Sri Lanka Institute of Information Technology

Programming Applications and Frameworks (IT3030)

Continuous Assignment - 2024, Semester 2

Final Documentation

GROUP ID: JUNE-WE-28

Written by:

Pramathilaka S.P.D.M (IT21185298)

Ekanayaka H.E.M.P.L (IT21185502)

Karunarathna J.H.H.N (IT21157950)

Wijenandana S.D. (IT21158254)

Contents

1. Introduction

Flexfit is an advanced social networking platform that was created with the interest of fitness enthusiasts in mind. It allows users to exchange workout plans, record their fitness journeys, and offer tips on leading healthy lives. Customized perfectly, this allows users to share multimedia files, images or videos, to highlight their exercise regimens, food preferences, and advancements. The application guarantees a thorough representation of customers' fitness endeavors with an intuitive interface that permits direct upload from personal devices and possibilities to provide complete descriptions accompanying media posts. In addition, users may post updates in real time about how their training is going.

Flexfit places a high value on user freedom by providing post management tools like changing and removing content, as well as by encouraging interaction with likes and comments on posts. Flexfit's following features and profile visibility help to create a vibrant fitness community that inspires and unites members with all levels of technical proficiency. Notable features like post-engagement notifications and simplified user interfaces highlight application's dedication to user-friendliness and guarantee that every participant on their fitness journey has a positive experience.

2. Functional Requirements

For Rest API [1]

- User authentication - Authenticated users should be able to access their own data and be granted permission to carry out specific tasks. They should also be able to securely register, log in, and log out.
- Data Manipulation and Retrieval - Users should be able to create, edit, remove, and retrieve their profiles, posts, and comments using an API.
- Media Upload and Storage - APIs ought to enable the effective uploading, saving, and retrieving of media files.
- Notifications - When users receive likes or comments on their postings, for example, the API ought to be able to notify them.
- User Interaction - The API ought to enable user interactions, like browsing and following the profiles of other users. It should be possible for users to look for posts or people.

For Client Web Application

- Authentication and User Registration - Users use their social network or email accounts to safely log in.
- Post sharing - Users are allowed to submit images or videos that can be seen by other users.
- Workout Status Update - Users utilize to exchange workout updates.
- Workout Plan Sharing - Exercise Plans are shared and customized by users through Workout Plan Sharing
- Meal Plan Sharing - Users exchange meal plans that include servings, nutrition facts, and recipes.
- Comment Management - Available to users, who can also remove comments on their posts and edit or delete them.
- User Interaction - Users can engage with posts by liking and commenting on them. Further they can search for other users and interact with them.
- User profile Management - Every user has a profile with posts and activities pertaining to fitness which the information can be updated or modified.

