year Project is structured into six main chapters, each addressing a different phase of the project.

For, Final Year Project (FYP1), Chapter 1 provides an overview of the report, including the project's background, the problem statement, the scope, the aims and objectives, the methodology, the significance of the project, the project schedule, the expected outcome, the project outline and the summary. Chapter 2 involves a literature review and the study of related online resources, such as research papers and similar web applications. Also, a comparison between this project and the existing systems will be analysed in this chapter. Chapter 3 outlines

the requirements and specifications for this project. System diagrams such as use case diagrams, activity diagrams, entity relationship diagram, graphical user interface designs, and system architecture design will be developed. Also, the data collection method, which is a Google Form questionnaire will be included.

For Final Year Project 2 (FYP2), Chapter 4 highlights the implementation of the system design through development and implementation. It also covers a detailed technical overview of the proposed system. Chapter 5 represents the test plan of the proposed system where every system test will have a pass or fail record. Chapter 6 describes the conclusion of this project and provides recommendations for possible future work on the proposed system.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter will focus on the literature review of existing systems related to the proposed system. Three relevant systems will be chosen from online sources for review and analysis. This process aims to comprehend the concepts, technologies, techniques, and implementations they cover. The insights from comparison will help identify the possible requirements, improvements, and limitations of the proposed system by comparing it with these selected systems. Following that, tools used to develop the proposed system are discussed.

2.2 Review on Similar Existing Systems

Three relevant systems are being reviewed and analysed in this section. Based on the searching on the Internet, “MyFitnessPal”, “MyNetDiary” and “ACE Fitness” are chosen as the sources for the literature review. Further details of the features are discussed in the section below.

2.2.1 MyFitnessPal

MyFitnessPal is a digital health and fitness platform founded in 2005 by Mike Lee. It serves as an automated tool that connects users with a comprehensive food database, enabling them to track their caloric intake and nutritional information efficiently (Gleantap, 2023). As of 2023, MyFitnessPal has over 200 million registered users and features a database that includes more than 7 million foods, making it one of the leading applications in the health and fitness sector (Business of Apps, 2024).

Below are some of the MyFitnessPal interface:

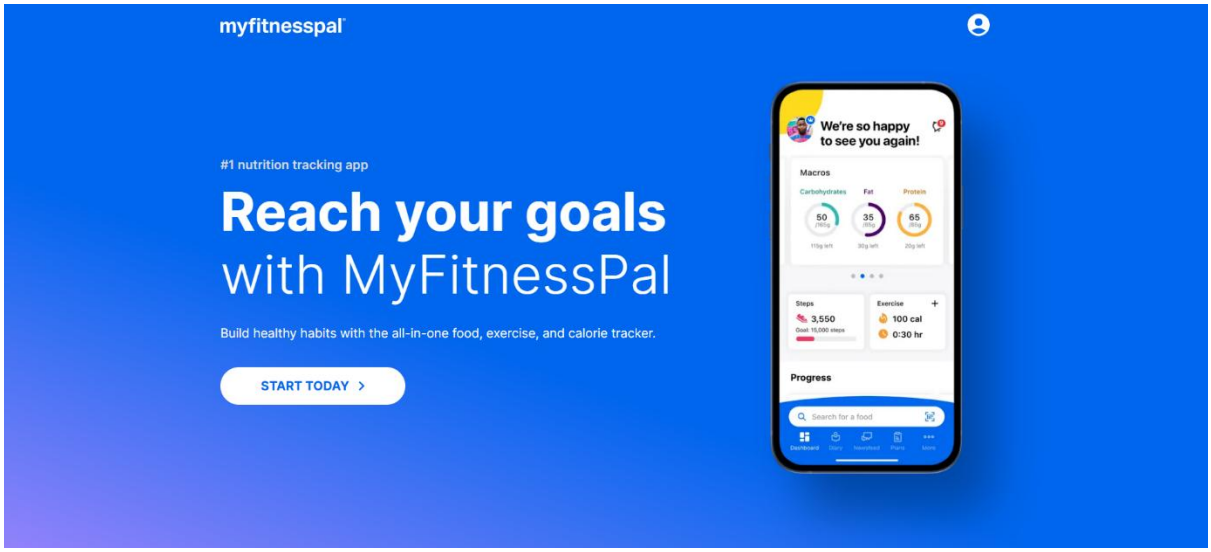


Figure 2.1: Dashboard of MyFitnessPal Before Sign In or Register as a Member.

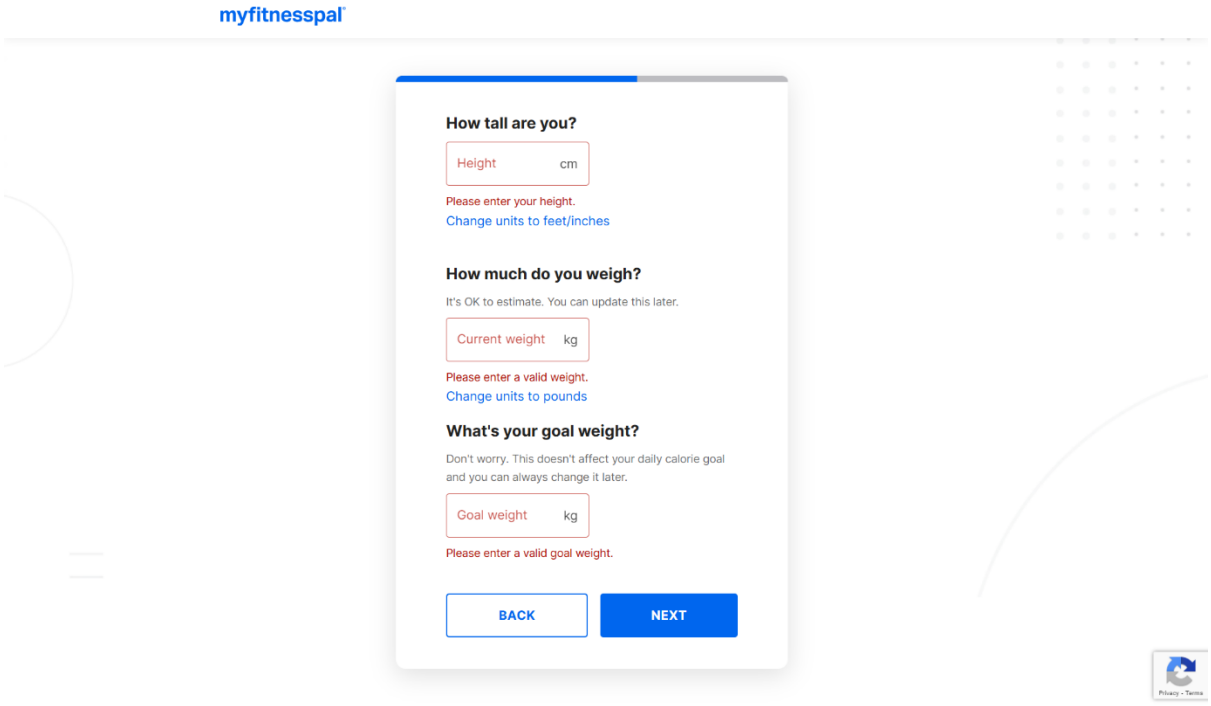


Figure 2.2: Input Personal Metrics and Goal Weight.

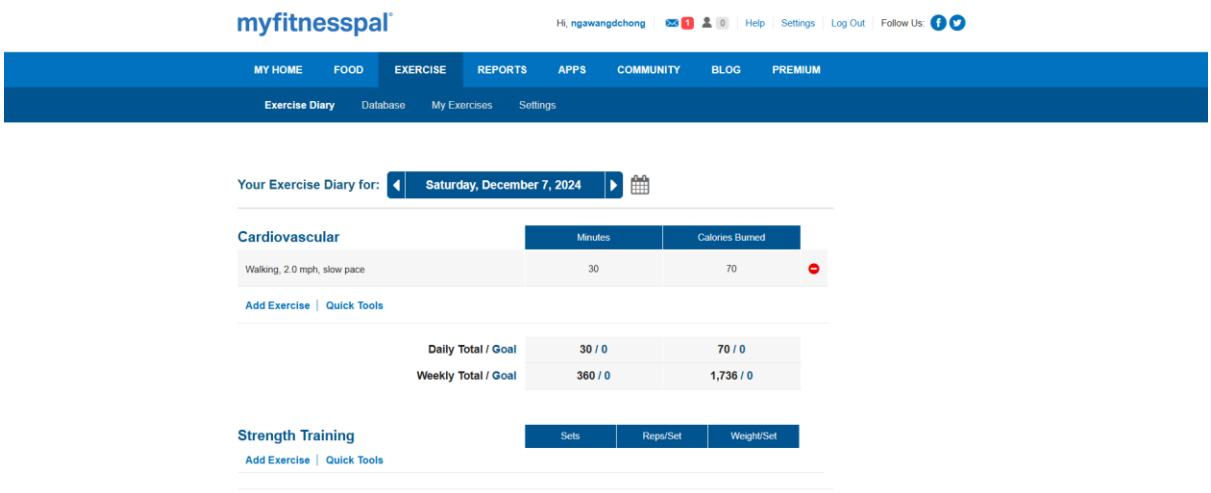


Figure 2.3: Interface of Physical Activity Log for a day.

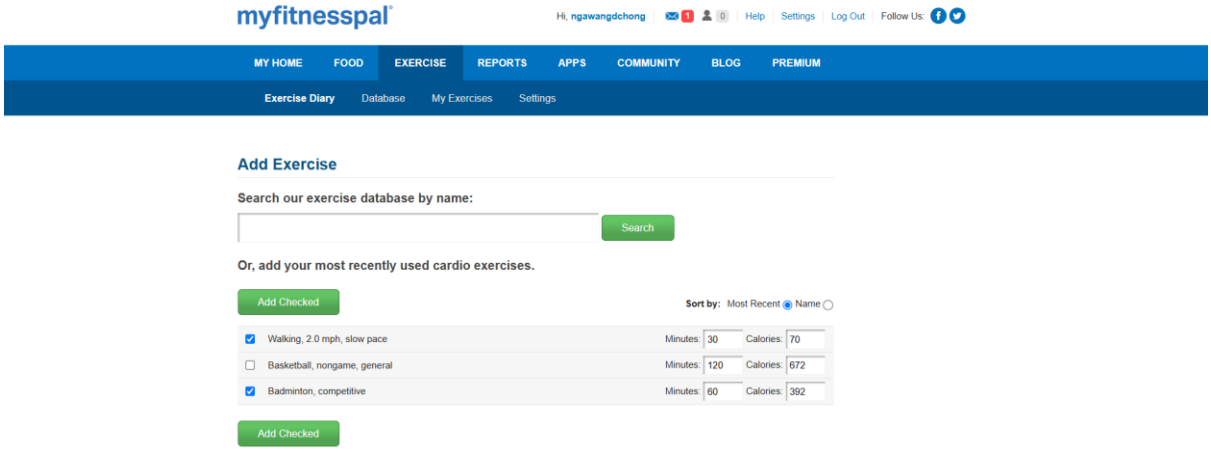


Figure 2.4: Quick Log Options of Recent Entries.

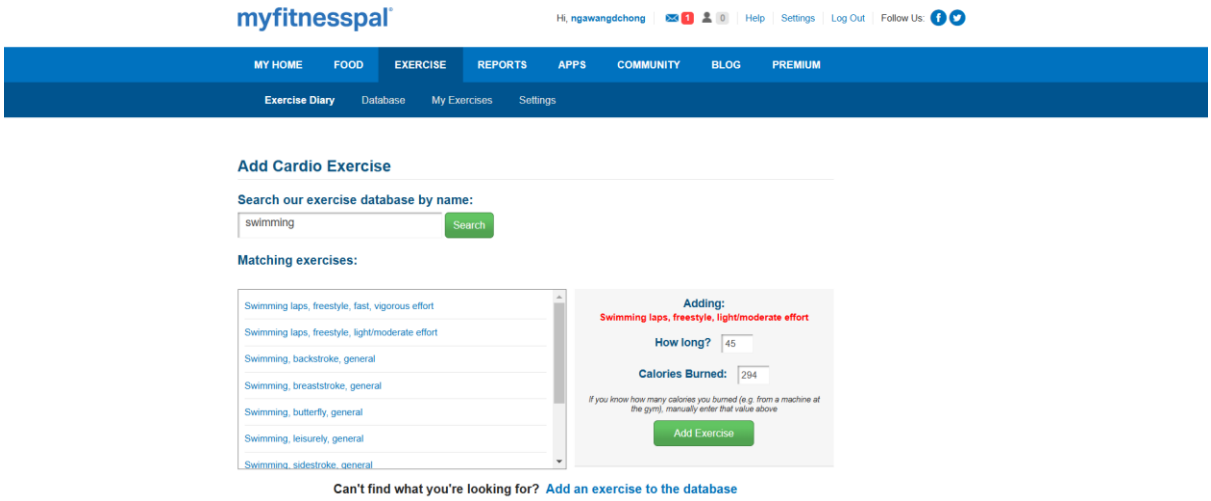


Figure 2.5: Add Exercise from Search into the Log.

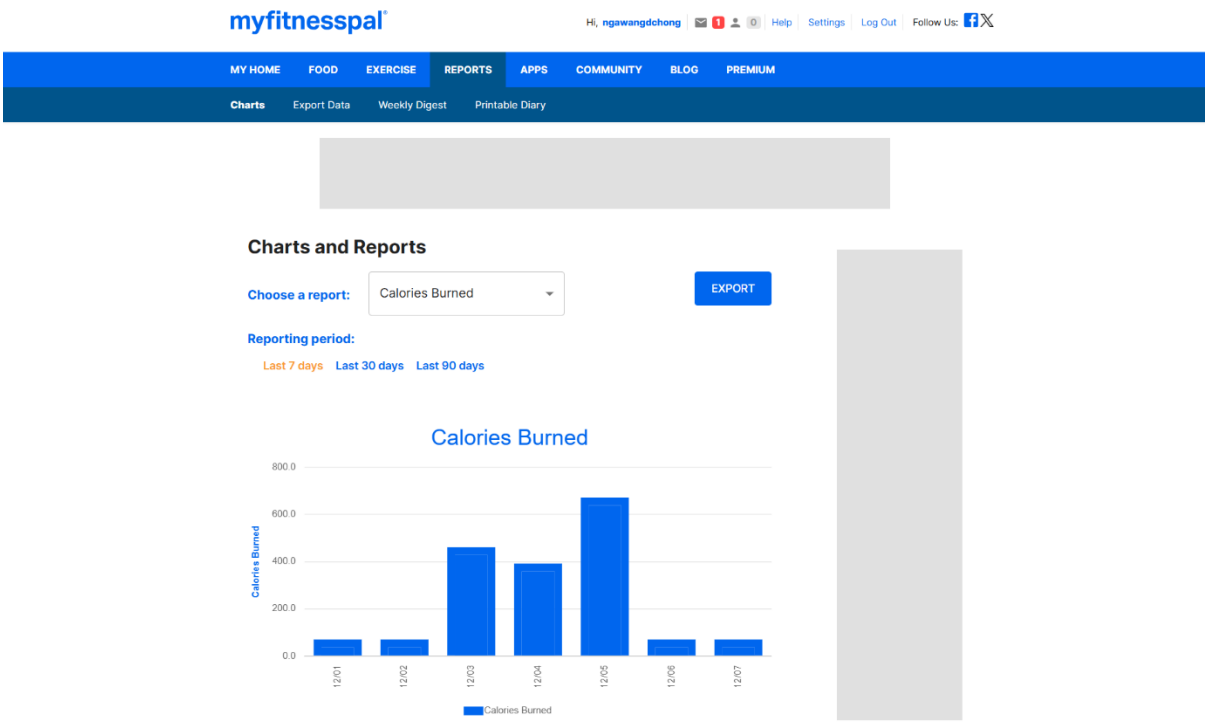


Figure 2.6: Chart Displaying Calorie Burned for the Last 7 Days.

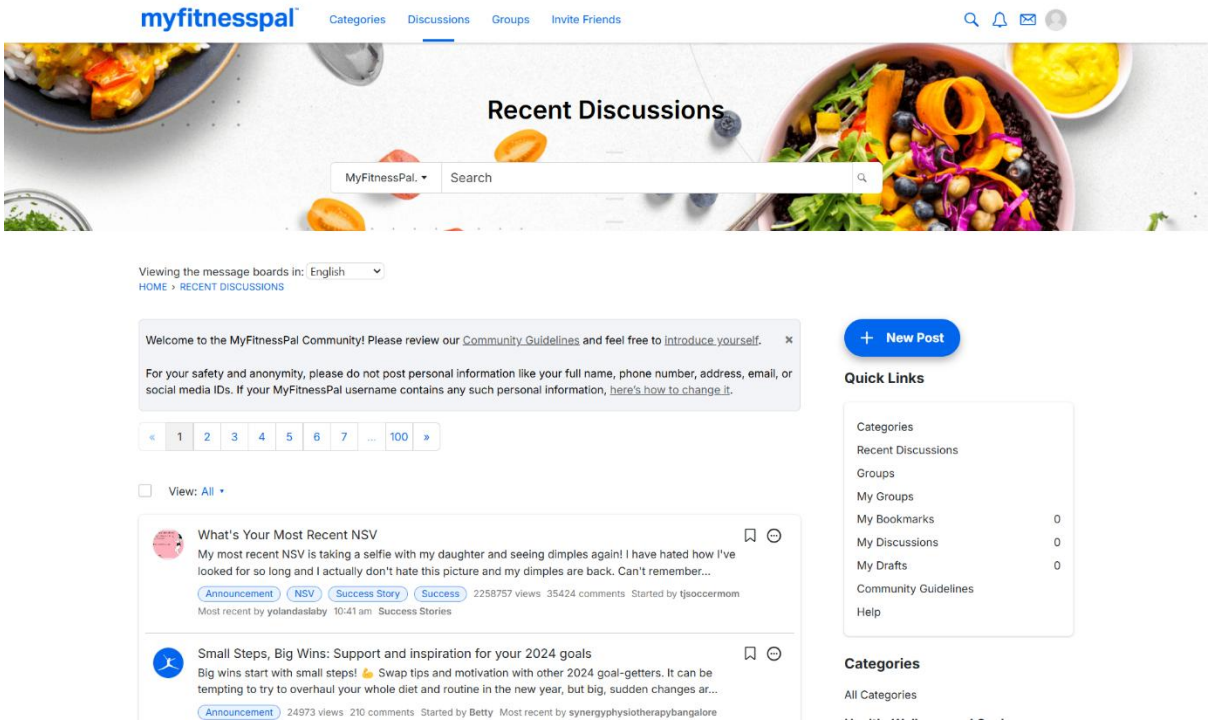


Figure 2.7: Discussion Community on MyFitnessPal.

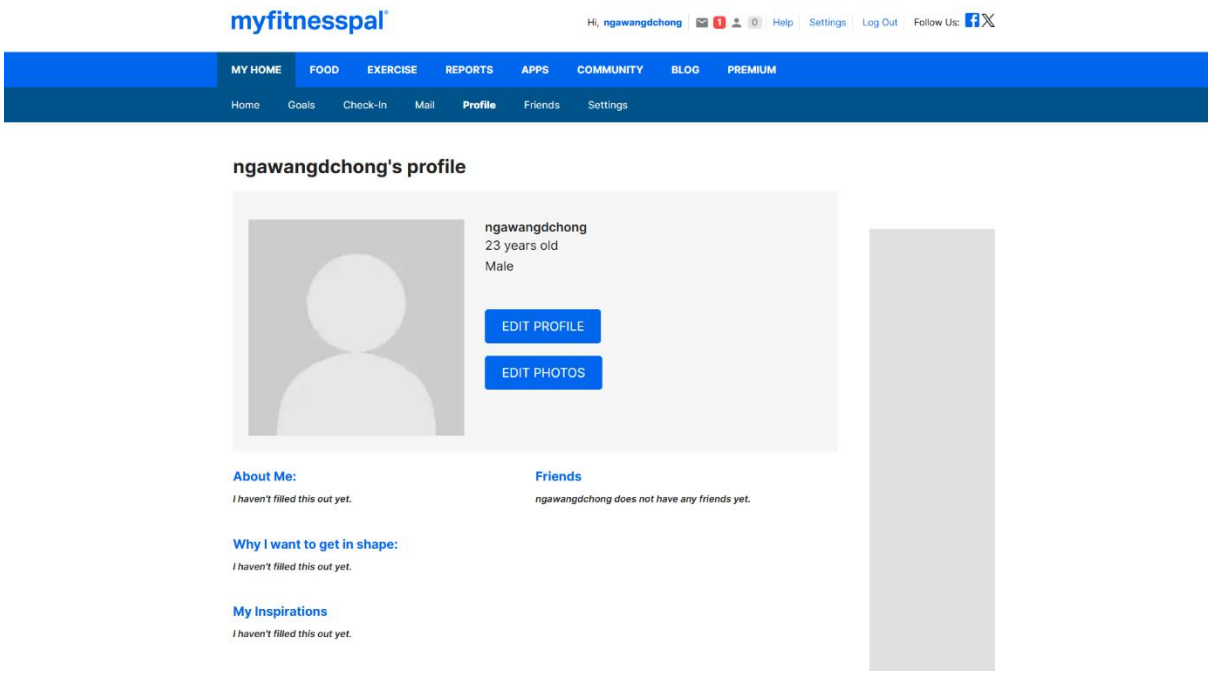


Figure 2.8: Profile of User.

There are some functions in MyFitnessPal will be listed in the table below:

Table 2.1: Functions in MyFitnessPal.

Functions	Descriptions
Register	*Figure 2.1* shows the dashboard of MyFitnessPal before the user sign in or register as a member. User can register by clicking the 'START TODAY' button.
Sign in	In *Figure 2.1*, user can sign in through clicking a profile image at the top right of the page. Users can sign in by entering their email and password, other options such as Google and Facebook sign in are provided as well.
Input Personal Metrics and Goal Weight	In *Figure 2.2*, users need to input their height, weight and goal weight during the registration process. MyFitnessPal uses the personal metrics inputted for calorie estimation.
Log Physical Activity	*Figure 2.3* shows the interface for users to log their physical activity completed in a particular day. After inputting the exercise and the exercise duration, the

	inputted duration and the calorie consumption estimated will be shown in the interface. *Figure 2.4* shows that when users want to add an exercise, there are quick log options of their recent entries. Users can also search for activity within the database to be inputted as shown in *Figure 2.5*.
Chart to display calorie consumption trends	In *Figure 2.6*, there is a chart displaying the calorie burned in a period (e.g. last 7 days) allowing users to make necessary adjustments based on the calorie consumption trends.
Discussion Community	*Figure 2.7* shows that MyFitnessPal has a discussion community that allows users to share their fitness experience.

Strengths of MyFitnessPal

- Free Basic Functionality:

  One of the most significant advantages of MyFitnessPal is that it offers a robust set of essential features without requiring a premium subscription. Users can easily track their calorie intake, log their food consumption, and monitor their physical activity through a comprehensive chart. This accessibility allows wide range of individuals to utilize the app effectively for their health and fitness goals.

- Community Support:

  MyFitnessPal includes social features that allow users to connect with friends and share progress, providing motivation and accountability through community engagement.

Weaknesses of MyFitnessPal

- No User Health Profile Presented

  As shown in *Figure 2.8*, rather than including the critical health metrics such as height, weight, Body Mass Index (BMI) and weight status categories, the user profile of

MyFitnessPal only shows the basic account details of the users. This means that while users can track their food intake and exercise, but they won't find an intuitive health snapshot readily available in their profile.

In summary, MyFitnessPal is a useful health and fitness application that offers a variety of features designed to help users manage their diet and exercise effectively. One of its standout strengths is the free basic functionality, which allows users to track calorie intake, log food consumption, and monitor physical activity without the need for a premium subscription. The activity choice for logging is also wide, allowing users to select from a diverse range of exercises. Additionally, MyFitnessPal fosters community support through social features that allow users to connect with friends, share progress, and motivate each other, enhancing the overall user experience. However, a significant limitation is the absence of a comprehensive user health profile. Instead of displaying essential health metrics like height, weight, and Body Mass Index (BMI), the user profile only includes basic account details.

2.2.2 MyNetDiary

MyNetDiary is a digital health application designed to assist users in tracking their food intake, exercise, and overall health metrics. It was introduced in 2009 and has gained recognition for its user-friendly interface and comprehensive features aimed at promoting weight loss and healthy living. The app allows users to set personalized weight goals and provides tools to monitor progress effectively (MyNetDiary, 2023).

Below are some of the MyNetDiary interface:

Figure 2.9: Dashboard of MyNetDiary Before Sign In or Register as a Member.

Figure 2.10: Dashboard of MyNetDiary After Sign In.

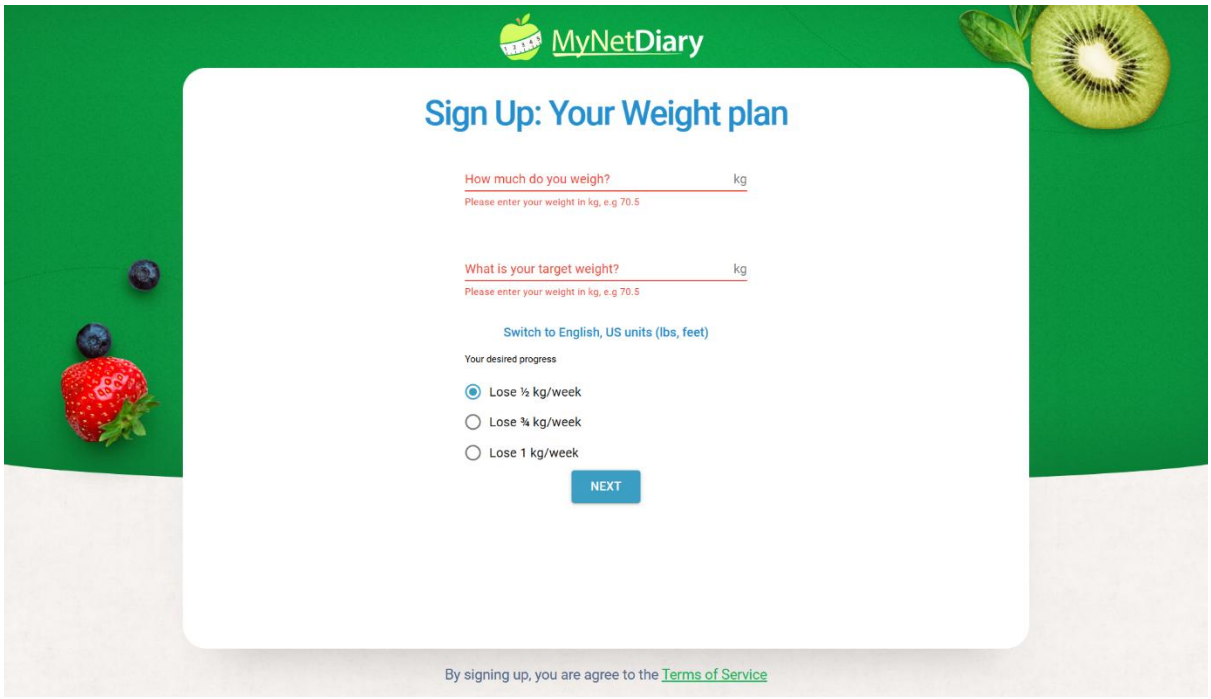


Figure 2.11: Input Personal Metrics and Fitness Goal.

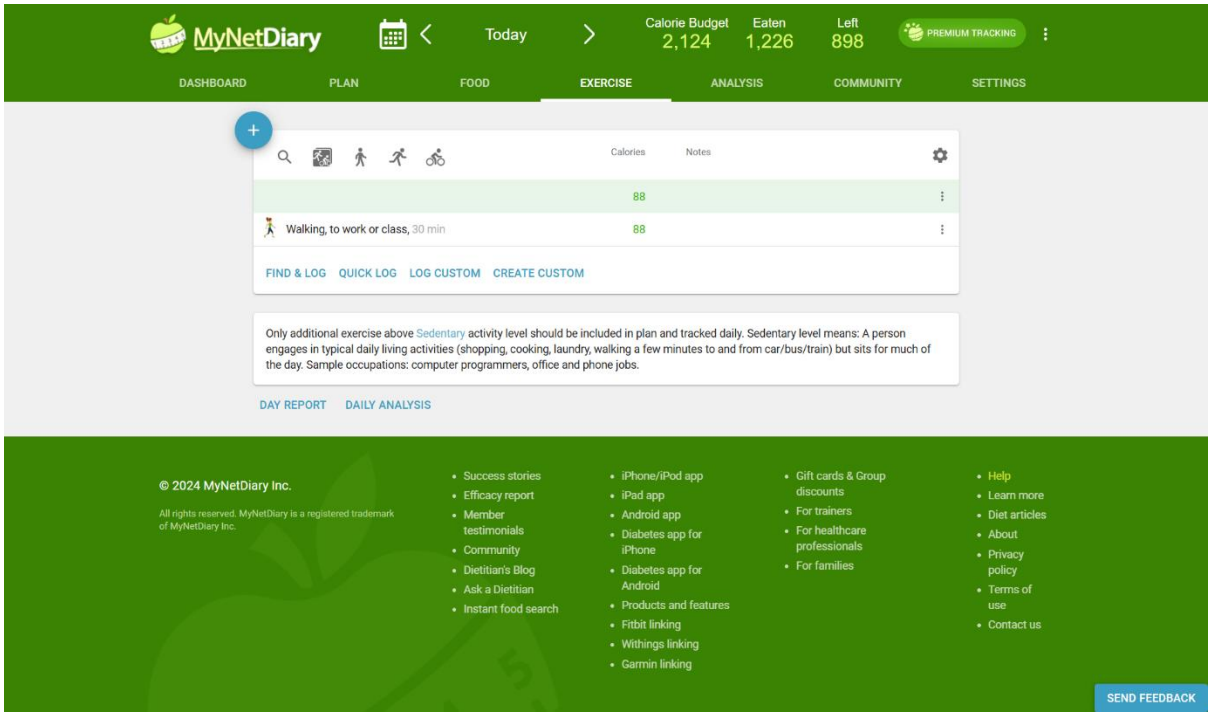


Figure 2.12: Interface of Physical Activity Log for a day.

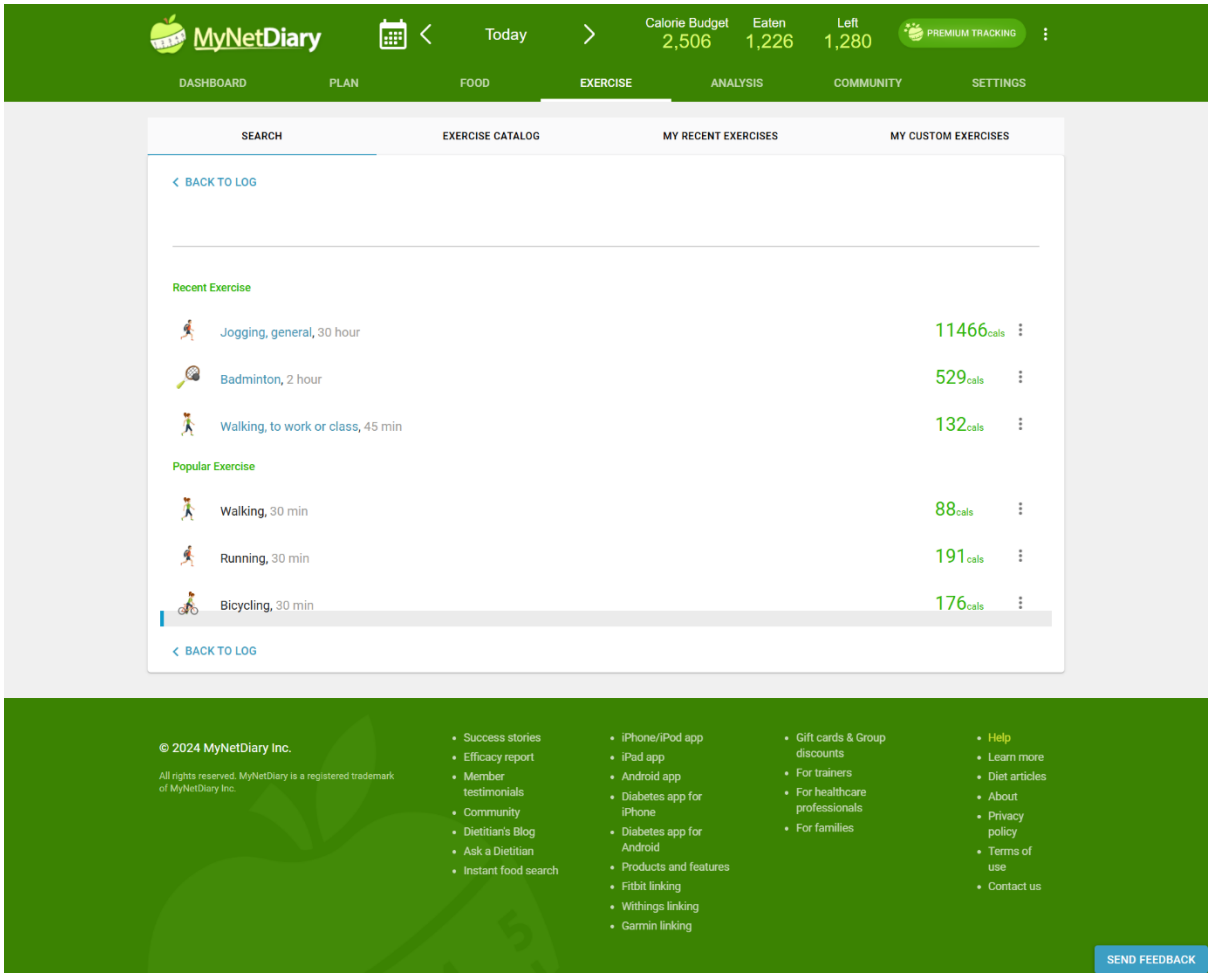


Figure 2.13: Quick Log Options of Recent Entries.

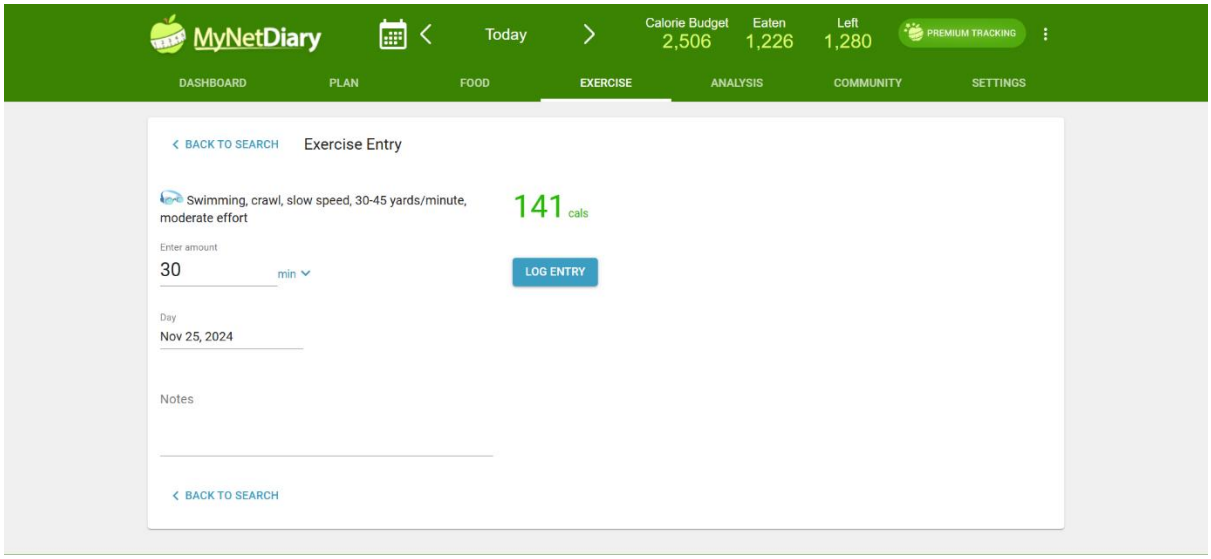


Figure 2.14: Add Exercise from Search into the Log.

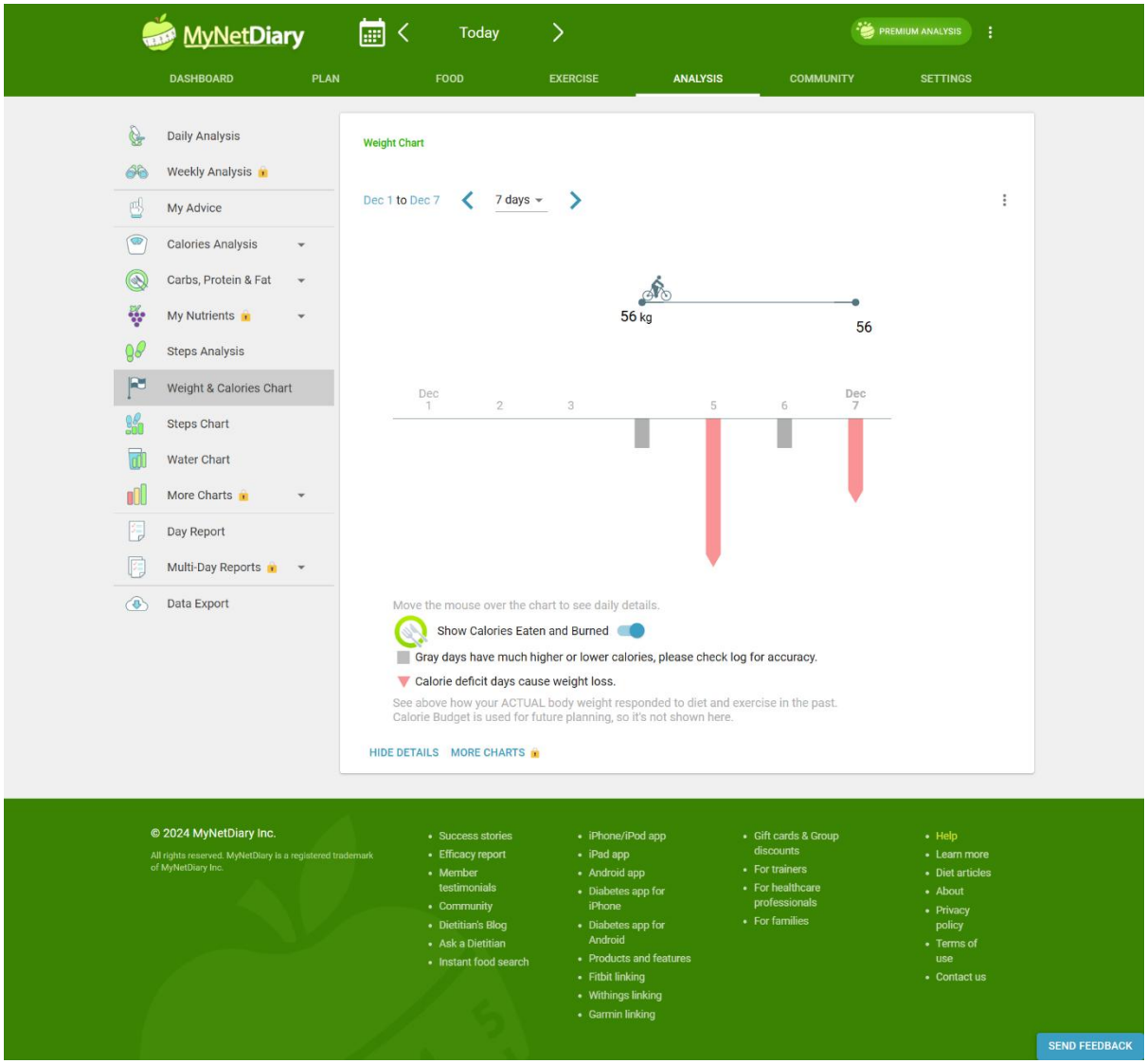


Figure 2.15: Chart Displaying Net Calorie for the Last 7 Days.

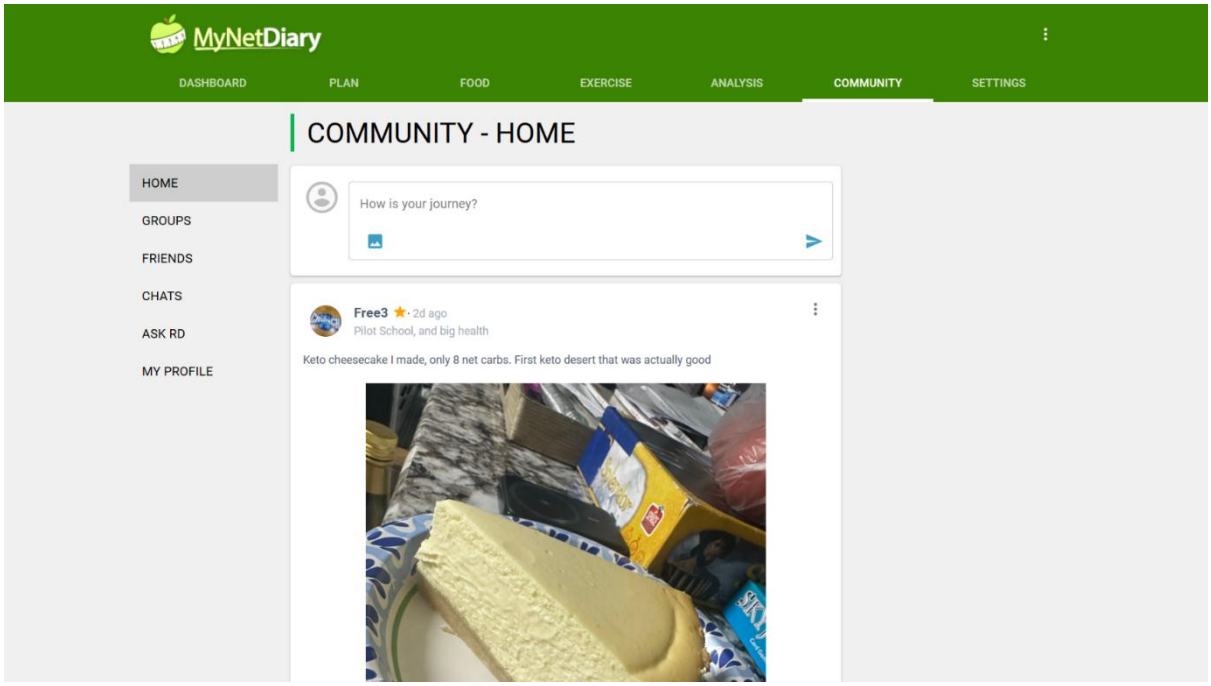


Figure 2.16: Discussion Community on MyNetDiary.

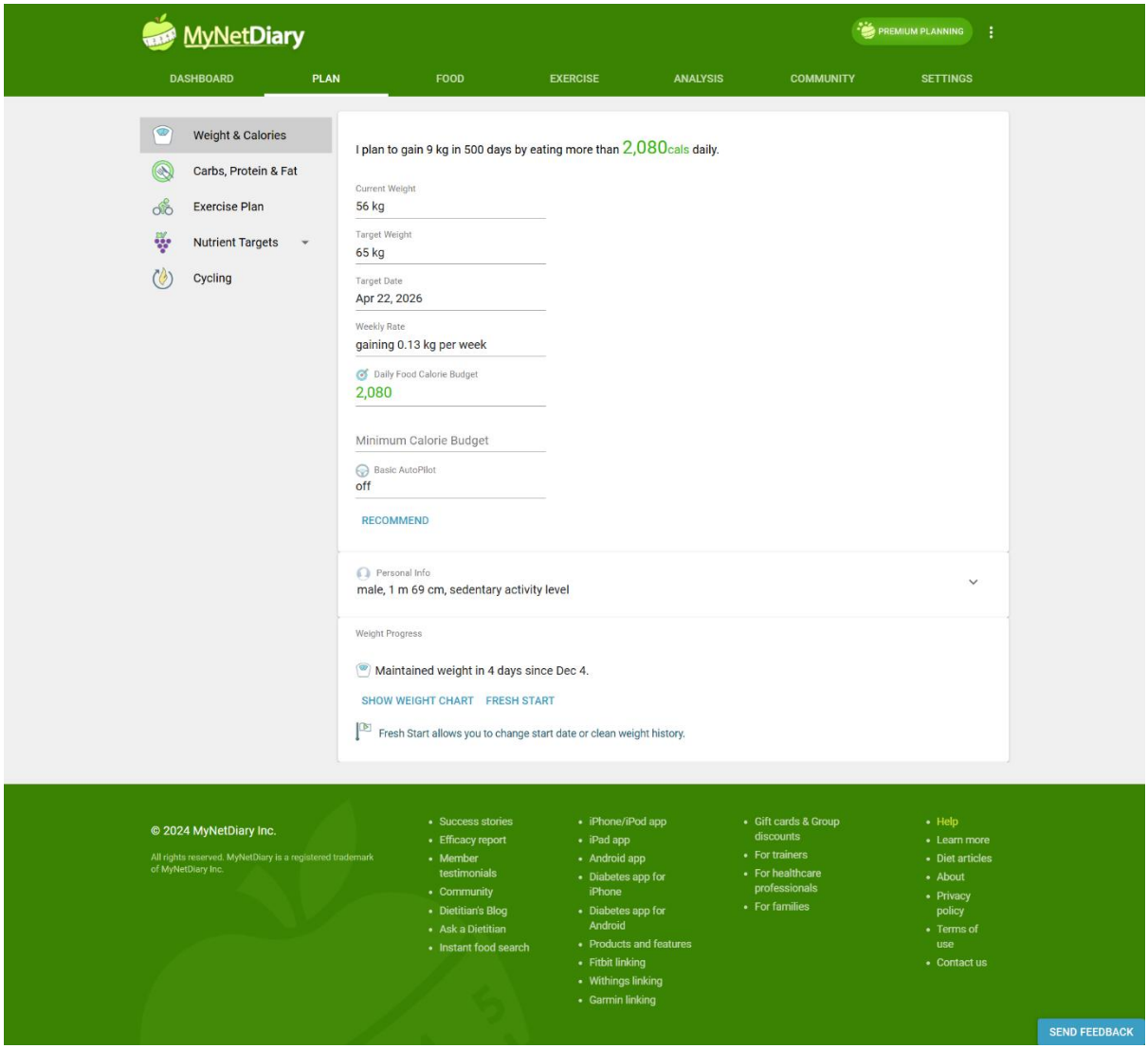


Figure 2.17: User Health Profile.

There are some functions in MyNetDiary will be listed in the table below:

Table 2.2: Functions in MyNetDiary.

Functions	Descriptions
Register	*Figure 2.9* shows the dashboard of MyNetDiary before the user sign in or register as a member. User can register by clicking the 'SIGN UP' button.
Sign in	*Figure 2.9* depicts the dashboard of MyNetDiary before the user sign in or register as a member. User can sign in by clicking the 'SIGN IN' button. Users can sign in by entering their email/account name and

	password, other selections such as Google, Facebook and Apple sign in are provided as well.
Input Personal Metrics and Goal Weight	In *Figure 2.11*, users need to input their height, weight and fitness goal which is their desired progress during the registration process. MyNetDiary utilizes the personal metrics stored for calorie estimation.
Log Physical Activity	*Figure 2.12* illustrates the user interface for logging daily physical activities. After entering the type of exercise and its duration, the interface displays the entered duration along with an estimated calorie consumption. In *Figure 2.13*, users are shown the quick log options for their recent entries when adding a new exercise. *Figure 2.14* depicts how users can search the database for activities to log.
Chart to display calorie trends	In *Figure 2.15* the chart shows the net calories of the user over a period (e.g., the last 7 days), enabling users to adjust their activities based on the observed calorie trends.
Discussion Community	In *Figure 2.16*, MyNetDiary is shown to have a discussion community where users can share their fitness experiences.
User Health Profile	In *Figure 2.17*, MyNetDiary showcases a user health profile, detailing a plan to gain weight. This profile outlines the user's target weight, daily calorie intake, current height and weight, and projected progress over a set period.

Strengths of MyNetDiary

- Integration with Fitness Trackers:

  MyNetDiary integrates smoothly with popular wearable devices and fitness apps, allowing for a cohesive health management experience. This integration helps users synchronize

their physical activity data with their dietary tracking, providing a comprehensive view of their health journey.

- User Health Profile

  One of the key strengths of MyNetDiary is its user health profile, which provides users with a comprehensive overview of their health metrics. This feature allows individuals to input personal data such as weight, height, and fitness goals, enabling the app to generate tailored insights and recommendations. By offering a clear snapshot of the user's health status, MyNetDiary allows users to track their progress effectively and make informed decisions regarding their diet and exercise routines. This personalized approach improves user engagement and motivation, making it easier for individuals to achieve their health and wellness objectives.

Weaknesses of MyNetDiary

- Cost of Premium Features:

  While the free version of MyNetDiary provides sufficient functionality for basic calorie tracking, many advanced features are locked behind a premium subscription that costs around $6.50 per month. This cost may discourage some users from accessing valuable tools like personalized feedback and analysis or advanced nutrient tracking.

MyNetDiary is a digital health application designed to help users track their food intake, exercise, and overall health metrics. MyNetDiary allows users to set personalized weight goals and provides effective tools for monitoring progress, making it a valuable resource for individuals seeking to improve their health. One of the standout strengths of MyNetDiary is its integration with fitness trackers, which enables seamless synchronization of physical activity data with dietary tracking. This feature provides users with a holistic view of their health journey, enhancing their ability to manage both diet and exercise effectively. Also, MyNetDiary includes a user health profile that offers a detailed overview of personal metrics, allowing users

to monitor their progress and adjust their goals accordingly. However, a notable weakness of MyNetDiary is the cost of premium features. The cost may deter some users from accessing valuable features that could enhance their experience and effectiveness in achieving their health goals.

2.2.3 ACE Fitness

The American Council on Exercise (ACE Fitness) is a prominent organization dedicated to improving public health through physical activity and fitness education. Established in 1985, ACE Fitness has become a leading authority in certifying personal trainers and health coaches globally. The organization emphasizes evidence-based practices and continuous research to enhance fitness programming and ensure the effectiveness of its certifications (ACE Fitness, 2023).

Below are some of the ACE Fitness interface:

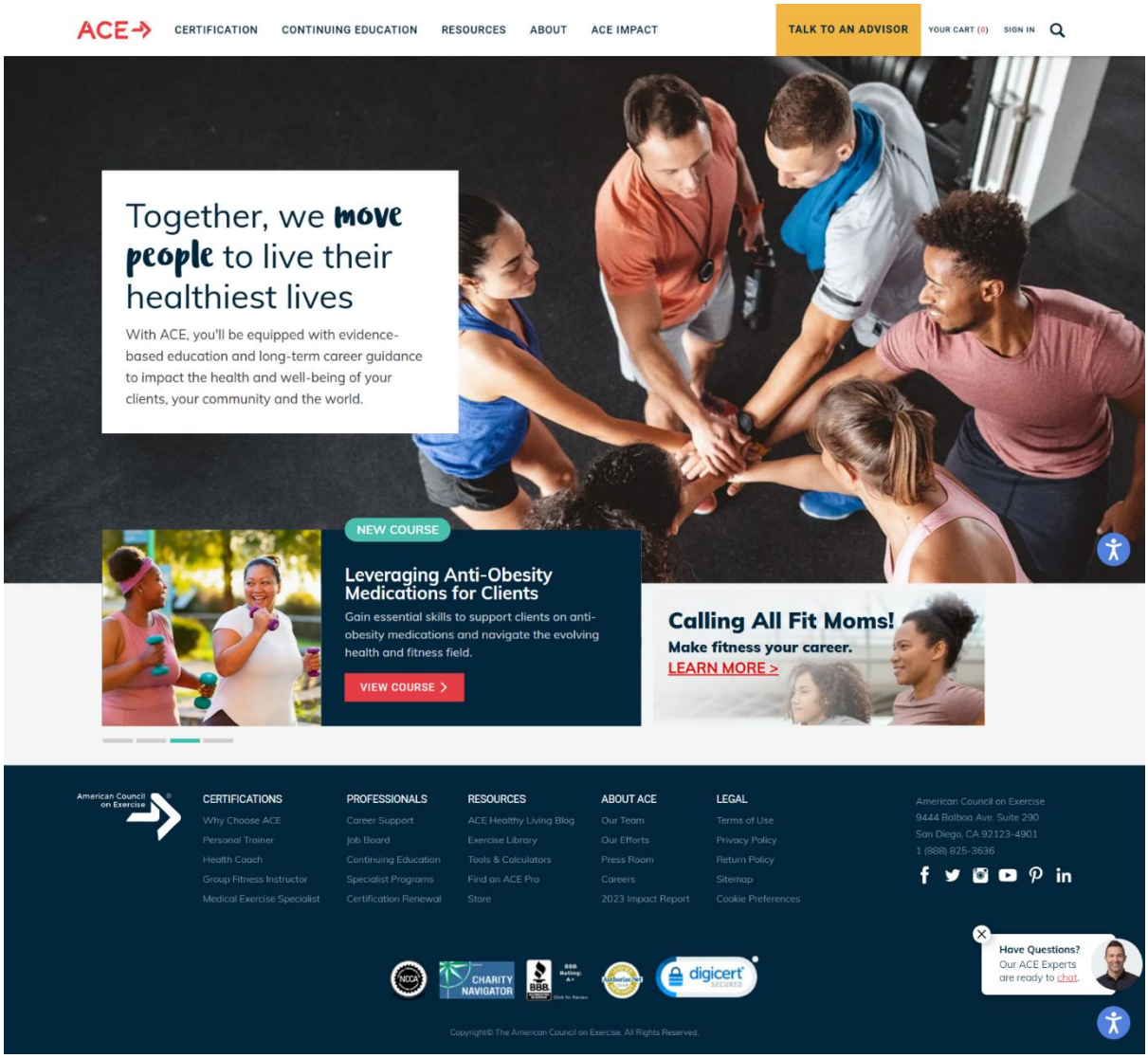


Figure 2.18: Dashboard of MyNetDiary Before Sign In or Register as a Member.