3. Non-Functional Requirements

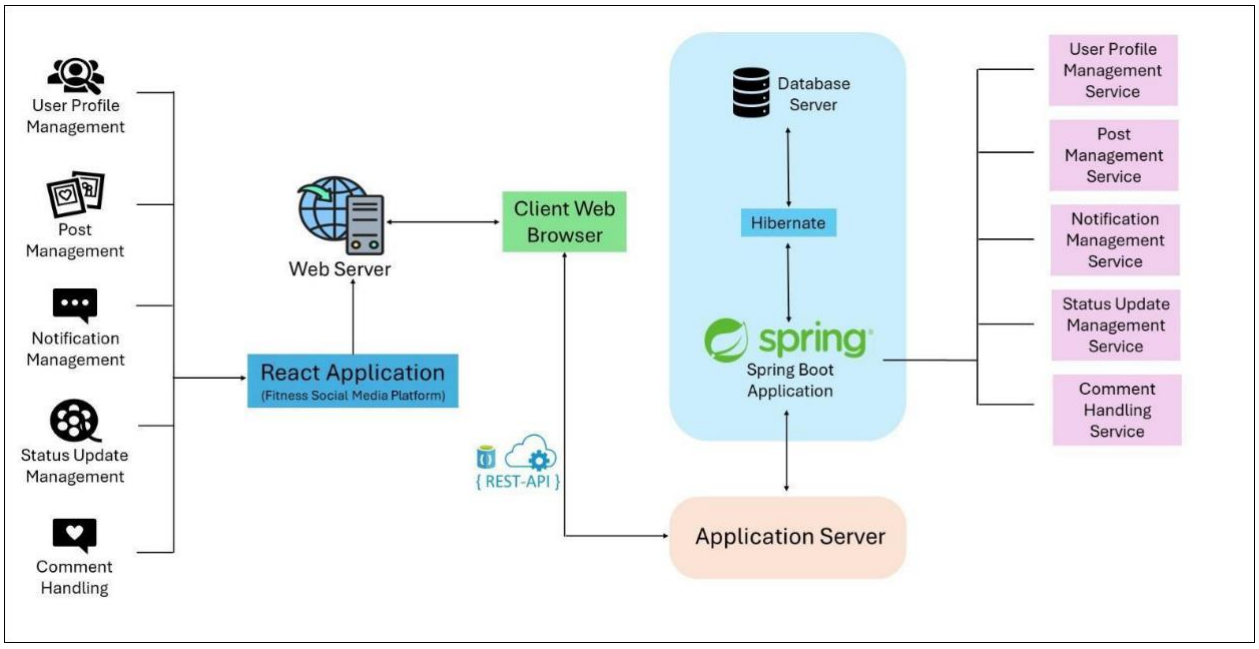
For Rest API

- Performance - A high volume of concurrent queries should be handled by the API with minimum delay. For a seamless user experience, API endpoint response times should be optimised.
- Security - To avoid unwanted access, user data should be sent securely over HTTPS. Strong authorization and authentication procedures are necessary to safeguard user privacy and stop data breaches.
- Scalability - To handle an expanding user base and rising traffic, the API should be built to scale horizontally.
- Reliability - To reduce downtime, the API should have redundant systems in place and be highly available (Disaster recovery plans).

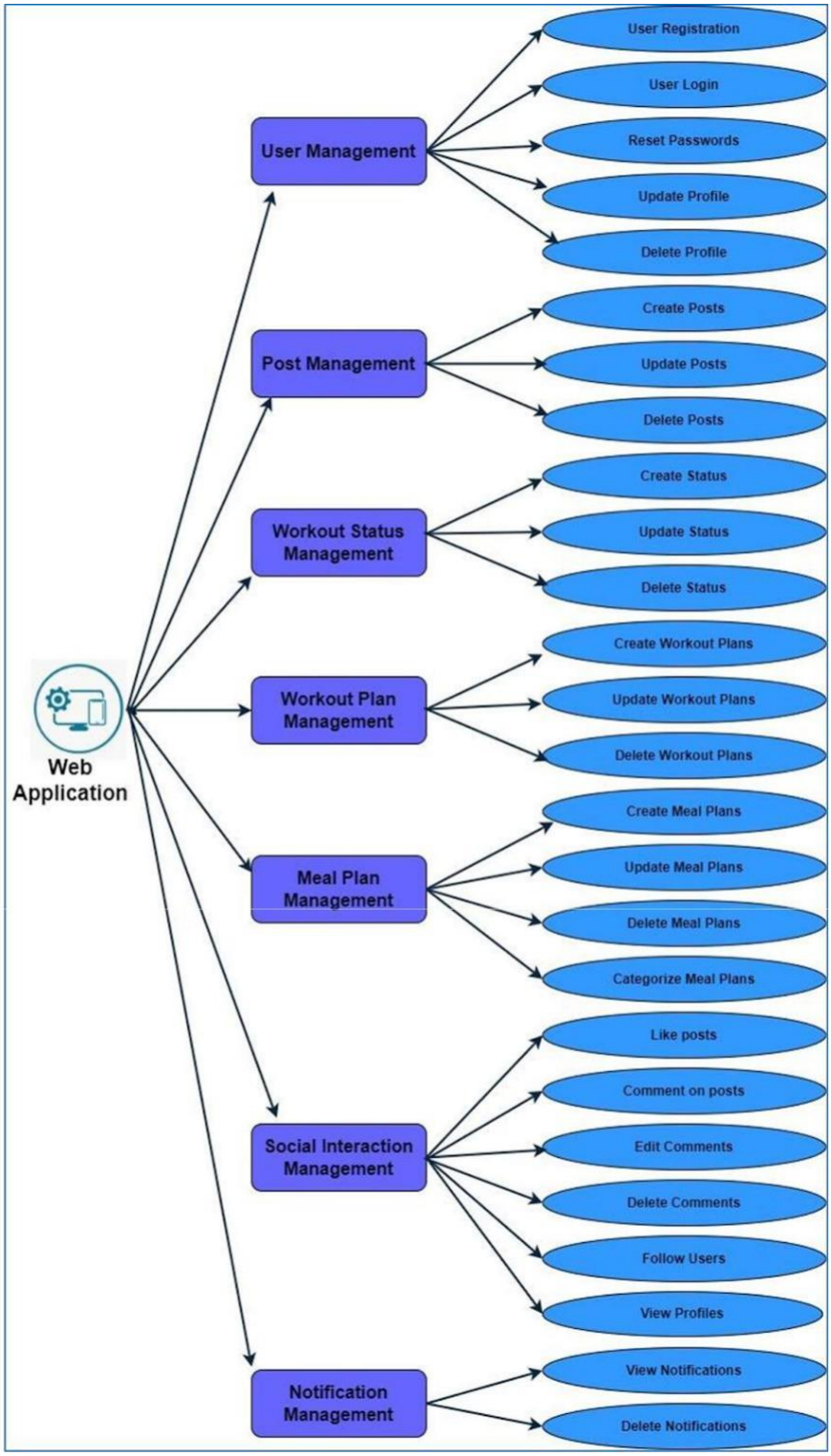
For Client Web Application

- Usability -A web application's user interface should be simple to use, visually appealing, and intuitive. It is important to have accessibility features to guarantee that users with impairments are included.
- Performance - Even on low-bandwidth connections, the page should load rapidly and react to user inputs right away.
- Compatibility - It should work with a variety of web browsers and gadgets, such as tablets, smartphones, and desktop computers.
- Security - Should adhere to online security best practices, which include defense against frequent flaws like XSS and CSRF attacks.