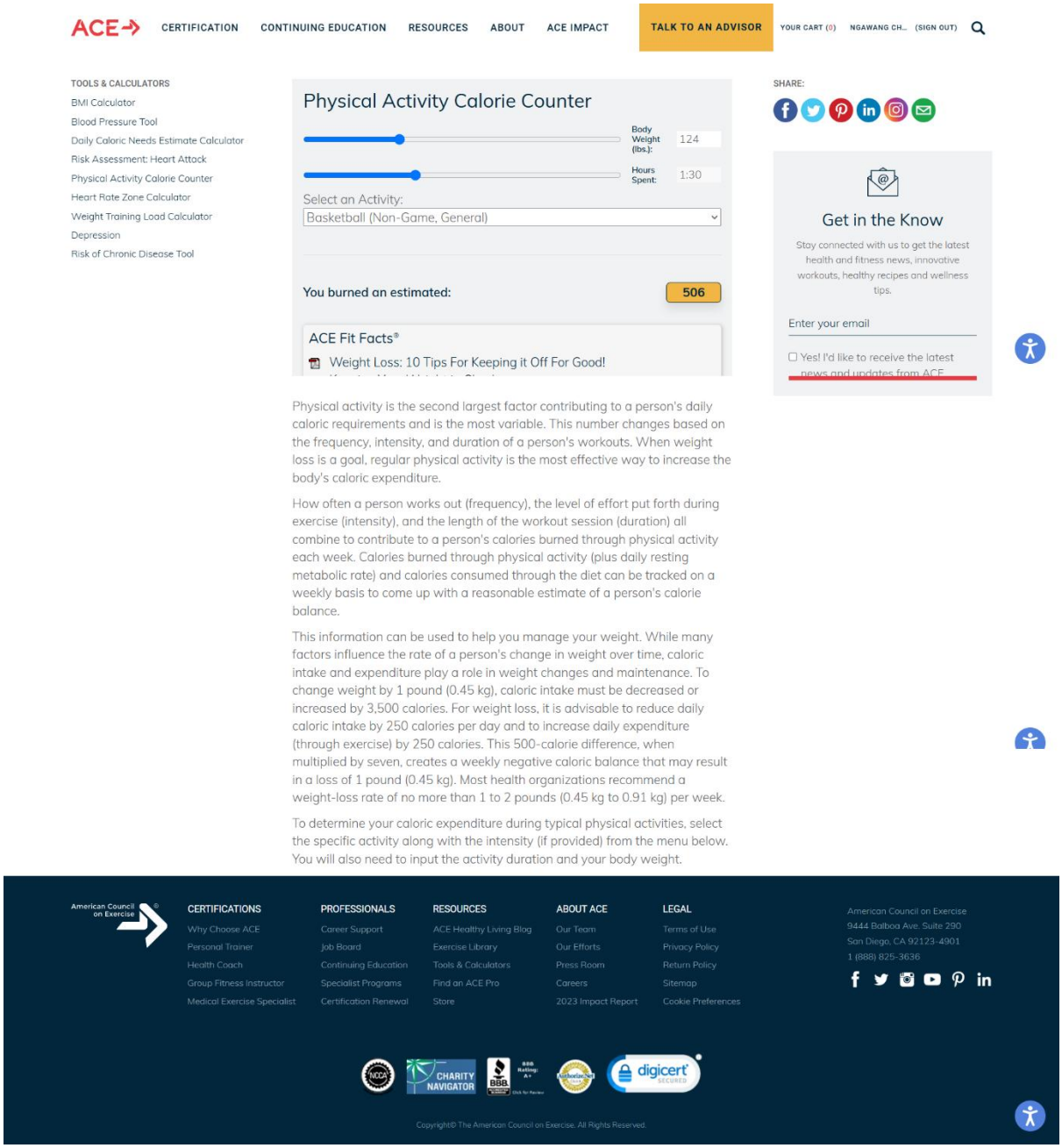


Figure 2.19: Interface of Calorie Estimation Based on Physical Activity.

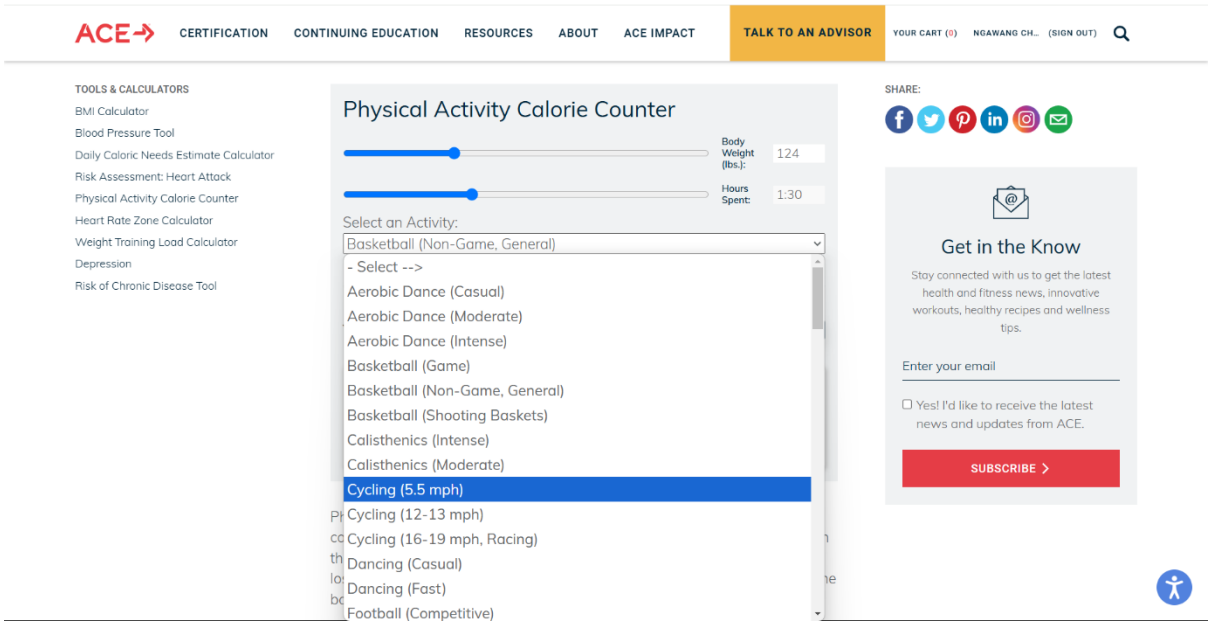


Figure 2.20: Select Physical Activity from Given Options.

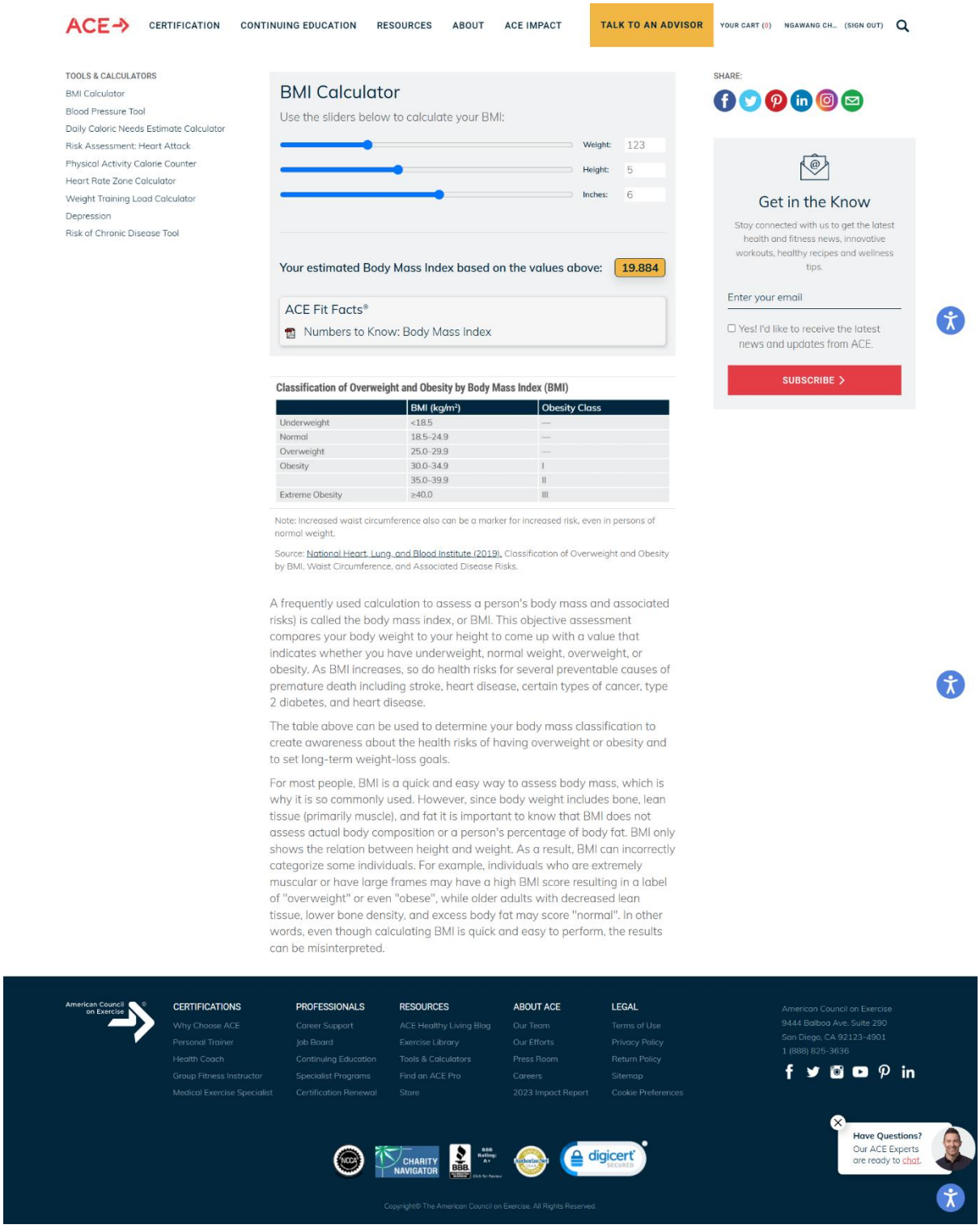


Figure 2.21: Compute body mass index (BMI) and classify weight status categories.

There are some functions in ACE Fitness will be listed in the table below:

Table 2.3: Functions in ACE Fitness.

Functions	Descriptions

Register	*Figure 2.18* shows the dashboard of ACE Fitness before the user sign in or register as a member. When user click into the 'SIGN IN' button, user can choose to sign in or to register as a member of ACE Fitness.
Sign in	*Figure 2.18* depicts the dashboard of ACE Fitness before the user sign in or register as a member. User can sign in by clicking the 'SIGN IN' button. Users can sign in by entering their email and password.
Input Physical Activity	*Figure 2.19* shows a tool that allows users to estimate the number of calories they burn through physical activity. The interface includes various input options for personalization, users need to specify the duration of their activity. As shown in *Figure 2.20*, a dropdown menu provides a selection of different activities. The tool then calculates and displays the estimated calories burned.
Inclusion of body weight for calorie counter	As shown in *Figure 2.19*, users can adjust their body weight (in lbs) using a slider. The calorie counter uses the body weight inputted for calorie estimation.
Compute body mass index (BMI) and classify weight status categories	*Figure 2.21* displays a BMI (Body Mass Index) calculator. The interface allows users to input their weight and height to calculate their BMI. There is a classification of weight status categories by BMI table, which provides ranges for different BMI categories.

Strengths of ACE Fitness

- Informational Text

  The ACE Fitness website provides comprehensive informational resources that emphasize the critical role of physical activity in weight management. One of the key strengths of the

site is its detailed explanation of how regular exercise contributes not only to caloric expenditure but also to overall health and well-being.

- BMI Calculator

  ACE Fitness provides a comprehensive Body Mass Index (BMI) calculator on its website, which serves as a valuable tool for individuals seeking to understand their weight status in relation to their height. The website also outlines specific weight status categories that based on BMI values.

Weaknesses of ACE Fitness

- Limited Interactive Features

  The weaknesses of ACE Fitness primarily lie in its limited interactive features and user engagement. It lacks the robust community support and social interaction features that many users seek in fitness applications. This absence may result in lesser user motivation and accountability, as individuals may need a sense of feeling connected to a community or receive encouragement from peers.

- Lack of feedback mechanisms

  Additionally, ACE Fitness does not provide personalized tracking or feedback mechanisms tailored to individual user goals, which can limit the effectiveness of its resources for users looking for more customized health and fitness plans. Without these interactive elements, users may find it challenging to maintain long-term engagement with the platform.

In summary, ACE Fitness stands out as an authoritative resource in the realm of physical activity and fitness education. Its commitment to improving public health through evidence-based practices and continuous research has established it as a leader in certifying personal trainers and health coaches worldwide. The website provides comprehensive informational resources that highlight the importance of physical activity in weight management, offering detailed explanations on how regular exercise contributes to overall health and well-being.

Another key feature is the BMI calculator, which serves as a valuable tool for individuals aiming to understand their weight status in relation to their height. By outlining specific weight status categories based on BMI values, ACE Fitness empowers users to make informed decisions regarding their health.

2.3 Comparison of the Existing Systems and the Proposed System

After reviewing and analysing existing systems, valuable insights into the potential features and functionalities of the proposed system can be identified. The table below provides a comparison between the features of the current systems and the proposed system.

Table 2.4: Comparison of the Features of Three Existing Systems and the Proposed System.

Feature \ System	MyFitnessPal	MyNetDiary	ACE Fitness	Proposed System
User health profiles	✗	✓	✗	✓
Calorie estimation based on food input	✓	✓	✗	✗
Calorie estimation based on physical activity input	✓	✓	✓	✓
Inclusion of personal metrics for calorie counter	✓	✓	✓	✓
Log physical activity for calorie tracking	✓	✓	✗	✓
Quick log options of recent entries	✓	✓	✗	✓
Graph/Chart to display calorie consumption trends	✓	✓	✗	✓
Body mass index (BMI) calculator and classification of weight status categories	✗	✗	✓	✓
Discussion community	✓	✓	✗	✓
Leaderboard	✗	✗	✗	✓

Free or Paid	Free/Paid	Free/Paid	Free	Free

Based on *Table 2.4*, MyNetDiary has the most features among the systems, while ACE Fitness has the least. MyNetDiary includes features such as a user health profile, calorie estimation based on food input, calorie estimation based on physical activity input, and personal metrics for the calorie counter. In contrast, ACE Fitness lacks features like a discussion community, quick log options, and calorie estimation based on food input. MyFitnessPal and MyNetDiary allow logging physical activity for calorie tracking. Furthermore, MyFitnessPal and MyNetDiary include a discussion community. ACE Fitness features a BMI calculator with weight status classification. Next, the proposed system offers a leaderboard feature. This feature is included in the proposed system, which can encourage user interaction and motivation. To improve the user experience and simplify health management, essential features listed in *Table 2.4* will be implemented in the proposed system except for the calorie estimation based on food input.

2.4 Review on Tools Used to Develop the Proposed System

2.4.1 Machine Learning

Machine learning is a branch of artificial intelligence that enables systems to learn from data and make predictions without being explicitly programmed. It is widely applied in various domains, including fitness, due to its ability to recognize and learn the data patterns. In the context of this project, machine learning is used to estimate the number of calories burned based on user input such as activity type, duration, and user weight. This allows for personalised and accurate calorie predictions, making the system more responsive and scalable. According to a study by Cierco Jimenez et al. (2022) mapped the capabilities of machine learning computational tools across different phases of data processing and decision-making. The review emphasized that tools offering end-to-end support from data preparation to model

interpretation are more suitable for practical applications. In addition, tools with user-friendly interfaces, such as graphical workflows, can significantly reduce the learning curve for non-technical users.

2.4.2 Backend and Frontend Framework

Django is an ideal framework for the proposed system due to its MVT architecture, which ensures a clean and scalable structure for managing user data and activity logs. Its built-in ORM simplifies database operations by eliminating the need for raw SQL, allowing efficient storage and retrieval of physical activity and calorie data (Index.dev, 2024). As a Python-based framework, Django integrates smoothly with machine learning libraries such as TensorFlow, PyTorch, and Scikit-learn, making it well-suited for predicting calorie consumption using trained models (Emanuilov, 2024).

Django also supports high traffic scalability and provides useful tools like its command-line interface for automating tasks such as migrations and project setup. For the frontend, HTML, CSS, JavaScript, and Django Template Language (DTL) were used to create dynamic and responsive interfaces. These languages combines layout design with programming to support user interactions and functionality. HTML structures the page content, CSS styles the layout and appearance, while JavaScript adds interactivity such as animations and real-time updates. These technologies work together to enhance the user experience for features such as logging activities, viewing calorie reports and so on.

2.4.3 Visual Studio Code

Developed by Microsoft, Visual Studio Code (VS Code) is a lightweight and powerful source code editor that has become a top choice among software developers. VS Code comes with integrated support for Node.js, TypeScript, and JavaScript, and offers a rich extension ecosystem for various languages such as Python, C++, Java, C#, PHP, and Go (Turing, 2024). VS Code provides comprehensive debugging capabilities that allow developers to set breakpoints, inspect variables, and step through code execution in real time. These features enable developers to closely analyse their code's behaviour, understand execution flow, and isolate the root causes of bugs and errors effectively (Microsoft, 2024)

2.5 Summary

This chapter reviewed and analysed existing systems relevant to the proposed calorie tracking system: MyFitnessPal, MyNetDiary, and ACE Fitness. Each system's features, strengths, and weaknesses were discussed. After reviewing and analysing existing systems, valuable insights into the potential features and functionalities are identified. Hence, the proposed system combines features like user health profiles, calorie estimation based on physical activity, BMI calculation, and a leaderboard for enhanced engagement. The tools and frameworks for developing the system, including Machine Learning, Django, DTL, HTML, CSS, JavaScript and Visual Studio Code were evaluated for their suitability in ensuring scalability, efficiency, and user experience.

CHAPTER 3:REQUIREMENT ANALYSIS AND DESIGN

3.1 Introduction

This chapter concentrates on requirement analysis and the planning phase, which are the first two stages of the Agile Model. The requirement analysis phase examines data from questionnaires and system requirements to understand user needs and ensure the system meets their expectations. In the design phase, the logical design covers the overall system architecture, use case diagram, use case specifications, activity diagrams, entity-relationship diagram and data dictionary which illustrate the structure and interactions of system components. The physical design, represented through wireframes, offers a visual blueprint of the user interface and its functionality.

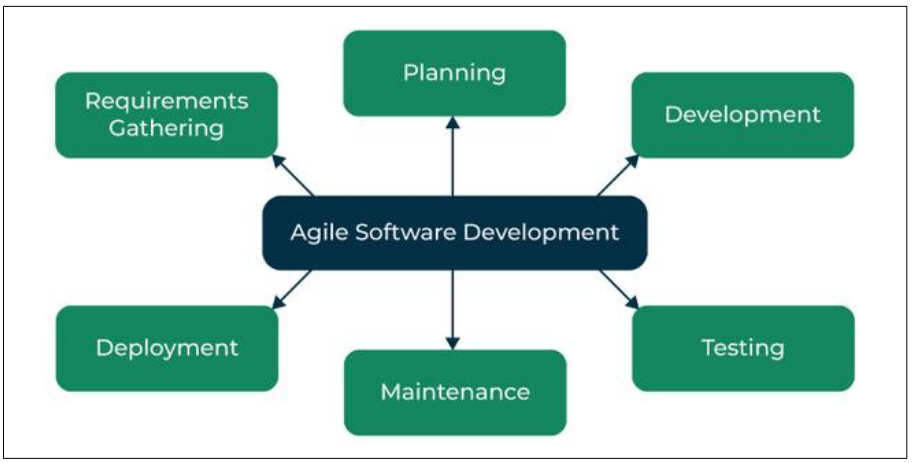


Figure 3.1: Agile Model.

3.2 Requirement Analysis Phase

This section will thoroughly discuss the data analysis performed through questionnaires and system requirements analysis, aiming to clearly and precisely outline the necessary requirements.

3.2.1 Data Analysis

A survey was conducted via Google Forms (attached in Appendix A) to gather demographic information, explore the motivations for developing the calorie tracking system, and identify

user requirements. The target audience for the survey included UNIMAS students and individuals interested in tracking their physical activities. Participants were encouraged to share their preferences and expectations to ensure the system aligns with user needs effectively. The survey was structured into three main sections: Section I Demographic, Section II Current Fitness Habits & Tools, and Section III Desired Features & Feedback. These sections were designed to collect comprehensive insights into user demographics, existing practices, and feature preferences. A total of 30 responses were collected, providing valuable data for analysis. The findings were subsequently analysed and presented through various visual representations to support the development and refinement of the system.

3.2.1.1 Analysis of Demographic

This survey incorporates two demographic-related questions in Section I. The charts below present an analysis of the demographic information from 30 respondents:

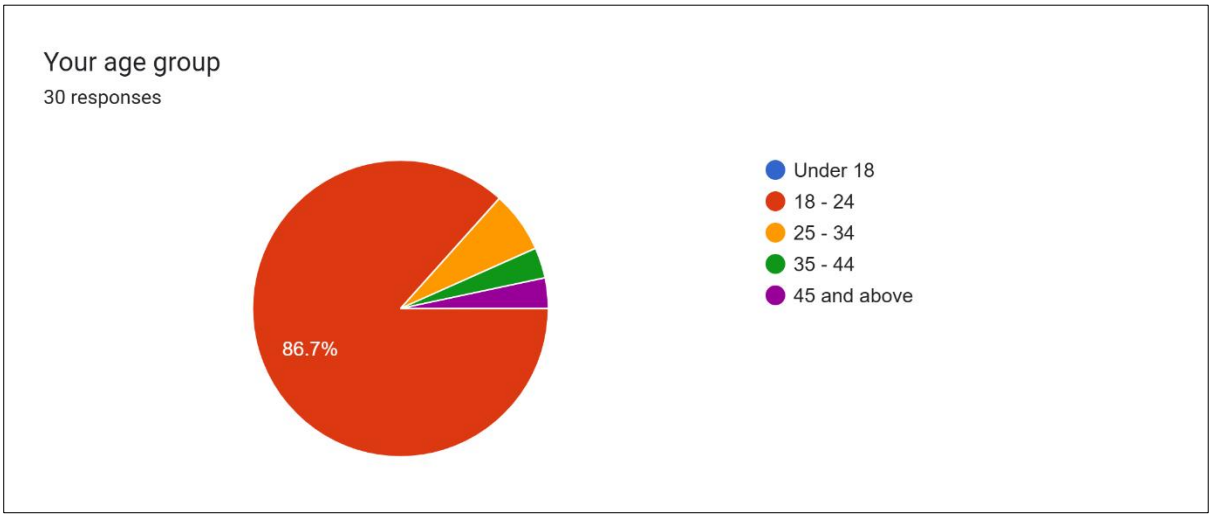


Figure 3.2: Age Analysis of Respondents.

As depicted in *Figure 3.2*, a significant majority of the respondents, 86.7%, are aged between 18 and 24, typically representing the demographic of university students. This is followed by 6.7% aged between 25 and 34, and 3.3% each for the age groups of 35 to 44 and 45 and above.

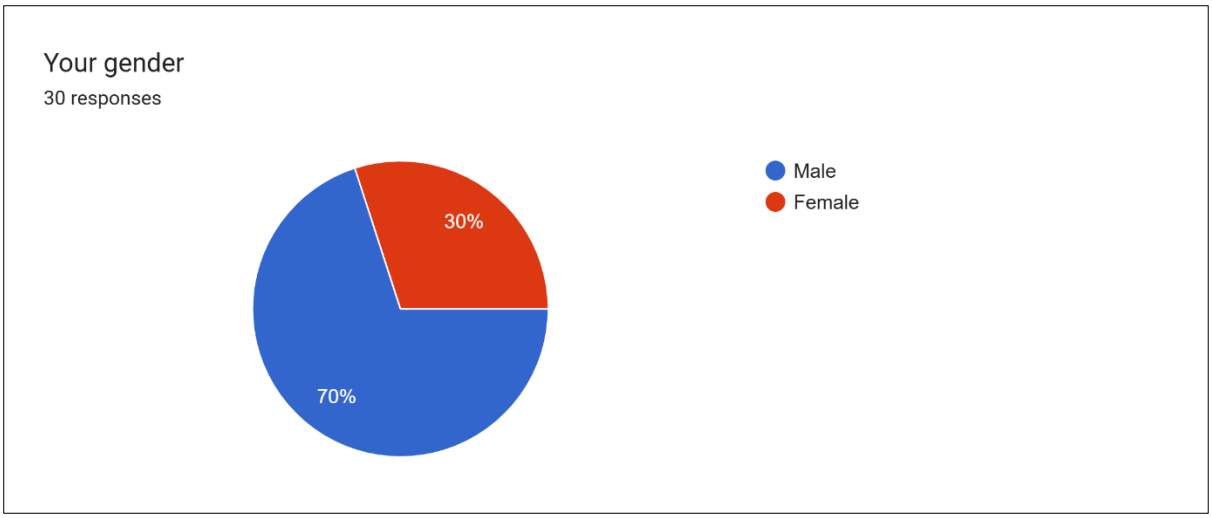


Figure 3.3: Gender Analysis of Respondents.

As depicted in *Figure 3.3*, the majority of respondents (70%) are male, while the remaining 30% are female. This distribution indicates a higher participation rate among males in the survey.

3.2.1.2 Analysis of Current Fitness Habit & Tools of Respondents

This section (Section II) focuses on the respondents' fitness routines and the tools they utilize for tracking and managing their physical activities. The analysis provides insights into current practices and preferences among the participants.

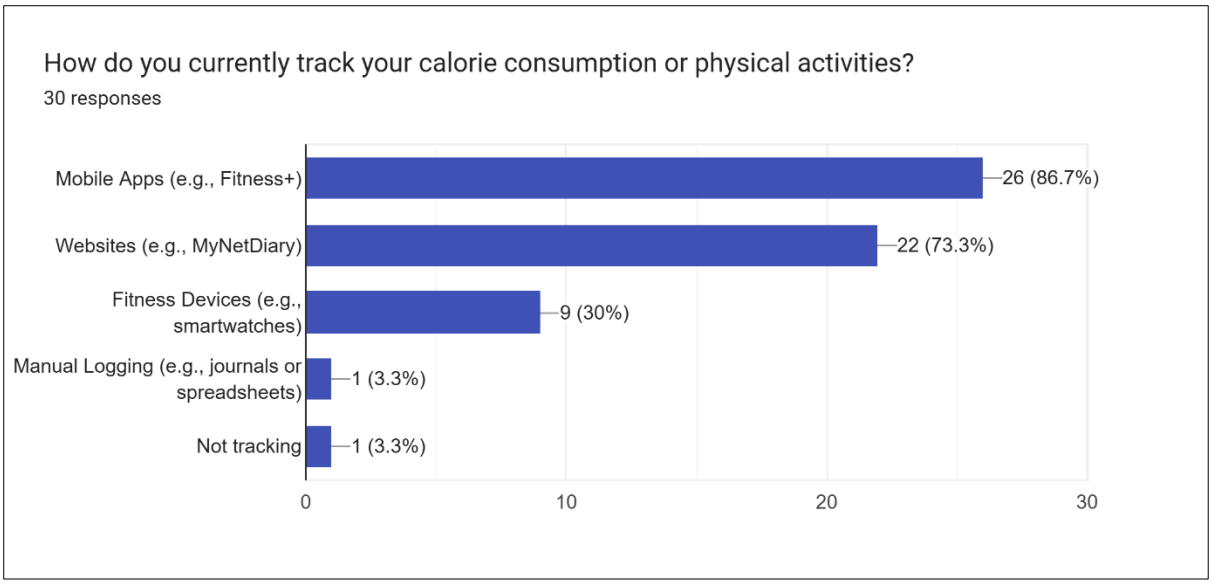


Figure 3.4: How do you currently track your calorie consumption or physical activities.

As depicted in *Figure 3.4*, the majority of respondents (86.7%) currently track their calorie consumption or physical activities using mobile apps. Websites are the second most popular option, used by 73.3% of respondents. Fitness devices, such as smartwatches, are utilized by 30% of respondents. Meanwhile, manual logging methods, such as journals or spreadsheets, and the other option (not tracking) each represent 3.3% of the responses. This data indicates a strong preference for digital tools in managing calorie tracking and physical activity monitoring.

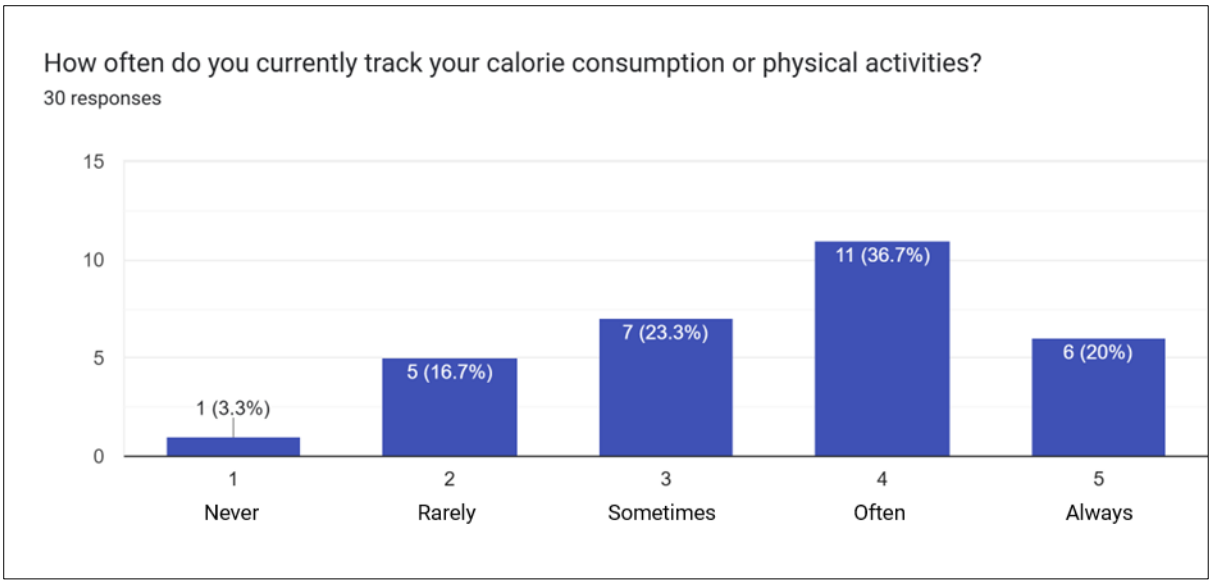


Figure 3.5: How often do you currently track your calorie consumption or physical activities.

As shown in *Figure 3.5*, respondents' frequency of tracking calorie consumption or physical activities varies. The largest group, 36.7%, indicated they track their activities often. This is followed by 23.3% who sometimes track their activities, and 20% who always track their activities. Meanwhile, 16.7% of respondents rarely track their activities, and a small minority of 3.3% do not track their activities at all. This suggests that most respondents have a regular or consistent approach to tracking their calorie consumption or physical activities.

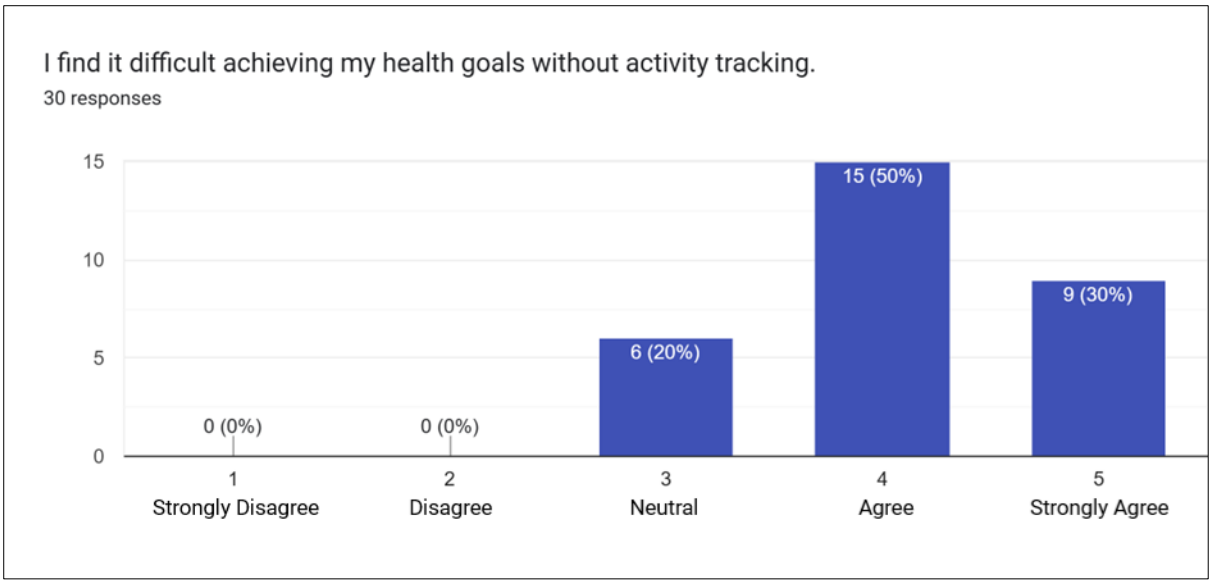


Figure 3.6: I find it difficult achieving my health goals without activity tracking.

As shown in *Figure 3.6*, the majority of respondents (50%) agree that they find it difficult to achieve their health goals without activity tracking, while 30% strongly agree with this statement. A smaller proportion, 20%, remain neutral, and no respondents indicated

disagreement or strong disagreement. This data highlights the importance of activity tracking as a perceived tool for achieving health-related goals among the respondents.

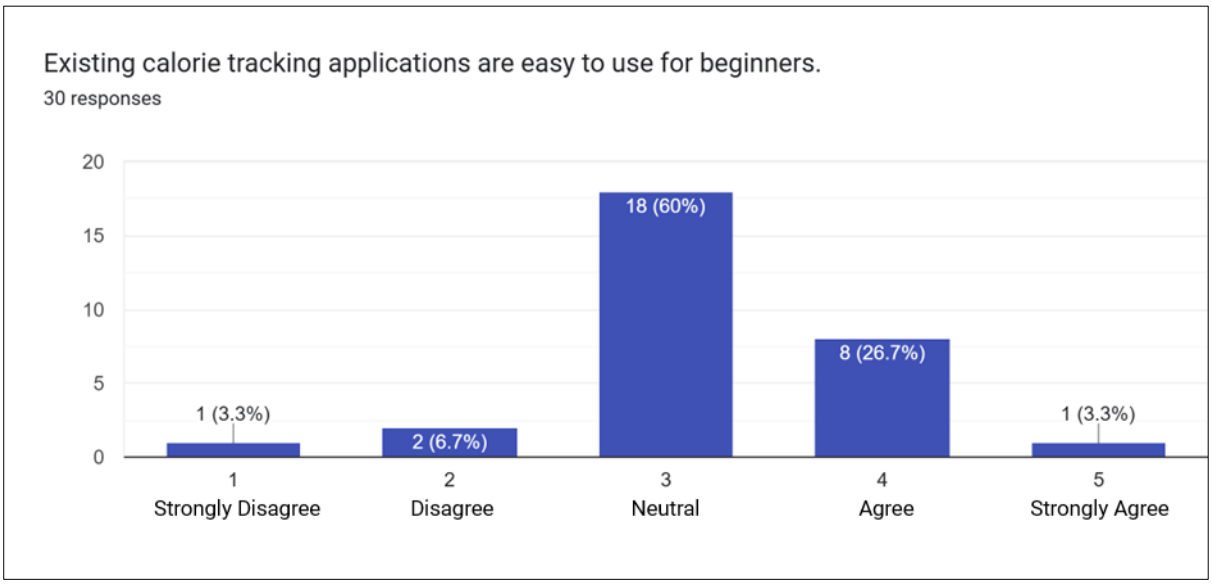


Figure 3.7: Existing calorie tracking applications are easy to use for beginners.

As shown in *Figure 3.7*, the majority of respondents (60%) remain neutral regarding the ease of use of existing calorie tracking applications for beginners. Meanwhile, 26.7% agree that these applications are easy to use, and 3.3% strongly agree. On the other hand, 6.7% disagree, and 3.3% strongly disagree. This suggests that while some respondents find the existing calorie-tracking applications are good to use for beginners, a significant portion either remains unsure or perceives them as less accessible for beginners.

3.2.1.2 Analysis of Desired Features & Feedback of Respondents

This section (Section III) explores the respondents' preferences for features and gathers feedback for the calorie tracking system. The analysis highlights key functionalities and user expectations to enhance the system's usability and engagement.

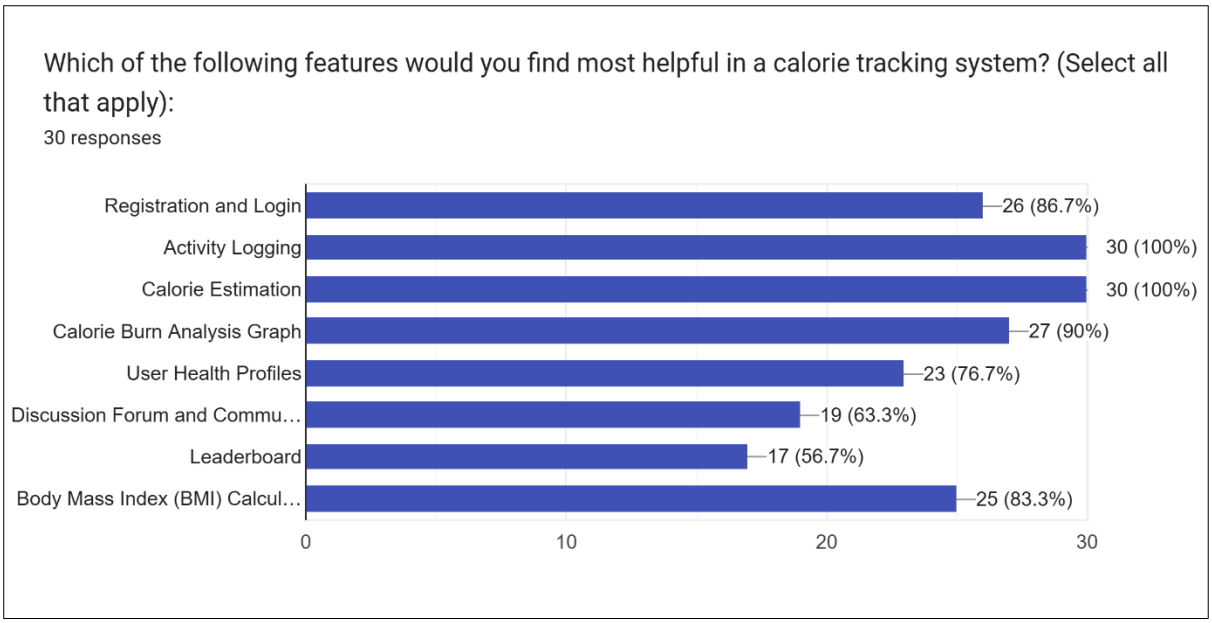


Figure 3.8: Which of the following features users find most helpful in calorie tracking system.

As shown in *Figure 3.8*, the most preferred features for a calorie tracking system are Activity Logging and Calorie Estimation, with both selected by all respondents (100%). The Calorie Burn Analysis Graph was the next most desired feature, chosen by 90% of respondents, followed by Body Mass Index (BMI) Calculator, selected by 83.3%. Registration and Login and User Health Profiles were also popular, chosen by 86.7% and 76.7% of respondents, respectively. Meanwhile, Discussion Forum and Community Features (63.3%) and Leaderboard (56.7%) were less commonly chosen but still significant. These preferences highlight the features that need to be included in a calorie tracking system.

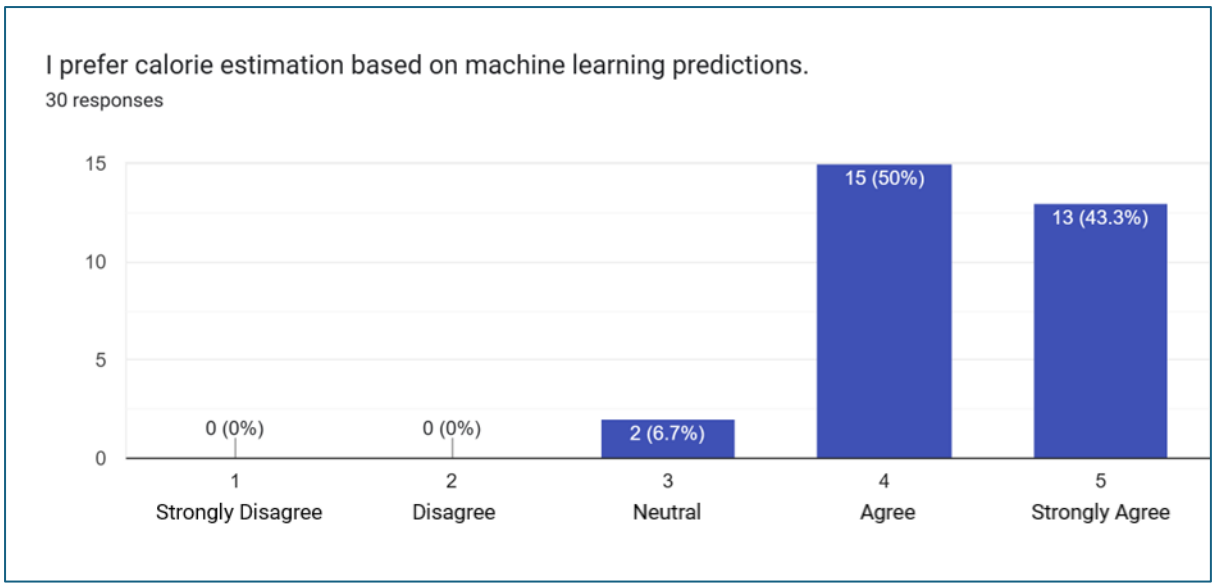


Figure 3.9: I prefer calorie estimation based on machine learning predictions.

As shown in *Figure 3.9*, a significant majority of respondents prefer calorie estimation based on machine learning predictions, with 50% agreeing and 43.3% strongly agreeing. Only 6.7% of respondents remain neutral, while none indicated disagreement or strong disagreement. This indicates a strong preference toward applying machine learning technologies for calorie estimation among the respondents.

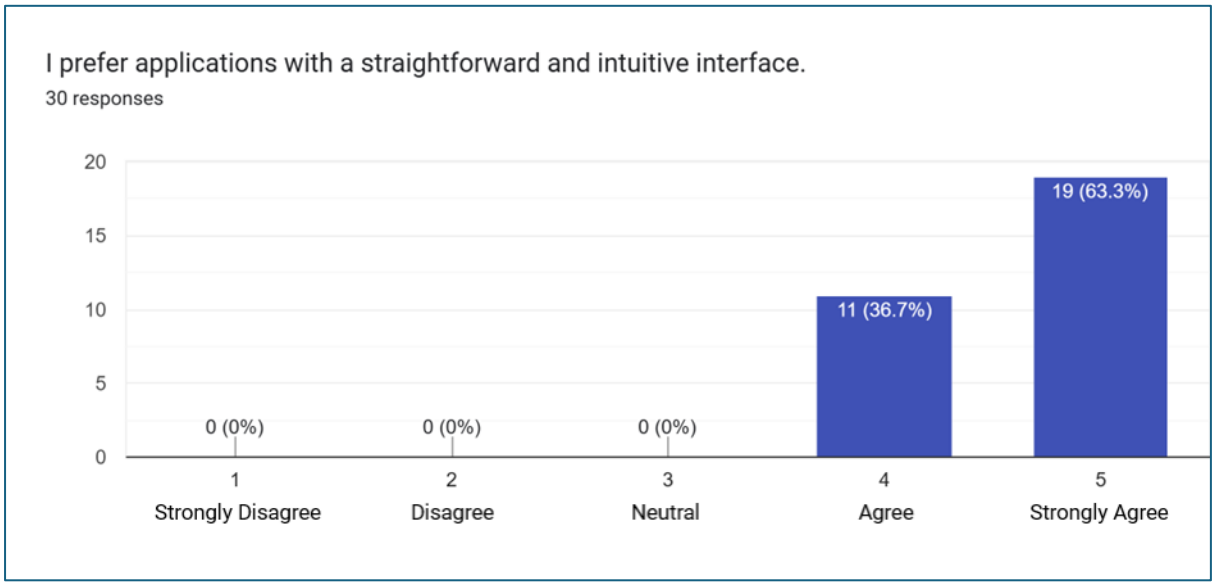


Figure 3.10: I prefer applications with a straightforward and intuitive interface.

As shown in *Figure 3.10*, a significant majority of respondents (63.3%) strongly agree and 36.7% agree that they prefer applications with a straightforward and intuitive interface. None of the respondents expressed neutrality, disagreement, or strong disagreement. This

highlights the importance of straightforward and intuitive interfaces in application design to meet user expectations effectively.

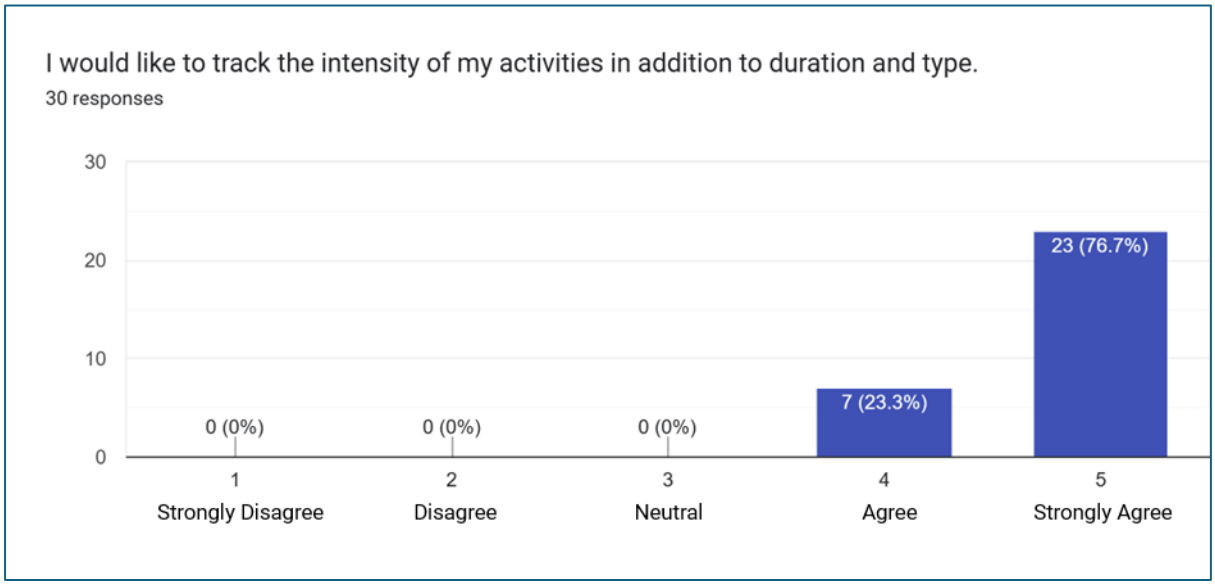


Figure 3.11: I would like to track the intensity of my activities.

As shown in *Figure 3.11*, the majority of respondents (76.7%) strongly agree that they would like to track the intensity of their activities in addition to duration and type. Another 23.3% agree with this statement. No respondents expressed neutrality, disagreement, or strong disagreement. This indicates a strong interest to include intensity when tracking physical activities.

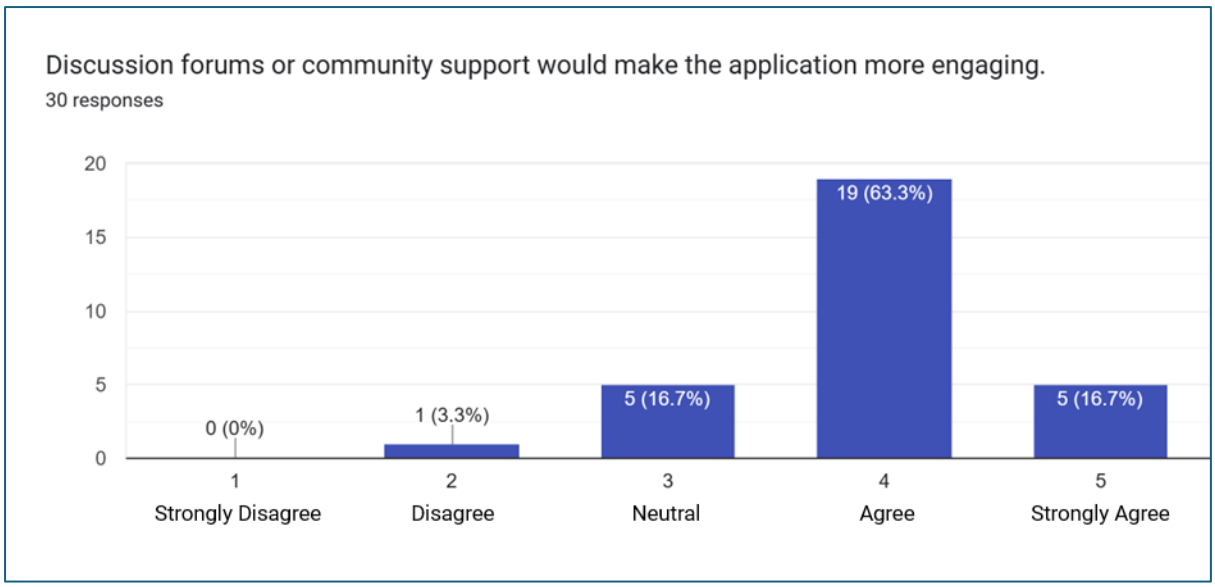


Figure 3.12: Discussion forums or community support would make the application more engaging.

As shown in *Figure 3.12*, the majority of respondents (63.3%) agree that discussion forums or community support would make the application more engaging, while 16.7% strongly agree. A smaller proportion, 16.7%, remain neutral, and 3.3% disagree. No respondents strongly disagreed. This indicates that integrating community support features could enhance user engagement within the calorie tracking application.

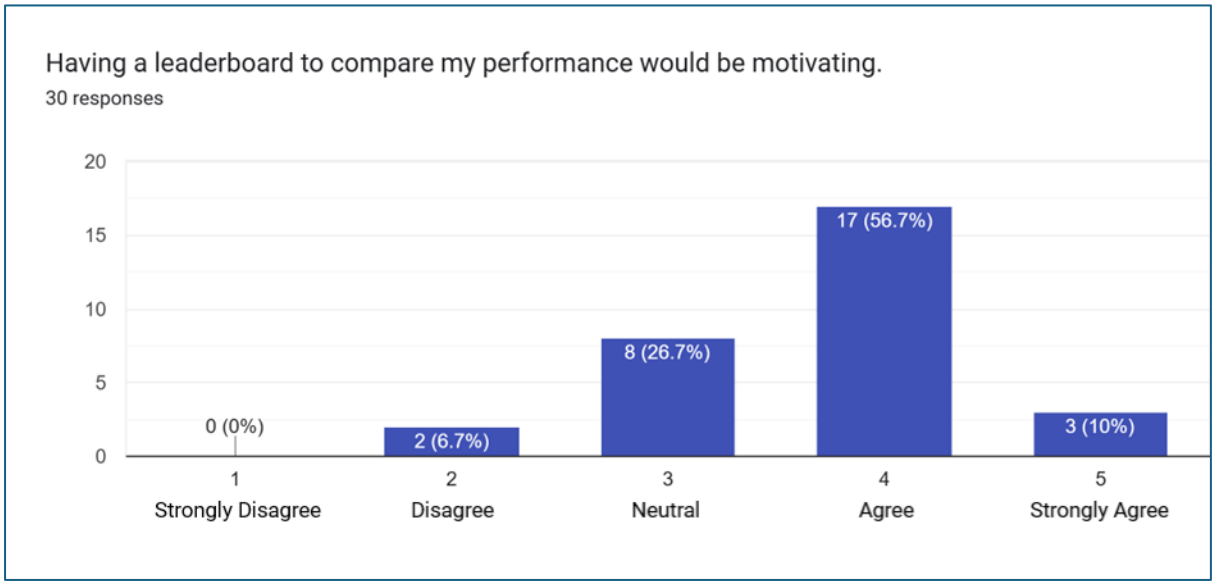


Figure 3.13: Having a leaderboard to compare my performance would be motivating.

As shown in *Figure 3.13*, a majority of respondents (56.7%) agree that having a leaderboard to compare their performance would be motivating, while 10% strongly agree. A

significant portion, 26.7%, remains neutral, and 6.7% disagree. No respondents strongly disagreed. This suggests that leaderboards could serve as a motivational feature for a substantial number of users.

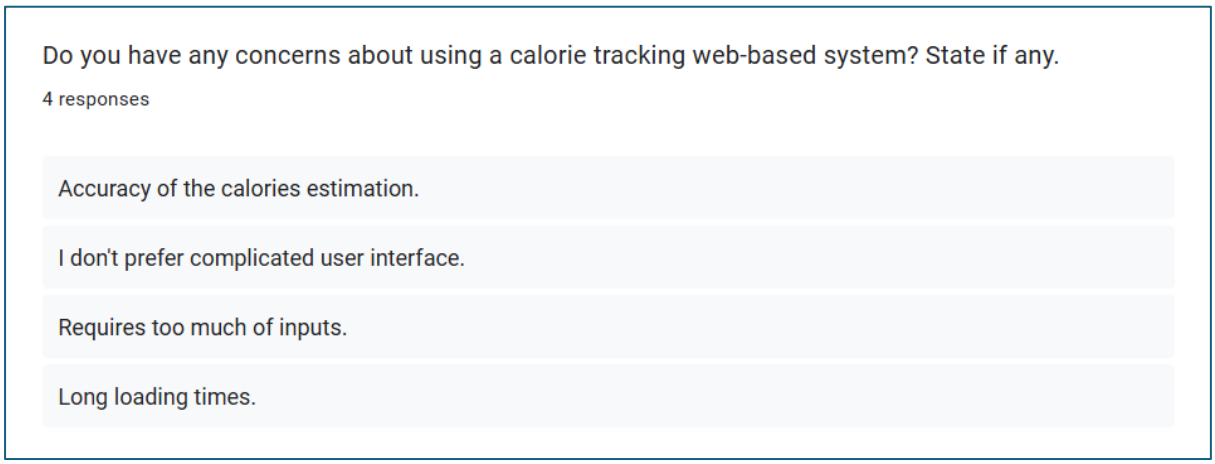


Figure 3.14: Do you have any concerns about using a calorie tracking web-based system.

Figure 3.14 shows an open and optional question allowed users to express their concerns about using a calorie tracking web-based system. The responses indicate the following concerns:

1. Accuracy of calorie estimation.
2. Preference for a straightforward user interface, avoiding complexity.
3. The system requires too much input from users.
4. Long loading times of the system.

3.2.2 Functional Requirements

The functional requirements of the proposed system including the system capabilities and features are as follows:

1. User Authentication and Management
    - The system allows users to create an account.
    - The system allows users to log into their registered accounts.
    - The system maintains a session for logged-in users.
    - The system allows users to view and update their personal details.
2. Activity Logging
    - The system allows users to log activities into the system for calorie tracking, with intensity incorporated as part of the activity type (e.g., "Badminton, competitive").

- The system computes estimated calories burned based on logged activities with their durations and distances (optional) using the machine learning model.

3. Calorie Report
    - The system generates and displays graphs showing calorie consumption over different periods (7/30/60 days).
    - The system generates a downloadable calorie report in pdf format displaying the list of all the logged activity and the graphs showing calorie consumption over different periods (7/30/60 days).
4. Discussion Forum
    - The system allows users to view all discussion posts and their replies.
    - The system allows users to create a discussion post with an optional image.
    - The system allows users to reply to an existing discussion post.
    - The system allows users to edit or delete their own posts and replies.
5. Leaderboard
    - The system displays a leaderboard ranking users based on total calories burned over a specific period.
    - The system updates leaderboard data dynamically as users log activities.
    - The system provides users the option to hide their ranking on the leaderboard, with visibility set to active by default.
6. BMI Calculator
    - The system calculates a user's BMI based on their current height and weight.
    - The system determines the weight status category based on the BMI table.
    - The system has a calculator for users to input values weight and height and calculates the BMI as well as showing the weight status category based on the value of BMI.

3.3 System Planning Phase

In the Agile model, the next step is the system design phase. This section discussed both the physical and logical design of the proposed system. The overall system architecture design, Use Case Diagram, Use Case Specifications, Activity Diagram, Entity Relationship Diagram, and Data Dictionary are part of the logical design. The wireframes of the proposed system are part of the physical design.

3.3.1 Logical Design

3.3.1.1 Overall System Architecture Design

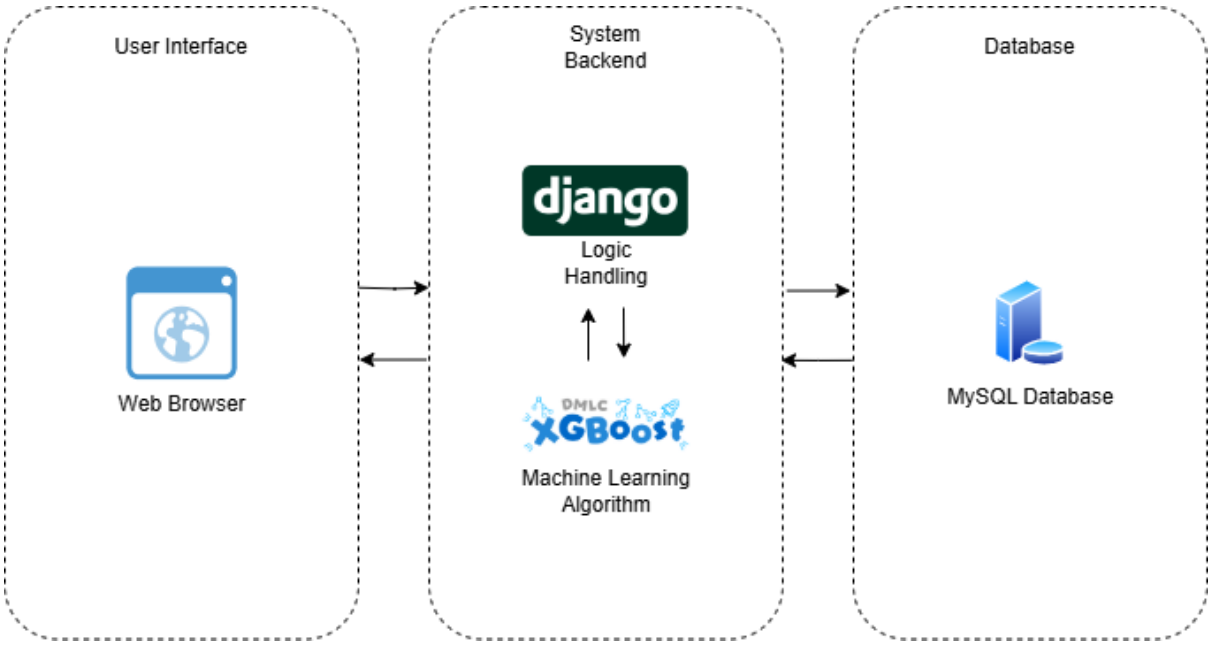


Figure 3.15: Overall System Architecture of the Proposed System.

The system architecture provides an overview of the components within the proposed system, illustrating their structure, relationships, and interactions. As shown in *Figure 3.15*, the overall system architecture is divided into three primary layers: Presentation Layer, Application Layer, and Data Layer.

1. Presentation Layer (User Interface)

   The layer serves as the communication interface between users and the system. This layer enables users to interact with the system through a web browser. The interface component of this tier is responsible for rendering interfaces using HTML, CSS, JavaScript and Django Template Language (DTL).

2. Application Layer (Backend Logic)

   The core functionalities of the system are handled in this layer. Django serves as the framework for backend handling, managing user requests, and processing data. Integrated with the machine learning algorithm, trained with XGBoost, which performs the estimations of calories burned based on user input activities.

3. Data Layer (Database)

   The MySQL database is responsible for storing and managing system data. It ensures efficient and secure data retrieval, supporting both the backend logic and the user interface.

3.3.1.2 Use Case Diagrams and Use Case Specifications

Use case diagram illustrates a set of use cases for a system, each use case demonstrates the different ways that a user might interact with a system.

Figure 3.16: Use Case Diagram of Proposed System.

Figure 3.16 illustrates the use case diagram of the proposed system. Each use case demonstrates the different ways that a user might interact with a system. Use case specification for each of the use cases are as follows:

Table 3.1: Use Case Specification: Login/Register.

Use Case	Login/Register
Actor	User
Summary Description	Allows users to register or log in to the system.
Pre-Condition	User is on the login/register page.
Basic Path	1. User provides username and password. 2. System validates credentials.

	3. If valid, the user is logged in. 4. When registering, the user provides additional details (e.g., email, personal metrics). 5. System calculates BMI based on height and weight, stores user information. 6. After successful registration, user is redirected to login page.
Post-Condition	User is successfully logged in or registered.

Table 3.2: Use Case Specification: Log Activity.

Use Case	Log Activity
Actor	User
Summary Description	Allows users to log physical activities for calorie tracking.
Pre-Condition	User is logged in.
Basic Path	1. User inputs activity details (type, duration). 2. The system validates the input. 3. The system calculates calories burned using the machine learning model. 4. System logs the activity in the database.
Post-Condition	Activity is logged, and calories burned are updated.

Table 3.3: Use Case Specification: View Calorie Consumption Trend.

Use Case	View Calorie Consumption Trend
Actor	User
Summary Description	Displays a graph of calorie consumption over time.
Pre-Condition	User has logged activities.

Basic Path	1. User navigates to the calorie trend page. 2. System retrieves calorie data for the selected time period (week/month). 3. System generates and displays the graph.
Post-Condition	Calorie trend graph is displayed.

Table 3.4: Use Case Specification: View Leaderboard.

Use Case	View Leaderboard
Actor	User
Summary Description	Displays a leaderboard ranking users by total calories burned.
Pre-Condition	Users have logged activities.
Basic Path	1. User navigates to the leaderboard page. 2. System retrieves leaderboard data (top users). 3. System displays the leaderboard.
Post-Condition	Leaderboard is displayed.

Table 3.5: Use Case Specification: View Discussion Forum.

Use Case	View Discussion Forum
Actor	User
Summary Description	Allows users to view discussion posts and replies.
Pre-Condition	User is logged in.
Basic Path	1. User navigates to the discussion forum page. 2. System retrieves discussion posts and replies. 3. System displays the posts and replies.
Post-Condition	Posts are displayed.
Extension: Create Discussion Post	1. User creates a new discussion post.

	2. System validates the input and saves the post. Post extends to: • Edit Post: User modifies the post content. • Delete Post: User removes the post.
Extension: Reply Discussion	1. User replies to an existing discussion post. 2. System validates the input and saves the reply. Reply extends to: • Edit Reply: User modifies the reply content. • Delete Reply: User removes the reply.

Table 3.6: Use Case Specification: View Profile.

Use Case	View Profile
Actor	User
Summary Description	Displays user profile information.
Pre-Condition	User is logged in.
Basic Path	1. User navigates to the profile page. 2. System retrieves and displays profile data including health metrics.
Post-Condition	Profile information is displayed.
Extension: Update Profile	1. User modifies profile details (e.g., weight). 2. System validates and updates the data.
Extension: Calculate BMI	1. User requests BMI calculation. 2. System computes and displays the BMI based on weight and height.

Table 3.7: Use Case Specification: Logout.

Use Case	Logout
Actor	User
Summary Description	Logs the user out of the system.
Pre-Condition	User is logged in.
Basic Path	1. User clicks the logout button. 2. System ends the session and redirects to the login page.
Post-Condition	User is logged out.

3.3.1.3 Activity Diagrams

The use case diagram in the previous section provides an overview of the end-to-end interaction between the user and the system. In this section, the activity diagram delves deeper into the system's backend processes. It illustrates the step-by-step flow of operations, detailing the sequence of actions and decisions that occur during specific functionalities. The activity diagrams highlight the interactions among user backend, System Backend, system database and machine learning (ML) algorithm, offering clarity on the operational details.

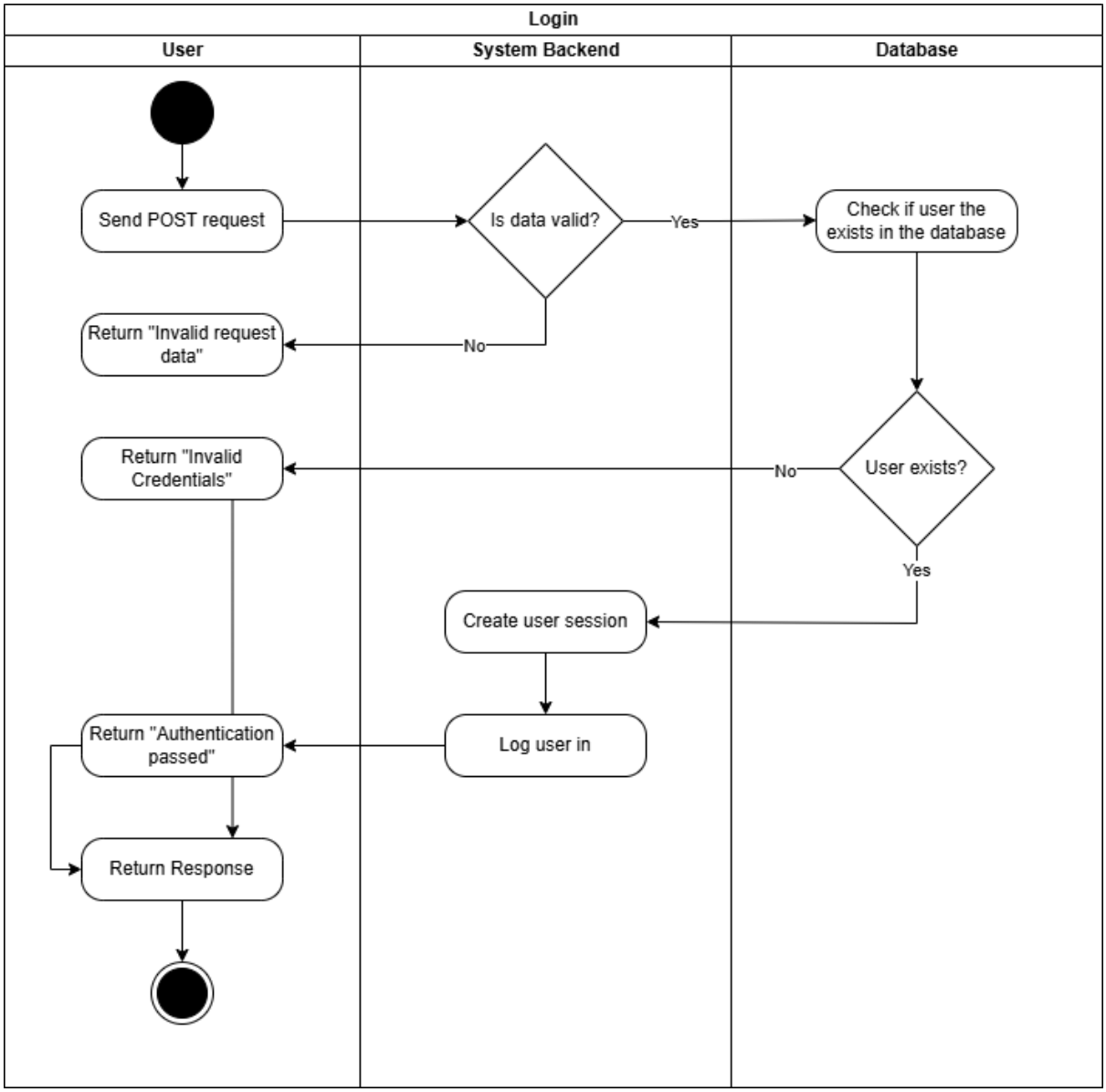


Figure 3.17: Activity Diagram: Login.

Figure 3.17 outlines the login process involving the User, System Backend, and Database. The User initiates the process by sending a POST request to the System Backend with login credentials. The backend first validates the request data; if invalid, it responds with "Invalid request data." If valid, the backend queries the Database to check if the username exists. If the user is not found, it responds with "Invalid Credentials." For valid users, System Backend creates a session, logs user in, and responds with "Authentication passed," completing the login process.

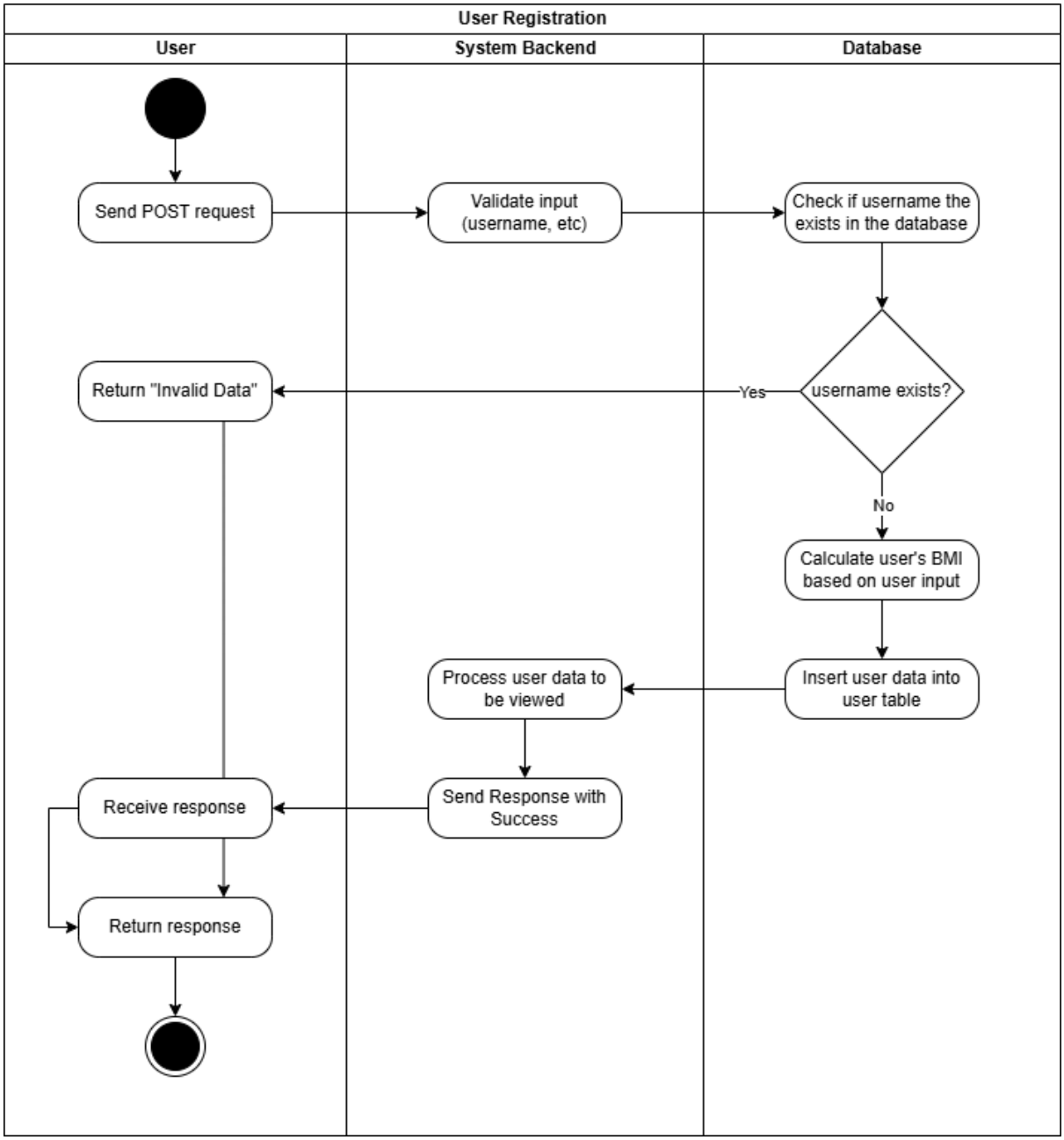


Figure 3.18: Activity Diagram: Registration.

Figure 3.18 describes the user registration process involving the User, System Backend, and Database. The User sends a POST request with registration details to the System Backend, which validates the input. If the data is invalid, the backend responds with "Invalid Data." If valid, the System Backend checks the Database to see if the username already exists. If it does, the process ends; otherwise, the backend then calculate the user's BMI based on inputs and inserts the user data into the database. The backend then processes the data, sends a success response, and the User receives confirmation of successful registration.

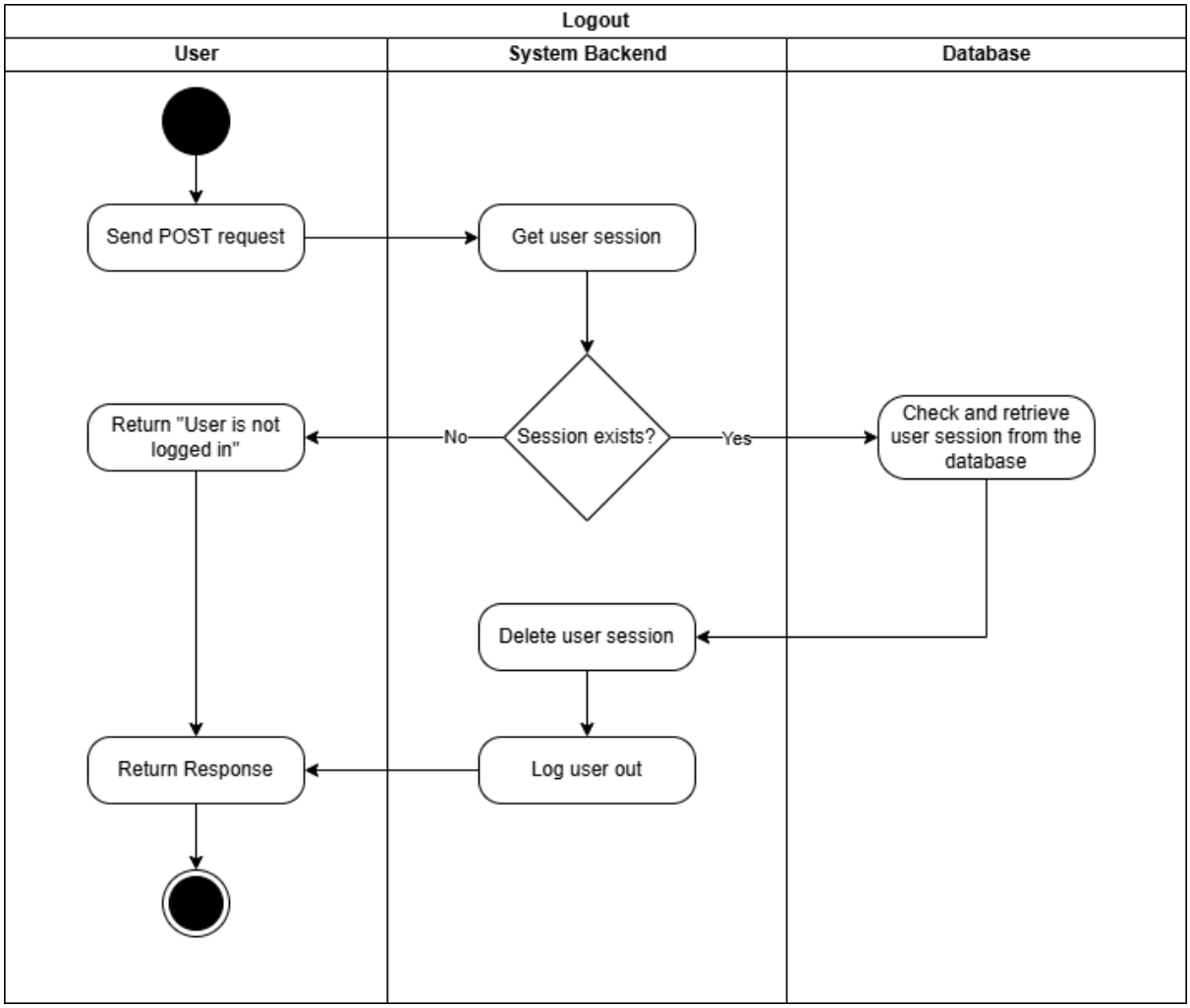


Figure 3.19: Activity Diagram: Logout.

Figure 3.19 describes the logout process involving the User, System Backend, and Database. The User sends a POST request to log out, prompting the System Backend to retrieve the user session. If no session exists, the backend responds with "User is not logged in." If a session is found, the System Backend retrieves it from the Database, deletes the session, logs the user's out, and sends a response confirming the logout. This process ensures proper session termination.

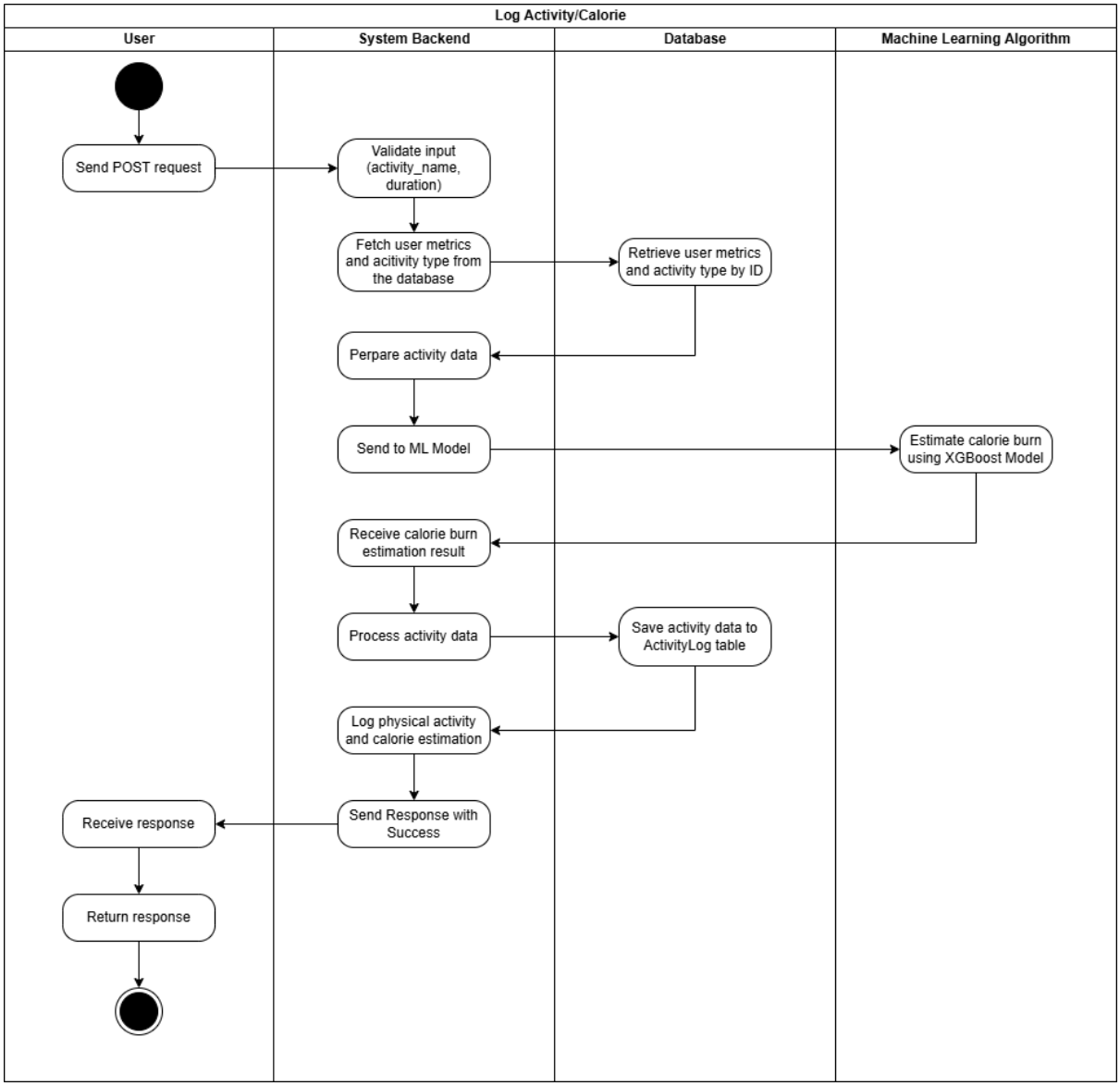


Figure 3.20: Activity Diagram: Log Activity/Calorie.

Figure 3.20 describes the process of logging an activity and estimating calorie burn involving the User, System Backend, Database, and a Machine Learning Algorithm. The User initiates the process by sending a POST request with activity details such as activity_name and duration. The System Backend validates the input, retrieves user metrics and activity type from the Database, and prepares the data. This data is then sent to the trained XGBoost model to estimate the calories burned. Upon receiving the estimated result, the System Backend processes the data, logs the activity along with the calorie estimation into the ActivityLog table in the Database, and sends a success response back to the User. This flow ensures accurate logging and calorie estimation for user activities.

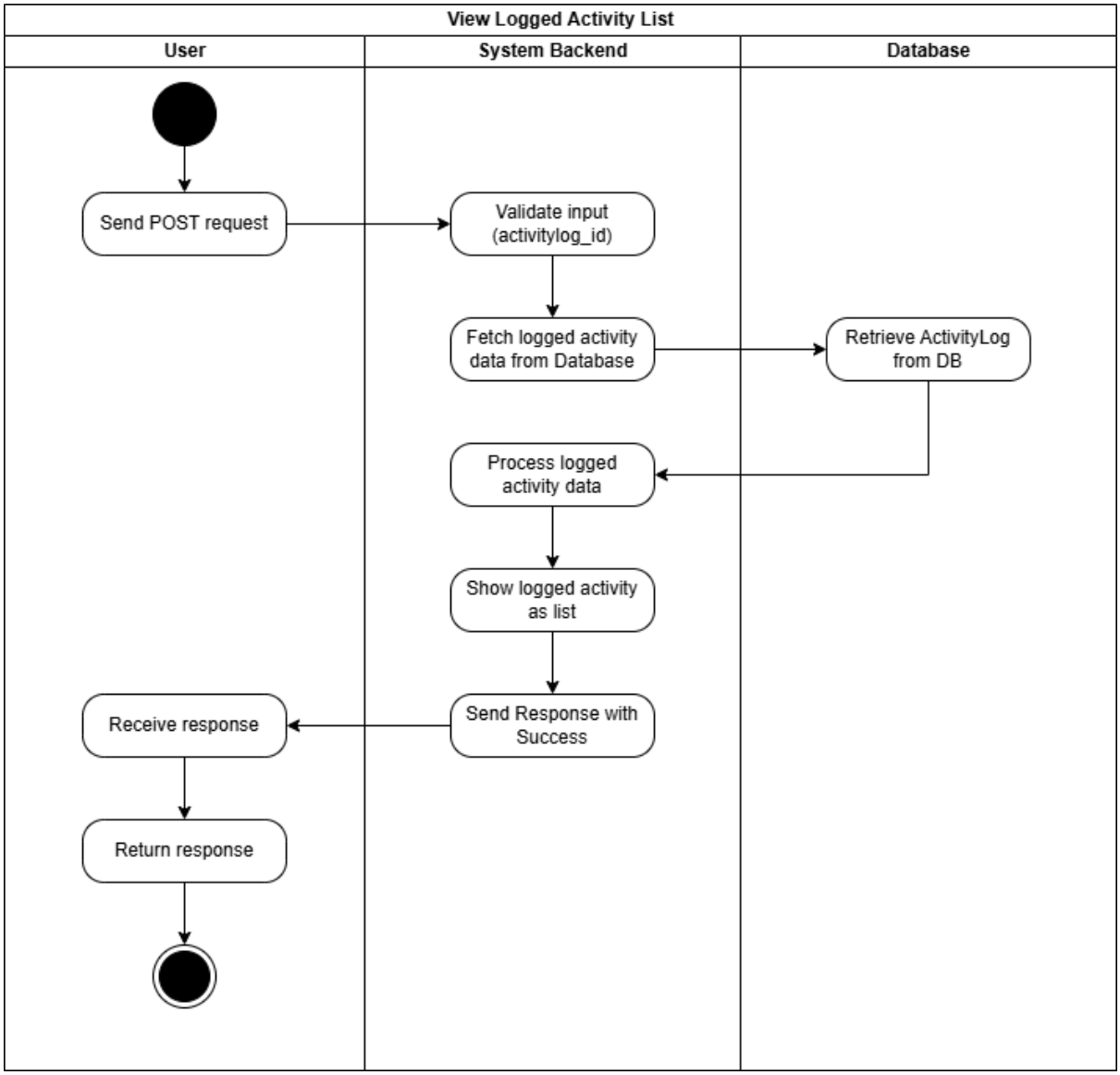


Figure 3.21: Activity Diagram: View Logged Activity List.

Figure 3.21 describes the process of viewing the logged activity list, involving the User, System Backend, and Database. The User initiates the request by sending a POST request containing the activitylog_id. The System Backend validates the input and retrieves the corresponding logged activity data from the ActivityLog table in the Database. Once the data is fetched, the System Backend processes it and displays the activity records in a list format. Finally, the backend sends a success response back to the User, ensuring retrieval and display of activity logs.

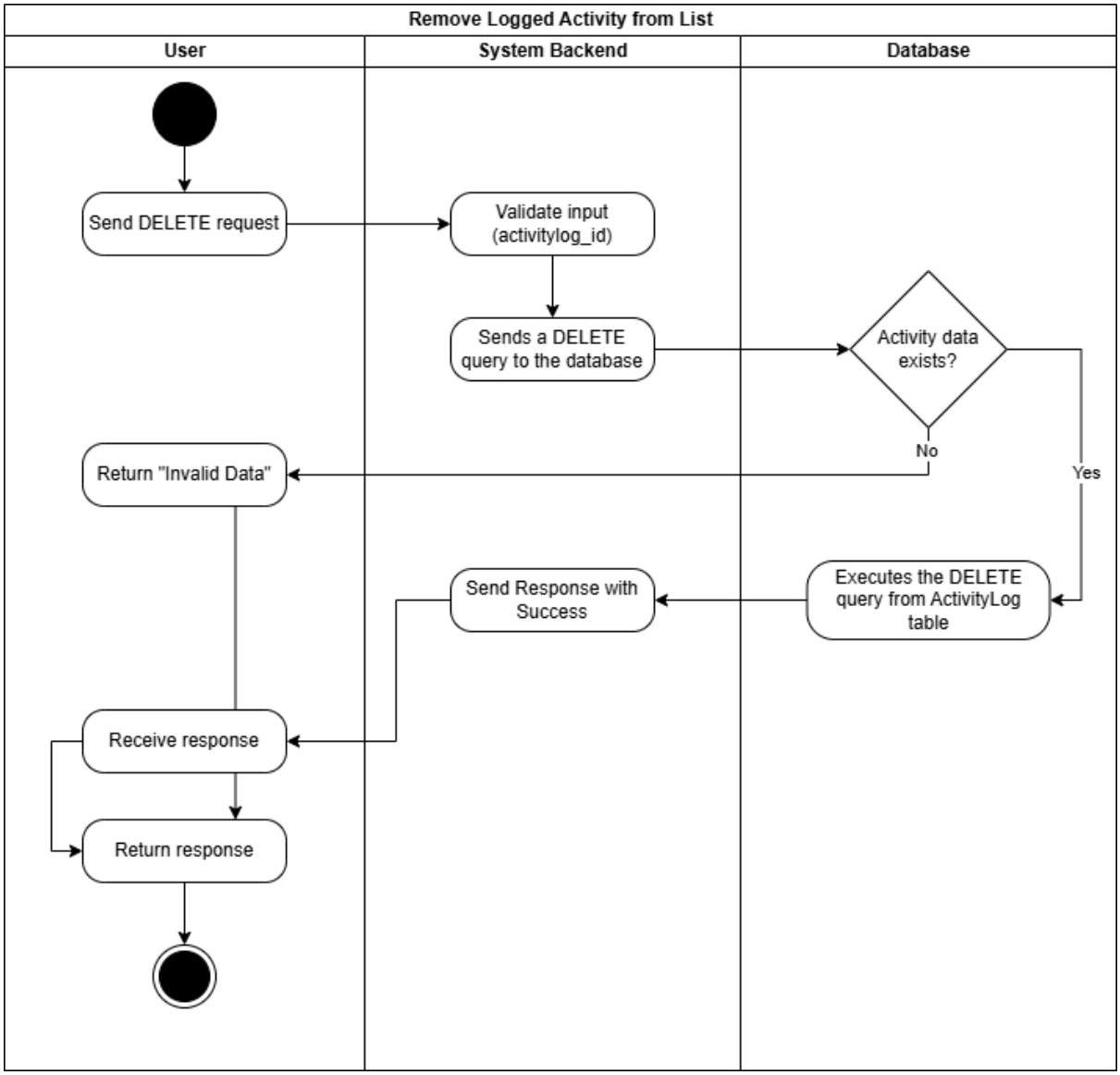


Figure 3.22: Activity Diagram: Remove Logged Activity from List.

Figure 3.22 describes the process of removing logged activity from the list, involving the User, System Backend, and Database. The User sends a DELETE request with the activitylog_id. The System Backend validates the input and sends a DELETE query to the Database. If the specified activity data exists in the ActivityLog table, the Database executes the DELETE query, and the System Backend sends a success response back to the User. If the data does not exist or the input is invalid, the System Backend returns an "Invalid Data" response to the User. This process ensures proper deletion of activity records while handling invalid requests.

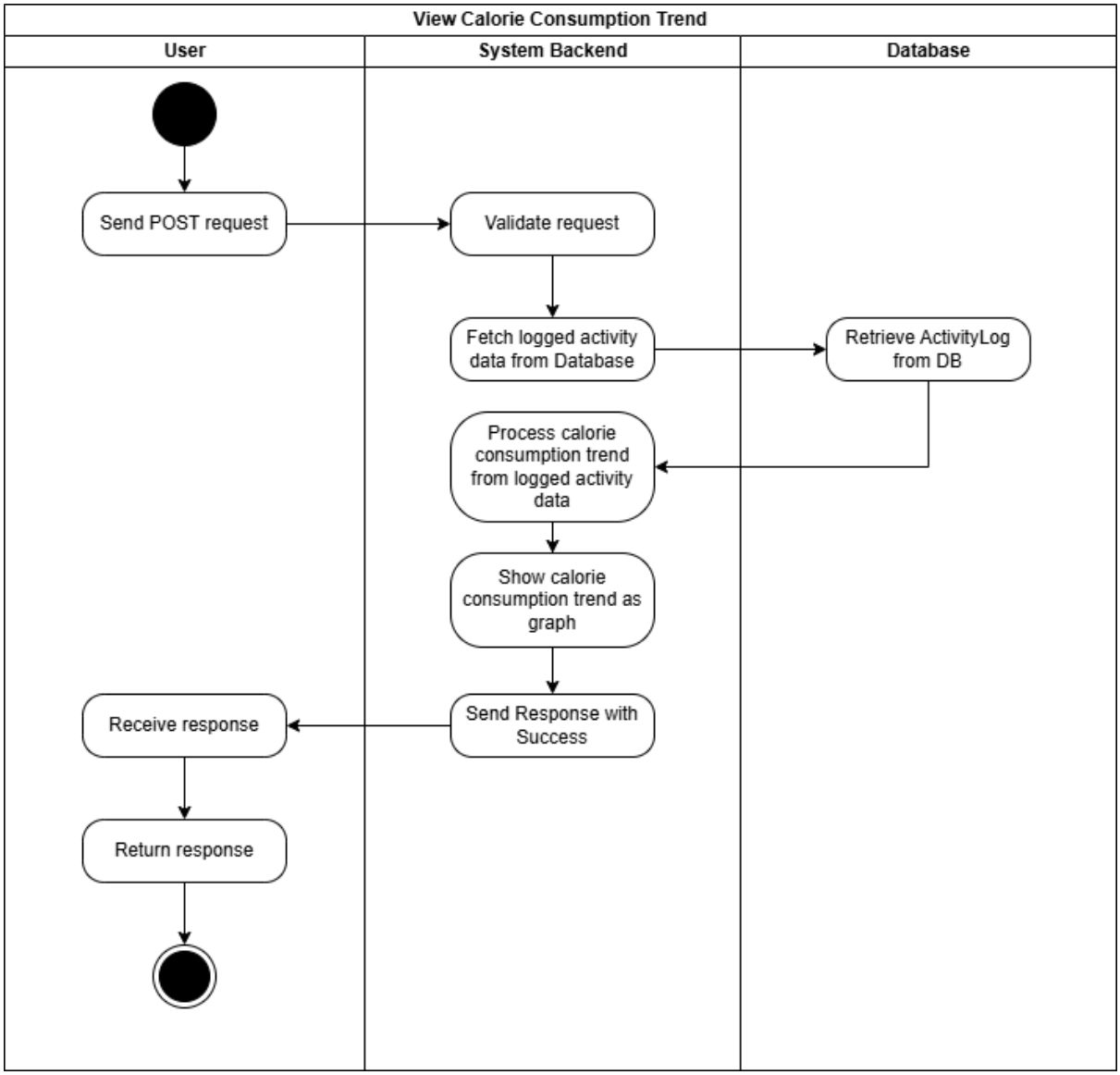


Figure 3.23: Activity Diagram: View Calorie Consumption Trend.

Figure 3.23 describes the process of viewing calorie consumption trends, involving the User, System Backend, and Database. The User sends a POST request to view the trend. The System Backend validates the request and retrieves logged activity data from the ActivityLog table in the Database. The retrieved data is processed to calculate and generate the calorie consumption trend, which is then displayed as a graph. Finally, the System Backend sends a success response back to the User.

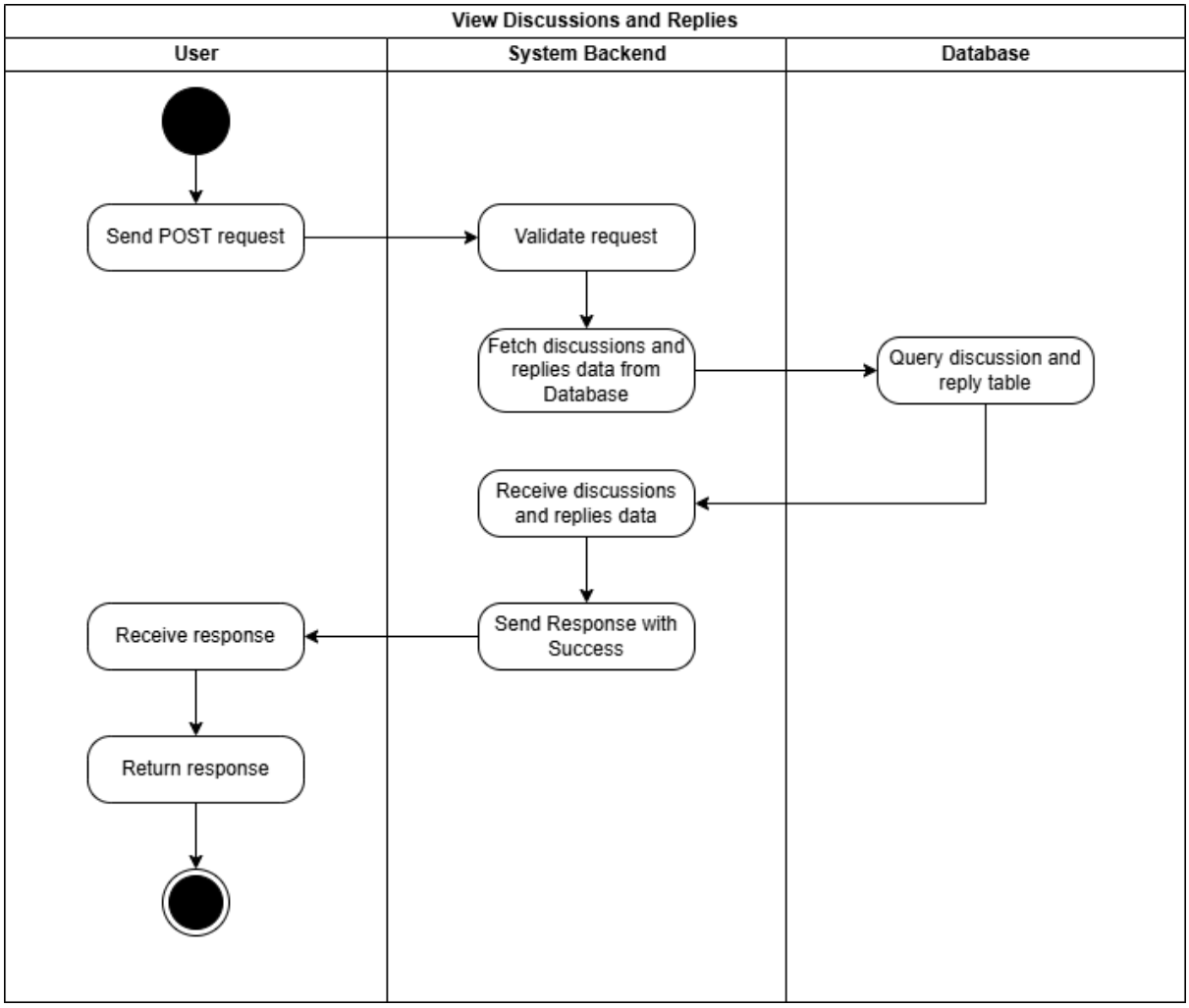


Figure 3.24: Activity Diagram: View Discussions and Replies.

Figure 3.24 describes the process of viewing discussions and replies, involving the User, System Backend, and Database. The User initiates the process by sending a POST request. The System Backend validates the request and queries the discussion table and reply table in the Database to fetch the respective data. Once the data is retrieved, the System Backend processes and sends the discussions and replies back to the User as part of a success response.

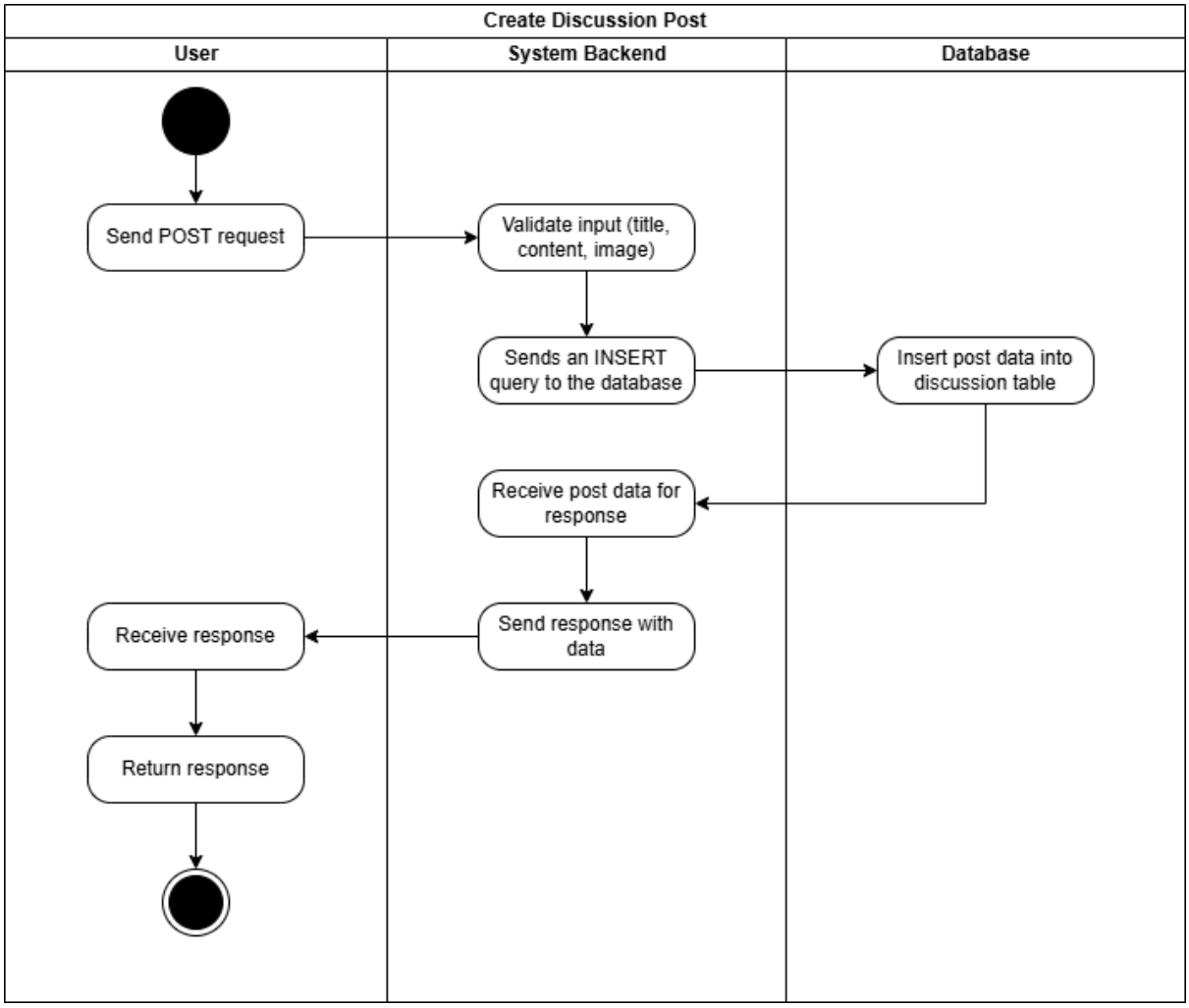


Figure 3.25: Activity Diagram: Create Discussion Post.

Figure 3.25 describes the process of creating a discussion post, involving the User, System Backend, and Database. The User sends a POST request containing input data such as title, content, and an optional image. The System Backend validates the input and sends an INSERT query to the Database to store the post in the discussion table. Once the data is successfully inserted, the Database returns the post data to the System Backend, which then sends a success response with the post details back to the User. This process ensures the proper creation and storage of discussion posts.

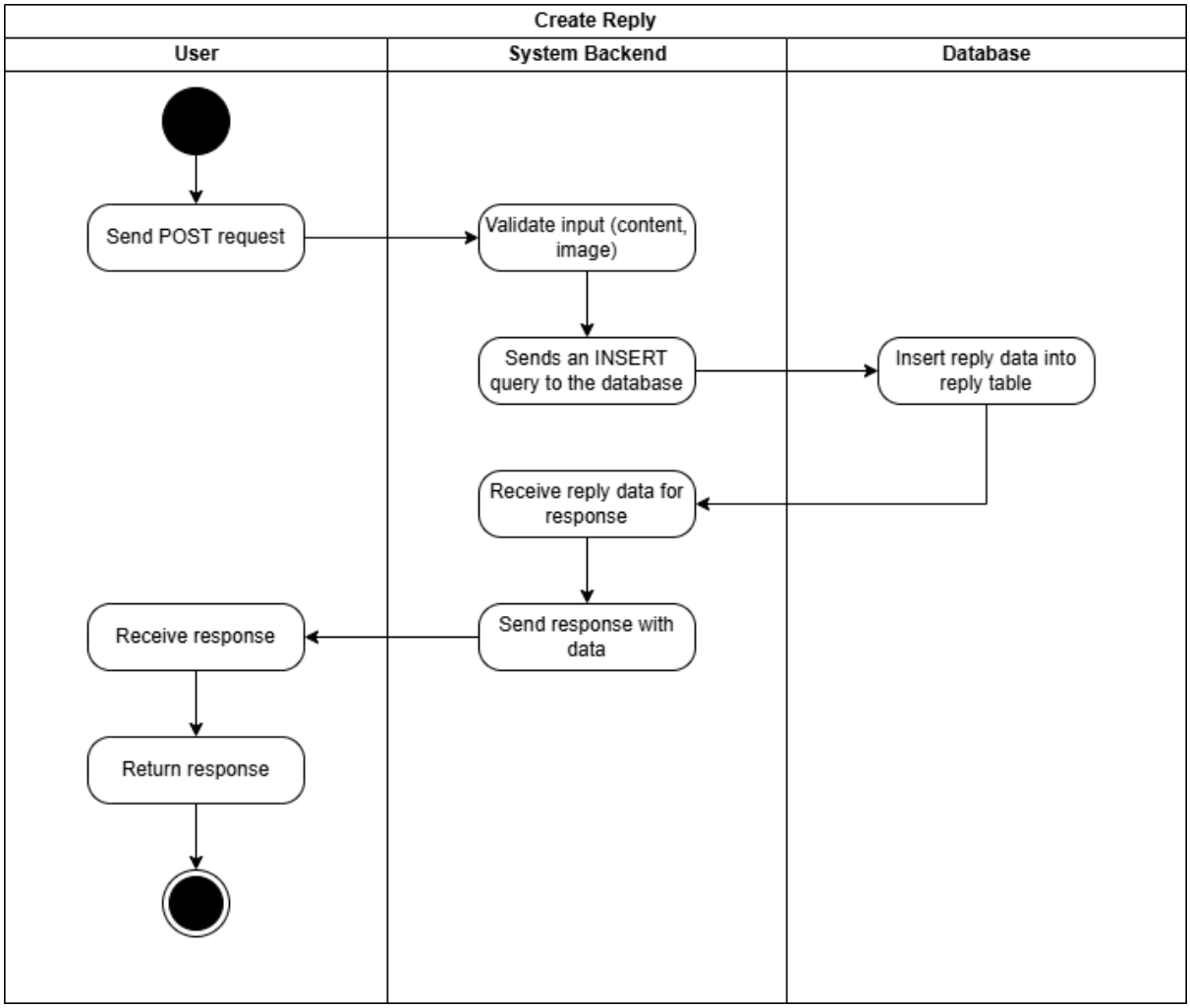


Figure 3.26: Activity Diagram: Create Reply.

Figure 3.26 describes the process of creating a reply to a discussion post, involving the User, System Backend, and Database. The User sends a POST request containing the reply data, including content and an optional image. The System Backend validates the input and sends an INSERT query to the Database to store the reply in the reply table. Once the data is successfully inserted, the Database returns the reply details to the System Backend, which then sends a success response with the reply data back to the User. This process ensures the proper creation and storage of replies.