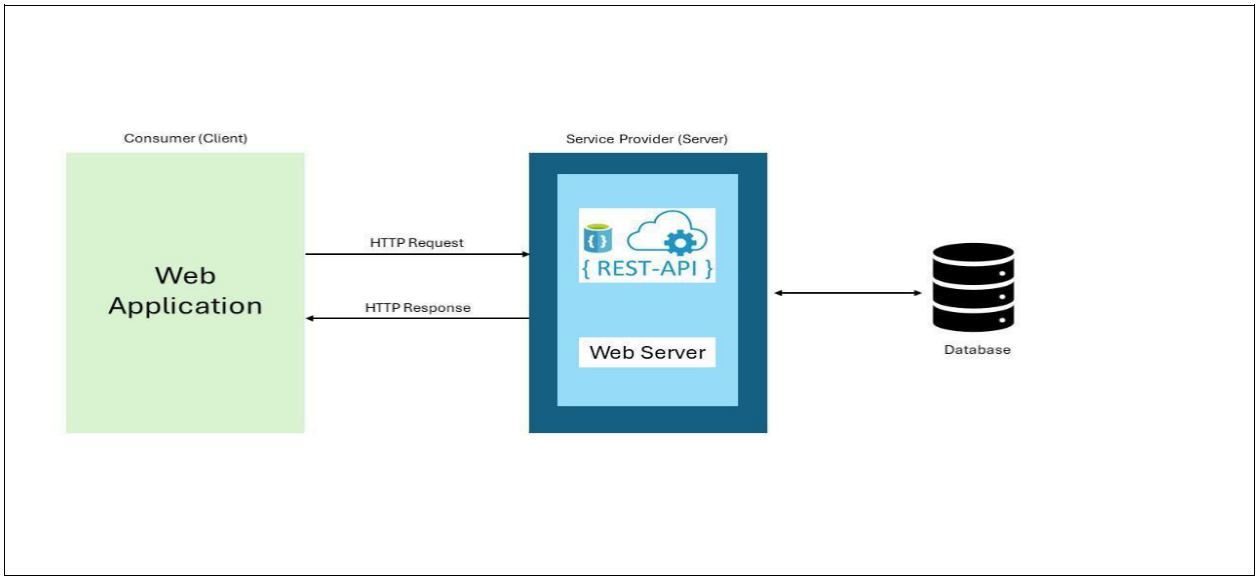
4. Overall Architecture Diagram



5. Architecture Diagram for the Client Web Application



6. Architecture Diagram for the Rest API



7. Design Details for Flexfit Application

Frontend Design

- **Authentication and User Registration:**

Social Login Integration: With social login integration, users are able to register or log in using their current Google accounts. Friction in the registration process is removed and this results in a seamless user experience.

Email Verification: Security for user accounts is ensured through email verification after they register and this prevents unauthorised access.

- **Post Sharing:**

Multimedia post Interface: Users can quickly post movies or pictures that highlight their workout routines. A straightforward drag-and-drop function makes information sharing effortless.

Rich Media Viewer: High-quality photos or videos accompany posts, boosting the platform's aesthetic appeal and promoting interaction.

- **Workout Status Update:**

Real-Time Updates: Users can share their workout progress in real-time, fostering a sense of community and accountability among members.

- **Workout and Meal Plan Sharing:**

Customization Options: Users are able to create and share meal plans and workout plans that are specific to their dietary requirements and fitness objectives. This stimulates the exchange of knowledge and the adoption of healthy living choices.

Nutrition Information Display: To help consumers make educated dietary choices, meal plans offer comprehensive nutritional information such as calorie counts, macronutrient breakdowns, and ingredient lists.

- **Comment Management:**

User interactions with comments: Viewers can express their opinions about posts by liking, and commenting on them. Meaningful conversations and interactions are made possible via a hierarchical comment system.

Tools for Comment Moderation: Users can control comments on their posts, including the power to remove or change them. This helps to maintain a courteous and positive environment for conversation.

- **Usability and Accessibility:**

Simple Navigation and Intuitive Controls: The interface is made to be easy to use, with controls that are easy to grasp. Inclusion for all users is ensured by accessibility features like screen reader compatibility.

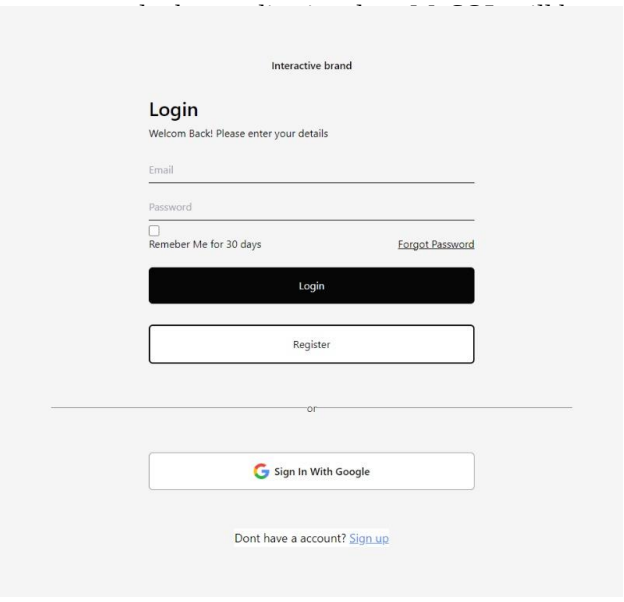
Backend Design

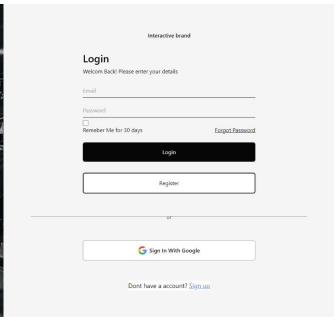
- **Authentication and User Registration:**

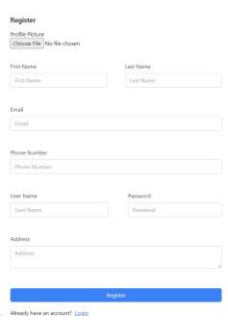
JWT Authentication: Encrypting user sessions and restricting unauthorised access to private information are two benefits of using JSON Web Tokens (JWT) for secure authentication.