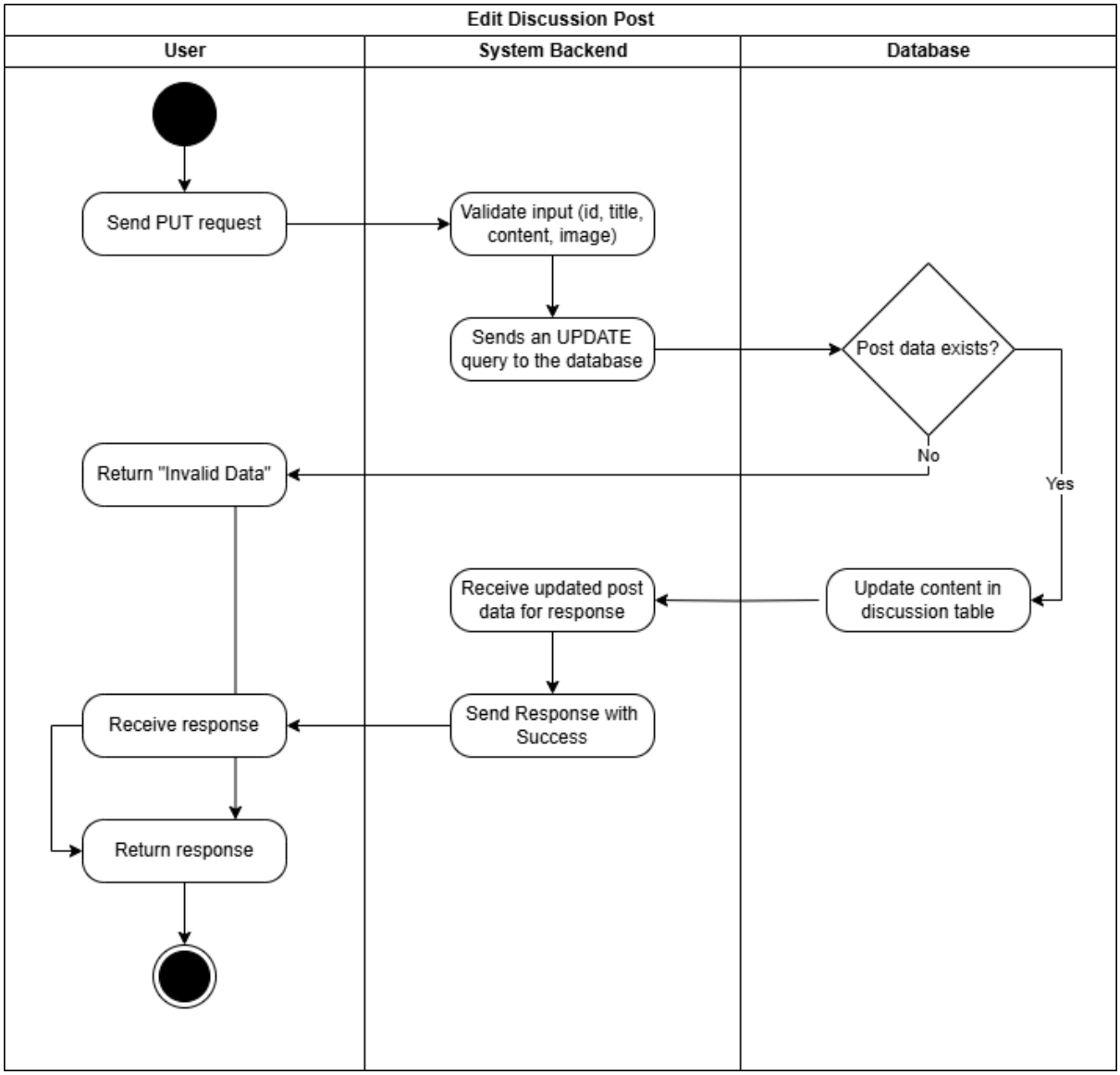


Figure 3.27: Activity Diagram: Edit Discussion Post.

Figure 3.27 describes the process of editing a discussion post, involving the User, System Backend, and Database. The User sends a PUT request containing the updated post data, including id, title, content, and an optional image. The System Backend validates the input and checks if the specified post exists in the Database. If the post is found, an UPDATE query is executed to modify the content in the discussion table. The updated post data is then returned to the System Backend, which sends a success response back to the User. If the post does not exist or the input is invalid, an "Invalid Data" response is returned to the User. This process ensures secure and accurate updates to discussion posts.

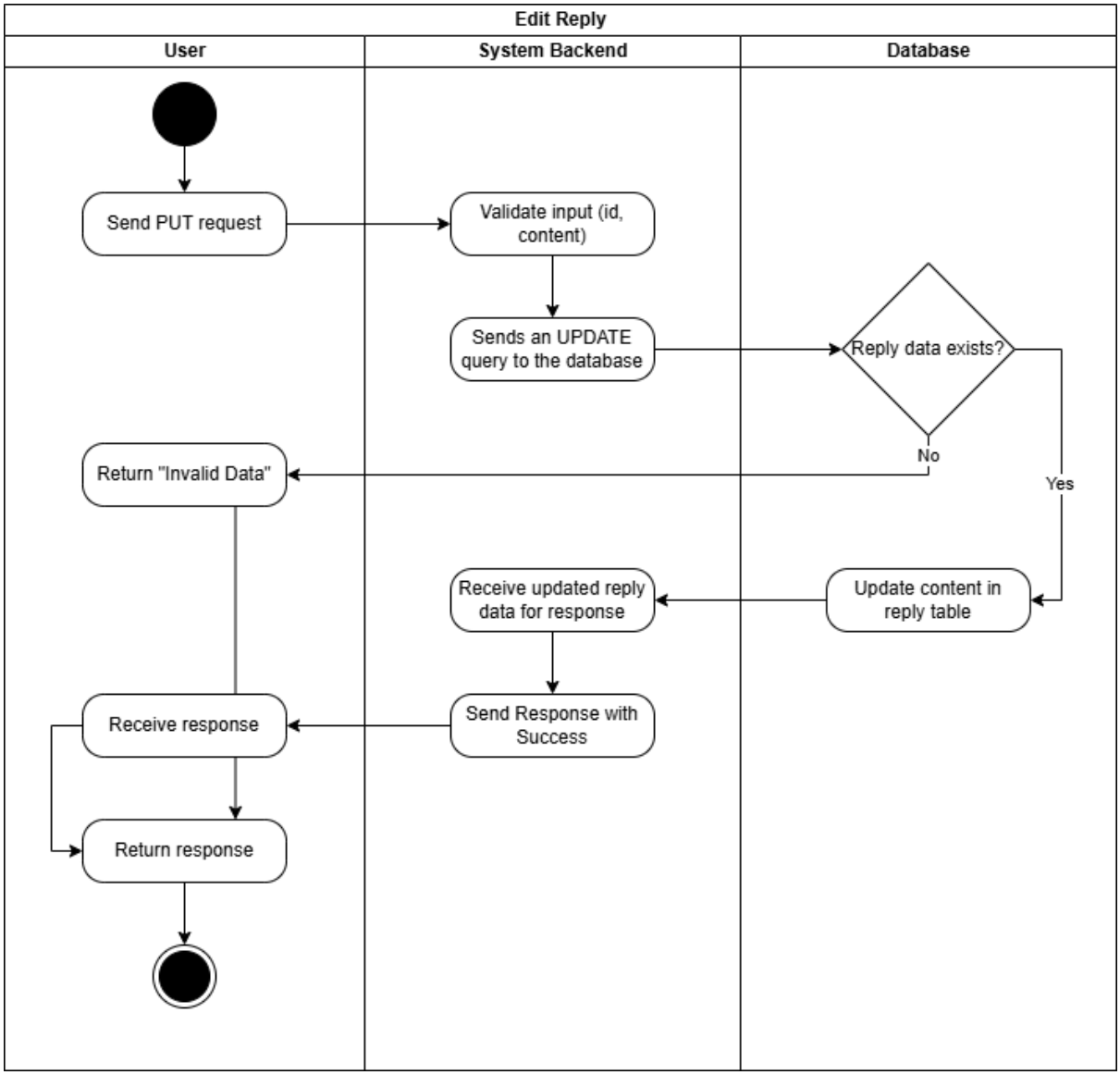


Figure 3.28: Activity Diagram: Edit Reply.

Figure 3.28 describes the process of editing a reply, involving the User, System Backend, and Database. The User sends a PUT request containing updated reply data, including the id and content. The System Backend validates the input and checks if the specified reply exists in the Database. If the reply is found, an UPDATE query is executed to modify the content in the reply table. The updated reply data is then returned to the System Backend, which sends a success response back to the User. If the reply does not exist or the input is invalid, an "Invalid Data" response is returned to the User. This process ensures secure and accurate updates to replies.

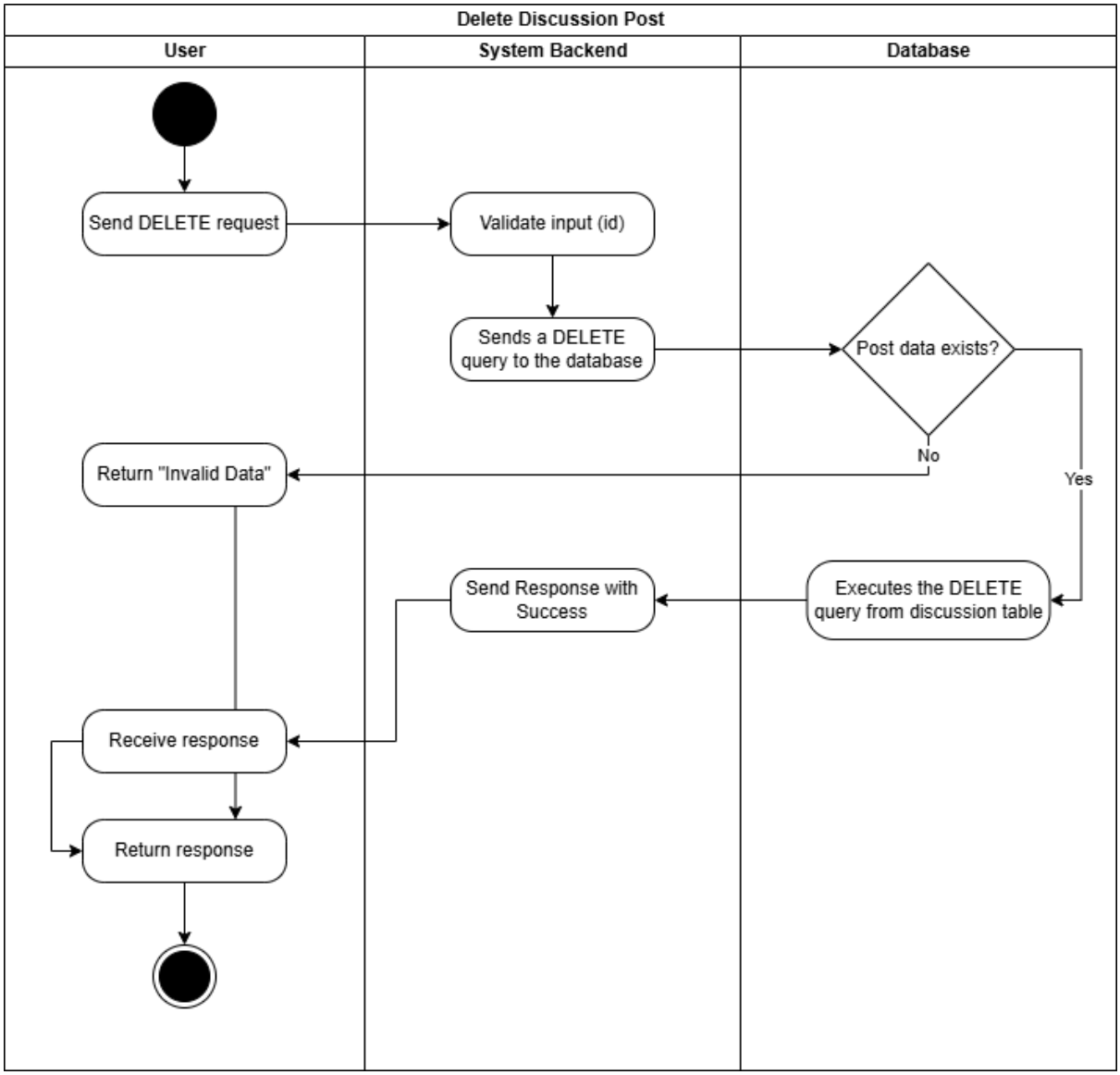


Figure 3.29: Activity Diagram: Delete Discussion Post.

Figure 3.29 describes the process of deleting a discussion post, involving the User, System Backend, and Database. The User sends a DELETE request containing the id of the post to be removed. The System Backend validates the input and checks if the specified post exists in the Database. If the post is found, a DELETE query is executed to remove the post from the discussion table. Upon successful deletion, the System Backend sends a success response back to the User. If the post does not exist or the input is invalid, the System Backend returns an "Invalid Data" response. This process ensures secure and proper deletion of discussion posts.

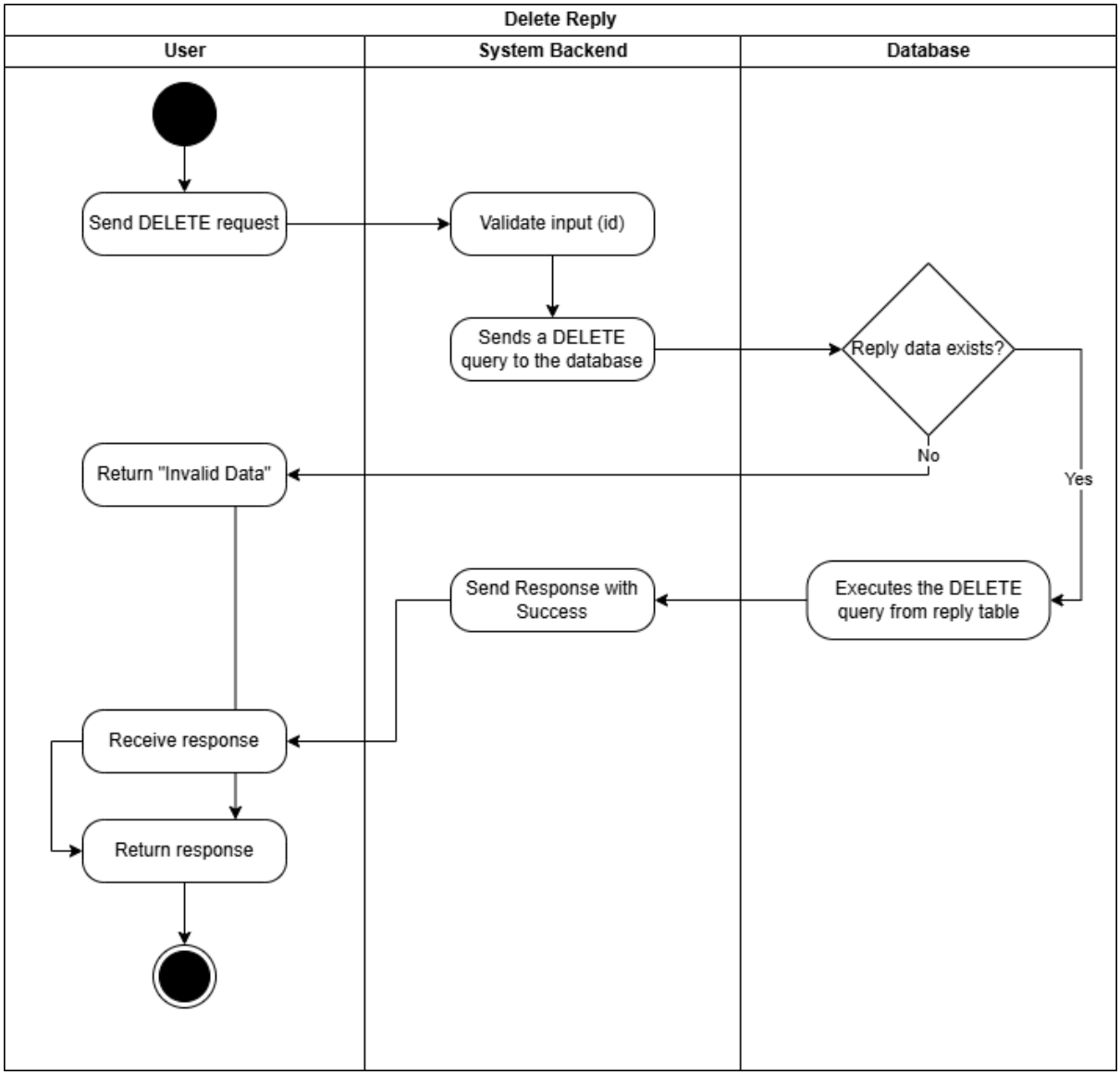


Figure 3.30: Activity Diagram: Delete Reply.

Figure 3.30 describes the process of deleting a reply, involving the User, System Backend, and Database. The User sends a DELETE request containing the id of the reply to be removed. The System Backend validates the input and checks if the specified reply exists in the Database. If the reply is found, a DELETE query is executed to remove the reply from the reply table. Upon successful deletion, the System Backend sends a success response back to the User. If the reply does not exist or the input is invalid, the System Backend returns an "Invalid Data" response. This ensures secure and proper deletion of replies.

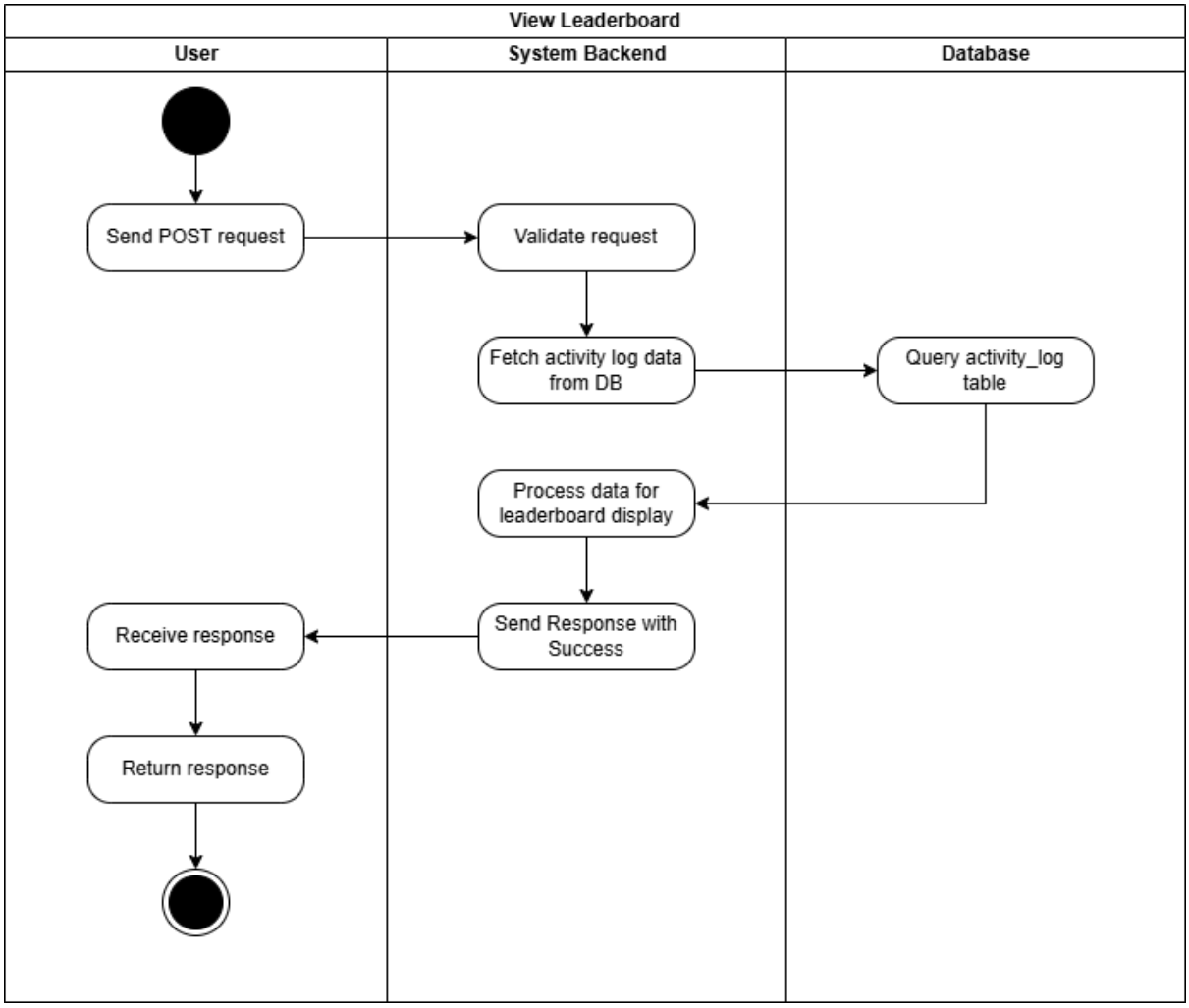


Figure 3.31: Activity Diagram: View Leaderboard.

Figure 3.31 describes the process of viewing the leaderboard, involving the User, System Backend and Database. The User sends a POST request to retrieve the leaderboard. The System Backend validates the request and queries the leaderboard table in the Database to fetch the leaderboard data. The leaderboard data is then sent back to the System Backend, which processes it and sends a success response with the leaderboard details to the User. This process ensures accurate retrieval of leaderboard information for the User.

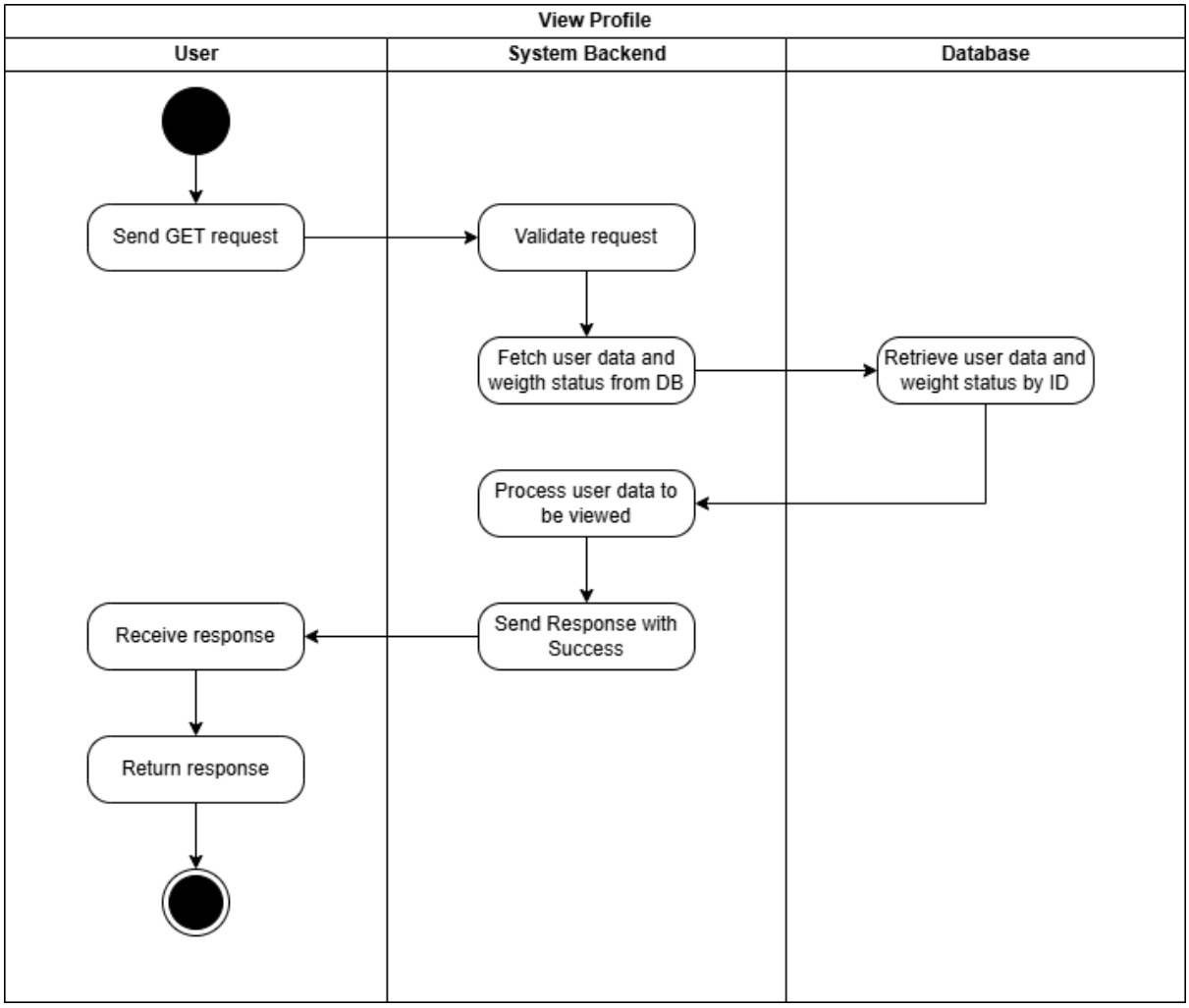


Figure 3.32: Activity Diagram: View Profile.

Figure 3.32 describes the process of viewing the user's profile, involving the User, System Backend, and Database. The User sends a POST request to retrieve profile data. The System Backend validates the request and checks if the user data exists in the Database. If the user data is found, it retrieves the data and processes it for display. The System Backend then sends a success response with the profile details back to the User. If the user data does not exist or the input is invalid, the System Backend returns an "Invalid Data" response. This process ensures proper validation and secure access to user profile information.

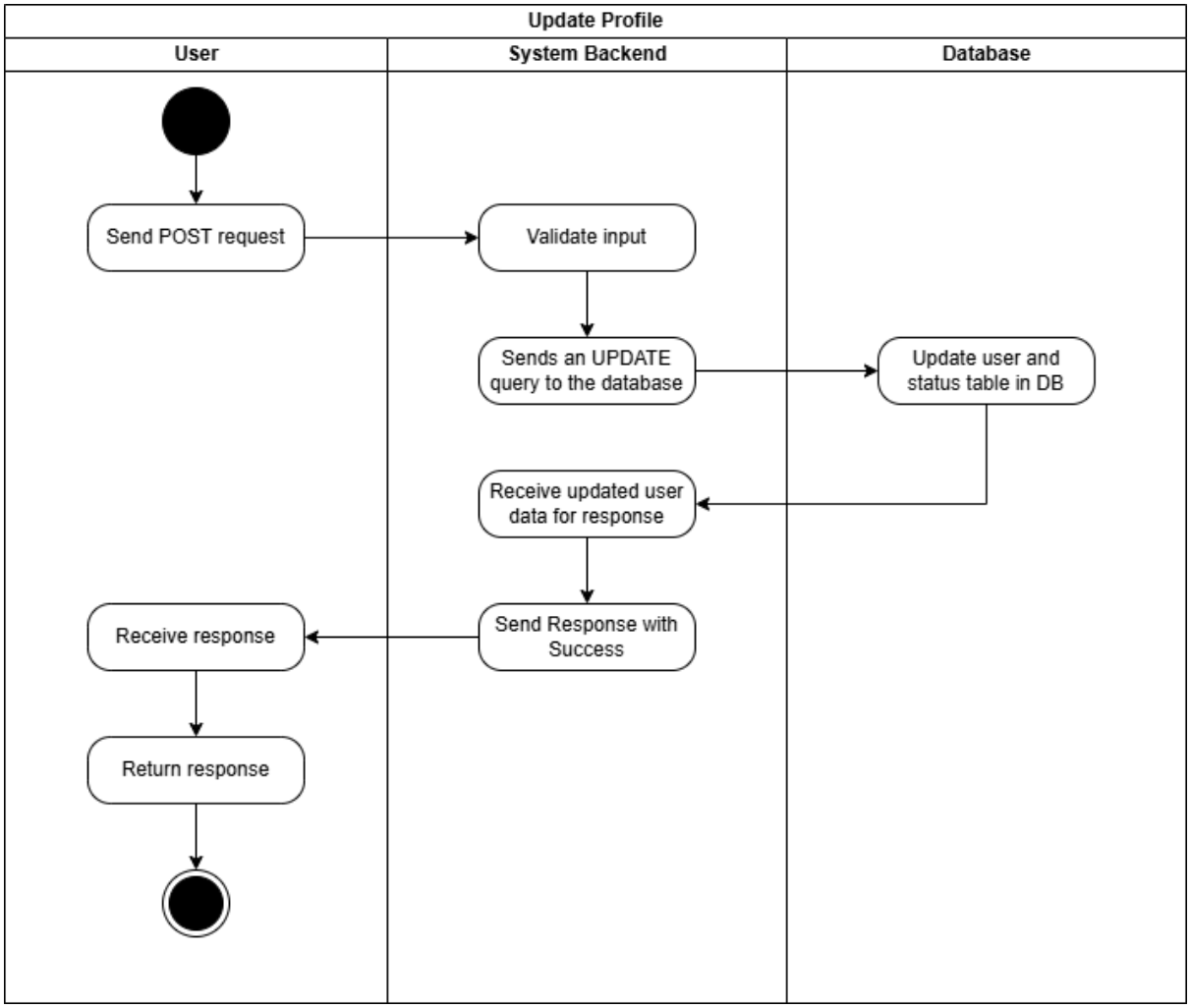


Figure 3.33: Activity Diagram: Update Profile.

Figure 3.33 describes the process of updating a user's profile, involving the User, System Backend, and Database. The User sends a PUT request containing the updated profile information and user_id. The System Backend validates the input and checks if the user data exists in the Database. If the user data is found, an UPDATE query is executed to modify the user's information in the user table. The updated data is then returned to the System Backend, which sends a success response back to the User. If the user data does not exist or the input is invalid, the System Backend returns an "Invalid Data" response. This process ensures proper validation and secure profile updates.

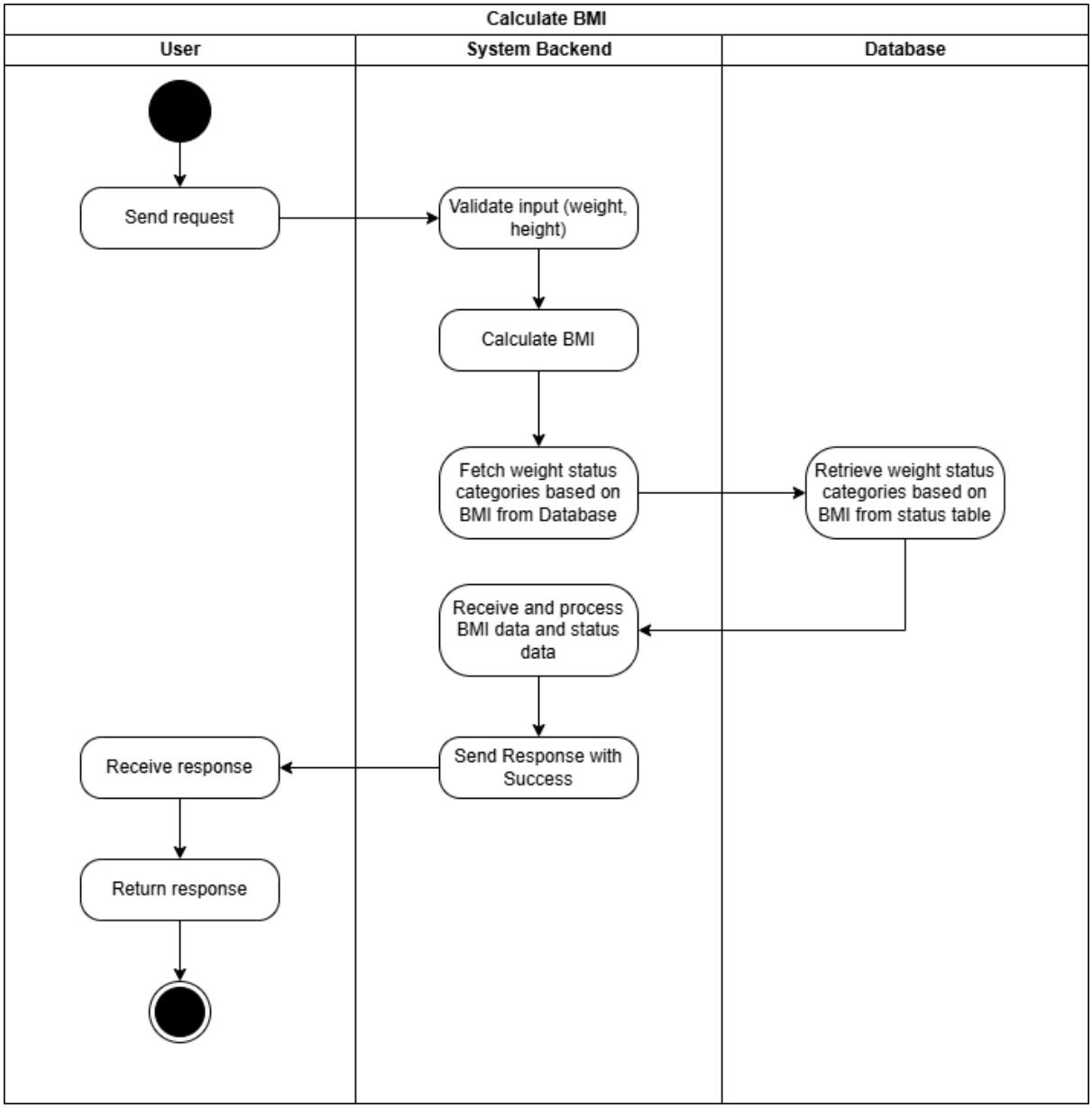


Figure 3.34: Activity Diagram: Calculate BMI.

Figure 3.34 describes the process of calculating BMI (Body Mass Index), involving the User, System Backend, and Database. The User sends a request with their weight and height. The System Backend validates the input and calculates the BMI using the provided data. It then queries the Database to retrieve weight status categories based on the calculated BMI from the status table. The System Backend processes the retrieved BMI data and weight status categories and sends a success response with the results back to the User. This process ensures accurate BMI calculation and provides meaningful insights for the users.

3.3.1.4 Entity Relationship Diagram (ERD) and Data Dictionary

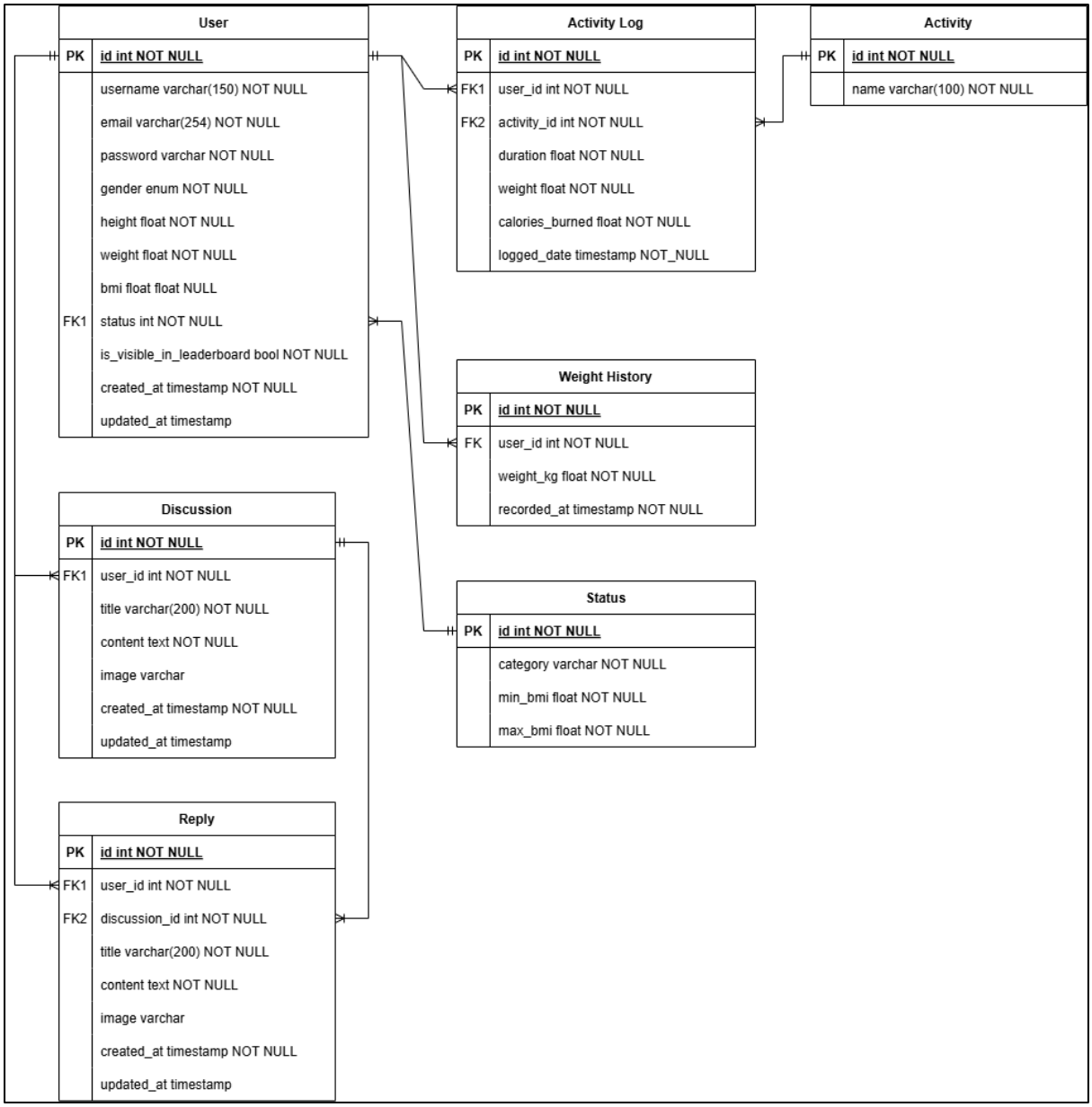


Figure 3.35: ERD of Proposed System.

The Entity-Relationship Diagram (ERD) in Figure 3.35 illustrates the updated database design for the proposed calorie tracking system, which integrates activity logging, historical weight tracking, performance monitoring, and social interaction. The system's database consists of seven tables: User, Status, Activity, Activity Log, Weight History, Discussion, and Reply.

The User table is the central entity, storing essential user information such as username, email, gender, height, weight, BMI (calculated), and the visibility status for leaderboard participation (is_visible_in_leaderboard). Timestamps (created_at, updated_at) track user

registration and profile updates. BMI classification is linked through the Status table using a foreign key relationship.

The Status table categorizes BMI values into defined ranges (min_bmi, max_bmi) for easier health status classification.

The Activity table maintains a list of predefined physical activities (e.g., Cycling, vigorous, Badminton, general). The Activity Log table records detailed information for each logged activity, including duration, weight used for estimation, calories burned, and the specific date of the logged activity (logged_date). This table connects to both User and Activity tables through foreign keys, allowing users to log multiple activity entries while referencing the predefined activity types.

The Weight History table stores historical weight records for users. Each user may only have one weight record per date. This design enables accurate calorie estimation when users log activities for past dates, as the system can reference the weight recorded on that particular day.

The Discussion table supports community engagement by allowing users to create discussion threads. Each discussion entry includes the title, content, optional image, and timestamps. The Reply table allows users to reply to discussion posts, maintaining a reference to both the discussion post and the user.

Foreign key constraints such as user_id, activity_id, and discussion_id establish referential integrity between tables, ensuring consistency and organization across the system. This design supports the system's core features: activity tracking, accurate calorie estimation with historical weight data, performance monitoring, and interactive community discussions.

The data dictionary for each entity are as follows:

Entity Name: User

Entity Description: Stores information about users who interact with the system. This includes personal metrics required for calorie tracking and profile management.

Table 3.8: Data Dictionary for User Table.

Column Name	**Column Description**	**Data Type**	**Key**	**Constraints**
id	Unique identifier for each user.	Integer	Primary Key	Auto-increment, NOT NULL

username	User's unique username.	String (150)	Unique Key	NOT NULL, Unique
email	User's email address.	String (254)	Unique Key	NOT NULL, Unique
password	Encrypted password.	String	-	NOT NULL
gender	User's gender ("Male", "Female") for personalization.	Enum	-	NOT NULL
height	User's height in meters.	Float	-	NOT NULL
weight	User's current weight in kilograms.	Float	-	NOT NULL
bmi	User's current body mass index in kilograms per meter square.	Float	-	NOT NULL
status_id	Reference to the id in the Status table.	Integer	Foreign Key	NOT NULL, Foreign Key
created_at	Timestamp when the user registered.	Timestamp	-	NOT NULL, Default=NOW
updated_at	Timestamp when the user metrics was updated.pas	Timestamp	-	Optional

Entity Name: Status

Entity Description: Stores weight status categories for each range of BMI.

Table 3.9: Data Dictionary for Status Table.

Column Name	**Column Description**	**Data Type**	**Key**	**Constraints**
id	Unique identifier for each weight status category.	Integer	Primary Key	NOT NULL
category	Weight status category ("Underweight", "Normal", "Overweight", "Obesity").	String (20)	Unique	NOT NULL, Unique
min_bmi	Minimum BMI value for the range.	Float	-	NOT NULL
max_bmi	Maximum BMI value for the range.	Float	-	NOT NULL

Entity Name: Activity

Entity Description: Contains predefined physical activity types for users to log.

Table 3.10: Data Dictionary for Activity Table.

Column Name	Column Description	Data Type	Key	Constraints
id	Unique identifier for the activity type.	Integer	Primary Key	Auto-increment, NOT NULL
name	Name of the activity (e.g., running).	String (100)	Unique Key	NOT NULL, Unique

Entity Name: Activity Log

Entity Description: Logs details of activities performed by users, including estimated calories burned.

Table 3.11: Data Dictionary for Activity Log Table.

Column Name	Column Description	Data Type	Key	Constraints
id	Unique identifier for each log.	Integer	Primary Key	Auto-increment, NOT NULL
user_id	Reference to the id in the User table.	Integer	Foreign Key	NOT NULL, Foreign Key
activity_id	Reference to the id in the Activity table.	Integer	Foreign Key	NOT NULL, Foreign Key
duration	Duration of activity in minutes.	Float	-	NOT NULL
weight_kg	Weight of the user for current log.	Float	-	NOT NULL
is_visible_in_leaderboard	Indicates whether the user is visible on the leaderboard. Default is visible.	Boolean	-	NOT NULL, Default=True
calories_burned	Estimated calories burned.	Float	-	NOT NULL
logged_date	Timestamp of when the activity was logged.	Timestamp	-	NOT NULL, Default=NOW

Entity Name: Discussion

Entity Description: Discussion posts created by users, optionally including images.

Table 3.12: Data Dictionary for Discussion Table.

Column Name	Column Description	Data Type	Key	Constraints
id	Unique identifier for each discussion/post.	Integer	Primary Key	Auto-increment, NOT NULL
user_id	Reference to the id in the User table.	Integer	Foreign Key	NOT NULL, Foreign Key
title	Title of the discussion.	String (200)	Foreign Key	NOT NULL, Foreign Key
content	Main content of the discussion post.	Text	-	NOT NULL
image	Path or URL of the image if added.	String (FilePath)	-	Optional
created_at	Timestamp when the discussion/post was created.	Timestamp	-	NOT NULL, Default=NOW
updated_at	Timestamp when the discussion/post was updated.	Timestamp	-	Optional

Entity Name: Reply

Entity Description: Replies to discussion posts by users.

Table 3.13: Data Dictionary for Reply Table.

Column Name	Column Description	Data Type	Key	Constraints
id	Unique identifier for each discussion/post.	Integer	Primary Key	Auto-increment, NOT NULL
discussion_id	Reference to the id in the Discussion table.	Integer	Foreign Key	NOT NULL, Foreign Key
user_id	Reference to the id in the User table.	Integer	Foreign Key	NOT NULL, Foreign Key
content	Main content of the discussion post.	Text	-	NOT NULL
image	Path or URL of the image if added.	String (FilePath)	-	Optional

created_at	Timestamp when the reply was created.	Timestamp	-	NOT NULL, Default=NOW
updated_at	Timestamp when the reply was updated.	Timestamp	-	Optional

Entity Name: Weight History

Entity Description: Stores historical weight records of users to enable accurate calorie estimation when logging activities for past dates.

Table 3.14: Data Dictionary for Weight History.

Column Name	Column Description	Data Type	Key	Constraints
id	Unique identifier for each weight record.	Integer	Primary Key	Auto-increment, NOT NULL
user_id	Reference to the id in the User table.	Integer	Foreign Key	NOT NULL, Foreign Key
weight_kg	Indicates whether the user is visible on the leaderboard. Default is visible.	Float	-	NOT NULL, Default=True
recorded_at	Username of the user.	Timestamp	-	NOT NULL

3.3.2 Physical Design

Wireframes are used as the physical design of the proposed system, serving as a blueprint that outlines the layout, structure, and functionality of the user interface. It helps visualize the placement of elements and ensures that the system's design aligns with user requirements and objectives in the design phase.

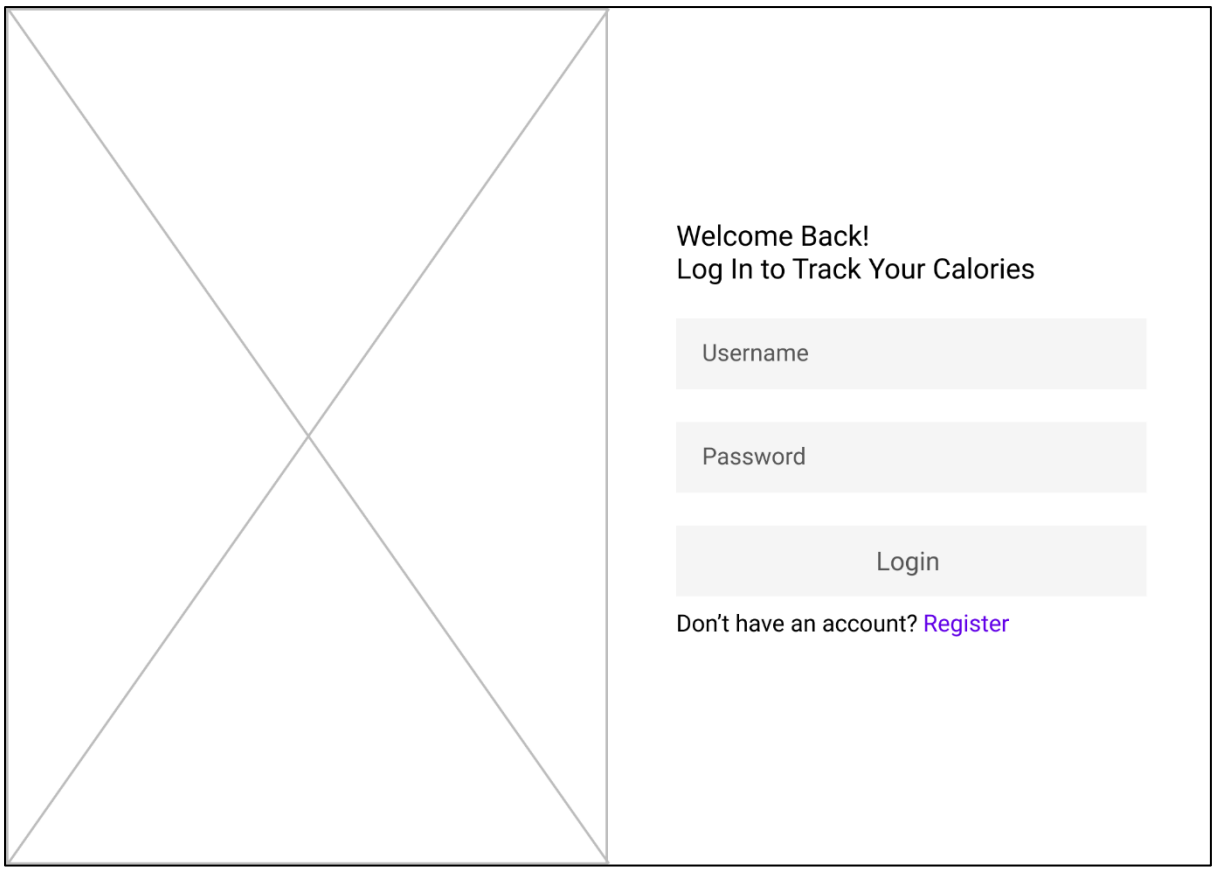


Figure 3.36: Login Page.

Figure 3.36 shows the login screen for the proposed system. The page includes standard input fields for username and password, and a login button. For new users, a link to the registration page is provided as a register button.

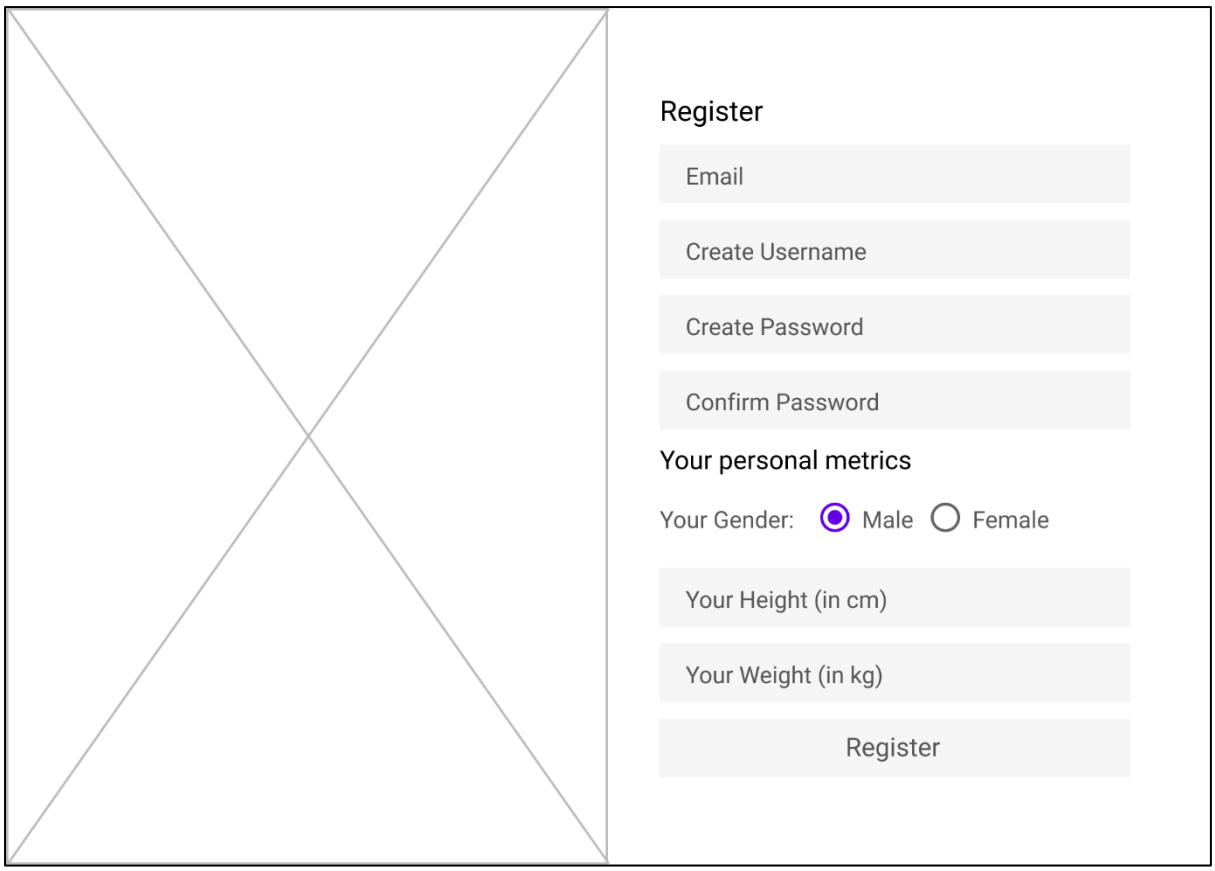


Figure 3.37: Registration Page.

Figure 3.37 shows a registration screen for the proposed system. The users are required to input all the included fields such as email, username, password, and password confirmation. Also, there is a section for collecting user's personal metrics such as gender, height, and weight, which are essential for personalized calorie tracking.

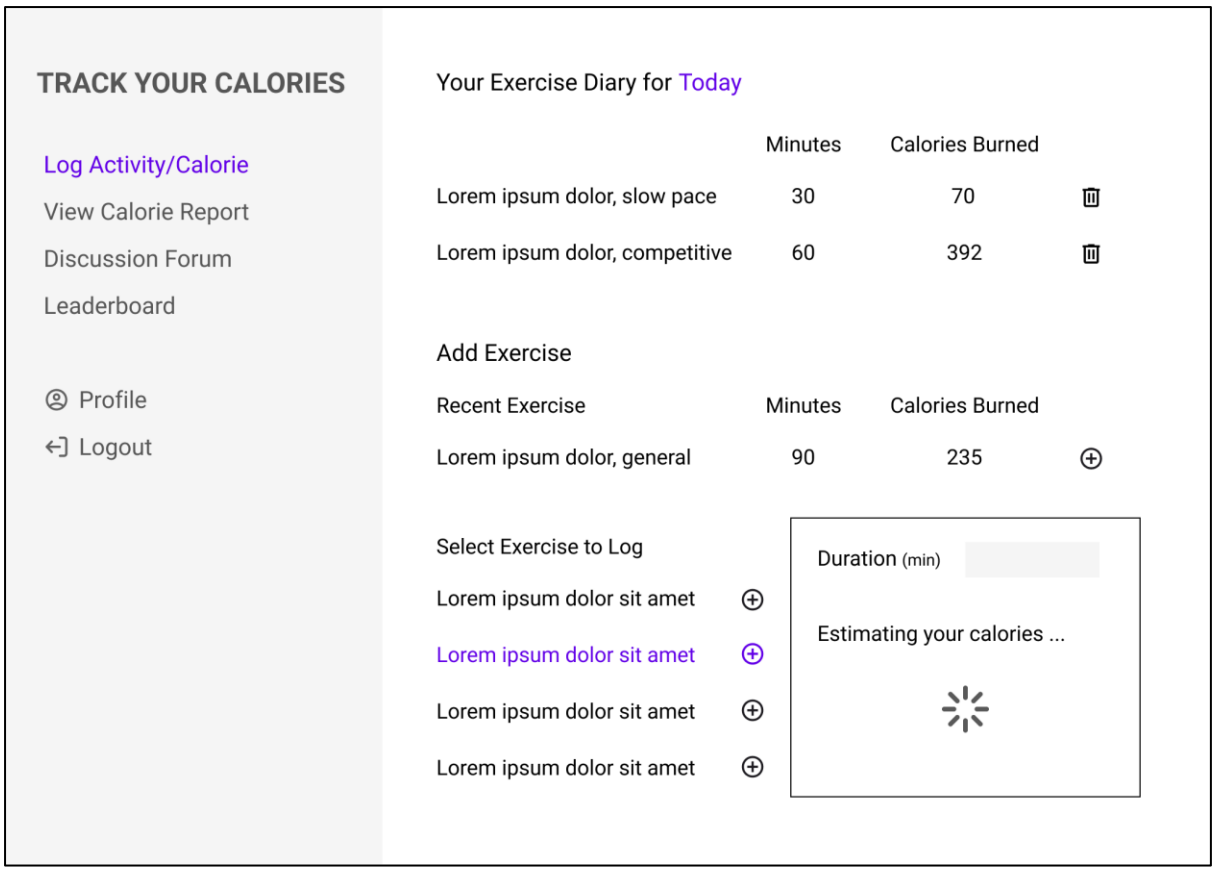


Figure 3.38: Log Activity and Calorie Estimation Page.

Figure 3.38 shows a page within the proposed system where users can manage their exercise data. In this page, users can view a history of their recent workouts, including duration and estimated calories burned. They can also log new activities by selecting from their recent log exercises or from a list of provided exercises and enter the duration and distance (if applicable), the system will estimate the calories burned from the inputs. The page allows for flexibility by enabling users to delete logged activities as needed.

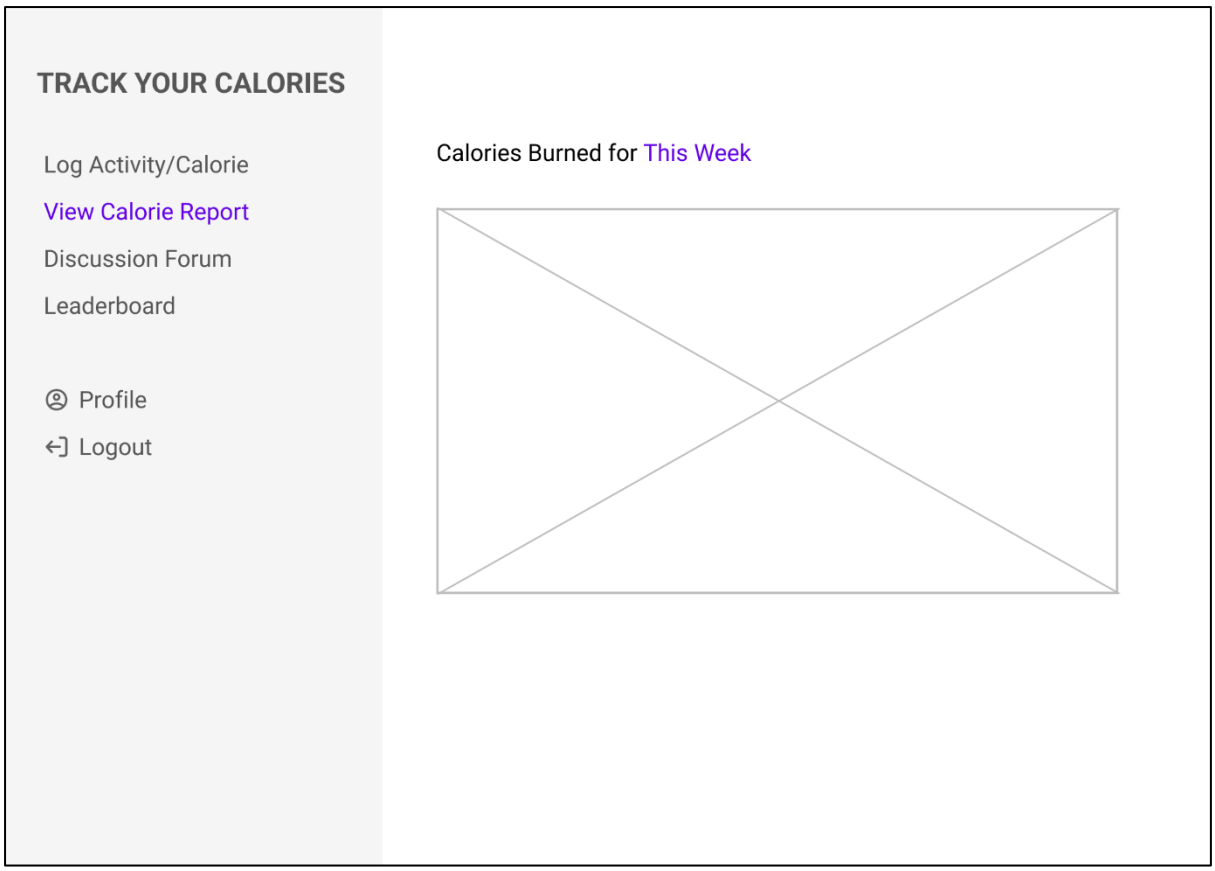


Figure 3.39: Calorie Report Page.

Figure 3.39 shows the calorie report page within the proposed system designed to help users track their calorie data. On this page, users can view a report (a graph) of their weekly calories burned, visually represented as the content placeholder.

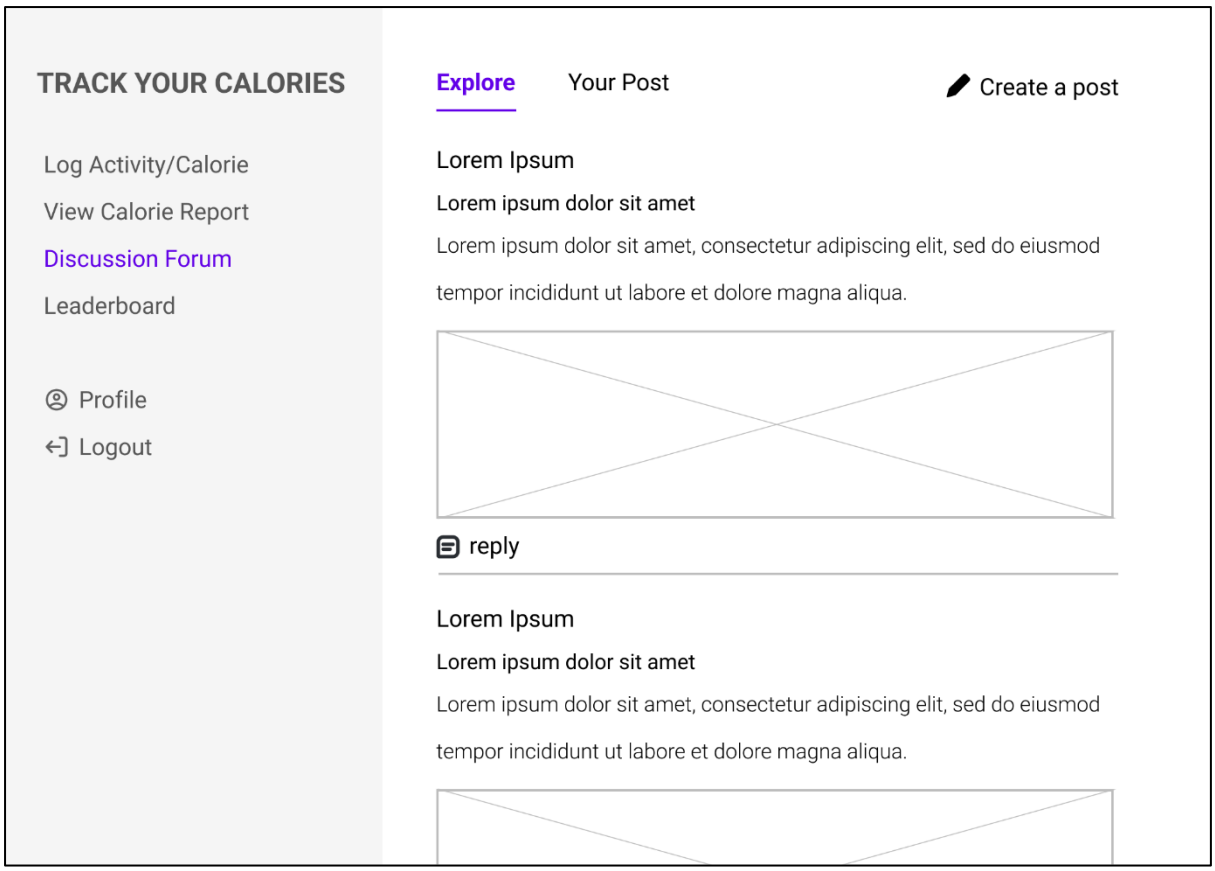


Figure 3.40: Discussion Forum Page.

Figure 3.40 shows the Discussion Forum page within the proposed system, where users can engage with the community by exploring posts, creating their own posts, and replying to existing ones.

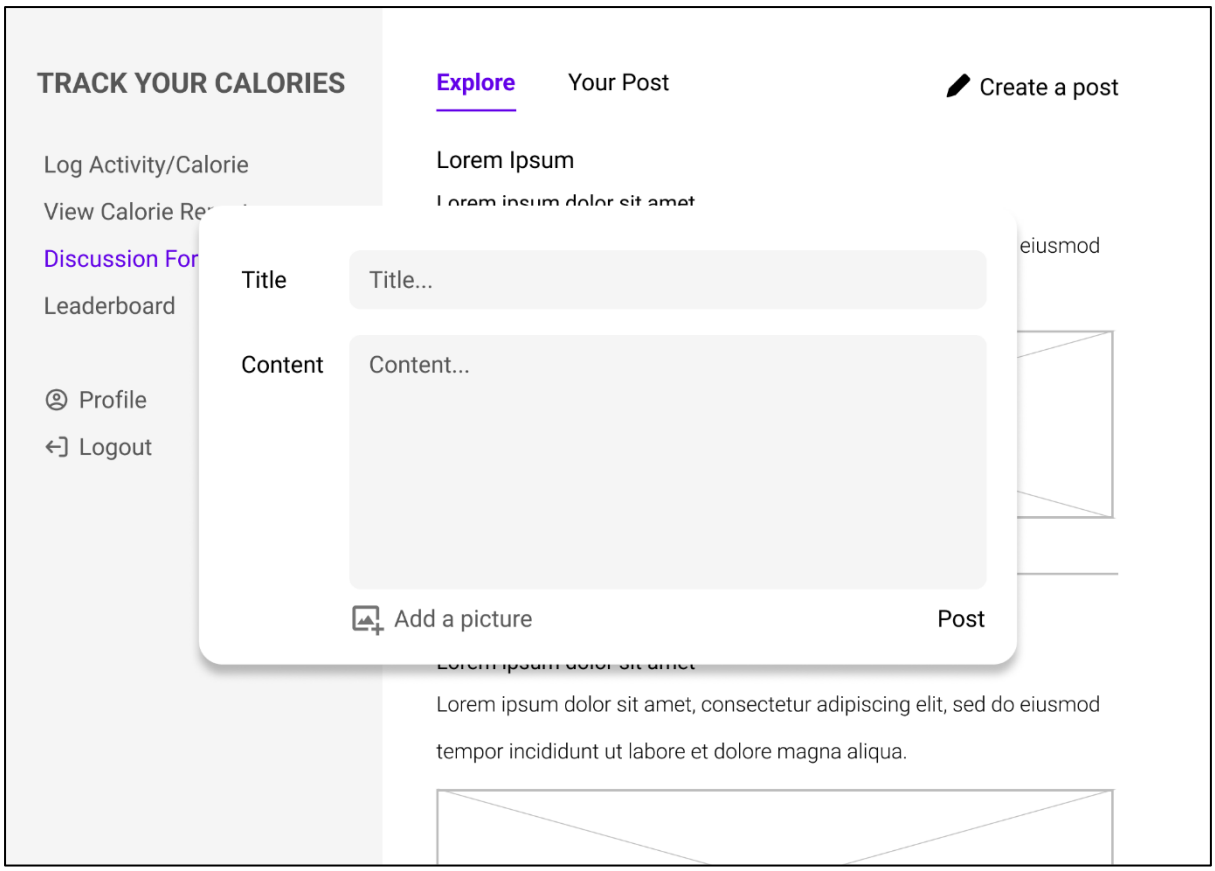


Figure 3.41: Create Post.

Figure 3.41 depicts the create post feature on the Discussion Forum page of the proposed system. This interface allows users to contribute new posts to the forum. A pop up is displayed, providing fields for entering a Title and Content for the post. Users can also attach image content by selecting the "Add a picture" option. Once all necessary information is entered, users can submit their post by clicking the Post button.

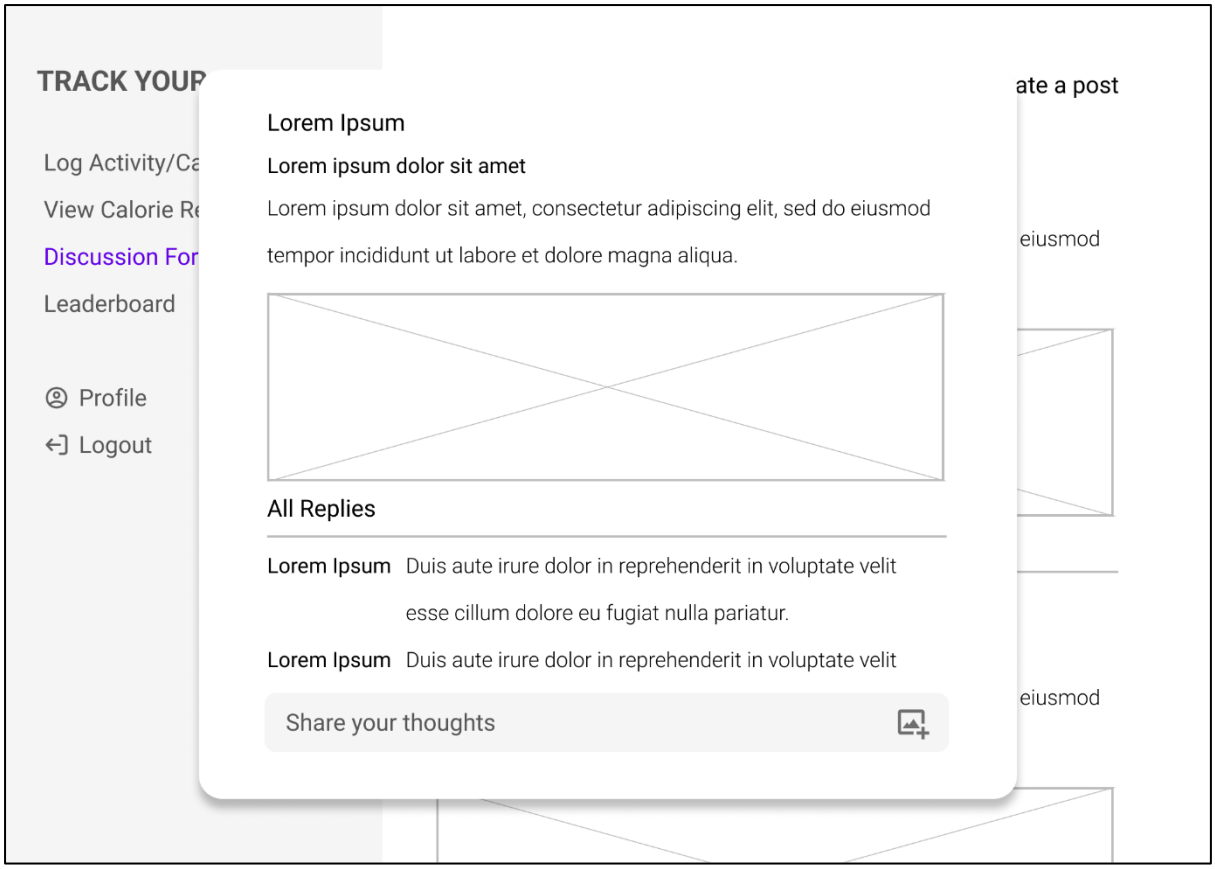


Figure 3.42: View Post and Reply.

Figure 3.42 depicts the view post and reply feature within the Discussion Forum page of the proposed system. The interface showcases a detailed view of a selected post, including its Title, Description, and an attached image. Below the post, a section titled All Replies displays responses from other users. Also, a reply field is provided with the option to attach image.

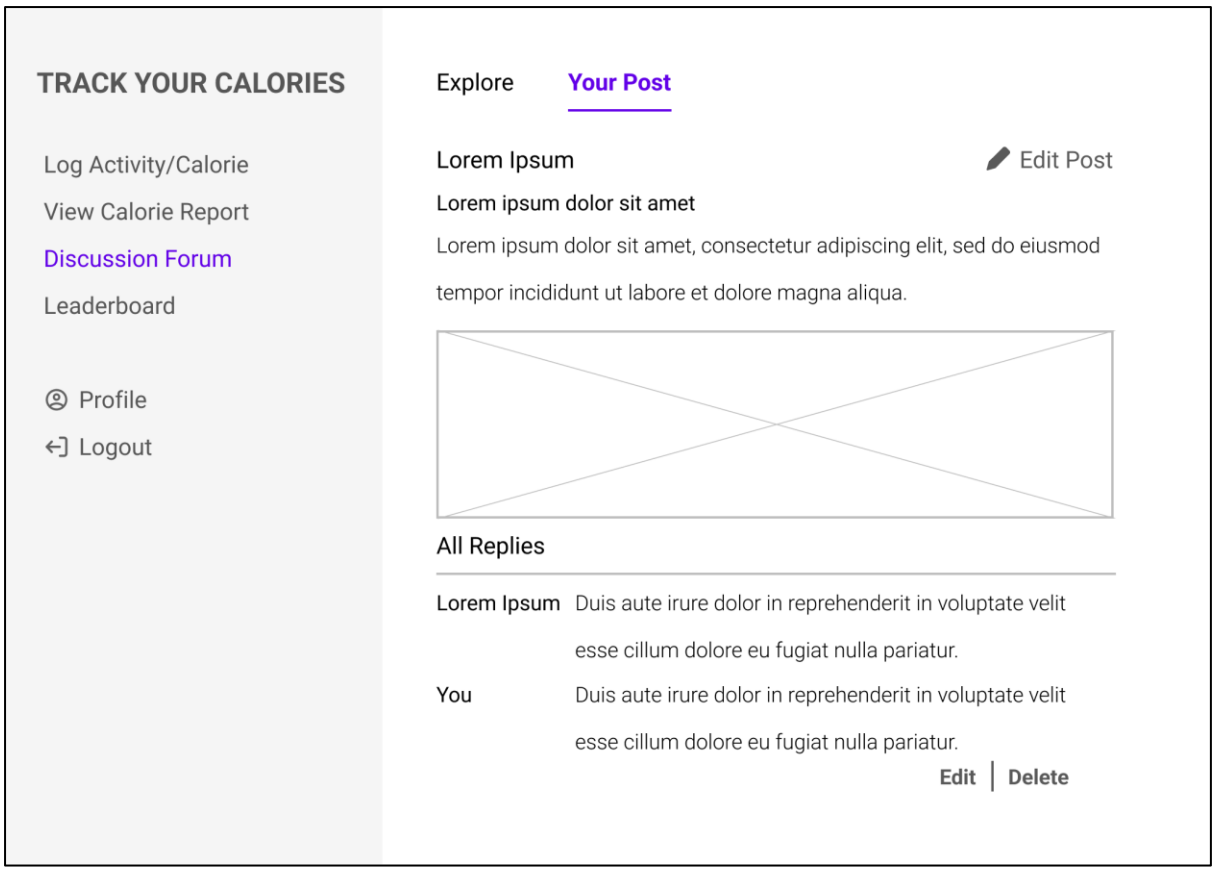


Figure 3.43: View Your Post and Reply Page.

Figure 3.43 shows the Your Post page within the Discussion Forum, where users can view their own post along with all replies. The figure shows that the users have the ability to edit or delete their post and replies they created.

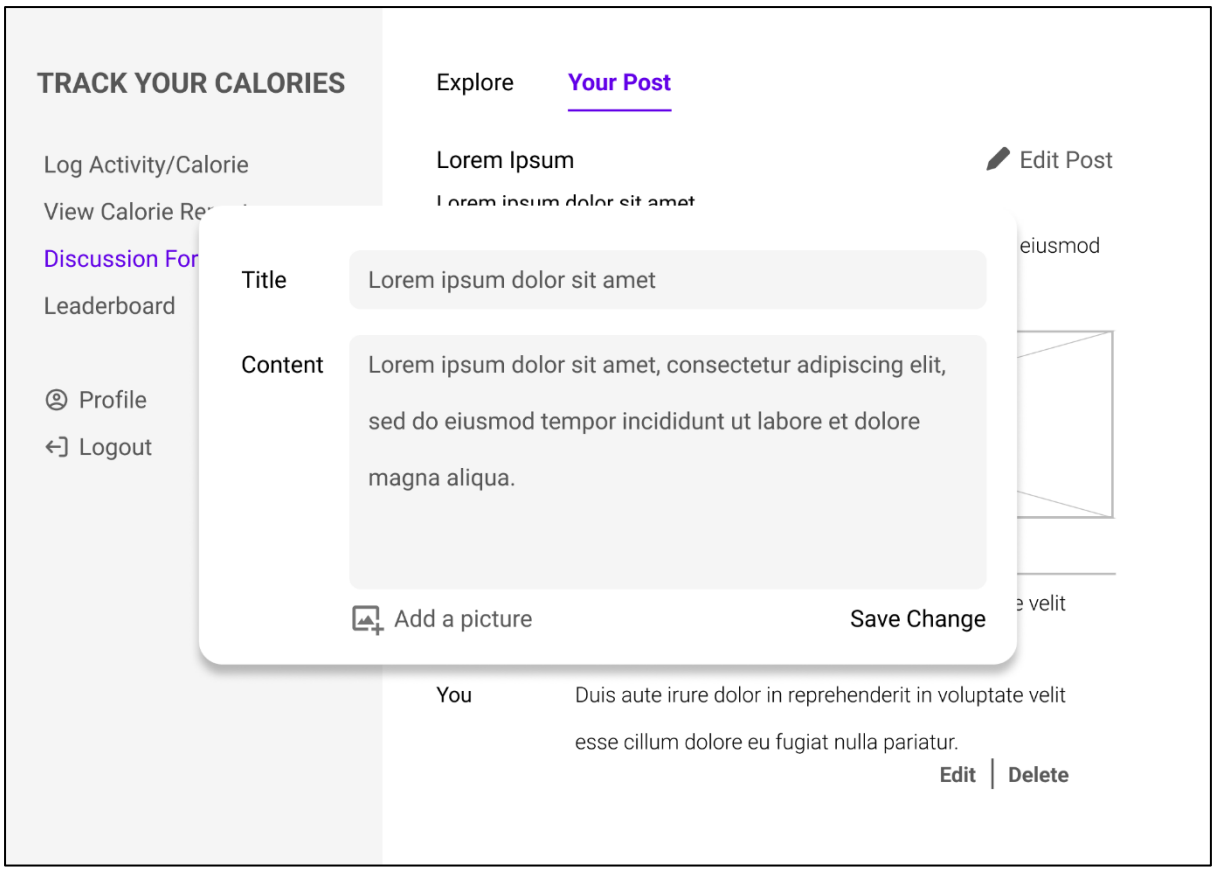


Figure 3.44: Edit Post.

Figure 3.44 shows the edit post feature, where users can modify the title, content, and attached image of their post. A pop up provides editable fields for these elements, along with an option to add a picture. Changes can be saved by clicking the Save Change button, allowing users to update their posts as needed.

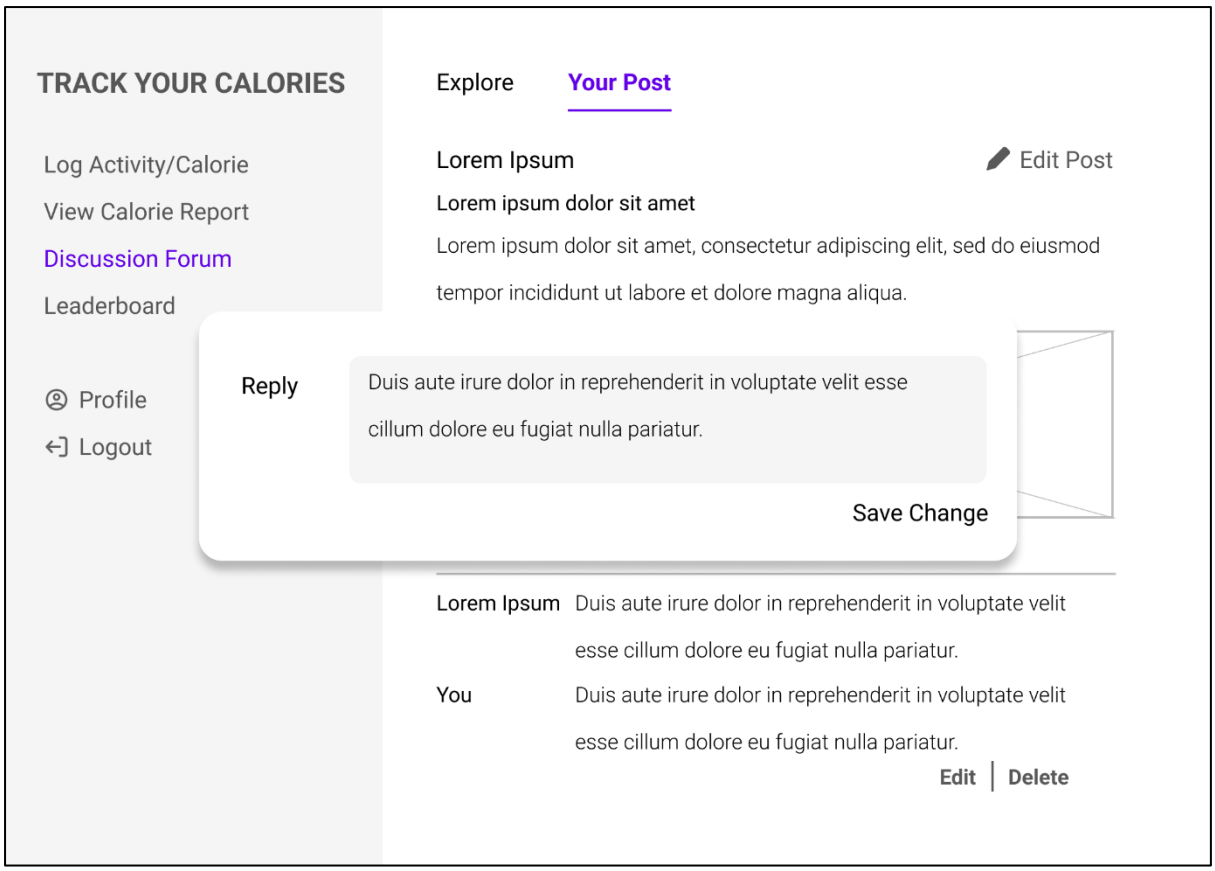


Figure 3.45: Edit Reply.

Figure 3.45 illustrates the edit reply feature, enabling users to update their responses to posts. A pop up displays an editable text field where users can revise the content of their reply. Once changes are made, users can save them by clicking the Save Change button.

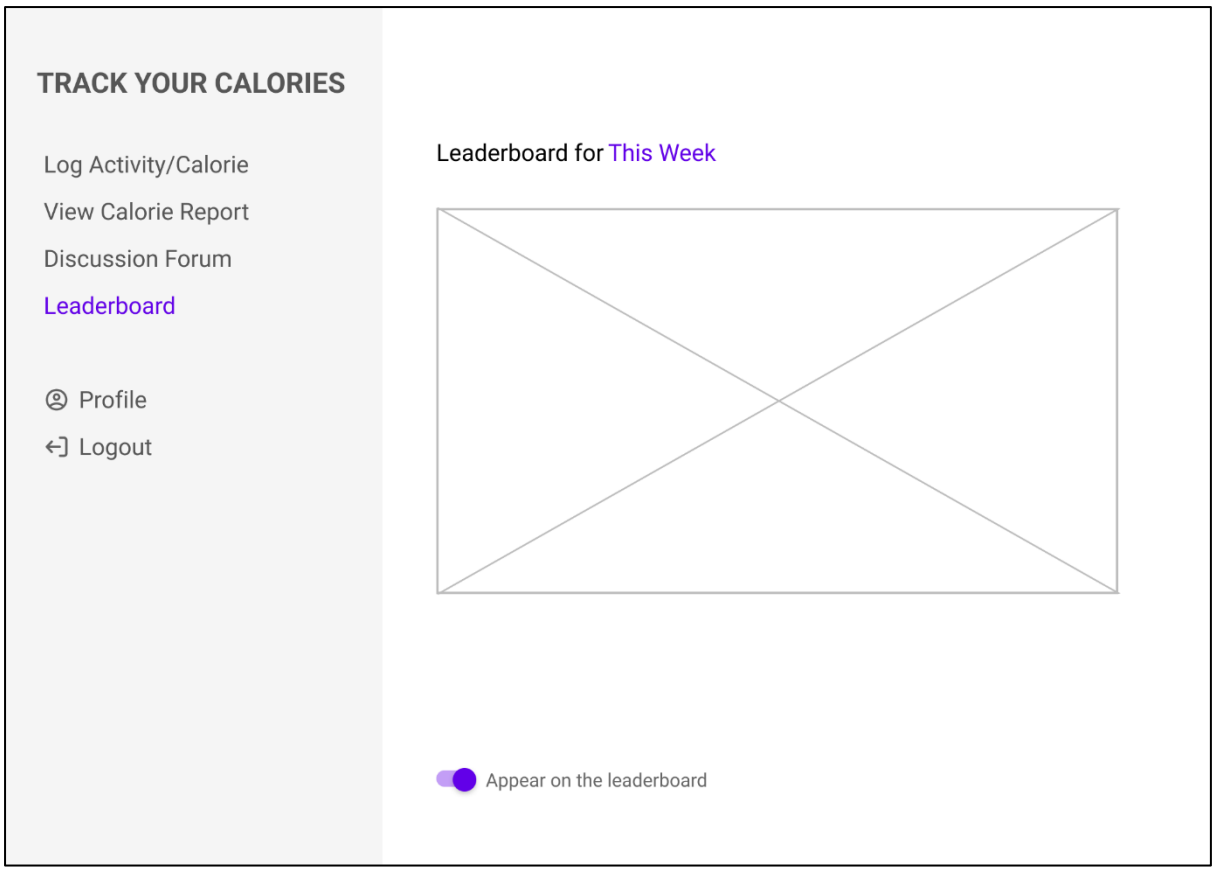


Figure 3.46: Leaderboard Page.

Figure 3.46 shows the Leaderboard Page. The content placeholder represents a ranked list of users based on their calorie-burning activities for the week. There is a toggle button "Appear on the leaderboard" provided for users to deselect if they do not wish to appear in the leaderboard, with visibility set to active by default. This page encourages users to track and compare their progress with others.

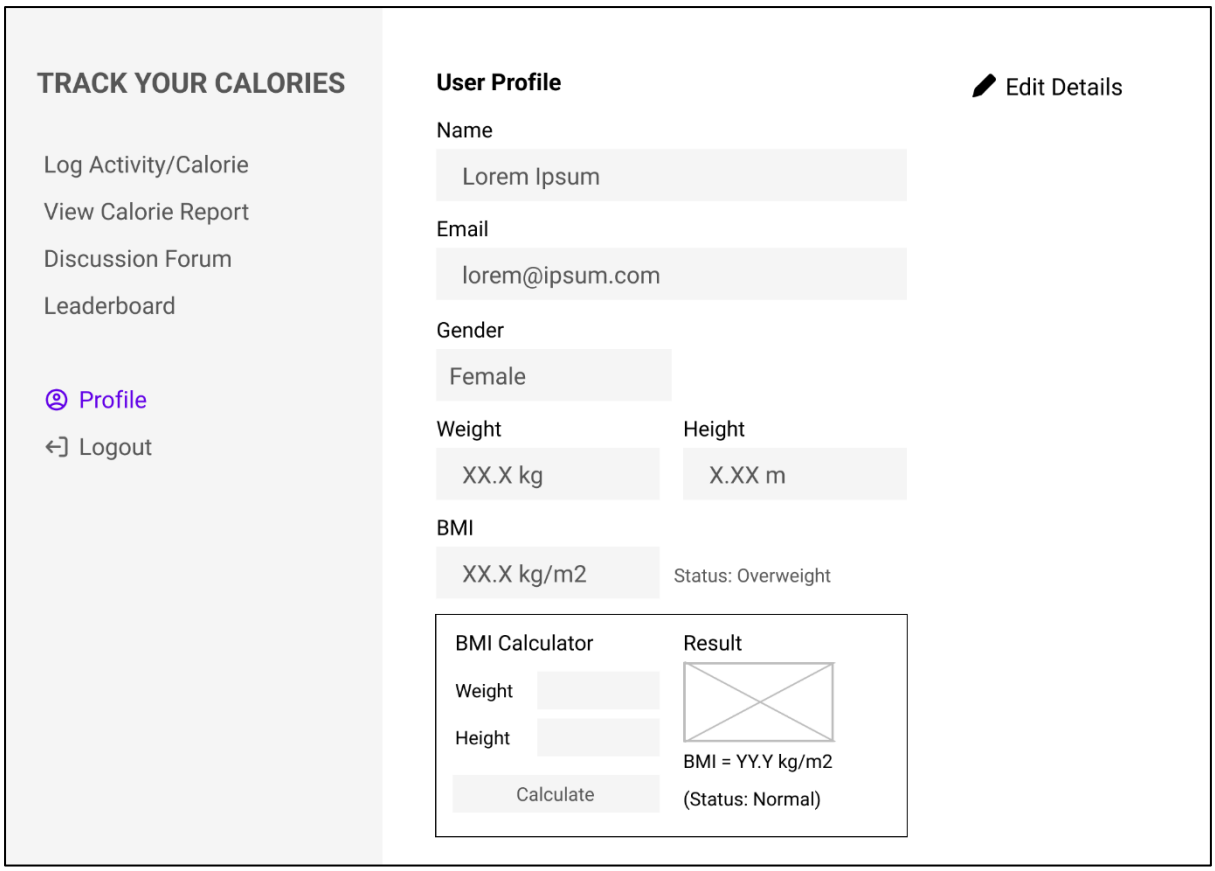


Figure 3.47: User Profile Page.

Figure 3.47 shows the User Profile Page, where users can view their personal information, including name, email, gender, weight, height, and BMI (Body Mass Index) along with its status (e.g., Normal or Overweight). The page features an Edit Details option for updating profile information. The BMI of the user will be shown based on the updated weight and height, the weight status categories will be updated based on the updated BMI. Additionally, another BMI Calculator is integrated, allowing users to input weight and height values, calculate their BMI, and view the corresponding weight status categories. This page provides a comprehensive overview and tools for users to manage and monitor their health-related data.

CHAPTER 4: IMPLEMENTATION

4.1 Introduction

This chapter details the implementation of the proposed system, building upon the design framework outlined in Chapter 3. Development is carried out using Visual Studio Code as the primary code editor and HeidiSQL for managing the MySQL database. Prior to implementation, these tools are set up to ensure a smooth development workflow. The chapter also showcases the outcomes of the system development through relevant screenshots, each accompanied by concise explanations to illustrate key functionalities and interface behaviour.

4.2 Development Tools

This section outlines the essential tools and technologies employed in building the proposed calorie tracking system. Each tool played a specific role in supporting system development, database management, machine learning integration, and overall project implementation.

4.2.1 Visual Studio Code

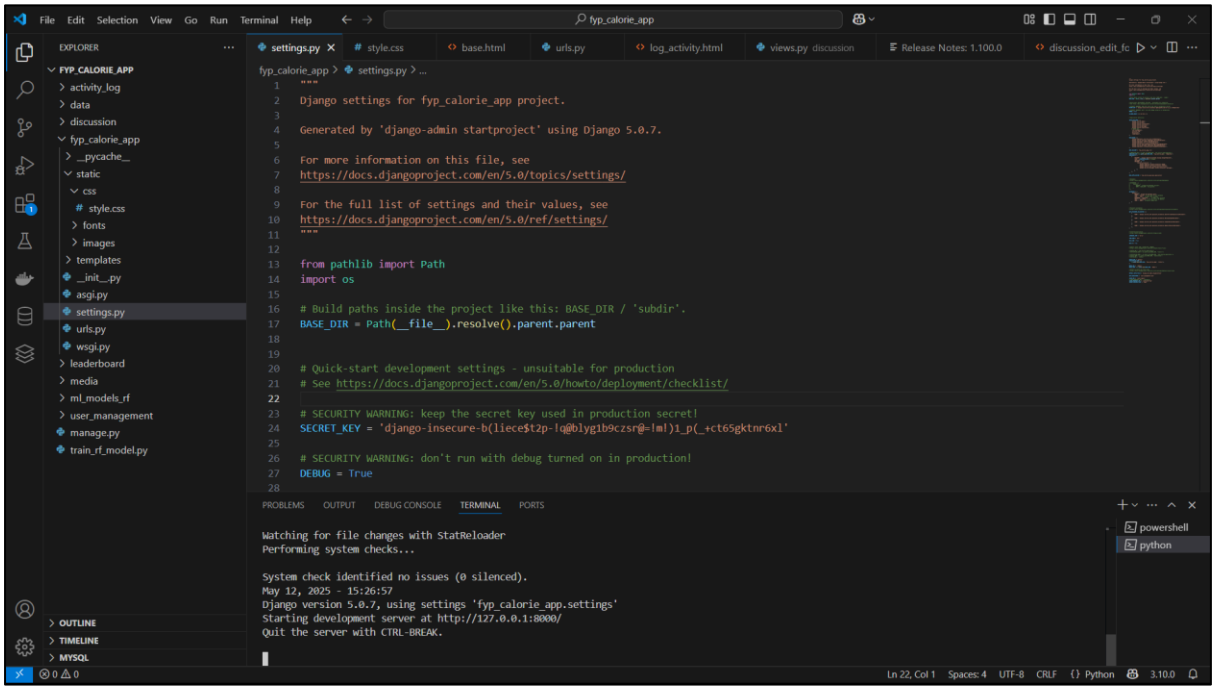


Figure 4.1: Visual Studio Code

Figure 4.1 displays the development environment within Visual Studio Code, actively used in implementing the proposed system. Visual Studio Code provides a powerful and developer-friendly interface for building Django-based applications. Its built-in terminal, file explorer, and syntax highlighting streamline the coding process, while features like real-time error detection significantly improve workflow efficiency. In this project, Visual Studio Code serves as the central workspace for managing multiple Django apps, template files, static assets, and machine learning scripts, all within a cohesive and organized structure.

4.2.2 HeidiSQL

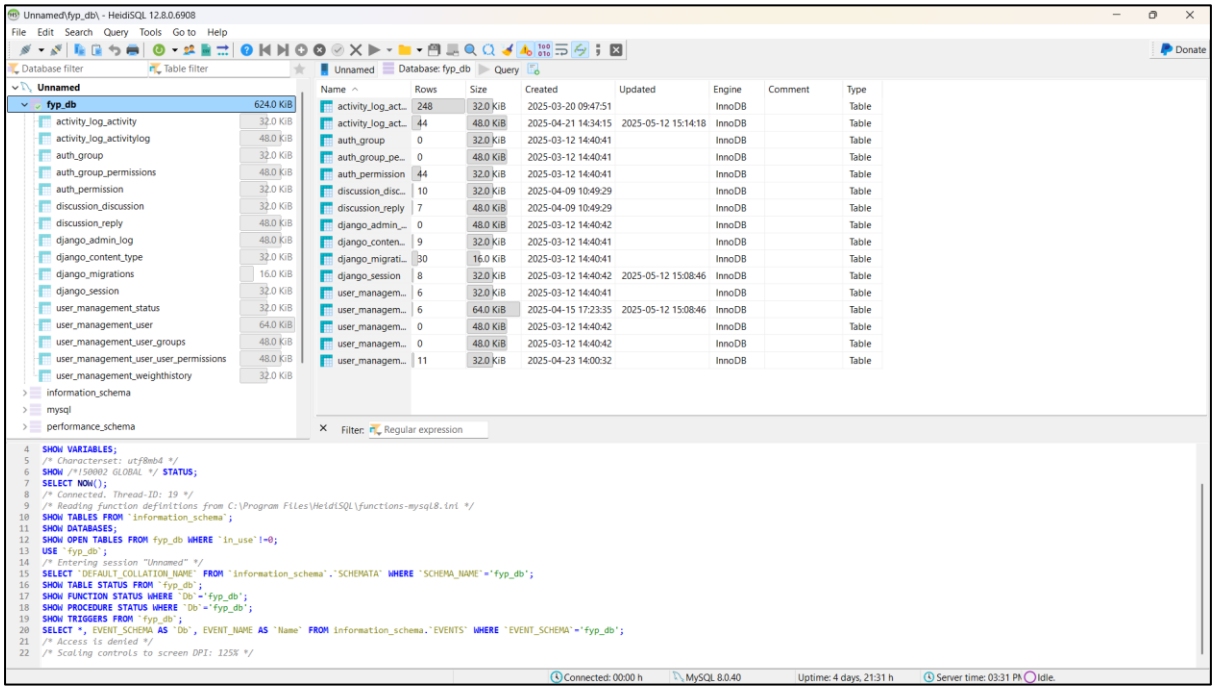


Figure 4.2: HeidiSQL

Figure 4.2 shows the database structure of the proposed system, managed using HeidiSQL. HeidiSQL serves as a lightweight and efficient graphical interface for interacting with the MySQL database. It allows for the visualization, creation, and manipulation of database tables, facilitating effective backend management throughout the development process. As shown in the figure, the database fyp_db contains multiple interrelated tables, including those for user management, activity logging, leaderboard tracking, and discussion modules. These tables are systematically designed to support key features of the calorie tracking system. HeidiSQL also supports direct SQL queries and table inspection, making it an essential tool for ensuring data consistency, verifying relationships, and debugging database-related functionalities during development.

4.2.3 Kaggle

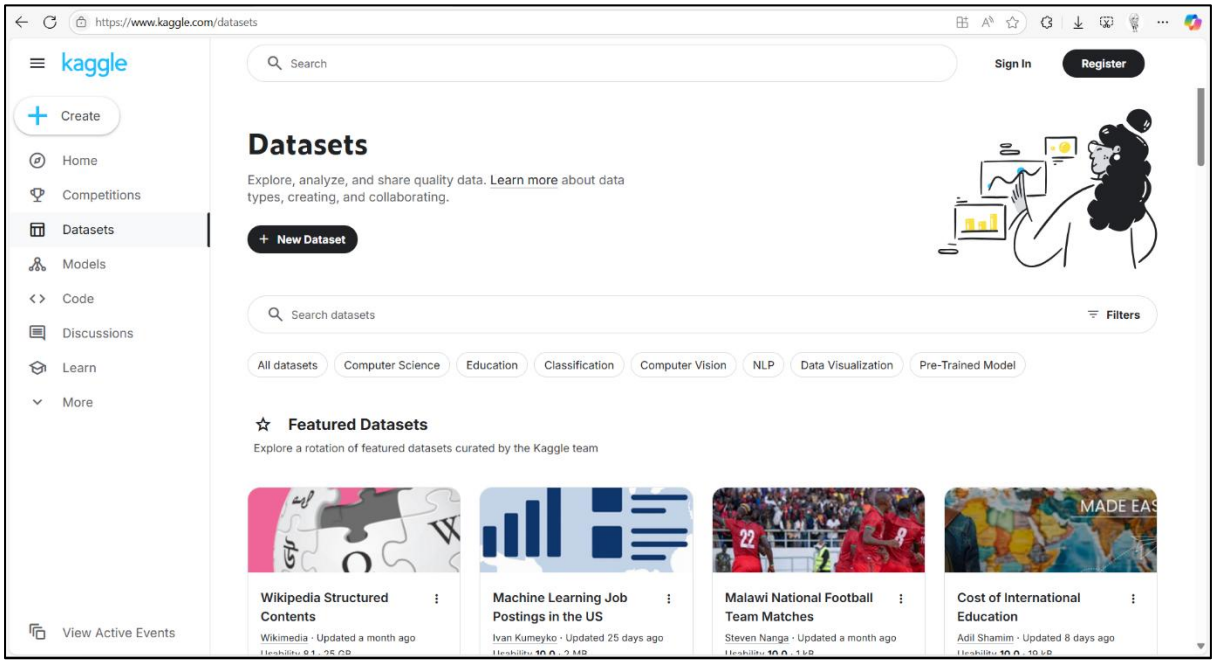


Figure 4.3: Kaggle

Kaggle was used as the primary source for collecting datasets related to activity-based calorie estimation. Kaggle is a public platform that hosts a wide range of datasets across various domains. Through the platform's search and filtering features, relevant datasets containing physical activity records and corresponding calorie burn data were identified. These datasets were used in the development and training of the calorie estimation module integrated into the system. The ability to access and analyze structured datasets from Kaggle supported the data-driven component of this project.

4.2.4 XGBoost

XGBoost (Extreme Gradient Boosting) was used as the core machine learning algorithm for estimating calorie consumption based on user activity data. XGBoost is known for its speed, accuracy, and efficiency in handling structured data, making it highly suitable for regression tasks like calorie prediction. Its ability to perform well even with limited data,

handle missing values, and apply regularization to prevent overfitting makes it a reliable choice for real-world applications.

In this project, XGBoost was chosen because it outperforms many traditional algorithms in terms of both prediction accuracy and training time. It also provides insights through feature importance scores, helping to identify key factors that influence calorie burn, such as activity type, duration, and user attributes. The model was trained using pre-processed datasets and integrated into the Django backend, allowing real-time calorie predictions based on user input. Its high efficiency and interpretability contribute significantly to the overall performance and reliability of the proposed system.

4.2.5 Django

The proposed system was developed using Django, a high-level Python web framework that promotes rapid development and clean design. Django was selected due to its built-in features such as an object-relational mapper (ORM), user authentication, URL routing, and admin interface, which streamlined the backend development process. The framework follows the Model-View-Template (MVT) architecture, allowing for a clear separation of logic, data, and presentation layers. This structure enabled efficient management of user data, activity records, and integration of the machine learning model used for calorie estimation. Django's modular approach also facilitated the creation of reusable apps within the system, including modules for user management, activity logging, and leaderboard functionality.

4.3 User Interface Design

This section introduces the core features and functionalities of the calorie tracking system. To provide clearer insight, the web-based user interfaces are included and referenced throughout the explanation. Each interface is discussed in detail to illustrate how users interact with the system and how various features are accessed and utilized.

4.3.1 User Login

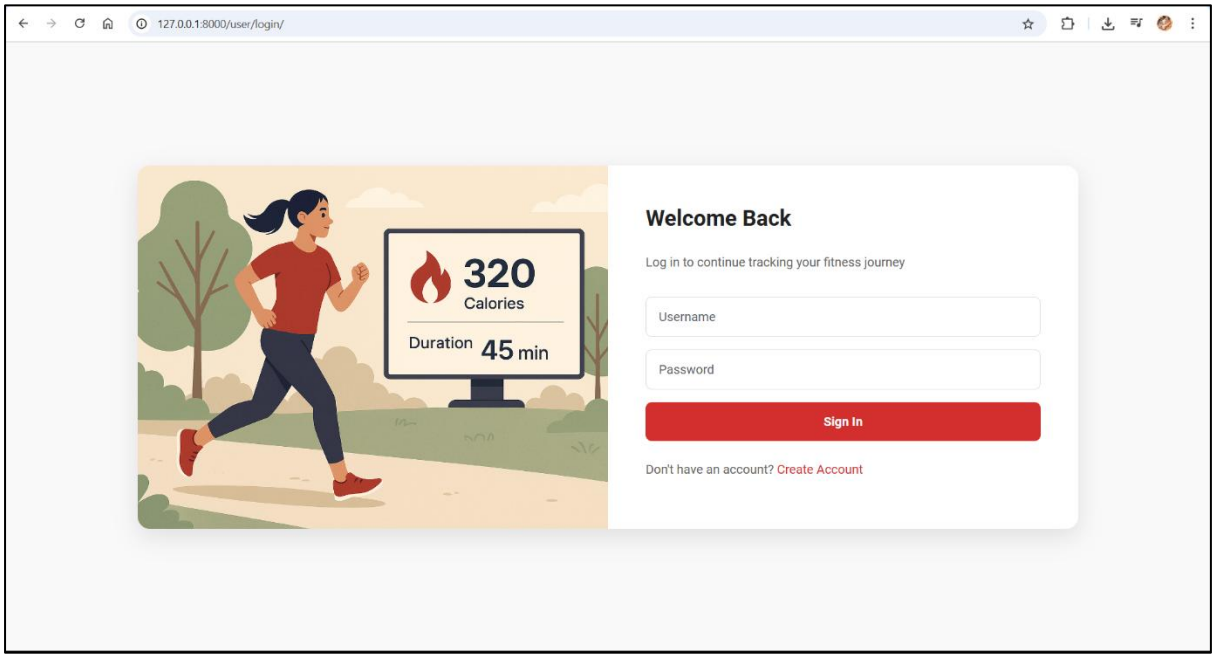


Figure 4.4: Login Page

Figure 4.4 shows the login page of the system. Users are required to enter their username and password in the form provided on the right. After filling in the credentials, clicking the "Sign In" button will authenticate the user and grant access to the system. A registration link is also provided for new users.

4.3.2 User Registration

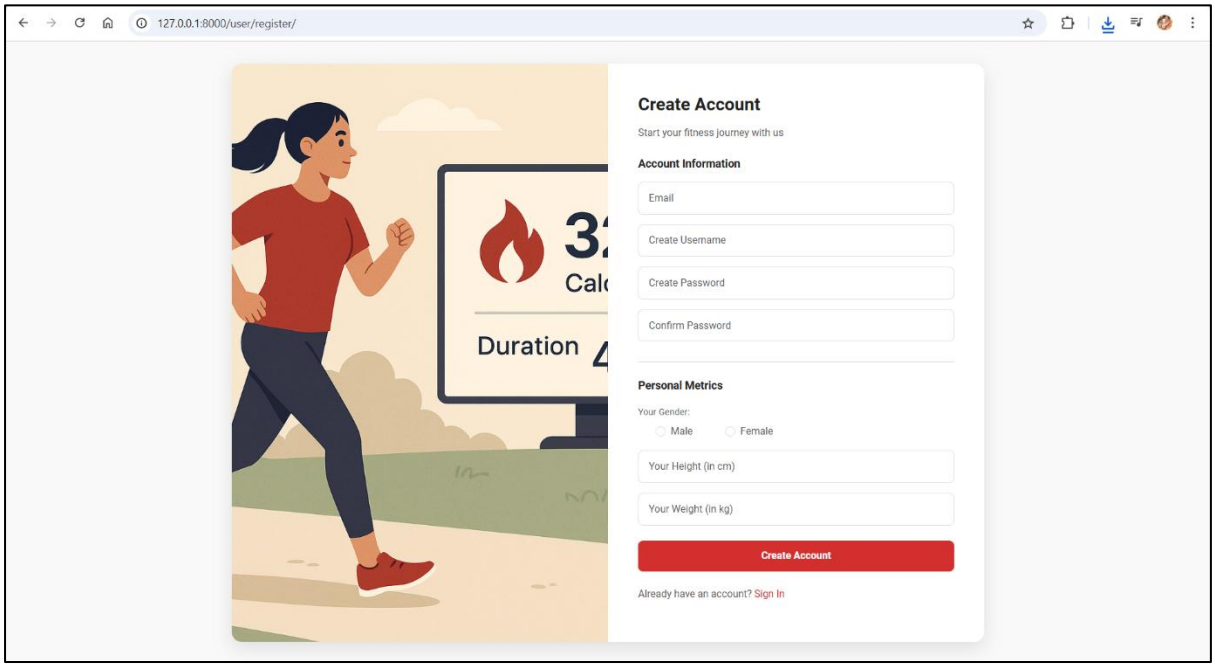


Figure 4.5: Registration Page

Figure 4.5 displays the registration page. New users can fill in the form on the right to create an account by providing email, username, password, gender, height, and weight. This information is used for calorie estimations and BMI calculations. The left section of the page contains a static image for visual consistency with the login screen. A “Create Account” button submits the form, and an alternate link is available to sign in.

4.3.3 User Profile

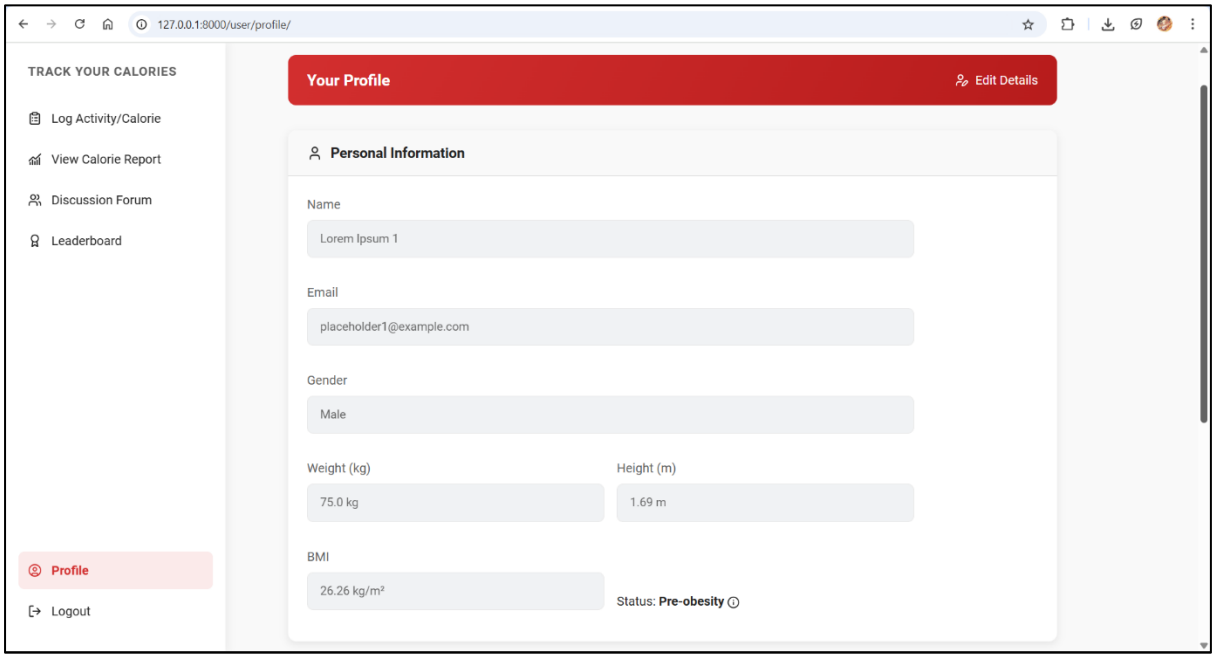


Figure 4.6: User Profile Page

Figure 4.6 shows the user profile page. It displays the user's name, email, gender, height, weight, and BMI. The data is presented in a non-editable format. Users can update this information by clicking the "Edit Details" button. The BMI value is calculated automatically based on the stored height and weight, and a corresponding status is displayed to indicate the user's BMI classification.