- **Data Manipulation and Retrieval:**

RESTful API Endpoints: By utilizing RESTful API endpoints for CRUD activities (Create, Read, Update, Delete), users may effectively manage their posts, comments, and profiles.

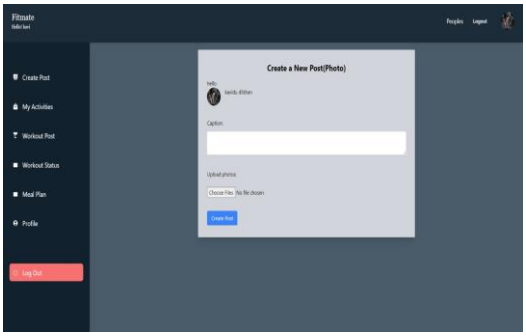


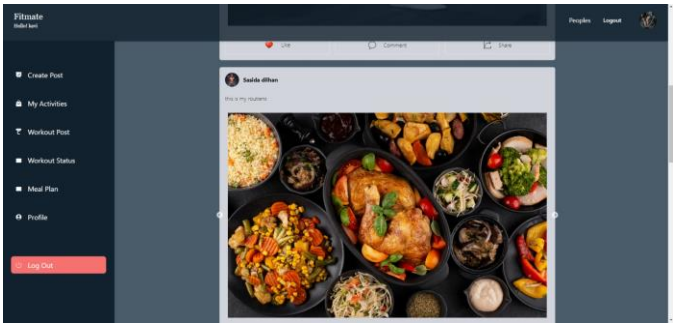




Post Management

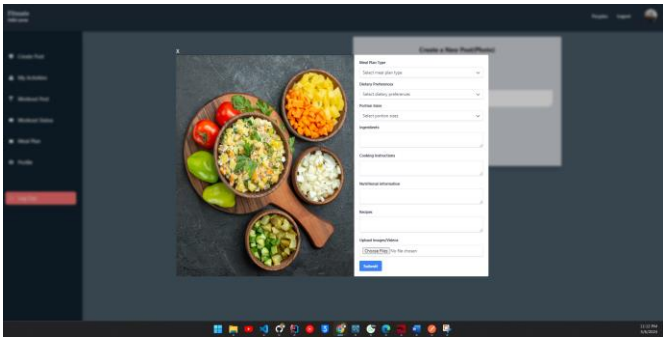
Allows users to post pictures & videos showcasing their fitness activities, workouts, healthy meals and progress. Once the users create posts of pictures or videos, they are displayed in the home page. The user management function also allows the users to edit or delete the posts they created.

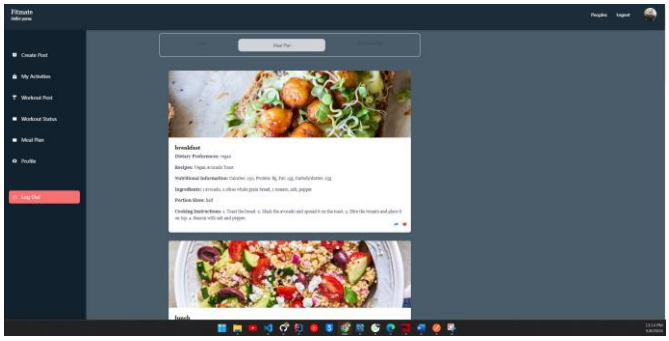




Meal Plan Management

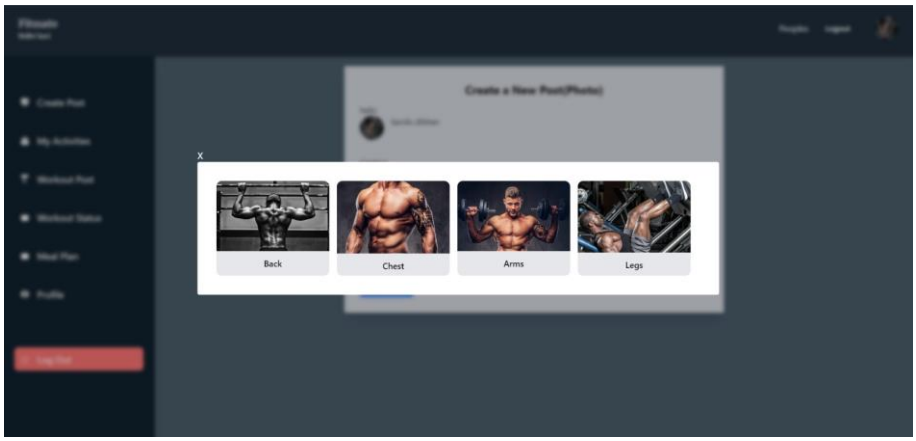
Users can share their meal plans, including recipes and nutritional information with their followers. All the created meal plan posts are displayed in the Meal Plans page. At any time, users have the ability to update or delete their meal plans.





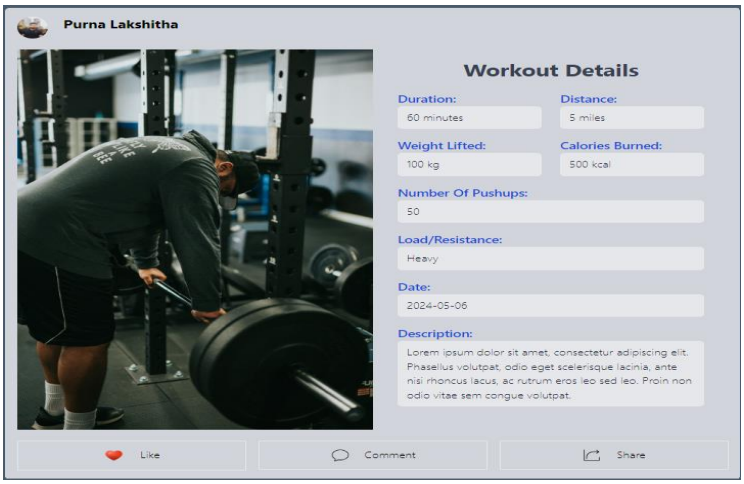
Workout Plan Management

Allows users to share their workout plans, including routines, exercises, sets and repetitions. Other followers can view these workout plans. Users can also edit and delete these plans whenever necessary.



Workout Status Management

Let the users create and share updates on their current workout status such as number of sets completed, distance run, weights lifted etc. These statuses can be viewed in the Workout Status Page, they can be updated and deleted at any time by the user.