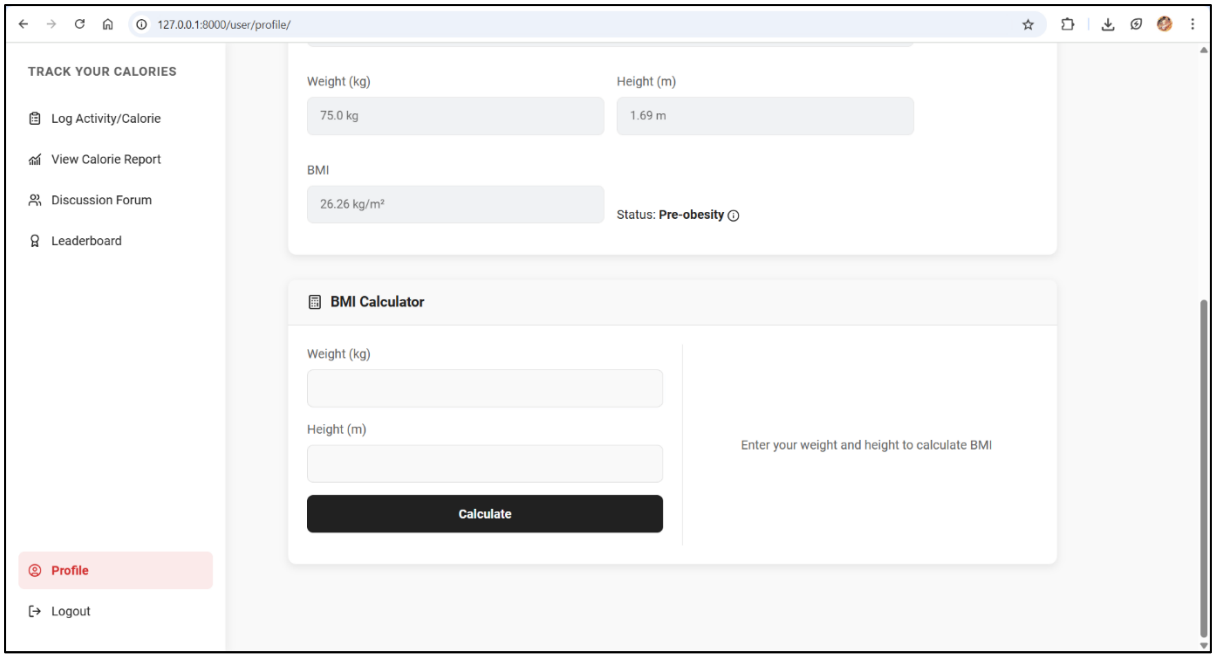


Figure 4.7: BMI Calculator (Initial View)

Figure 4.7 presents the initial view of the BMI calculator below the profile section. Users can enter their current weight and height to calculate BMI without changing their stored profile data.

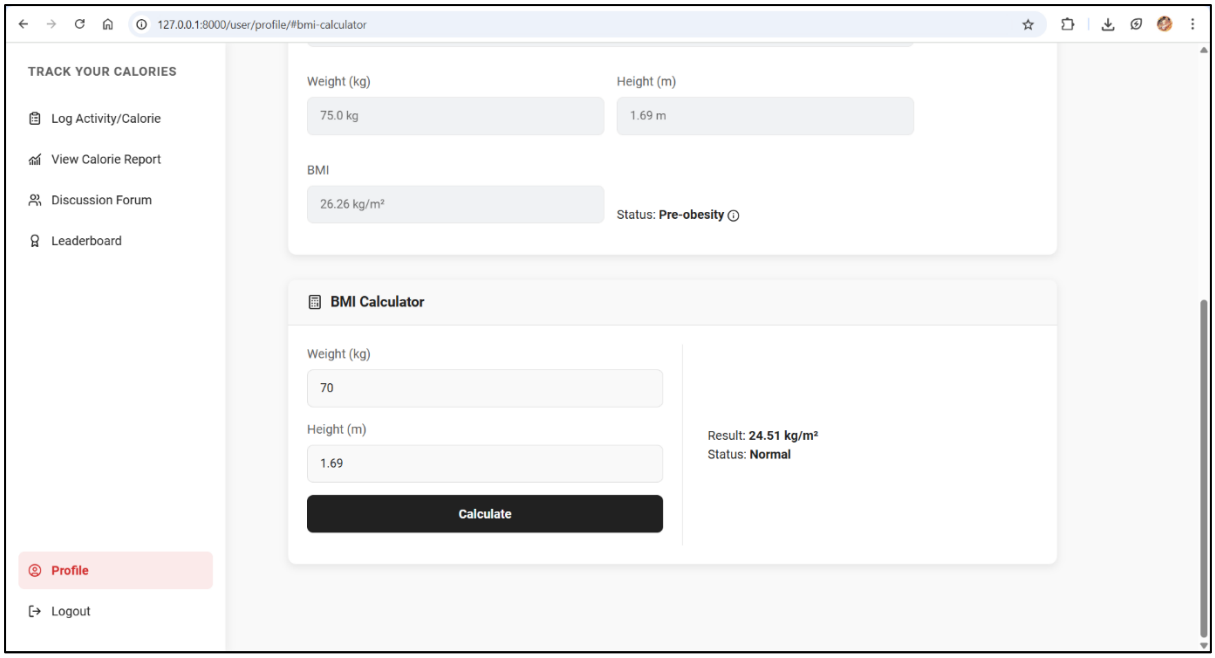


Figure 4.8: BMI Calculator (With Result)

Figure 4.8 shows the BMI calculator after a calculation is completed. The BMI result and the corresponding category (e.g., normal, overweight) are displayed below the input fields.

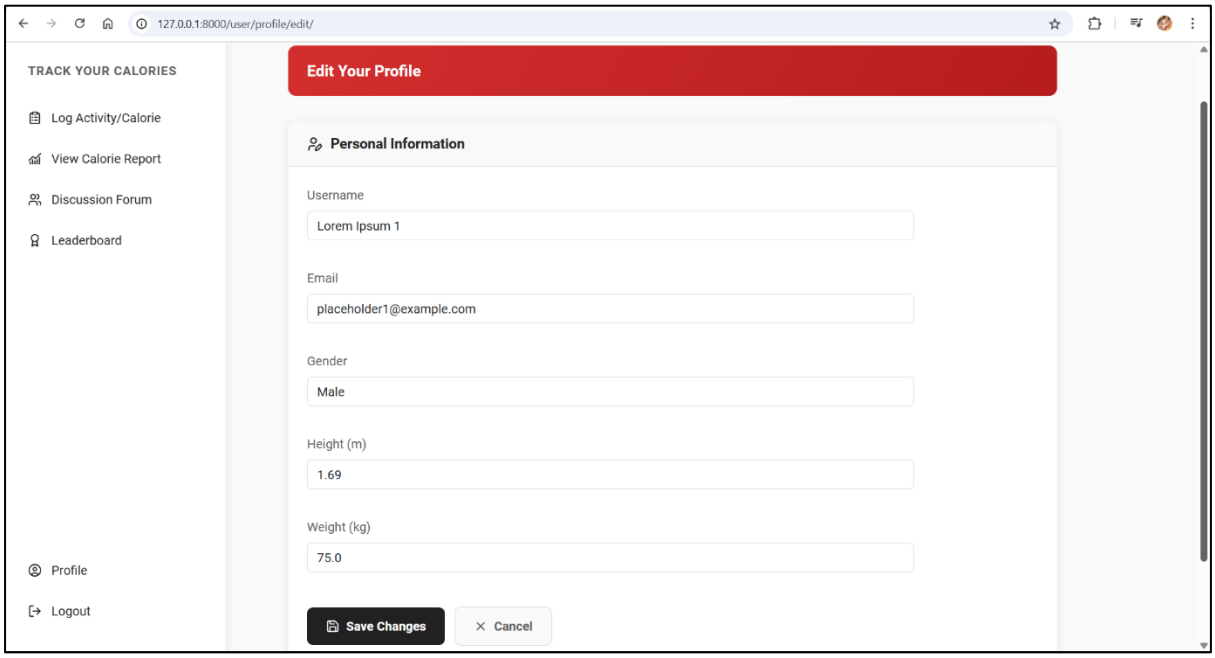


Figure 4.9: Edit Profile Page

Figure 4.9 shows the Edit Profile page. Here, users can modify their personal details, including email, gender, height, and weight. After editing, they can click "Save Changes" to

update the information or "Cancel" to discard the changes. Updates made here affect the BMI calculation and any related health metrics used in the system.

4.3.4 Log Activity

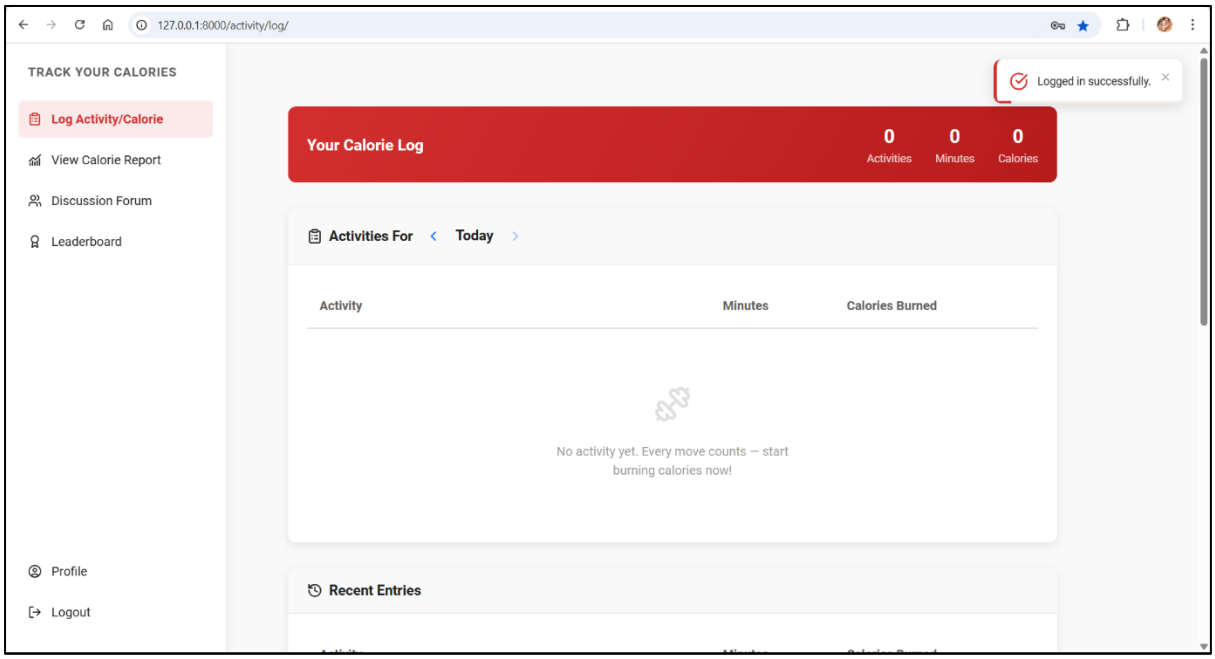


Figure 4.10: Log Activity Page (Empty State)

Figure 4.10 shows the Log Activity page immediately after login, with no activities recorded. The summary banner at the top indicates zero activities, minutes, and calories. A message prompts the user to start logging their activities for the day.

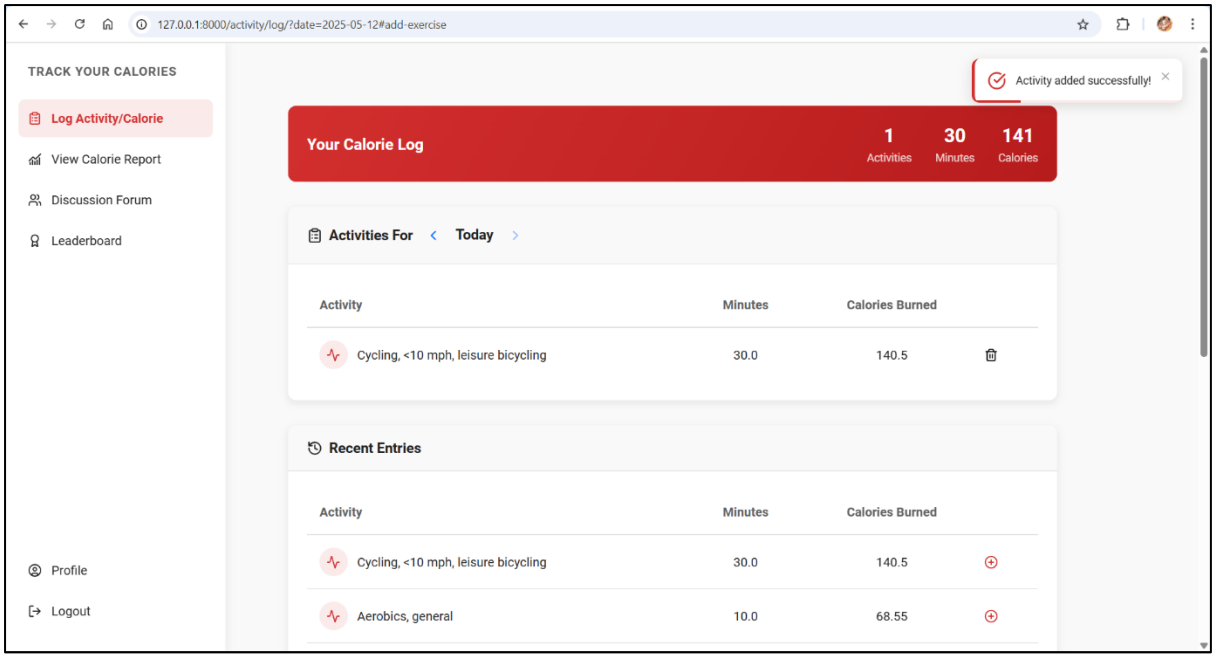


Figure 4.11: Log Activity Page (Activity Recorded)

Figure 4.11 displays the Log Activity page after a user has logged an activity. The top banner is updated with the total number of activities, time spent, and calories burned. Logged entries are listed with activity names, durations, and calorie estimates. Each entry includes a delete icon, and a toast message confirms successful logging.

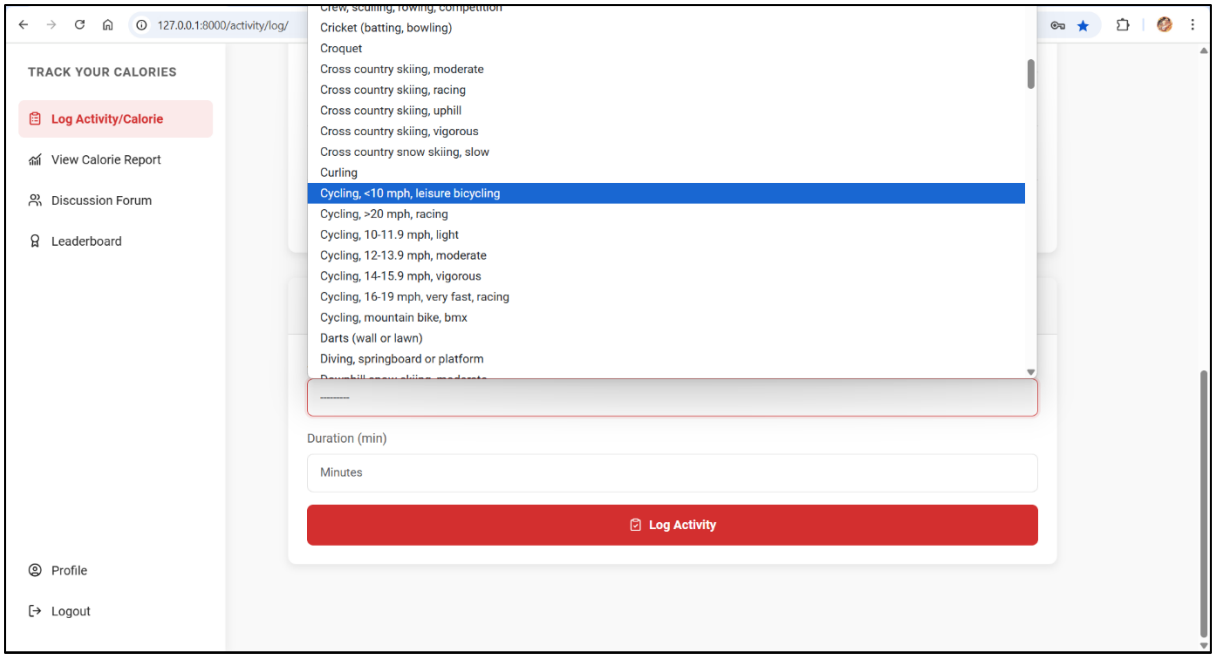


Figure 4.12: Log Activity Page (Log Activity Option)

Figure 4.12 shows the dropdown menu for selecting an activity type. The options include various categories such as sports, walking, running, and more. Users must select an activity before logging into it for a specified duration. This ensures consistency in activity categorization and enables accurate calorie estimation during the logging process.

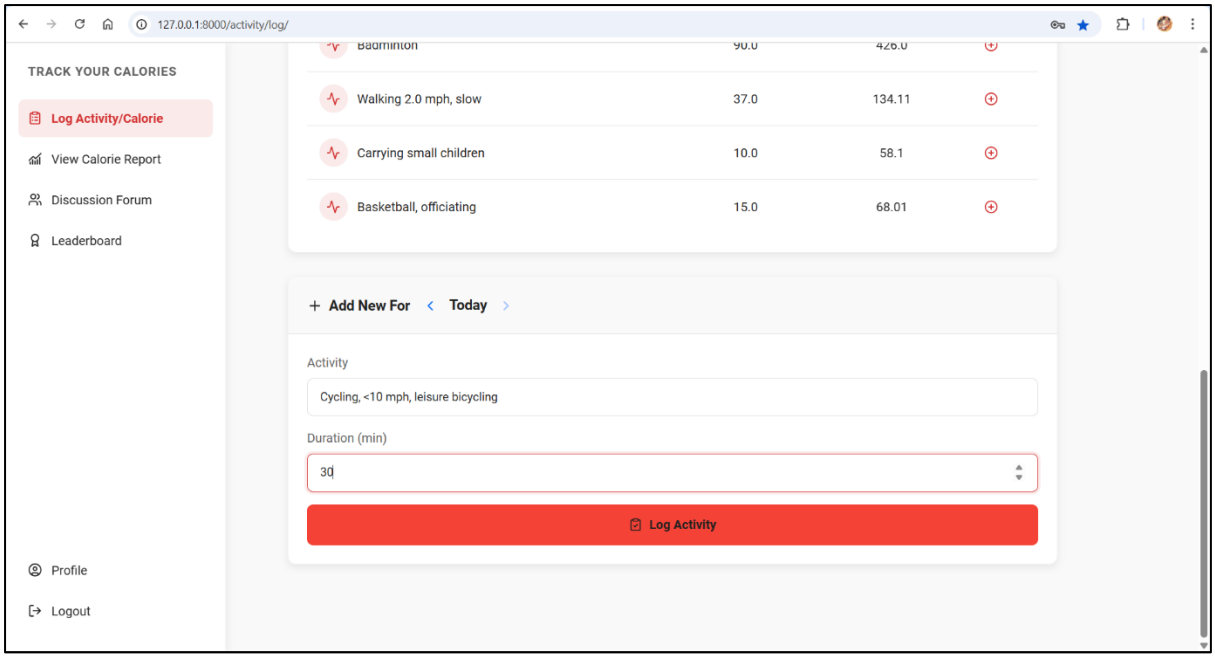


Figure 4.13: Log Activity Page (Log Activity)

Figure 4.13 presents the log activity form. Users select an activity, enter the duration in minutes, and click "Log Activity." The system then estimates calories burned using the built-in machine learning model.

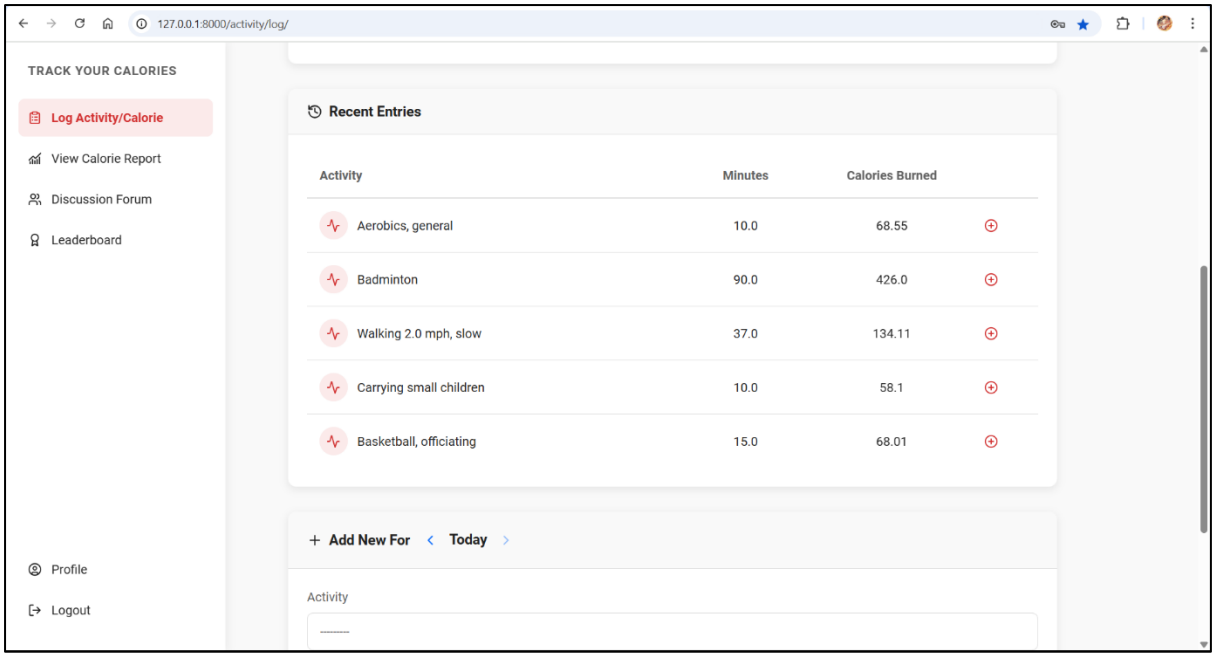


Figure 4.14: Log Activity Page (Recent Activities)

Figure 4.14 displays the "Recent Entries" section. Previously logged activities were listed with a plus icon, allowing users to quickly re-add them to the current day's log. This feature promotes convenience and encourages consistency in daily tracking by simplifying repeated entries.

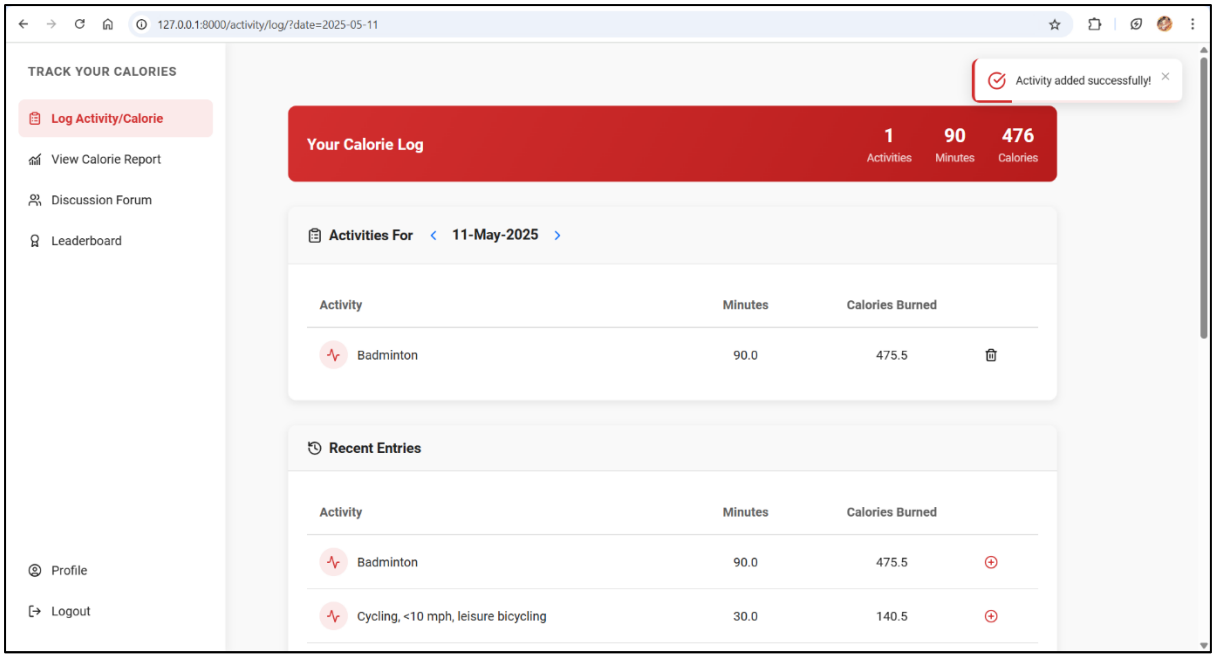


Figure 4.15: Activity Log (Past Date)

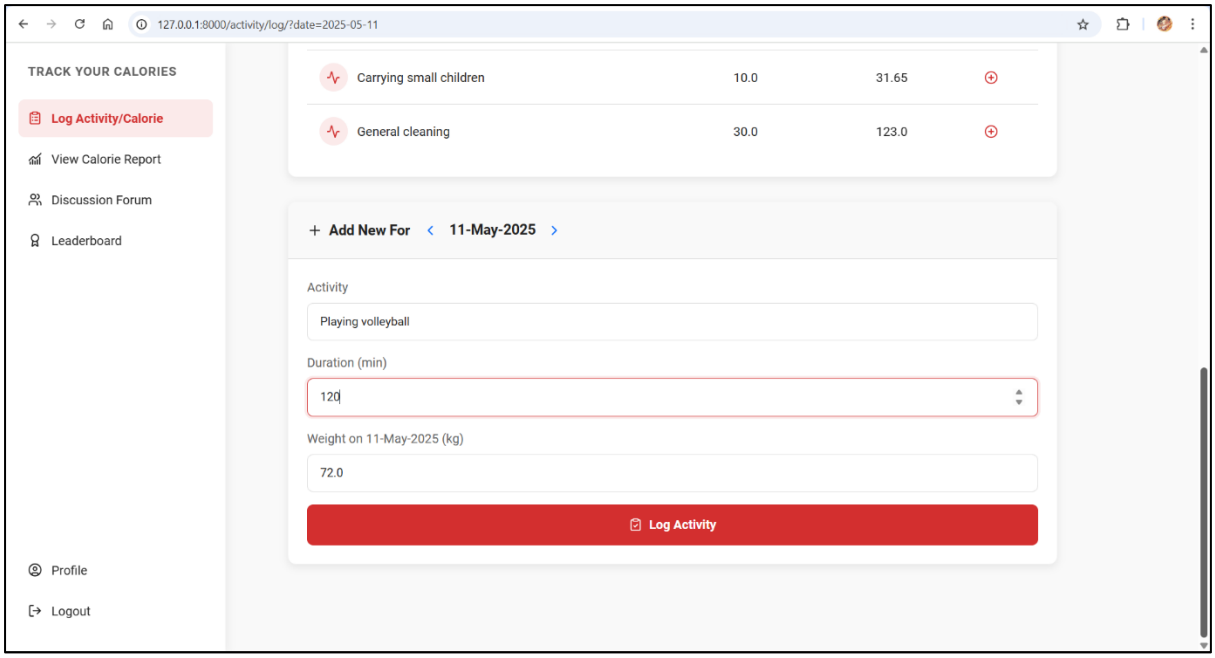


Figure 4.16: Log Activity Page (Log Activity for Past Date)

Figure 4.15 and *Figure 4.16* show how users log activities for a past date. Users select the desired date from a calendar, then input the activity type, duration, and body weight for that date. The system calculates the calories burned and adds the entry to that specific day's log.

This feature supports accurate historical tracking by allowing users to maintain a complete and dated record of their physical activity over time.

4.3.5 Calorie Report

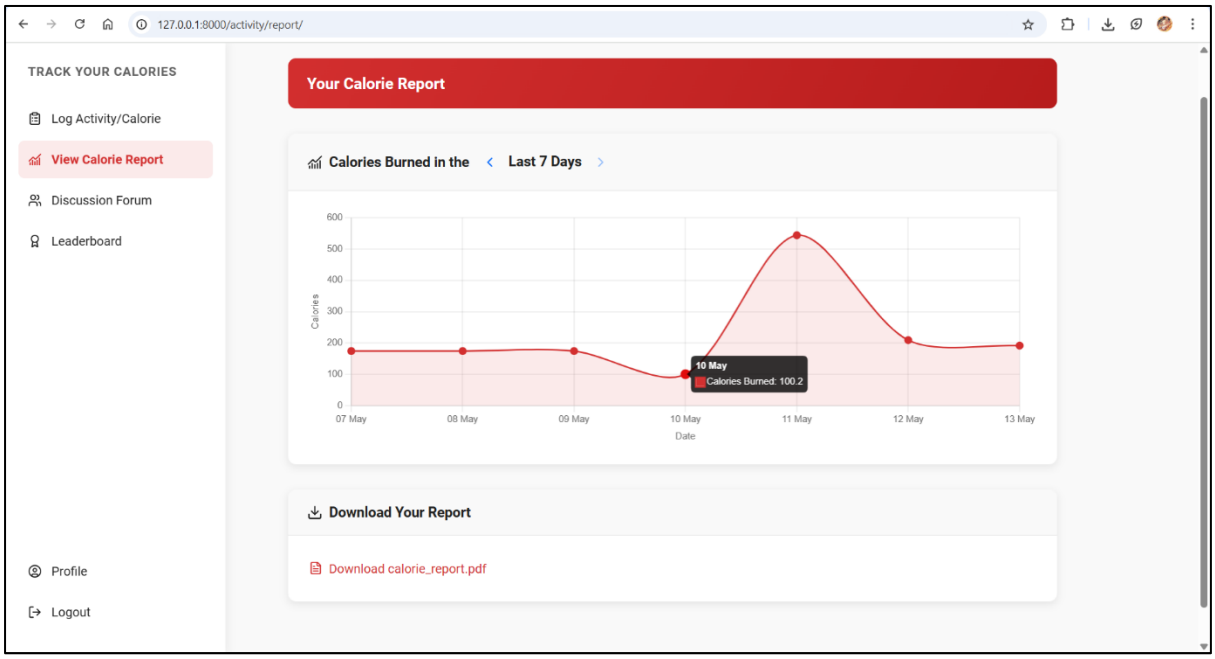


Figure 4.17: Calorie Report (Last 7 Days)

This interface presents a graphical summary of calories burned over the past 7 days. A line chart displays daily calorie totals, with each data point representing the total calories burned on a specific day. Users can hover over each point to view the exact value. This feature allows users to monitor short-term trends in their physical activity and calorie expenditure. Below the chart, a download button is provided to generate and retrieve the report in PDF format for record-keeping or external review.

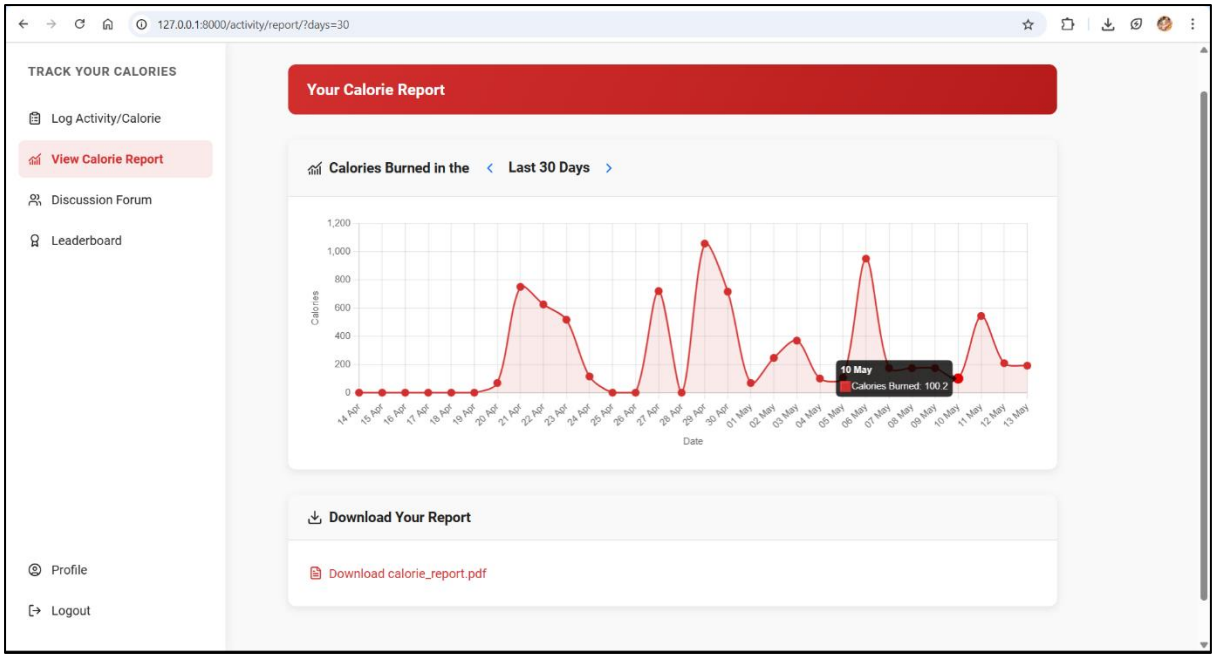


Figure 4.18: Calorie Report (Last 30 Days)

Figure 4.18 extends the same report to cover 30 days range. The longer timeframe helps users identify broader trends and fluctuations. Data is plotted by day, and the export option is still available.

4.3.6 Discussion Forum

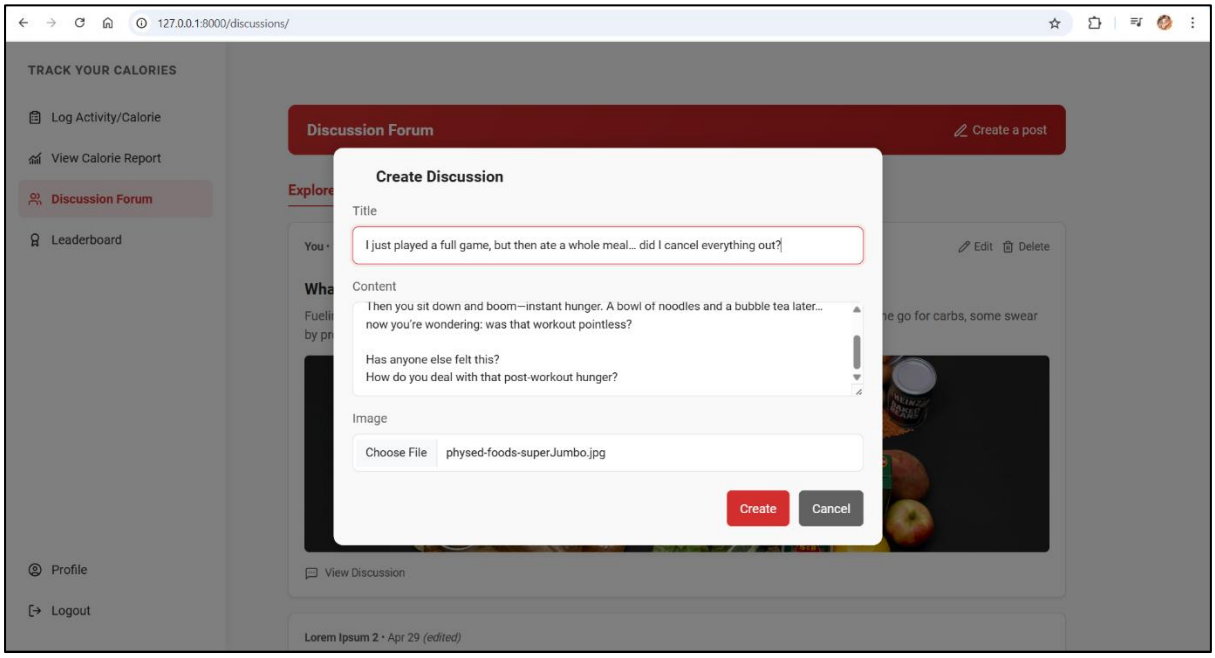


Figure 4.19: Create Discussion Modal

Figure 4.19 shows the Create Discussion modal. When users click “Create a post,” a modal appears where they can enter a title, content, and optionally attach an image.

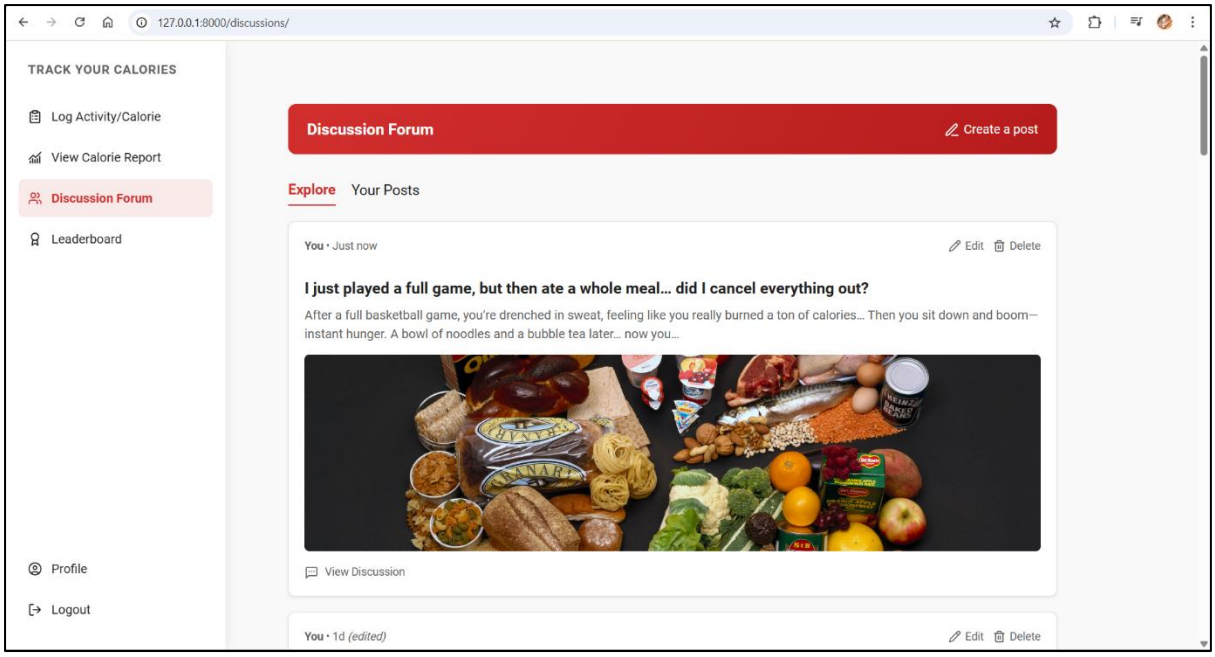


Figure 4.20: Discussion Post Created in Explore Tab

Figure 4.20 shows the post immediately after it is published. The post appears in the "Explore" tab with the title, content, and image. If it's the user's post, edit and delete options are visible, and the user is labeled as "You."

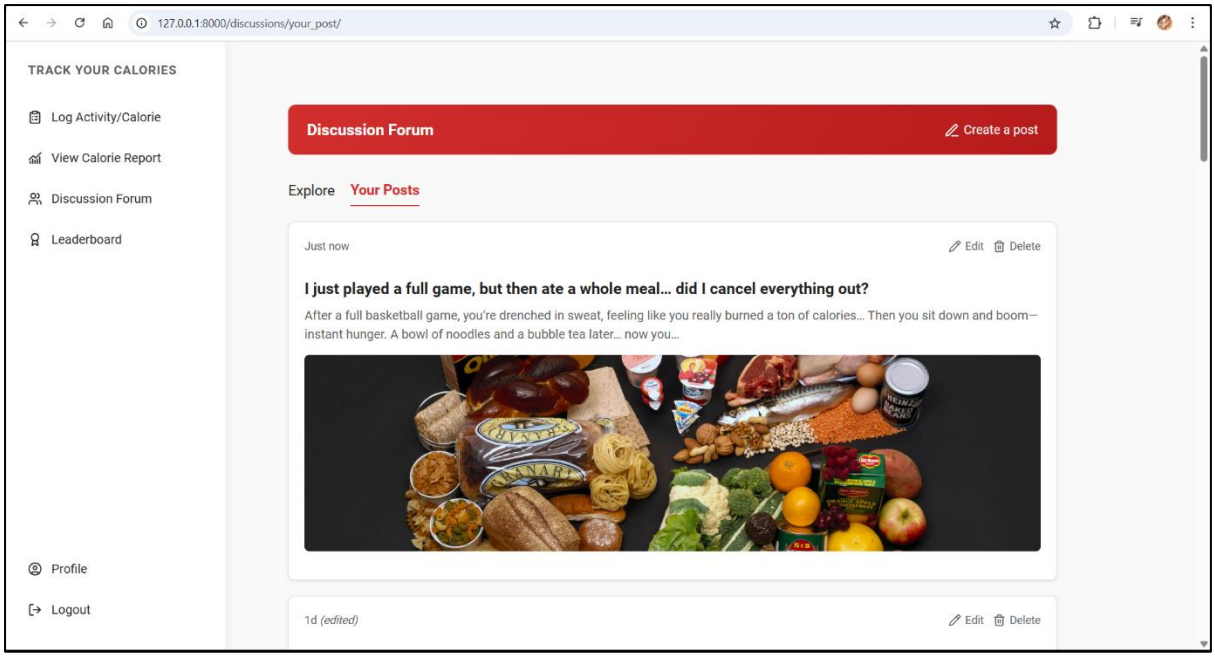


Figure 4.21: Discussion Post Appearing in 'Your Posts' Tab

Figure 4.21 displays the same post under the "Your Posts" tab, which serves as a dedicated space for users to manage their own discussions. Here, users can keep track of all their contributions, view replies from others, and perform further actions such as editing or removing a post.

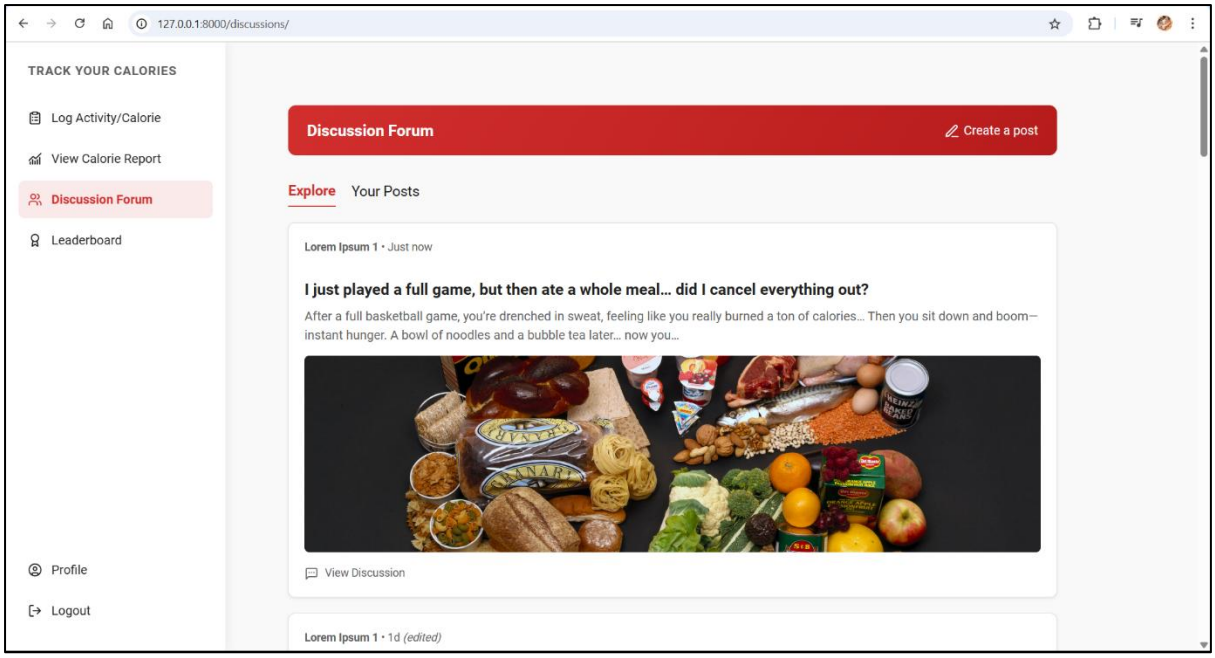


Figure 4.22: Post Seen by Another User in Explore Tab

Figure 4.22 shows how other users see the post in the "Explore" tab. They can read the post, view the author's name, and click the comment button. Edit/delete options are not visible to them.

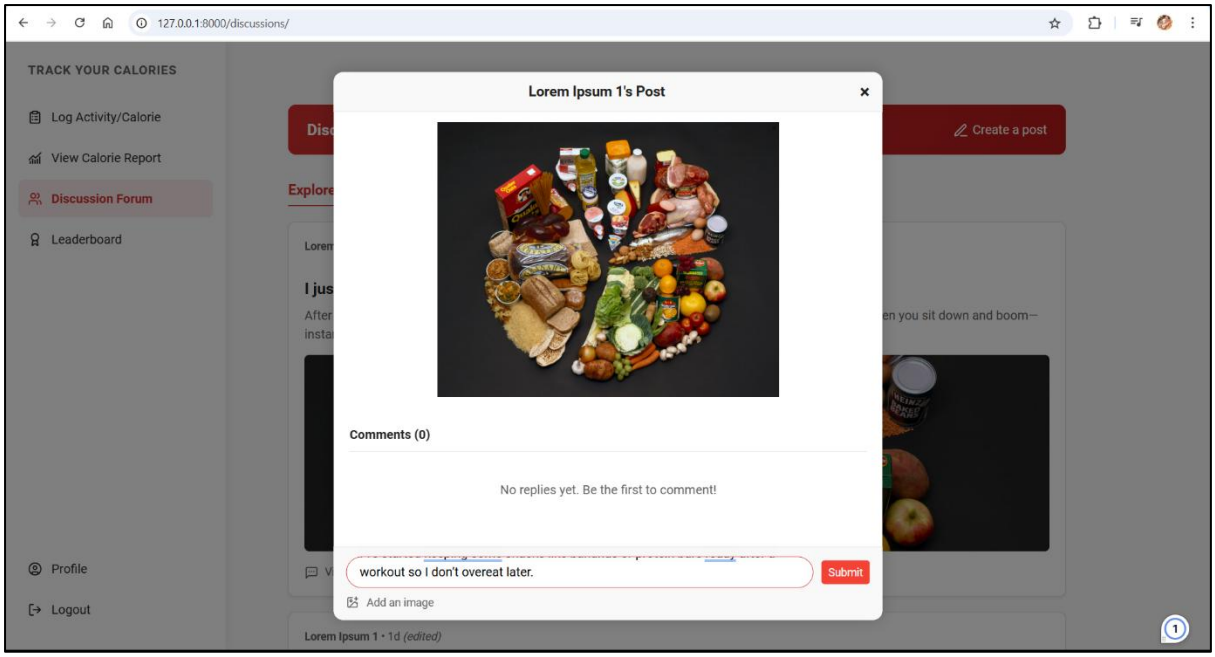


Figure 4.23: Viewing into a Discussion Post

Figure 4.23 displays the full discussion view after a post is clicked. Users see the complete content, any attached image, and a comment section. The author's name and edit history are also shown.

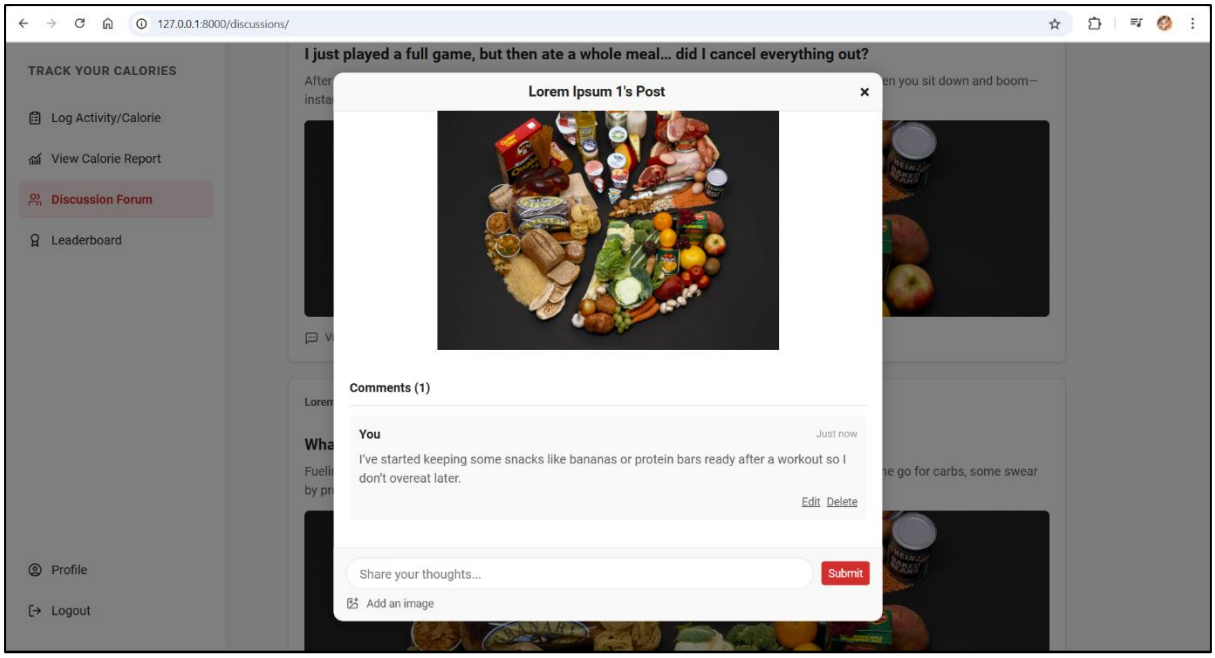


Figure 4.24: Comment Submitted on a Discussion Post

Figure 4.24 shows how a comment looks after being submitted. The reply includes the commenter's name, timestamp, and image if any. The commenter can edit or delete their reply.

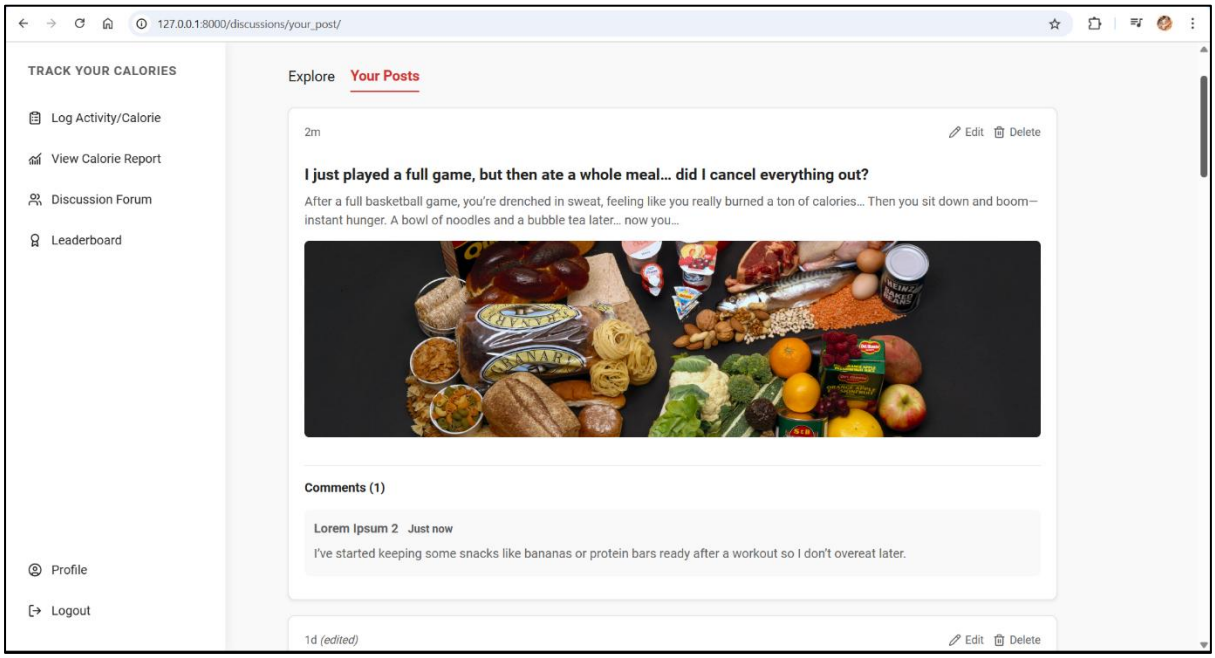


Figure 4.25: Updated 'Your Posts' View with a Comment

Figure 4.25 shows how a post appears in "Your Posts" after receiving comments. Replies are visible directly under the post card for quick review and engagement.