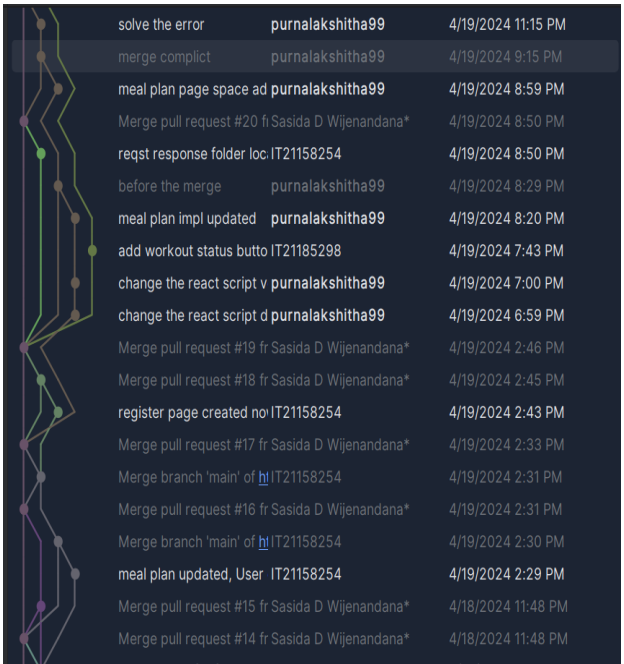
Comment Management

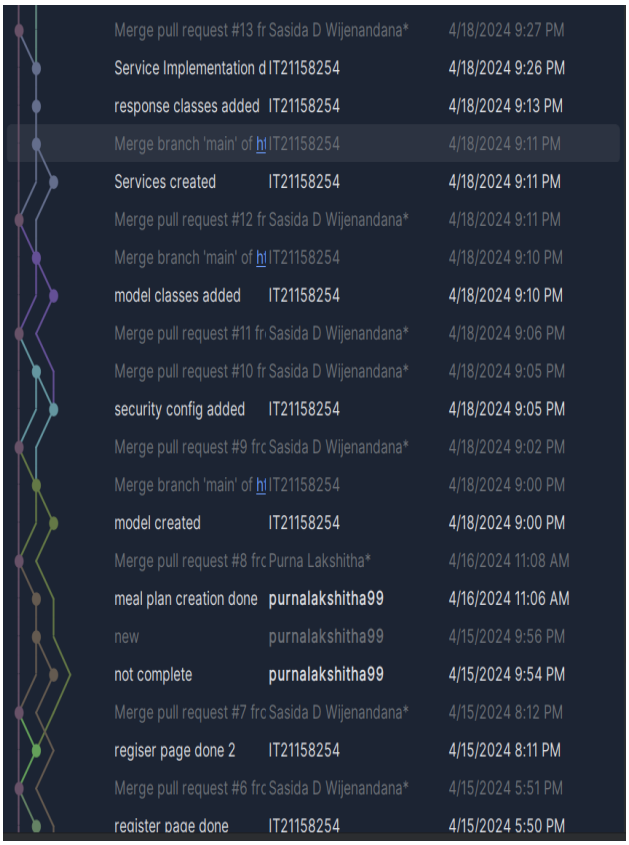
Users can easily create comments, which are then displayed in an organized and user-friendly format. Moderators have the ability to delete inappropriate comments, ensuring a safe environment for all users. Additionally, comments can be updated and deleted in real-time.

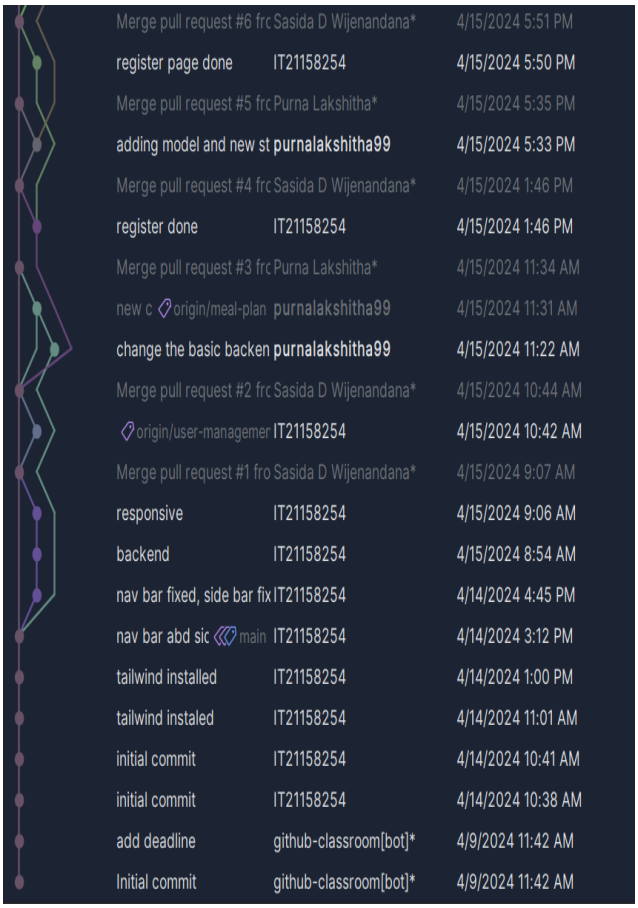
9. Branching & Commit Tree

GIT REPOSITORY LINK –

https://github.com/PAF-IT3030/paf-assignment-2024-jun_we_28.git




Merge pull request #13 fr Sasida D Wijenandana* 4/18/2024 9:27 PM
Service Implementation d IT21158254 4/18/2024 9:26 PM
response classes added IT21158254 4/18/2024 9:13 PM
Merge branch 'main' of ht IT21158254 4/18/2024 9:11 PM
Services created IT21158254 4/18/2024 9:11 PM
Merge pull request #12 fr Sasida D Wijenandana* 4/18/2024 9:11 PM
Merge branch 'main' of ht IT21158254 4/18/2024 9:10 PM
model classes added IT21158254 4/18/2024 9:10 PM
Merge pull request #11 fr Sasida D Wijenandana* 4/18/2024 9:06 PM
Merge pull request #10 fr Sasida D Wijenandana* 4/18/2024 9:05 PM
security config added IT21158254 4/18/2024 9:05 PM
Merge pull request #9 frc Sasida D Wijenandana* 4/18/2024 9:02 PM
Merge branch 'main' of ht IT21158254 4/18/2024 9:00 PM
model created IT21158254 4/18/2024 9:00 PM
Merge pull request #8 frc Purna Lakshitha* 4/16/2024 11:08 AM
meal plan creation done purnalakshitha99 4/16/2024 11:06 AM
new purnalakshitha99 4/15/2024 9:56 PM
not complete purnalakshitha99 4/15/2024 9:54 PM
Merge pull request #7 frc Sasida D Wijenandana* 4/15/2024 8:12 PM
regiser page done 2 IT21158254 4/15/2024 8:11 PM
Merge pull request #6 frc Sasida D Wijenandana* 4/15/2024 5:51 PM
register page done IT21158254 4/15/2024 5:50 PM


Merge pull request #6 frc Sasida D Wijenandana* 4/15/2024 5:51 PM
register page done IT21158254 4/15/2024 5:50 PM
Merge pull request #5 frc Purna Lakshitha* 4/15/2024 5:35 PM
adding model and new st purnalakshitha99 4/15/2024 5:33 PM
Merge pull request #4 frc Sasida D Wijenandana* 4/15/2024 1:46 PM
register done IT21158254 4/15/2024 1:46 PM
Merge pull request #3 frc Purna Lakshitha* 4/15/2024 11:34 AM
new c origin/meal-plan purnalakshitha99 4/15/2024 11:31 AM
change the basic backen purnalakshitha99 4/15/2024 11:22 AM
Merge pull request #2 frc Sasida D Wijenandana* 4/15/2024 10:44 AM
origin/user-managemer IT21158254 4/15/2024 10:42 AM
Merge pull request #1 fro Sasida D Wijenandana* 4/15/2024 9:07 AM
responsive IT21158254 4/15/2024 9:06 AM
backend IT21158254 4/15/2024 8:54 AM
nav bar fixed, side bar fix IT21158254 4/14/2024 4:45 PM
nav bar abd sic main IT21158254 4/14/2024 3:12 PM
tailwind installed IT21158254 4/14/2024 1:00 PM
tailwind instaled IT21158254 4/14/2024 11:01 AM
initial commit IT21158254 4/14/2024 10:41 AM
initial commit IT21158254 4/14/2024 10:38 AM
add deadline github-classroom[bot]* 4/9/2024 11:42 AM
Initial commit github-classroom[bot]* 4/9/2024 11:42 AM

10. References

[1] Akana, "How to define API requirements," 20 September 2020. [Online]. Available: https://www.akana.com/blog/api-requirements-what-consider.

11. Individual Contribution

Member	Contribution
Pramathilaka S.P.D.M (IT21185298)	Workout Status Management Side Bar
Karunarathna J.H.H.N(IT21157950)	Workout Plan Management
Ekanayaka H.E.M.P.L IT21185502	Meal Plan management Sign in
Wijanandana S.D(IT21158254)	Post Management User Management