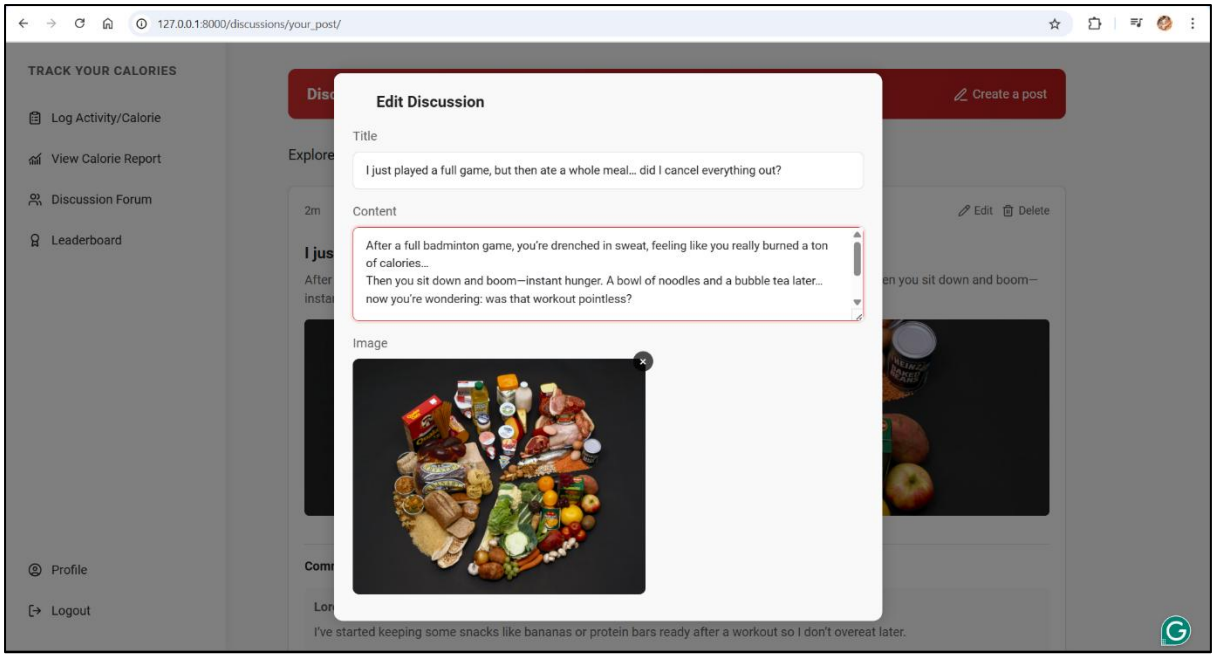


Figure 4.26: Editing a Discussion Post

Figure 4.26 shows the Edit Post modal. Users can modify the title, content, and image of their post. The layout allows for easy updates.

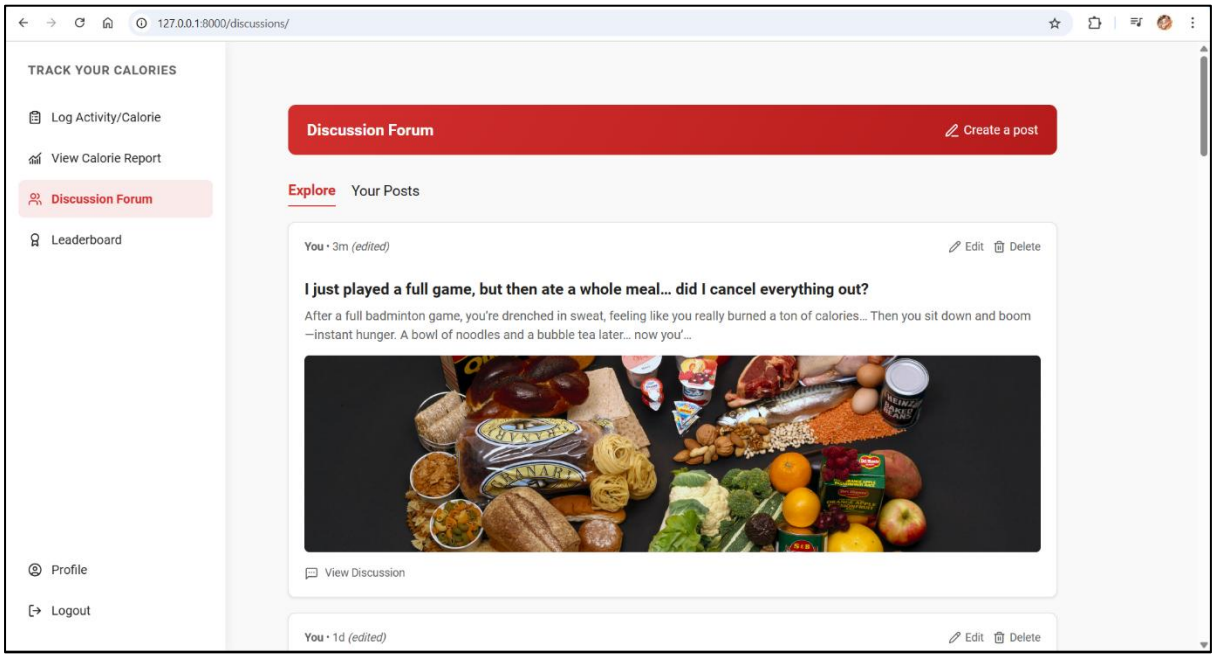


Figure 4.27: Post After Being Edited

Figure 4.27 shows how the edited post appears. The updated content is displayed with an "(edited)" label, keeping the change transparent for other users.

4.3.7 Leaderboard

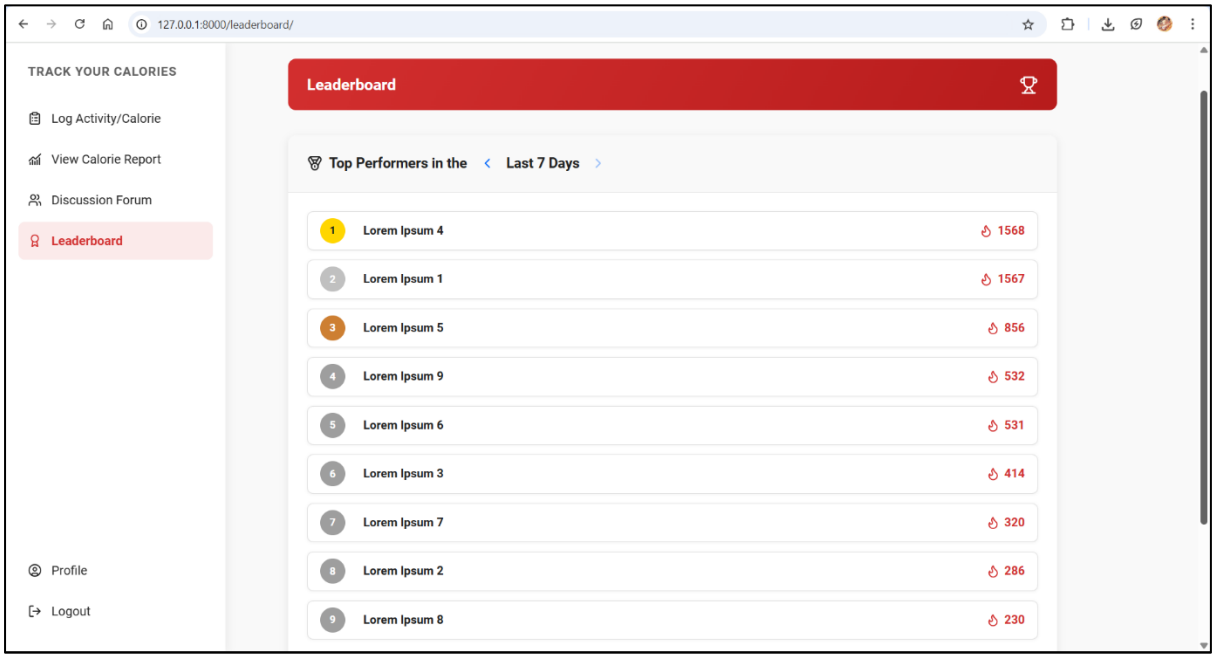


Figure 4.28: Leaderboard Page

Figure 4.28 presents the Leaderboard page. Users are ranked by total calories burned over a selected time range. The top three users are highlighted with gold, silver, and bronze icons. Other users are listed with their ranks and calorie totals.

4.4 Summary

This chapter presented the complete implementation process of the proposed calorie tracking system. It described the development tools utilized, including Visual Studio Code, HeidiSQL, Kaggle, Django, and XGBoost, highlighting their specific roles in facilitating system development and machine learning integration. The chapter also provided a walkthrough of the user interface, covering key functionalities such as user registration, profile management, activity logging, calorie reporting, discussion forum interaction, and leaderboard tracking. These implementations collectively demonstrate the system's ability to support personalised activity monitoring, real-time calorie estimation, and interactive user engagement within an integrated web-based platform.

CHAPTER 5: TESTING

5.1 Introduction

This chapter outlines the testing processes conducted to verify the functionality, reliability, and performance of the proposed Calorie Tracking Web Application. Testing is a crucial phase in the development cycle, allowing developers to identify and rectify defects, ensure that features behave as intended, and improve the system's overall quality before full deployment.

The primary goal of this application is to assist users in logging physical activities, estimating calories burned based on input parameters, and visualizing health trends. As such, comprehensive testing was conducted to evaluate both the frontend and backend components of the system, including user registration, activity logging, BMI calculation, discussion forum interactions, and calorie reporting features.

This chapter includes detailed documentation of functional testing, as well as relevant usability testing to assess the application's performance from the user's perspective. Each test case includes scenarios, expected outcomes, and actual results, which are validated to ensure the system performs reliably and accurately.

5.2 Functional Testing

Functional testing verifies that the application meets all defined requirements and specifications. It focuses on confirming whether each feature delivers the correct output based on given inputs. The testing process for this project includes input validation, form processing, session handling, activity tracking, report generation, and error message verification.

Each module was tested individually with both valid and invalid scenarios to ensure expected system behavior. The following sections provide a comprehensive breakdown of the tested modules, methods, and outcomes:

Table 5.1: Functional Testing on User Registration

Test Case ID	001			
Test Case Module	User Registration			
Test Case Description	Ensure users can register successfully with valid input (email, username, password, gender, user height and user weight).			
Pre-requisite	N/A			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successful registration with valid data	1. Navigate to the registration page. 2. Fill in all fields with valid data (email, username, password, gender, user height and user weight). 3. Submit the form.	• User account is created successfully. • User is redirected to the login page.	As expected.	**PASS**
Registration attempt with missing required fields	1. Navigate to the registration page. 2. Leave a required field (e.g., email) empty. 3. Submit the form.	• Inline error message displayed: "Please fill out this field." • Form submission is prevented.	As expected.	**PASS**
Registration attempt with mismatched passwords	1. Navigate to the registration page. 2. Enter valid data but intentionally mismatch the password and confirm password fields. 3. Submit the form.	• Error message displayed: "Please correct the errors in the form." • Form submission is prevented.	As expected.	**PASS**

Table 5.2: Functional Testing on User Login

Test Case ID	002
Test Case Name	User Login
Test Case Description	Ensure users can log in successfully with valid credentials.
Pre-requisite	User account exists.

Scenario	Input / Test Steps	Expected Result	Actual Result	Status (Pass / Fail)
Login with valid credentials	1. Navigate to the login page. 2. Enter correct username and password. 3. Submit the form.	• User successfully logs in. • User is redirected to the log activity page.	As expected.	**PASS**
Login with invalid credentials	1. Navigate to the login page. 2. Enter an existing username with an incorrect password. 3. Submit the form.	• Error message displayed: "Invalid username or password." • Form submission is prevented.	As expected.	**PASS**

Table 5.3: Functional Testing on Logout

Test Case ID	003			
Test Case Name	Logout			
Test Case Description	Verify that a logged-in user can log out from the system.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successful Logout	1. Click the logout button on the navigation bar.	• User is logged out successfully. • User is redirected to the login page. • Notification message displayed: "You have been logged out."	As expected.	**PASS**

Table 5.4: Functional Testing on View User Profile

Test Case ID	004

Test Case Name	View User Profile			
Test Case Description	Ensure logged-in users can view their profile details clearly.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
View profile information.	1. Navigate to profile page.	• Profile page displays the user's name, email, gender, height, weight, and BMI with status.	As expected.	**PASS**

Table 5.5: Functional Testing on Edit Profile

Test Case ID	005			
Test Case Name	Edit Profile			
Test Case Description	Verify users can update and save their personal profile information.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successful profile update	1. Navigate to profile page. 2. Click the "Edit Profile" button. 3. Modify personal details (e.g., height or weight). 4. Click "Save Changes."	• Changes are successfully saved. • User is redirected back to profile page with updated values shown.	As expected.	**PASS**
Cancel profile edit	1. Navigate to profile page. 2. Click the "Edit Profile" button. 3. Modify personal details (e.g., height or weight). 4. Click "Cancel".	• User is redirected back to profile page. • No changes are saved.	As expected.	**PASS**

Table 5.6: Functional Testing on BMI Calculator

Test Case ID	006			
Test Case Name	BMI Calculator			
Test Case Description	Ensure the BMI calculation on the user profile page functions correctly.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Calculate BMI with valid inputs	1. Navigate to the profile page. 2. Locate the BMI calculator section. 3. Enter a valid numeric value for weight and height in the BMI calculator section. 4. Submit the form.	• BMI value is calculated and displayed accurately. • Corresponding weight status (e.g., Underweight, Normal, Overweight) is correctly indicated.	As expected.	**PASS**
Input invalid values	1. Navigate to the profile page. 2. Locate the BMI calculator section. 3. Enter invalid value for weight and height in the BMI calculator section. 4. Submit the form.	• Error message displayed: “Please check your input.” • BMI is not calculated.	As expected.	**PASS**

Table 5.7: Functional Testing on View Activity Log

Test Case ID	007			
Test Case Name	View Activity Log			
Test Case Description	Verify correct display of activity logs for selected dates, handling empty states appropriately.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
View today's activity log (with existing activities)	1. Navigate to the log activity page. 2. Check the "Activities for Today" section.	• All logged activities for today are displayed clearly, including activity type, duration	As expected.	**PASS**

		(minutes), and calories burned.		
View today's activity log (no activities logged)	1. Navigate to the log activity page. 2. Ensure no activities exist for today.	• Informational message displayed clearly: “No activity yet. Every move counts — start burning calories now!”	As expected.	**PASS**
View activity log for past date.	1. Navigate to the log activity page. 2. Select a past date using the date picker or navigation arrows.	• Displays all activities for the selected past date, including activity type, duration (minutes), and calories burned.	As expected.	**PASS**

Table 5.8: Functional Testing on Delete from Activity Log

Test Case ID	008			
Test Case Name	Delete from Activity Log			
Test Case Description	Verify users can delete their logged activities and that changes are immediately reflected.			
Pre-requisite	• User is logged in. • Activity logs exist.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Delete today’s activity log	1. Navigate to the log activity page. 2. Locate an existing log entry in the “Activities for Today” section. 3. Click the delete (trash) icon next to the activity.	• Selected activity is deleted immediately. • Total duration and calorie summary are updated accordingly.	As expected.	**PASS**
Delete activity log from a past date	1. Navigate to the log activity page. 2. Select a past date with existing activities.	• Selected activity is deleted immediately. • Total duration and	As expected.	**PASS**

	3. Click the delete (trash) icon next to the activity to remove.	calorie summary reflect the updated data.		

Table 5.9: Functional Testing on View Recent Logs

Test Case ID	009			
Test Case Name	View Recent Logs			
Test Case Description	Verify display of the user's five most recent unique activities, ensuring clear handling when no recent logs exist.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Display recent activities (existing logs)	1. Navigate to log activity page. 2. Scroll to the "Recent Entries" section.	• Shows the five most recent unique activities clearly including activity types, duration (min) and calories burned.	As expected.	**PASS**
No recent activities exist	1. Navigate to log activity page. 2. Ensure no recent logs exist.	• Informational message displayed: "No recent activities found."	As expected.	**PASS**

Table 5.10: Functional Testing on Add Activity Log from Recent Logs

Test Case ID	010			
Test Case Name	Add Activity Log from Recent Logs			
Test Case Description	Verify users can quickly reuse recent activities to log them again for the selected date.			
Pre-requisite	• User is logged in. • Recent activity logs exist.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**

Reuse recent activity for today	1. Navigate to log activity page. 2. Scroll to “Recent Entries” section. 3. Click “+” icon beside the desired recent activity.	• Selected recent activity is logged for today immediately. • Correct duration and calculated calories are displayed.	As expected.	**PASS**
Reuse recent activity for a past date	1. Navigate to log activity page. 2. Select a past date. 3. Scroll to “Recent Entries” section. 4. Click “+” icon beside the desired recent activity.	• Selected recent activity is logged correctly for the chosen past date. • Duration and calories reflect accurate calculation.	As expected.	**PASS**

Table 5.11: Functional Testing on Log New Activity

Test Case ID	011			
Test Case Name	Log New Activity			
Test Case Description	Verify the activity logging feature accurately handles input validation, data saving, and calorie calculation for both current and past dates.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Log activity for today with valid inputs	1. Navigate to log activity page. 2. Scroll to “Add New For Today” section. 3. Select an activity. 4. Enter the duration. 5. Click “Log Activity”	• Activity is logged immediately for today. • Correct duration and calorie calculation are shown.	As expected.	**PASS**
Attempt logging with missing inputs	1. Navigate to log activity page. 2. Scroll to “Add New For Today” section. 3. Leave required fields empty. 4. Click “Log Activity”	• Inline error message displayed: "Please fill out this field." • Form submission	As expected.	**PASS**

		is prevented.		
Log activity for a past date	1. Navigate to log activity page. 2. Scroll to “Add New For Today” section. 3. Select a past date. 4. Select an activity. 5. Enter duration. 6. Confirm weight on selected date. 7. Click “Log Activity”	• Activity is logged for the selected past date. • Correct details (activity type, duration, calories) are shown.	As expected.	**PASS**

Table 5.12: Functional Testing on View Calorie Report

Test Case ID	012			
Test Case Name	View Calorie Report			
Test Case Description	Ensure the calorie report visualizes data accurately over selectable date ranges (7, 30, or 60 days).			
Pre-requisite	• User is logged in. • Activity logs exist.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
View report (7 days)	1. Navigate to the calorie report page.	• Graph correctly displays calories burned over the past 7 days.	As expected.	**PASS**
View report (30 or 60 days)	1. Navigate to the calorie report page. 2. Adjust date range using provided controls.	• Graph dynamically updates to correctly reflect the selected range.	As expected.	**PASS**

Table 5.13: Functional Testing on Download Calorie Report

Test Case ID	013
Test Case Name	Download Calorie Report
Test Case Description	Validate PDF download of calorie report with accurate user data, logs, and visualizations.

Pre-requisite	• User is logged in. • Activity logs exist.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Download report as PDF	1. Navigate to the calorie report page. 2. Scroll to "Download Your Report." section. 3. Click the "Download calorie_report.pdf" button.	• PDF report is correctly generated and downloaded. • PDF includes accurate user data, activity logs, and graphical summary.	As expected.	**PASS**

Table 5.14: Functional Testing on View Discussion & Reply

Test Case ID	014			
Test Case Name	View Discussion & Reply			
Test Case Description	Ensure users can clearly view discussions and corresponding replies.			
Pre-requisite	• User is logged in. • At least one discussion exists.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
View discussion with replies	1. Navigate to the discussions page. 2. Click on a discussion from the list.	• Discussion content is displayed clearly in a modal window. • All associated replies are listed chronologically.	As expected.	**PASS**
View discussion with no replies	1. Navigate to the discussions page. 2. Click on a discussion that has no replies.	• Discussion content is displayed clearly. • An indication of "No replies yet." is displayed.	As expected.	**PASS**

Table 5.15: Functional Testing on Create New Discussion

Test Case ID	015			
Test Case Name	Create New Discussion			
Test Case Description	Validate users can successfully create a new discussion, with or without an attached image.			
Pre-requisite	User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Create a discussion without an image	1. Navigate to discussion page. 2. Click “Create Discussion”. 3. Fill in the title and content fields. 4. Click “Post”.	• New discussion is successfully created. • Discussion appears at the top of the discussions list immediately.	As expected.	**PASS**
Create a discussion with image	1. Navigate to discussion page. 2. Fill in the title and content fields. 3. Attach a valid image file. 4. Click “Post”	• New discussion with attached image is created successfully. • Thumbnail image displayed alongside discussion title.	As expected.	**PASS**
Attempt to submit an empty discussion	1. Navigate to discussion page. 2. Click “Create Discussion” 3. Leave the required fields empty 4. Click “Post”	• Inline error message displayed: “Please fill out this field.” • Discussion submission is prevented.	As expected.	**PASS**

Table 5.16: Functional Testing on Edit Existing Discussion

Test Case ID	016
Test Case Name	Edit Existing Discussion
Test Case Description	Verify that users can successfully edit their previously created discussions.

Pre-requisite	• User is logged in. • User has an existing discussion.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successfully edit discussion content	1. Navigate to discussion page. 2. Navigate to the user's existing discussions. 3. Click the "Edit" button on the targeted discussion. 4. Modify the title or content. 5. Click "Save."	• Changes are successfully updated. • Updated details are reflected immediately. • An indicator "(edited)" appears next to the discussion timestamp.	As expected.	**PASS**
Cancel edit without saving	1. Navigate to discussion page. 2. Navigate to the user's existing discussions. 3. Click "Edit" on an existing discussion. 4. Make changes to content or title. 5. Click "Cancel."	• Changes are not saved. • Original discussion content remains unchanged.	As expected.	**PASS**

Table 5.17: Functional Testing on Delete Existing Discussion

Test Case ID	017			
Test Case Name	Delete Existing Discussion			
Test Case Description	Ensure users can delete their discussions.			
Pre-requisite	• User is logged in. • User has existing discussions.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successfully delete a discussion	1. Navigate to discussion page. 2. Navigate to the user's existing discussions. 3. Click the "Delete" button on a targeted discussion. 4. Confirm deletion in the prompt dialog.	• Discussion is permanently deleted. • Discussion is immediately removed from the list.	As expected.	**PASS**

Table 5.18: Functional Testing on Reply to Discussion

Test Case ID	018			
Test Case Name	Reply to Discussion			
Test Case Description	Ensure users can reply to existing discussions.			
Pre-requisite	• User is logged in. • A discussion exists.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Reply successfully to a discussion	1. Navigate to discussion page. 2. Navigate to an existing discussion. 3. Click on the discussion to open it. 4. Enter reply text into the reply input field. 5. Click “Reply.”	• Reply is immediately posted and displayed beneath the discussion. • Reply includes correct author name, content, and timestamp.	As expected.	**PASS**
Attempt to reply with empty content	1. Navigate to discussion page. 2. Navigate to an existing discussion. 3. Click on the discussion to open it. 4. Leave the reply input field empty. 5. Click “Reply.”	• Inline error message displayed: “Please fill out this field.” • Reply submission is prevented.	As expected.	**PASS**

Table 5.19: Functional Testing on Edit Reply

Test Case ID	019			
Test Case Name	Edit Reply			
Test Case Description	Verify users can edit their replies to discussions.			
Pre-requisite	• User is logged in. • User has previously replied to a discussion.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Successfully edit a reply	1. Navigate to discussion page.	• Reply content updates immediately.	As expected.	**PASS**

	2. Navigate to an existing discussion containing user's reply. 3. Click "Edit" on targeted reply. 4. Modify reply content. 5. Click "Save."	• An indicator "(edited)" appears next to the reply timestamp.		
Cancel reply edit	1. Navigate to discussion page. 2. Navigate to an existing discussion containing user's reply. 3. Click "Edit" on targeted reply. 4. Modify reply content. 5. Click "Cancel."	• Original reply remains unchanged. • No modifications saved.	As expected.	**PASS**

Table 5.20: Functional Testing on Delete Reply

Test Case ID	020			
Test Case Name	Delete Reply			
Test Case Description	Ensure users can delete their replies.			
Pre-requisite	• User is logged in. • User has previously replied to a discussion.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Delete reply successfully	1. Navigate to discussion page. 2. Navigate to an existing discussion containing user's reply. 3. Click "Delete" on targeted reply. 4. Confirm deletion in the prompt dialog.	• Reply is immediately removed from the discussion.	As expected.	**PASS**

Table 5.21: Functional Testing on View Leaderboard

Test Case ID	021
Test Case Name	View leaderboard
Test Case Description	Ensure users can view the leaderboard and see the top users ranked by calories burned within a selected range (7, 30, or 60 days).
Pre-requisite	• User is logged in. • Activity logs from multiple users exist in the selected range.

Scenario	Input / Test Steps	Expected Result	Actual Result	Status (Pass / Fail)
View leaderboard for the default (7-day) range	1. Navigate to the leaderboard page. 2. Observe the list of ranked users.	• Leaderboard displays top users sorted by total calories burned. • Each entry includes rank, username, and calories burned. • Users not visible in leaderboard (is_visible_in_leaderboard=False) are excluded.	As expected.	**PASS**
Change leaderboard range	1. Navigate to the leaderboard page. 2. Click the left/right arrow buttons to change the leaderboard range (7, 30, or 60 days). 3. Observe the list of ranked users.	• The list reloads with data for the selected range. • Range label updates accordingly (e.g., “Last 30 Days”).	As expected.	**PASS**

Table 5.22: Functional Testing on Toggle Leaderboard Visibility

Test Case ID	022			
Test Case Name	Toggle Leaderboard Visibility			
Test Case Description	Verify that users can toggle their visibility on the leaderboard using the privacy switch.			
Pre-requisite	• User is logged in.			
Scenario	**Input / Test Steps**	**Expected Result**	**Actual Result**	**Status (Pass / Fail)**
Make profile visible on the leaderboard	1. Navigate to the leaderboard page. 2. Enable the "Appear on the Leaderboard" toggle switch.	• User appears in the leaderboard if they have logged activities.	As expected.	**PASS**
Hide profile from leaderboard	1. Navigate to the leaderboard page. 2. Disable the "Appear on the Leaderboard" toggle switch.	• User is no longer visible in the leaderboard, regardless of activity logs.	As expected.	**PASS**

5.3 Usability Testing

To evaluate the system's usability and user experience, a usability testing questionnaire was conducted among 30 respondents. These participants tested the system and shared their feedback regarding various features including registration, login, profile management, BMI calculator, activity logging, calorie report, discussion forum and leaderboard. The results are summarized as follows:

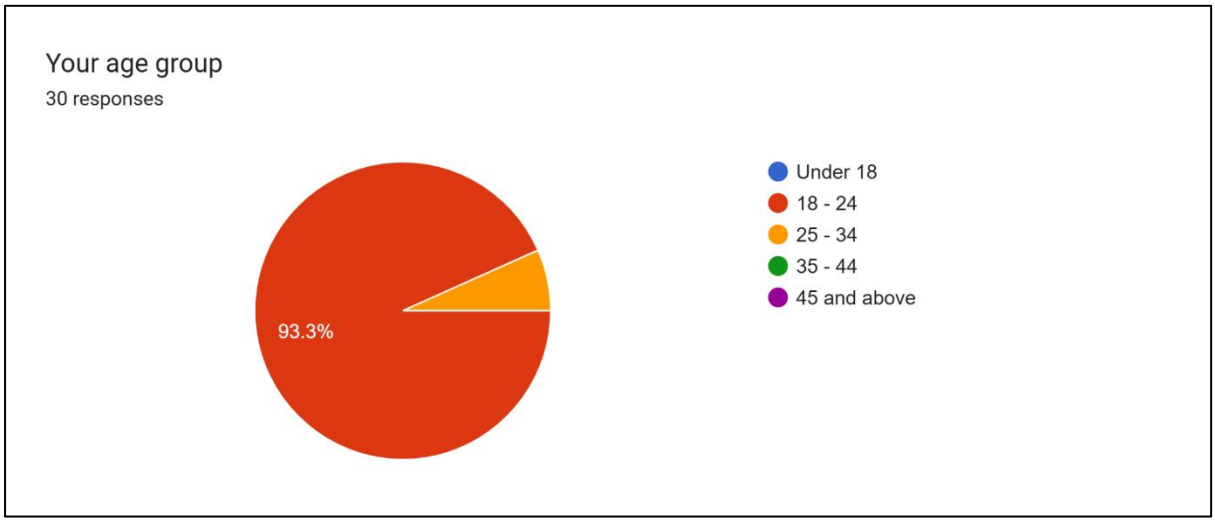


Figure 5.1: Your Age Group

As illustrated in *Figure 5.1*, the majority of respondents (93.3%) were aged between 18 and 24, followed by 6.7% aged 25 to 34. No respondents were under 18 or over 34. This age distribution indicates that the system primarily reached university-age users, aligning well with the target audience of UNIMAS students.

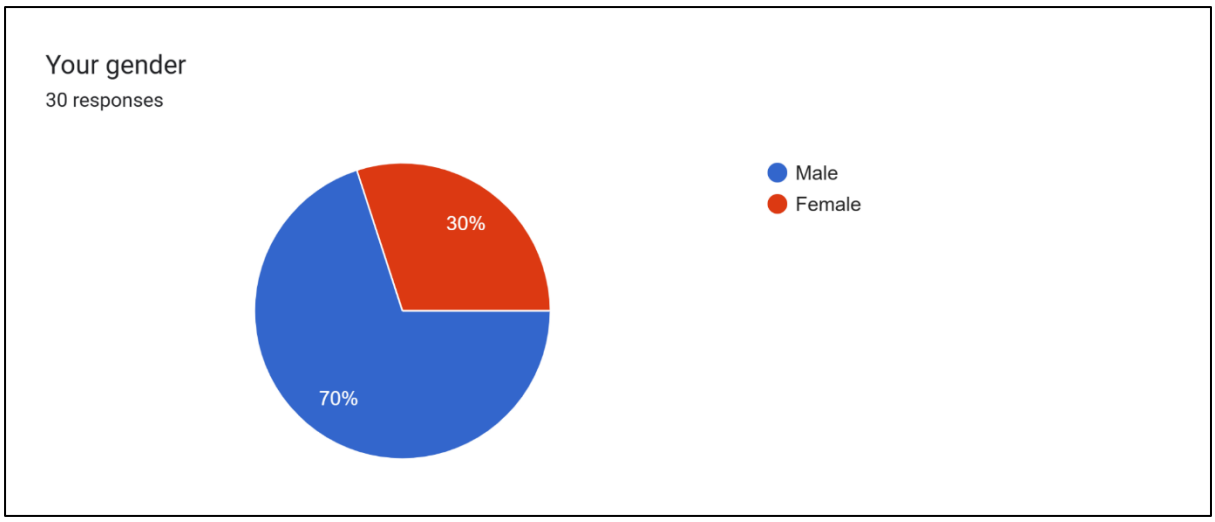


Figure 5.2: Your Gender

Figure 5.2 shows that 70% of the respondents were male, while 30% were female. This gender distribution reflects balanced participation and provides meaningful insights from both user perspectives.

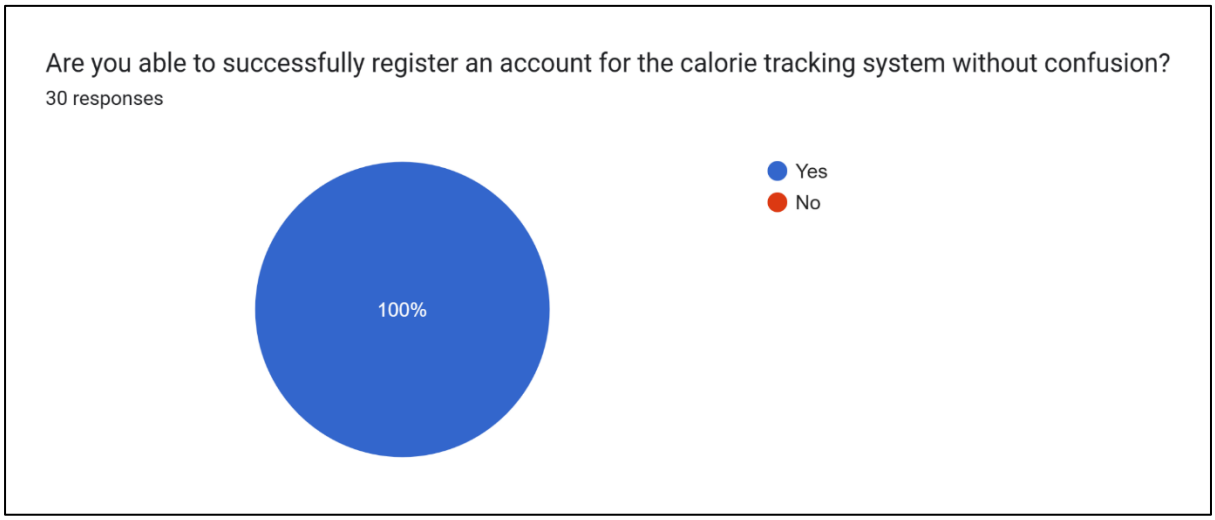


Figure 5.3: Are you able to successfully register an account for the calorie tracking system without confusion?

All respondents (100%) reported being able to register successfully without any confusion. This indicates that the registration process was intuitive, clearly labeled, and free from usability errors.

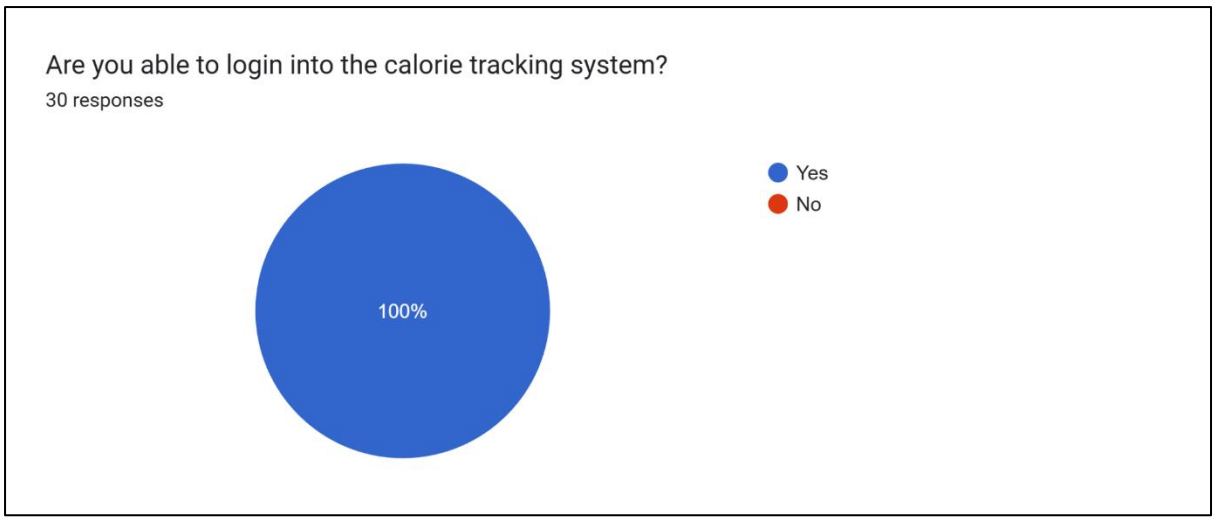


Figure 5.4: Are you able to login into the calorie tracking system?

Similarly, 100% of users indicated successful login without issues. This shows that the authentication process, including input validation and feedback for errors, was functioning correctly and was user-friendly.

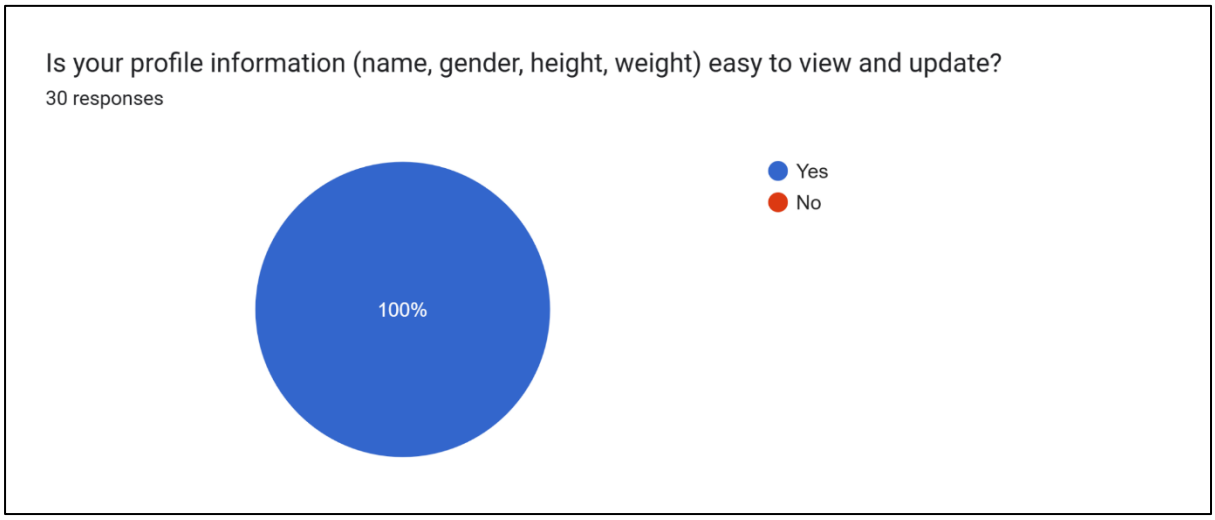


Figure 5.5: Is your profile information (name, gender, height, weight) easy to view and update?

All respondents (100%) agreed that the profile section was easy to navigate and update. This affirms that the layout and form design of the profile page met usability expectations, allowing users to manage their health metrics effortlessly.

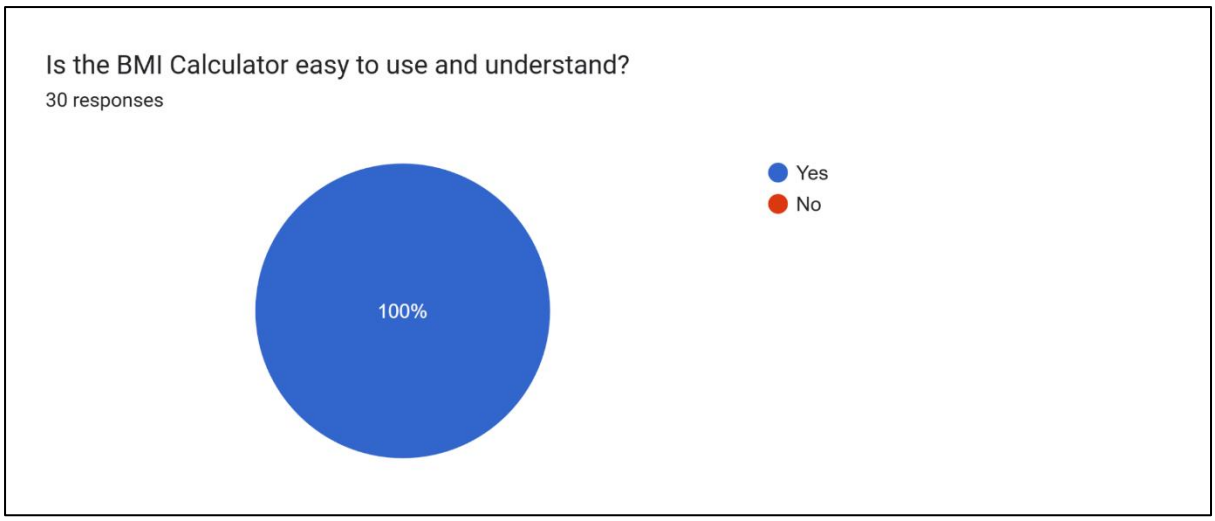


Figure 5.6: Is the BMI Calculator easy to use and understand?

Every respondent (100%) found the BMI calculator simple and easy to understand. This suggests the input design, validation logic, and feedback presentation were effectively implemented, helping users calculate and interpret their BMI without confusion.

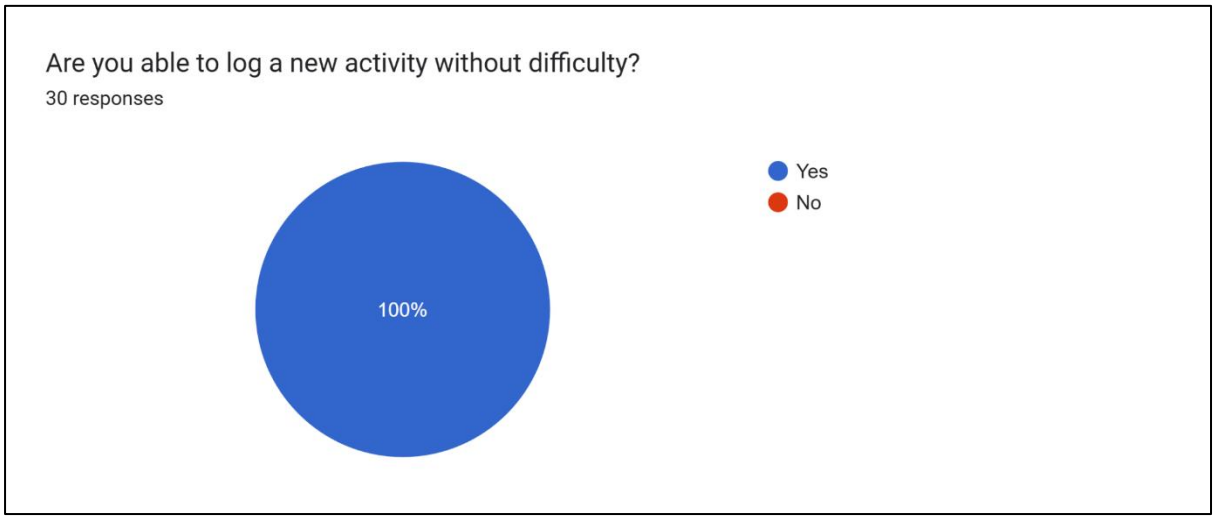


Figure 5.7: Are you able to log a new activity without difficulty?

All respondent (100%) were able to log a new activity successfully. This reflects well on the accessibility and flow of the activity logging form, as well as the responsiveness of the system's feedback.

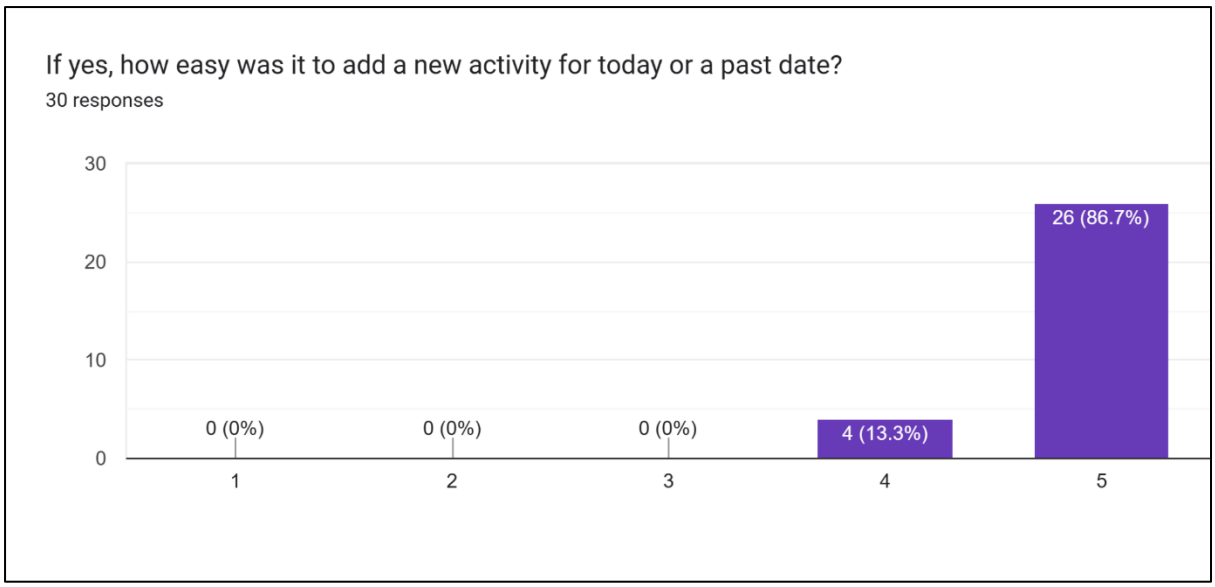


Figure 5.8: If yes, how easy was it to add a new activity for today or a past date?

As shown in *Figure 5.8*, 86.7% of respondents rated the experience of logging activities for today or past dates as "Very Easy" (5), while 13.3% rated it as "Easy" (4). No respondents reported difficulty (1–3). This confirms that users found the activity logging process highly intuitive, regardless of the date selected.

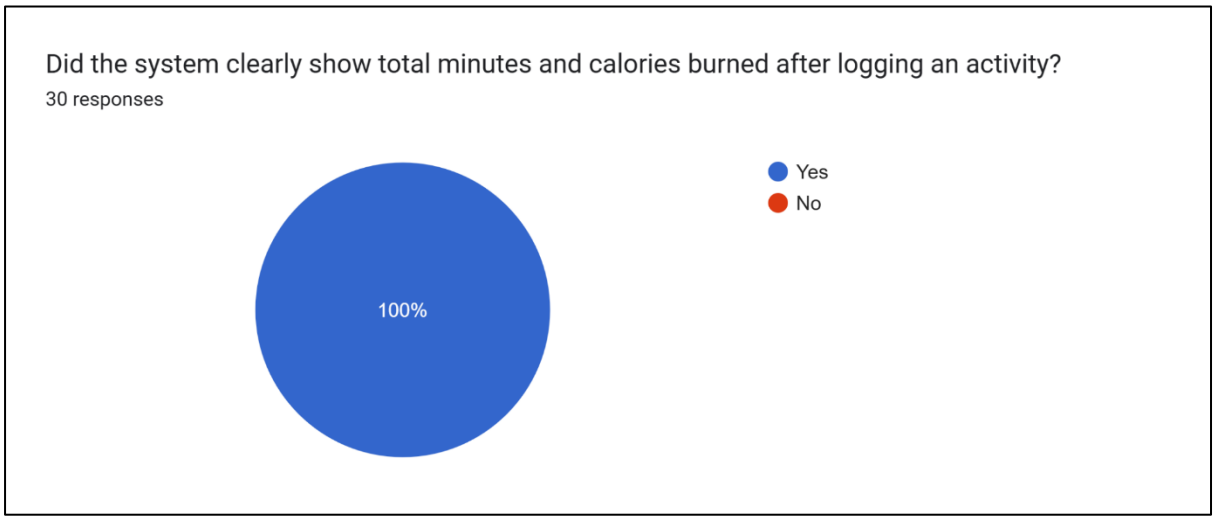


Figure 5.9: Did the system clearly show total minutes and calories burned after logging an activity?

According to *Figure 5.9*, all respondents (100%) confirmed that the system clearly displayed the total minutes and calories burned. This indicates that the visual representation and data summaries were effective in helping users monitor their activity results.

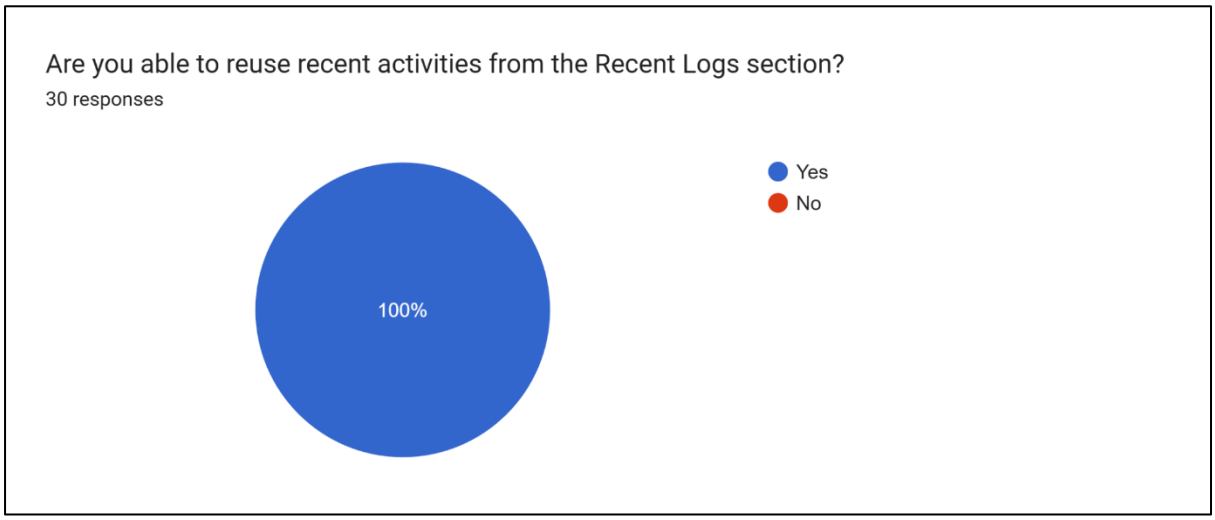


Figure 5.10: Are you able to reuse recent activities from the Recent Logs section?

As shown in Figure 5.10, all respondents (100%) confirmed that they were able to reuse recent activity entries from the Recent Logs section. This demonstrates that the system's feature for re-logging previously performed activities is both functional and easily accessible to users.

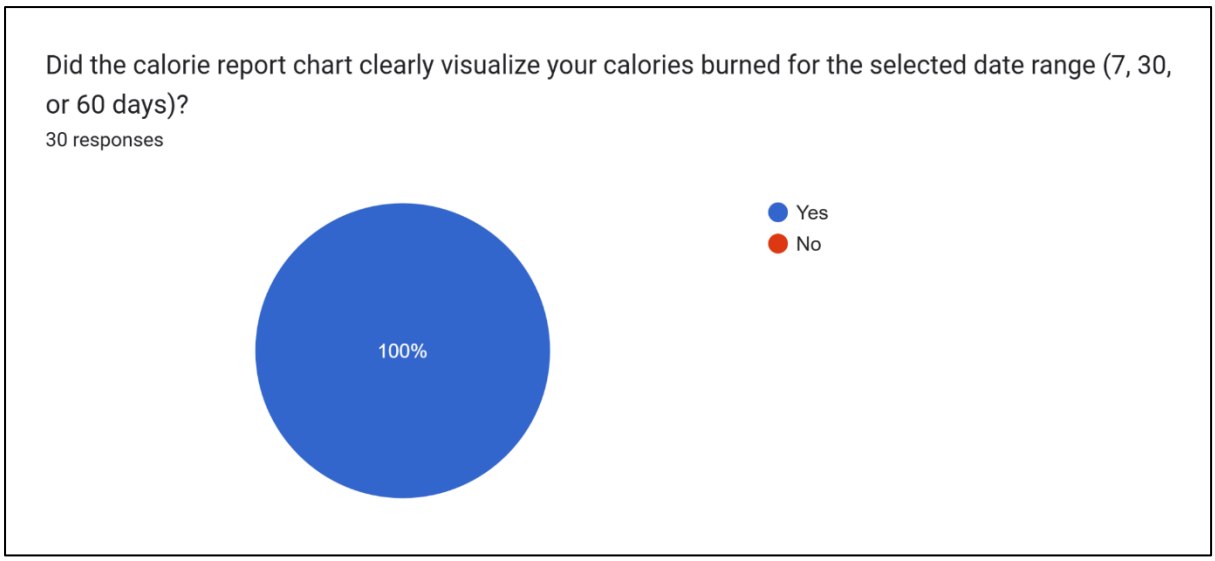


Figure 5.11: Did the calorie report chart clearly visualize your calories burned for the selected date range (7, 30, or 60 days)?

Figure 5.11 shows that every respondent (100%) found the calorie report chart clear and informative. This validates that the graph component is well-designed to communicate calorie-burning trends effectively over various date ranges.

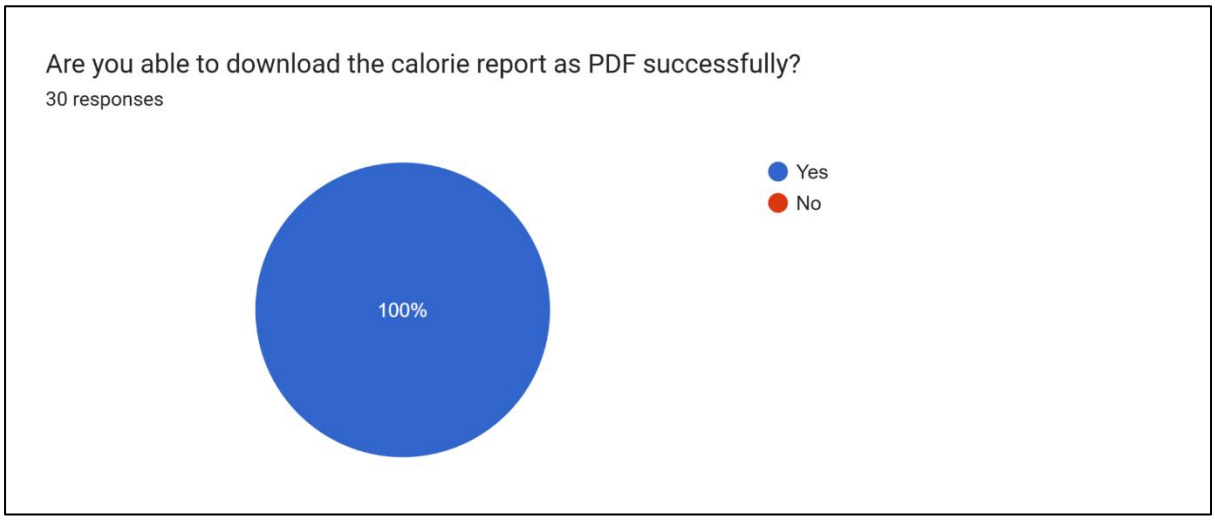


Figure 5.12: Are you able to download the calorie report as PDF successfully?

All participants (100%) indicated successful download of the PDF report. This confirms that the PDF generation functionality, including embedded charts and user data, is reliably executed by the system.

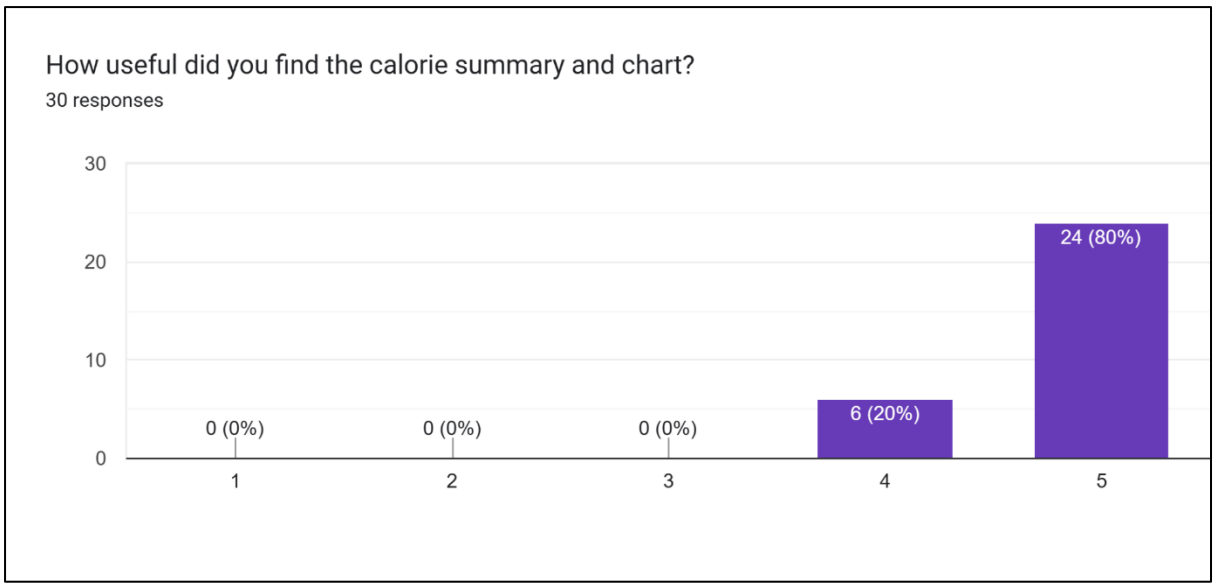


Figure 5.13: How useful did you find the calorie summary and chart?

As illustrated in *Figure 5.13*, 80% of respondents rated the calorie summary and chart as “Very Useful” (5), while the remaining 20% rated it as “Useful” (4). None gave it a lower rating. This affirms the strong value of visual summaries in helping users interpret their activity data.

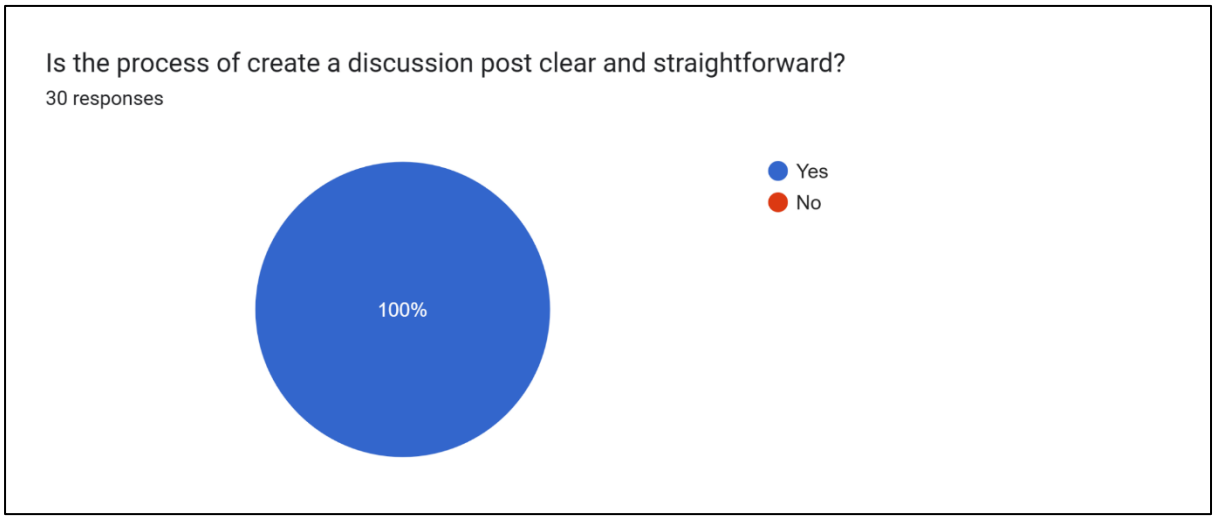


Figure 5.14: Is the process of create a discussion post clear and straightforward?

According to *Figure 5.14*, 100% of users found the discussion creation process easy and straightforward. This indicates the form for creating posts, including text and optional image uploads, was intuitive and met user expectations.

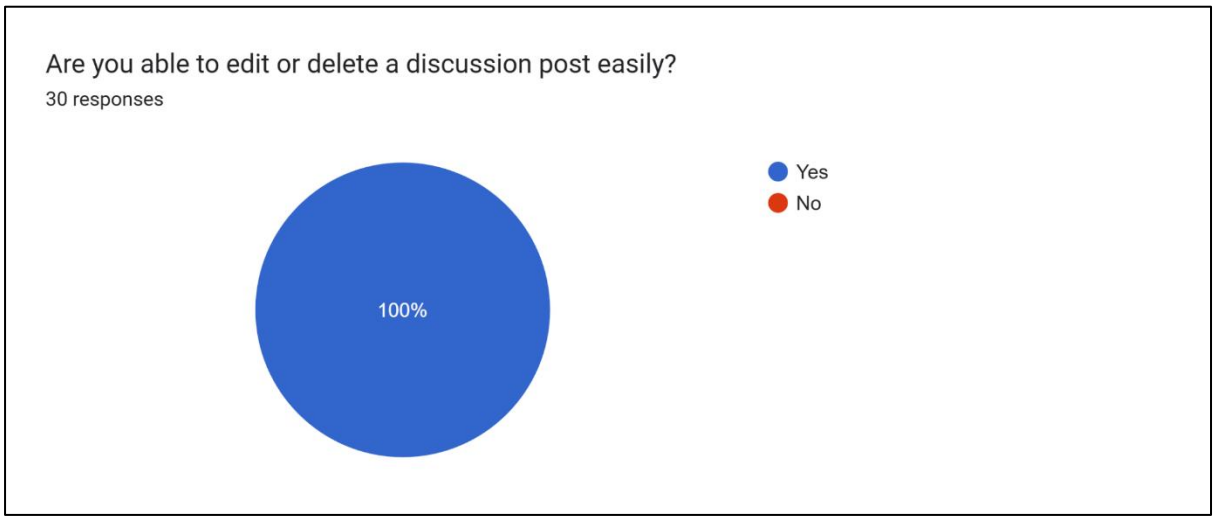


Figure 5.15: Are you able to edit or delete a discussion post easily?

Figure 5.15 reveals that all respondents were able to edit or delete their discussion posts without difficulty. This suggests that the UI controls for post management were implemented in a user-friendly and accessible manner.

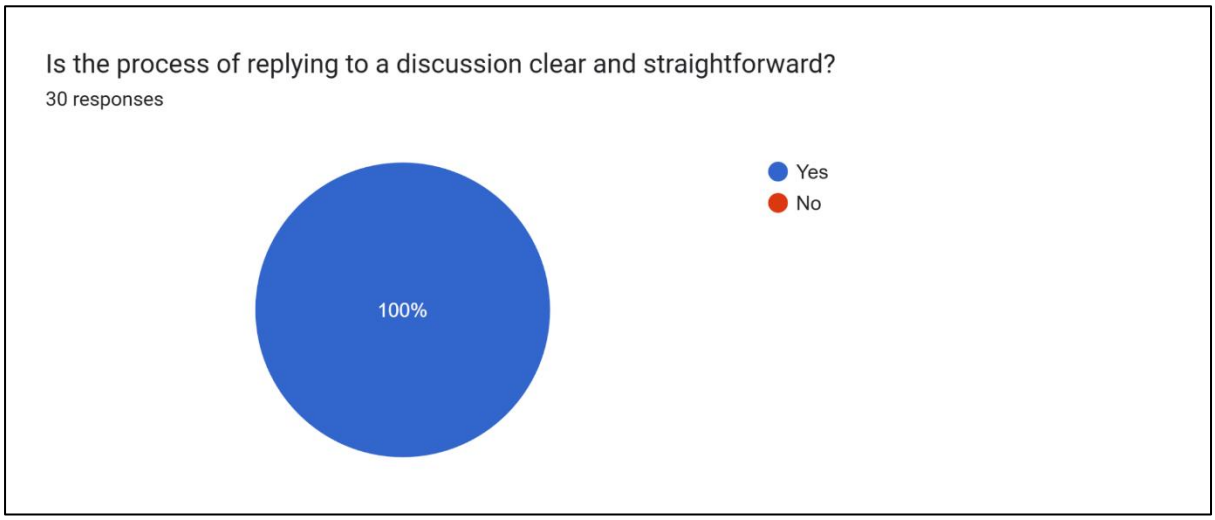


Figure 5.16: Is the process of replying to a discussion clear and straightforward?

All respondents (100%) reported no issues when replying to a discussion post. This validates the simplicity of the reply input field and the reliability of the system's interaction model for threaded conversations.

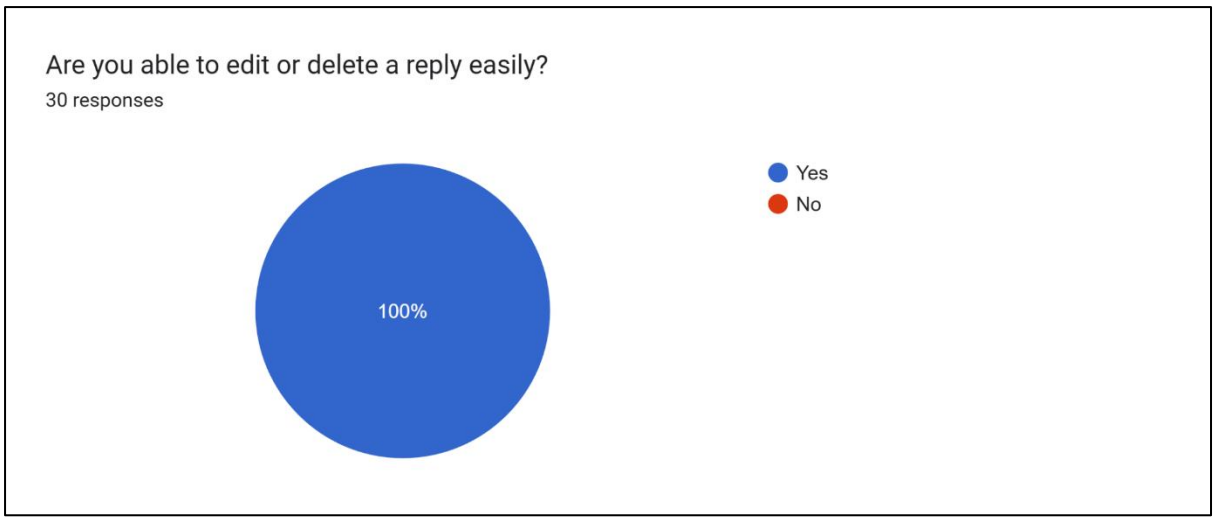


Figure 5.17: Are you able to edit or delete a reply easily?

As shown in *Figure 5.17*, 100% of users could successfully edit or delete their replies. This reflects effective implementation of reply-level permissions and update/delete options within the discussion forum.

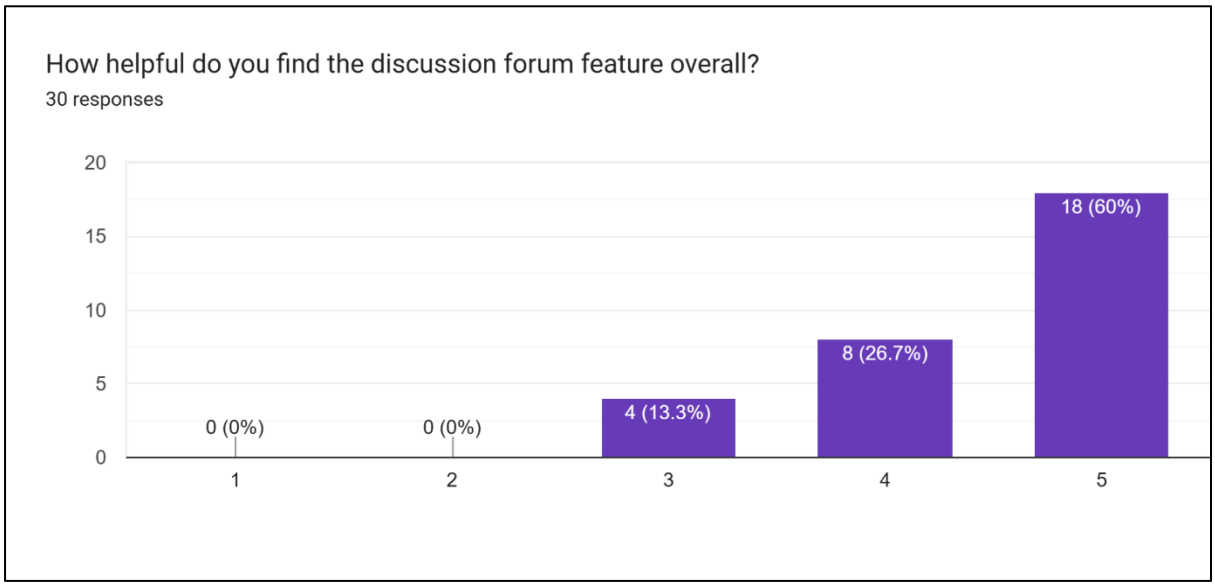


Figure 5.18: How helpful do you find the discussion forum feature overall?

Figure 5.18 shows that 60% of users rated the forum feature as "Very Helpful" (5), while 26.7% rated it "Helpful" (4), and 13.3% selected "Neutral" (3). No respondents found the feature unhelpful. This indicates the discussion forum is seen as a valuable component of the application for community engagement and support.

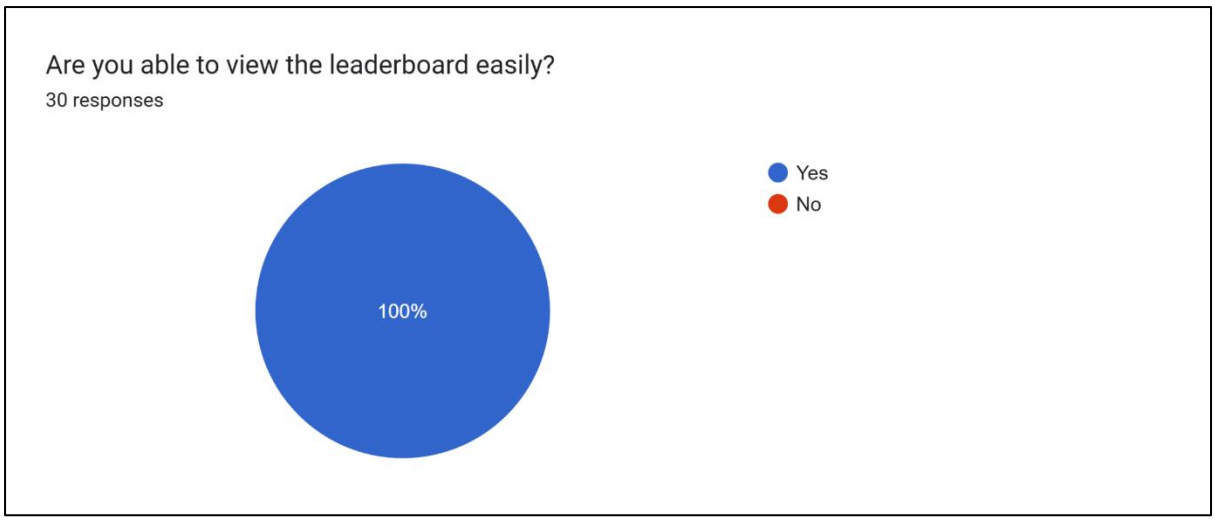


Figure 5.19: Are you able to view the leaderboard easily?

According to *Figure 5.19*, all participants (100%) were able to view the leaderboard with ease. This confirms that the ranking page is not only functional but also easy to locate and interpret by users.

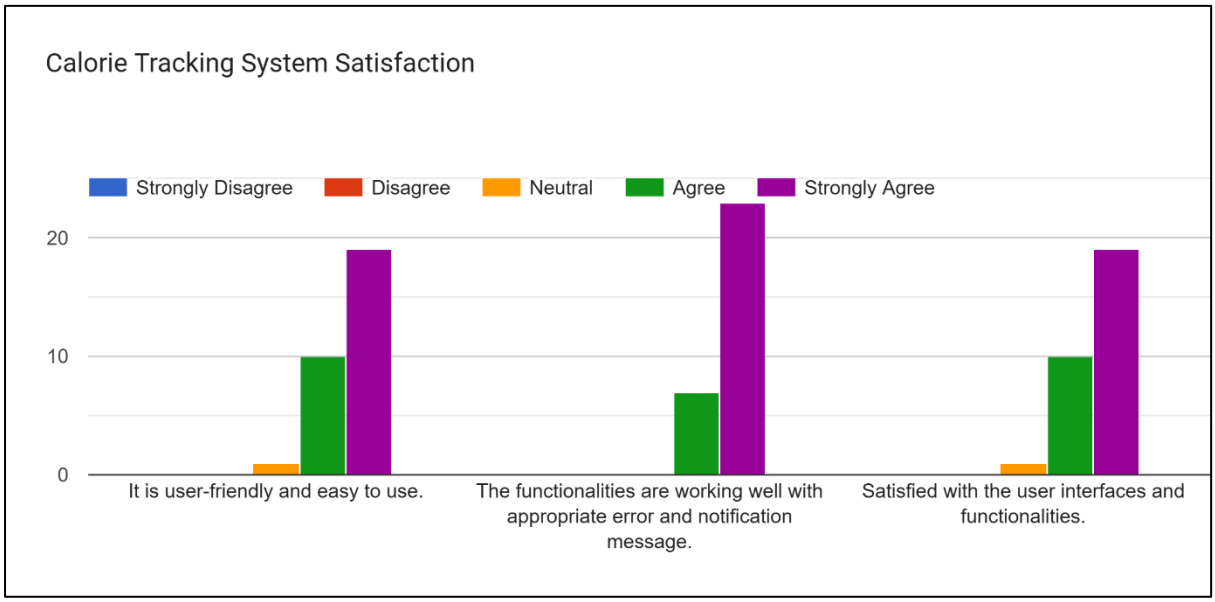


Figure 5.20: Calorie Tracking System Satisfaction

Below are the questions related to users' satisfaction with the system:

- It is user-friendly and easy to use.
- The functionalities are working well with appropriate error and notification messages.

- Satisfied with the user interfaces and functionalities.

The usability result for user satisfaction is illustrated in *Figure 5.20*. Overall, the majority of respondents strongly agreed that the system is user-friendly and easy to use, the functionalities work as intended with proper feedback, and they are satisfied with the user interfaces and overall features provided by the application. This positive feedback reflects a high level of satisfaction and confirms that the system meets user expectations effectively.

5.4 Summary

This chapter documented the testing process taken to make sure the functionality, reliability, and usability of the proposed calorie tracking system is working as expected. Functional testing validated all system modules, including registration, login, profile management, activity logging, calorie reporting, discussion forum, and leaderboard features, ensuring they operated according to specified requirements. Also, usability testing conducted with 30 participants, suggested that the system was working as expected and met expectations across various components.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Introduction

This chapter presents the project's achievements, identifies existing limitations, and suggests future enhancements. The outcomes of the project are evaluated against its initial objectives to determine overall success. Additionally, this section highlights unresolved limitations, which serve as opportunities for future improvements and adaptations.

6.2 Project Achievement

As stated in Chapter 1, the objective of this project is to develop a web-based calorie tracking application tailored to the needs of UNIMAS students. The system enables students to manually log their physical activities, receive calorie burn predictions, and visualize their progress through an intuitive interface.

The objectives and the achievements are summarised in *Table 6.1*.

Table 6.1 Summaries of the Objectives and Achievements.

Objective	**Achievement**
Design a web-based calorie tracking application tailored to the needs of UNIMAS students, which allows them to manually log their physical activities and receive calorie burn predictions.	The system was successfully designed and implemented with a proper calorie logging interface that allows students to input their activities, including duration, intensity, and activity type. Calorie burn predictions are generated based on a trained machine learning model.
Develop a web-based calorie tracking application with a comprehensive interface that enables UNIMAS students to log various physical activities and visualize their calorie consumption trends through intuitive graphs.	The system enables UNIMAS students to log various physical activities through a structured web interface and visualize their calorie consumption trends over time using intuitive graphs based on the logged data.
Evaluate the usability and functionality of the proposed application.	All main features were implemented and tested successfully. Usability testing with a group of users gathered positive feedback,

	confirming that the application meets expectations in terms of design, functionality, and overall performance.

6.3 Project Limitation

Although the project successfully met its stated objectives, several technical limitations related to backend architecture and system scalability have been identified:

- The current implementation relies on a traditional relational database structure, potentially resulting in degraded query performance and slower response times as the user base and activity data scale over time.
- Another notable limitation is the absence of a dedicated caching mechanism which leads to redundant database operations, further limiting the system's responsiveness and efficiency.
- While the system uses XGBoost for calorie prediction, the model only estimated based on the user's weight hence only limitedly reflect individual metabolic differences.
- The system currently lacks an administrative interface, limiting effective user account management, data monitoring, and routine maintenance tasks.

6.4 Future Work

Future improvements should focus on enhancing the system's technical architecture to support scalability, personalization, and administrative control:

- Implementing advanced database optimization techniques such as query indexing, normalisation, and restructuring will improve performance under higher data loads. Additionally, integrating a caching solution like Redis will reduce redundant database hits and improve response times.
- Incorporating user-specific features (gender, fitness level, etc.) into the calorie prediction model will enable more accurate and personalised estimations. This could involve

retraining the model using clustered user profiles or incorporating dynamic learning mechanisms.

- Developing a comprehensive admin interface will allow authorised personnel to monitor system usage, manage user accounts, analyse aggregate activity data, and perform maintenance tasks through an accessible and well-structured backend system.

These enhancements will not only address current technical constraints but also lay the foundation for a more robust, scalable, and adaptable system in future iterations.

REFERENCES

ACE Fitness. (2023). Does the ACE Integrated Fitness Training® Model really work? Year 3 study. https://www.acefitness.org/continuing-education/certified/june-2023/8354/ace-supported-research-does-the-ace-integrated-fitness-training-model-really-work-year-3-study/

Bhole, K., Singh, S., Patel, P., Patil, A., & Hariharan, S. (2024). CALORYSUM: Calorie Tracker Web Application. International Research Journal of Modernization in Engineering, Technology, and Science, 6(4). https://doi.org/10.56726/IRJMETS53439

Business of Apps. (2024). MyFitnessPal statistics and facts. https://www.businessofapps.com/data/myfitnesspal-statistics/

Carballo-Fazanes, A., Rodríguez-Fernández, J. E., Rico-Díaz, J., Varela-Casal, C., Barcala-Furelos, R., Rey, E., & Abelairas-Gómez, C. (2020). Physical activity habits and determinants, sedentary behaviour and lifestyle in university students. International Journal of Environmental Research and Public Health, 17(9), 3272. https://doi.org/10.3390/ijerph17093272

Emanuilov, S. (2024). Why Django is the perfect platform for AI web applications. Medium. https://medium.com/@simeon.emanuilov/why-django-is-the-perfect-platform-for-ai-web-applications-d0c36d1c8244

GeeksforGeeks. (2025, May 1). Agile Software Development – Software Engineering. In GeeksforGeeks. https://www.geeksforgeeks.org/software-engineering-agile-software-development/

Gleantap. (2023). MyFitnessPal: A comprehensive guide to the app's features and benefits. https://www.gleantap.com/myfitnesspal-guide

Index.dev. (2024, June 4). Maximizing efficiency across engineering, product, and hiring with Python Django. https://www.index.dev/blog/python-django-maximizing-efficiency-across-engineering-product-and-hiring

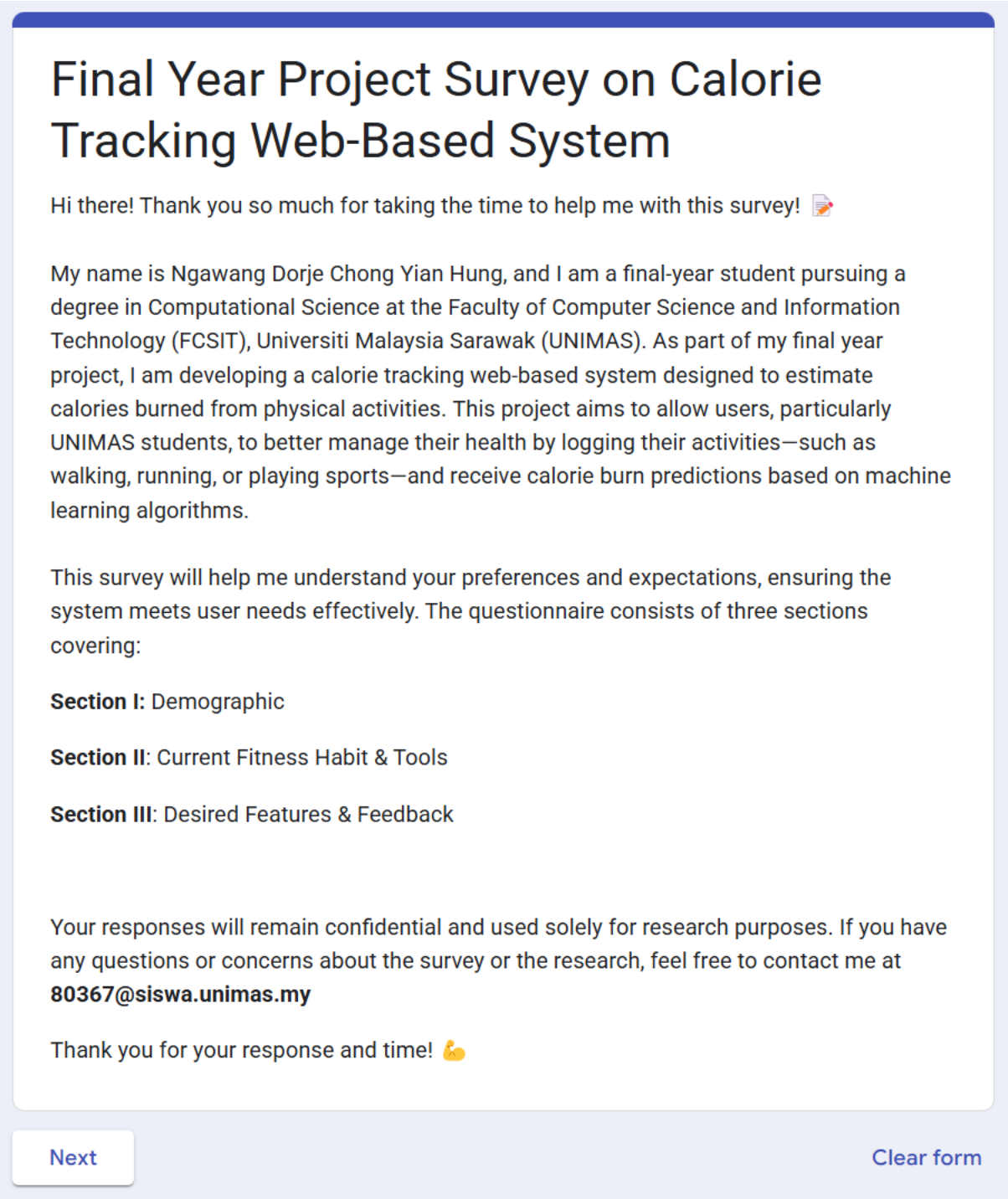
Jain, A. (2024, August 1). Top 20 Frontend Web Development Technologies (2024 List). WsCube Tech Blog. https://www.wscubetech.com/blog/front-end-technologies/

Microsoft. (2024). Debugging in Visual Studio Code. Retrieved from https://code.visualstudio.com/docs/editor/debugging

MyNetDiary. (2023). MyNetDiary efficacy. https://www.mynetdiary.com/mynetdiary_efficacy.html

Turing. (2024). Ultimate guide: Visual Studio vs. Visual Studio Code. Turing. https://www.turing.com/kb/ultimate-guide-visual-studio-vs-visual-studio-code

APPENDIX A



Final Year Project Survey on Calorie Tracking Web-Based System

Hi there! Thank you so much for taking the time to help me with this survey!

My name is Ngawang Dorje Chong Yian Hung, and I am a final-year student pursuing a degree in Computational Science at the Faculty of Computer Science and Information Technology (FCSIT), Universiti Malaysia Sarawak (UNIMAS). As part of my final year project, I am developing a calorie tracking web-based system designed to estimate calories burned from physical activities. This project aims to allow users, particularly UNIMAS students, to better manage their health by logging their activities—such as walking, running, or playing sports—and receive calorie burn predictions based on machine learning algorithms.

This survey will help me understand your preferences and expectations, ensuring the system meets user needs effectively. The questionnaire consists of three sections covering:

Section I: Demographic

Section II: Current Fitness Habit & Tools

Section III: Desired Features & Feedback

Your responses will remain confidential and used solely for research purposes. If you have any questions or concerns about the survey or the research, feel free to contact me at **80367@siswa.unimas.my**

Thank you for your response and time!

Next

Clear form

Figure A.1: Survey Form Description.

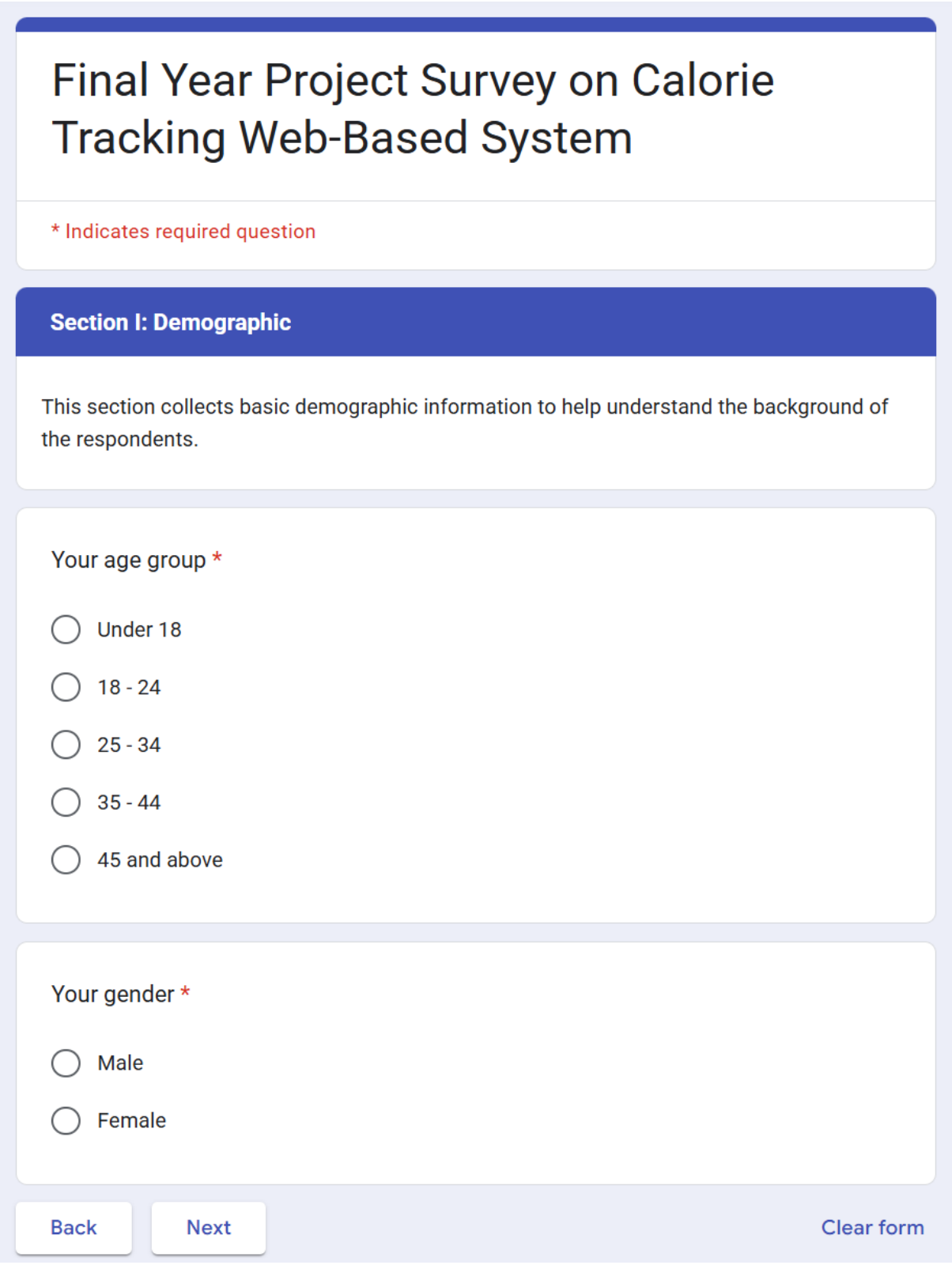
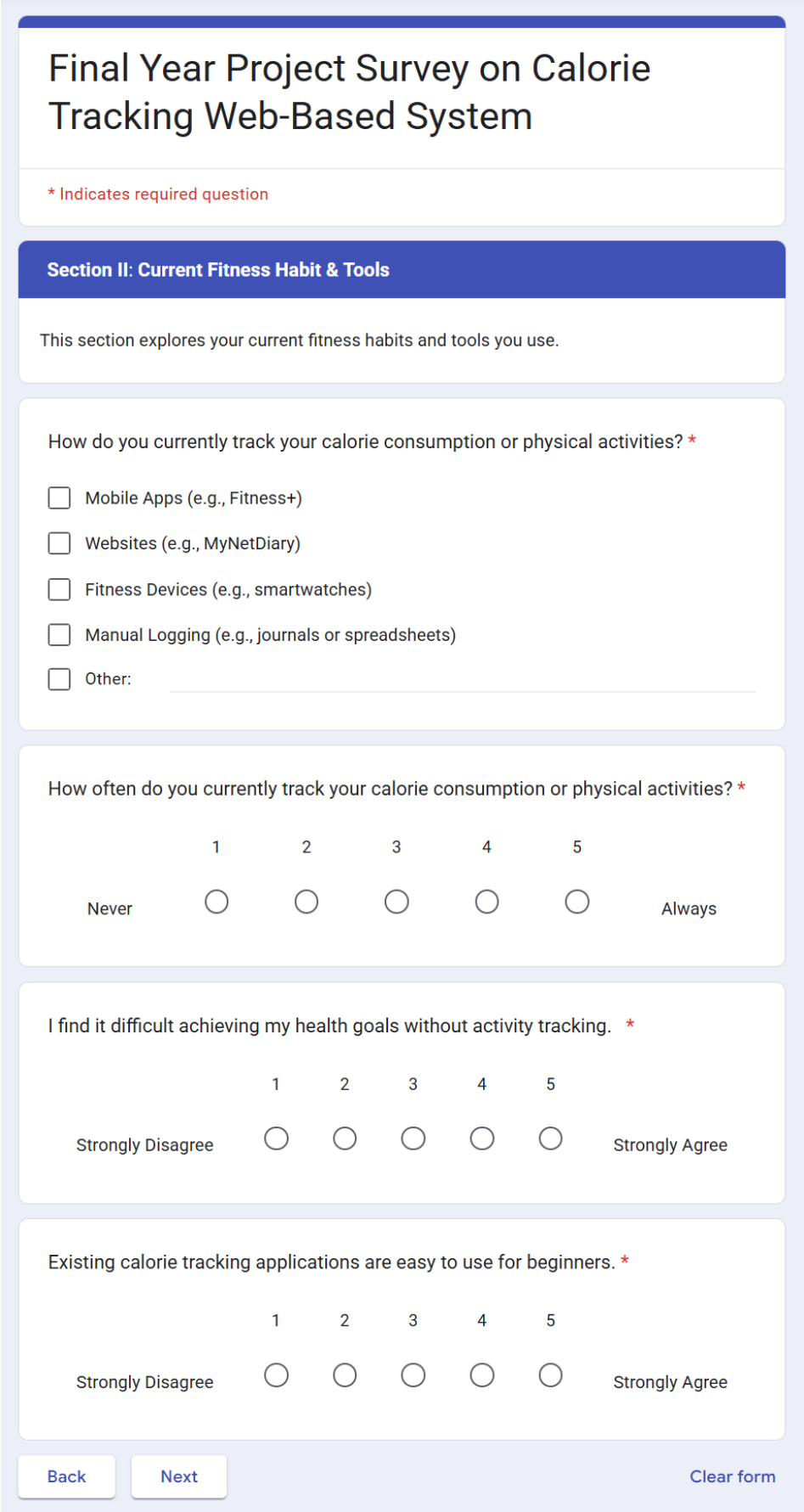


Figure A.2: Survey Section I: Demographic.


Final Year Project Survey on Calorie Tracking Web-Based System

* Indicates required question

Section II: Current Fitness Habit & Tools

This section explores your current fitness habits and tools you use.

How do you currently track your calorie consumption or physical activities? *

- [ ] Mobile Apps (e.g., Fitness+)
- [ ] Websites (e.g., MyNetDiary)
- [ ] Fitness Devices (e.g., smartwatches)
- [ ] Manual Logging (e.g., journals or spreadsheets)
- [ ] Other:

How often do you currently track your calorie consumption or physical activities? *

	1	2	3	4	5	
Never	○	○	○	○	○	Always

I find it difficult achieving my health goals without activity tracking. *

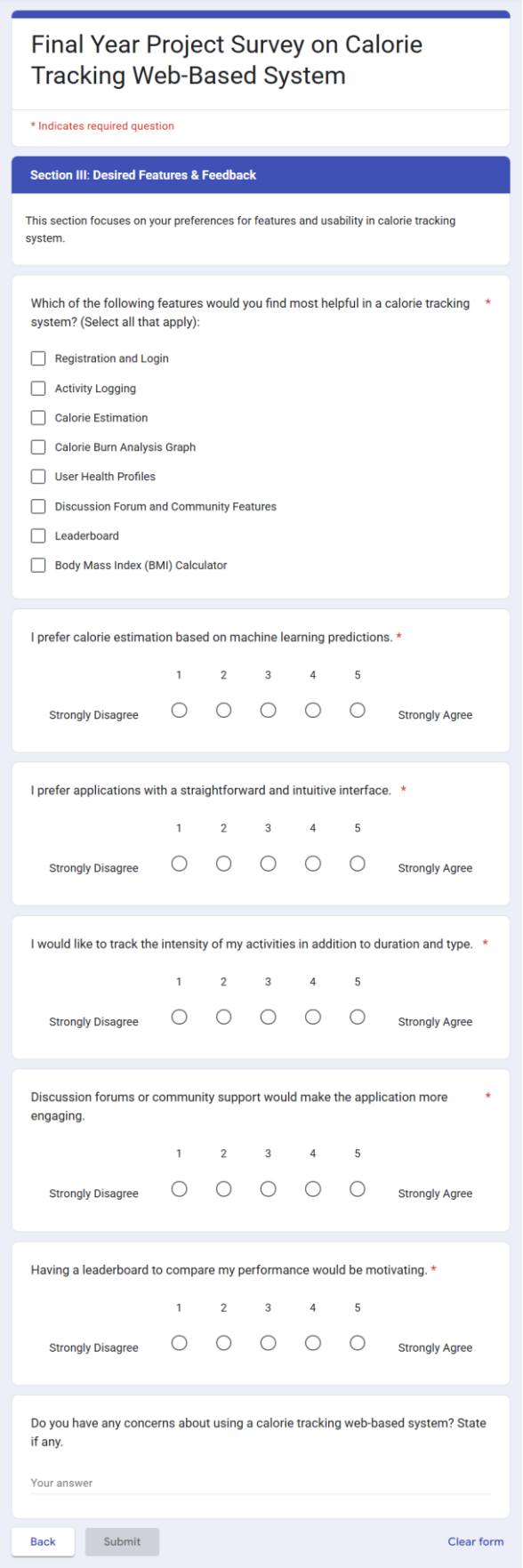
	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Existing calorie tracking applications are easy to use for beginners. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Back Next Clear form

Figure A.3: Survey Section II: Current Fitness Habit & Tools.



Final Year Project Survey on Calorie Tracking Web-Based System

* Indicates required question

Section III: Desired Features & Feedback

This section focuses on your preferences for features and usability in calorie tracking system.

Which of the following features would you find most helpful in a calorie tracking system? (Select all that apply): *

- [ ] Registration and Login
- [ ] Activity Logging
- [ ] Calorie Estimation
- [ ] Calorie Burn Analysis Graph
- [ ] User Health Profiles
- [ ] Discussion Forum and Community Features
- [ ] Leaderboard
- [ ] Body Mass Index (BMI) Calculator

I prefer calorie estimation based on machine learning predictions. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

I prefer applications with a straightforward and intuitive interface. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

I would like to track the intensity of my activities in addition to duration and type. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Discussion forums or community support would make the application more engaging. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Having a leaderboard to compare my performance would be motivating. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Do you have any concerns about using a calorie tracking web-based system? State if any.

Your answer

Back Submit Clear form

Figure A.4: Survey Section III: Desired Features & Feedback.

APPENDIX B

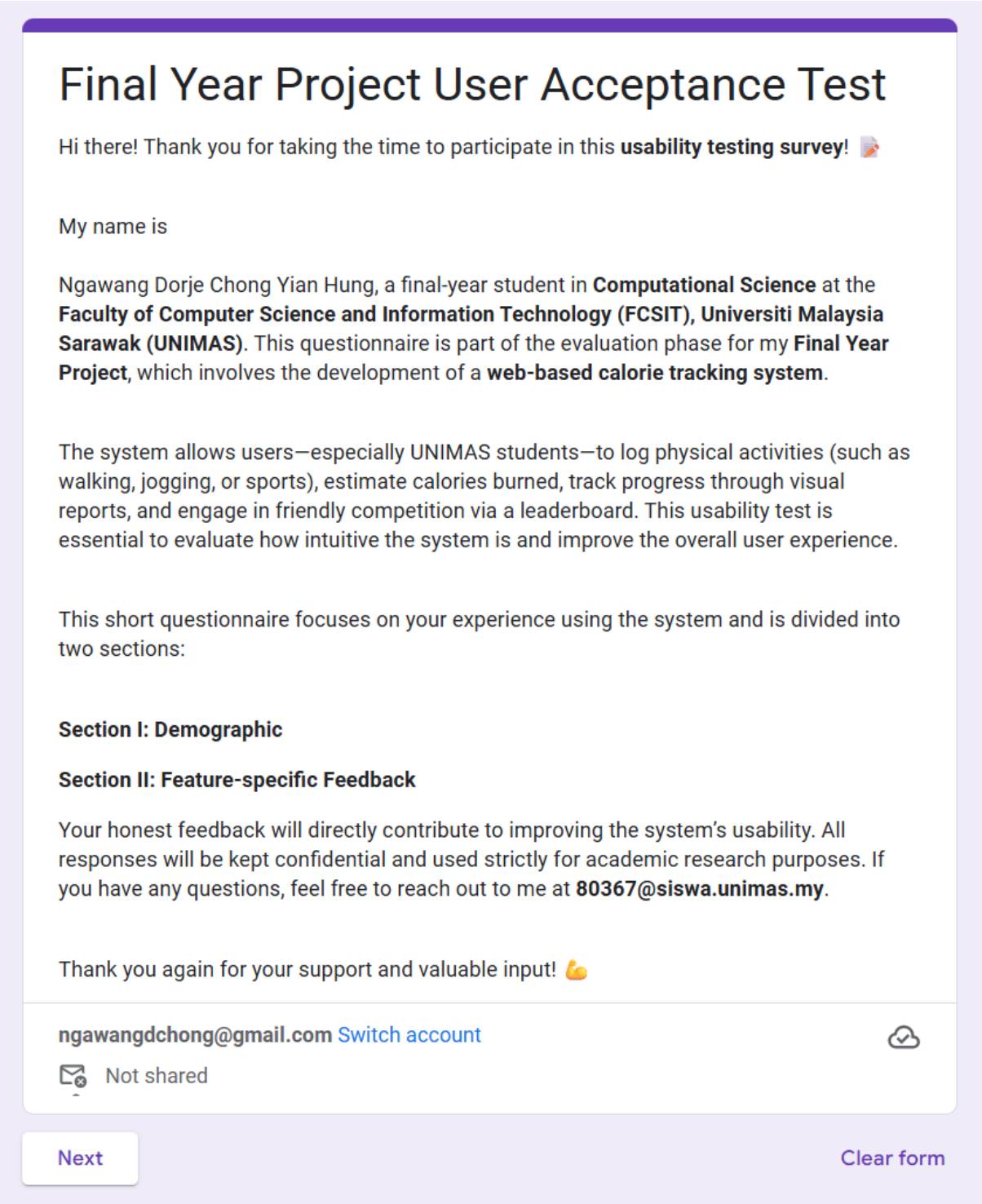


Figure B.1: Survey Form Description.

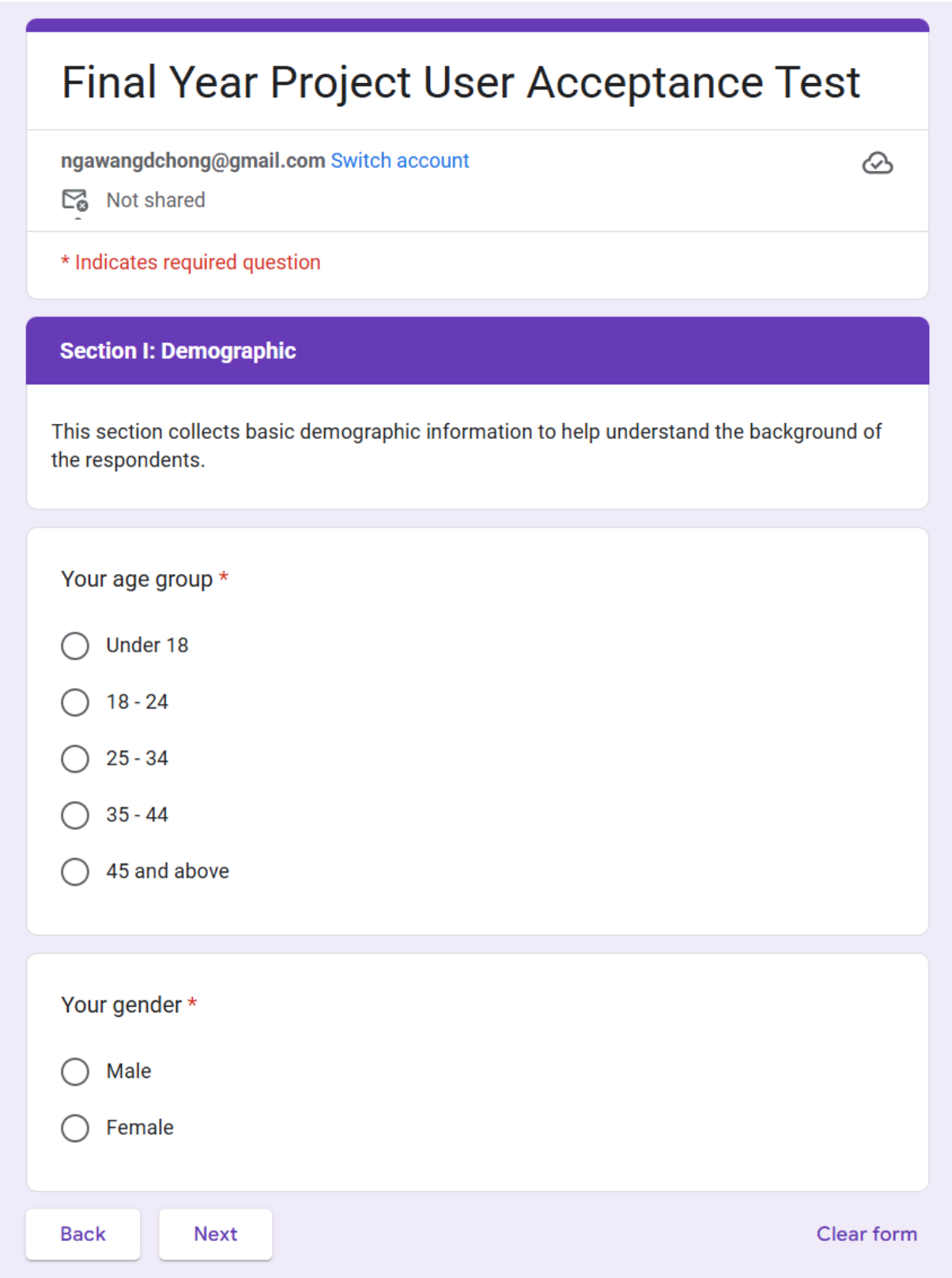


Figure B.2: Survey Section I: Demographic.

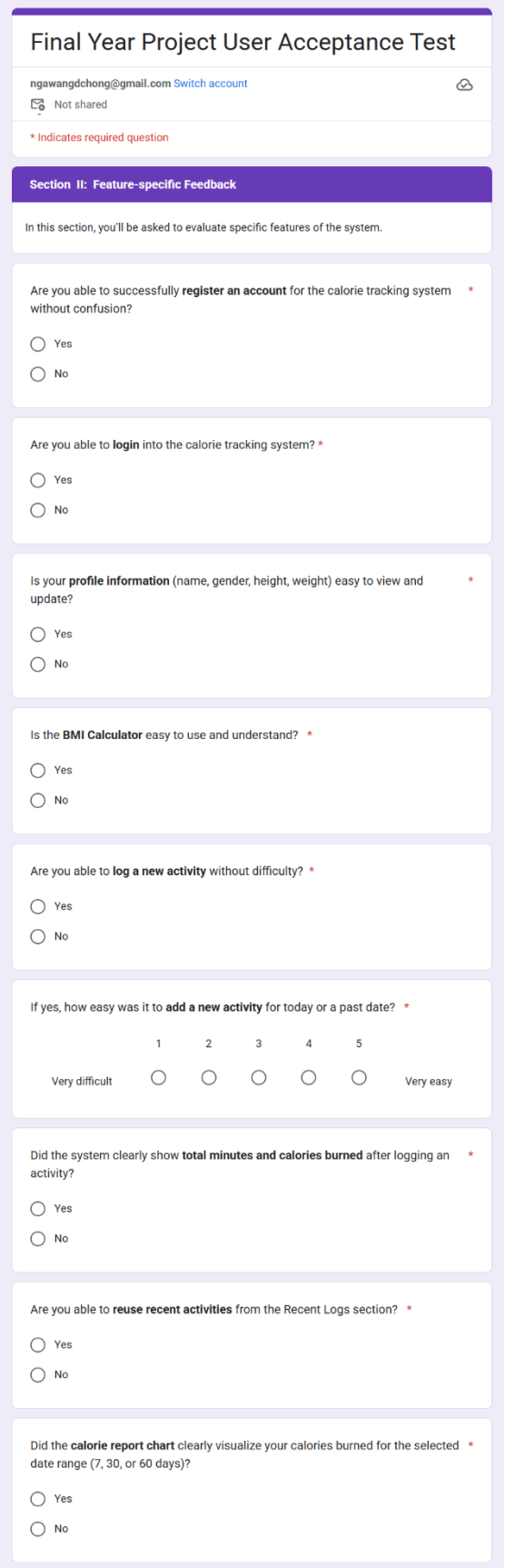


Figure B.3: Survey Section II: Feature-specific Feedback.

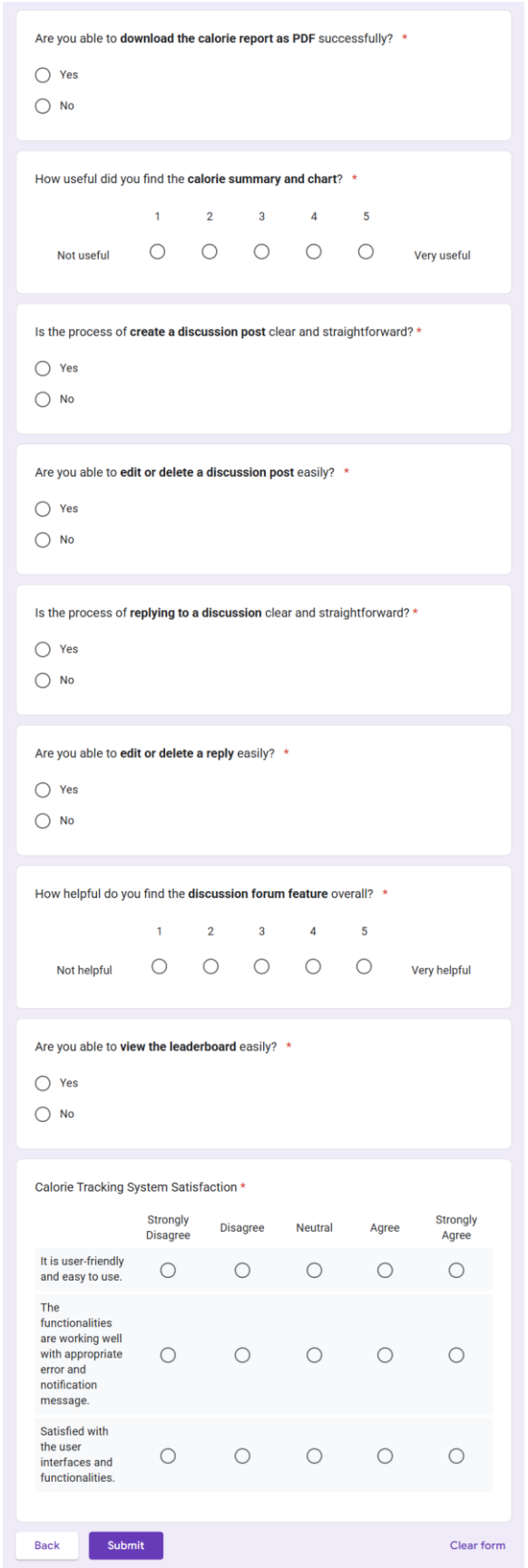


Figure B.4: Survey Section II: Feature-specific Feedback.