Programming Applications and Frameworks (IT3030)

Assignment – 2024, Semester 1

Important details

- This assignment carries 30% of the final mark for the IT3030 module.
- This assessment is to be carried out as a group work, but each group members contribution will be assessed individually. Therefore, members of the same group may get different marks depending on their performance.
- Assignment release date: **16th March 2024**
- Submission deadline: **5.00pm, 3rd May 2023 (GMT +5.30) via Courseweb**

Description of the assignment

The end-goal in this assignment is to design and implement,

1. A Java (Spring Boot) based REST API for the given business scenario, utilizing the best practices learned throughout the module.
2. A React based client web application to let the intended users fulfill their needs by utilizing the designed system.

The business scenario

Your team, a software development company, has been approached by a client with an idea to develop a social media platform tailored for fitness enthusiasts to share their fitness journey, workouts, and healthy lifestyle tips.

The idea of the client is as follows.

- Users can post pictures and videos:
    - Allow users to upload up to 3 photos or videos (max length: 30 seconds) showcasing their fitness activities, workouts, healthy meals, and progress.
    - Provide an interface for users to select and upload media files directly from their devices.
    - Include options for users to add descriptions to accompany their media posts.
- Current workout status updates:
    - Enable users to create and share updates on their current workout status, such as distance ran, number of pushups completed, weight lifted, etc.
    - Provide predefined templates for users to input their workout metrics, making it easy to track and share their progress.
    - Each post should include a brief description of the fitness achievement (provided by the user).
- Workout plan sharing:
    - Allow users to share their workout plans, including routines, exercises, sets, and repetitions.
    - Provide templates for users to create and customize their workout plans within the platform.
    - Enable users to update their workout plans over time as their fitness goals and preferences evolve.

- Meal plan sharing:
    - Enable users to share their meal plans, including recipes, nutritional information, and portion sizes.
    - Provide tools for users to create and document their meal plans, including options for adding ingredients, cooking instructions, and photos of prepared meals.
    - Implement features for users to categorize their meal plans based on dietary preferences, such as vegetarian, vegan, keto, etc.
- Users must have the ability to edit or delete their posts as needed.
- Other users should be able to interact with a user's posts by liking and commenting.
- Profiles should be available for users for displaying all their fitness-related posts and activities.
- Users should be able to view and follow other users' profiles, with all profiles being publicly visible.
- Features for liking and commenting on posts, with the ability for users to edit or delete their comments.
- Post owners should have the ability to delete comments on their posts.
- Users should receive notifications for likes and comments on their posts.
- The platform should prioritize user-friendliness and simplicity for individuals with varying levels of technical expertise.
- The client intends to make the social media platform available as a client web application, as well as a mobile application (iOS/ Android). However, a separate company will be tasked with developing the mobile application. You are to be contracted to develop the REST API and the client web application of the platform. The REST API that is to be created by your company will be consumed by both the client web application and the mobile applications. Therefore, it is important that it is well engineered.
- Allow the users to login to the platform with their existing users accounts on well-known services such as social media.

The tasks in the assignment

1. Identify,
    a. The functional requirements
    b. The non-functional requirements

    for the REST API and the client web application separately.

2. Create an overall architecture diagram for the entire system you are contracted to design and implement (you can ignore the mobile applications).
3. Create detailed architecture diagram(s) for the REST API.
4. Create detailed architecture diagram(s) for the client web application. You should decide upon suitable front-end architecture.
5. Develop the REST API and the client web application to satisfy the identified requirements.
6. Test and validate that the,
    a. REST API is working independently as well as with the client application as per the requirements.
    b. The client application works as per the requirements.

Technical requirements

1. Spring Boot to be used for the REST API implementation.
2. React to be used to develop the client application.
3. Authentication/ Authorization should be handled using Spring Security and OAuth 2.0. You could use social logins.

Other requirements

1. The assignment should properly be version controlled using Git, on GitHub with GitHub Workflow.
2. A team is provided with the freedom to be creative with how the system is to be designed and developed, especially going beyond the given basic requirements in order to make the application more secure, stable, and maintainable from a technical perspective with good UI/ UX.
3. Each member must create at least four (4) REST endpoints, each with a distinct HTTP method (e.g., GET, POST, PUT, DELETE).

Mark breakdown

Documentation	Total: 15 marks
• Initial document (23rd March)	05 marks
• Final Documentation (3rd May)	10 marks
In-Class Progress Review	Total: 10 marks
• Progress review (one week starting 15th April onwards)	10 marks
REST API	Total: 30 marks
• Proper endpoint naming	05 marks
• Follows the six REST architectural styles	10 marks
• Proper usage of HTTP methods and status codes	05 marks
• Good code quality following Java/ Spring coding conventions	05 marks
• Requirements satisfaction	05 marks
Client Web Application	Total: 15 marks
• Proper architectural design and implementation	05 marks
• Requirements satisfaction	05 marks
• Good UI/ UX	05 marks
Version controlling	Total: 10 marks
• Proper usage of Git (good commit messages, proper branching etc.)	05 marks
• Proper usage of the GitHub Workflow	05 marks
Implementing OAuth Authentication	10 marks
Overall Creativity	10 marks
Total	**100 marks**

Special Notes

- All students are strictly expected to maintain academic integrity and honesty.
- The goal of this assignment is to check a team's ability to create a simple product as per the modern standards discussed in the module.
- The group can decide how the work breakdown is done among group members. Each member of the team will be graded individually. Therefore, make sure that each member has a sufficient workload.
- Each group should submit an Initial document on tasks 1 to 4, as outlined in the "The tasks in the assignment" section. The designs made here are understood to be temporary, and if there is a change it can be mentioned in progress documents and in the final document.
- During the in-class progress review, you are expected to be able to show your progress and explain the design and technical decisions that led to your project's current status.
- Use of AI generated code for your submission e.g. GitHub CoPilot, ChatGPT, Gemini, GhostWriter or any similar tools is allowed. However, it is required to disclose this information in the in-class progress review and the viva. Keep records of detailed prompts and responses you have used for the AI Agent. You are expected to be able to **clearly explain** what AI generated code does during your progress review and viva.
- Version controlling should happen on a GitHub classroom repository which will be provided by the PAF team. **Do not** use your private GitHub accounts for this purpose. Instructions on this will be shared separately.
- For final assessment submissions, a group must provide,
    - A final report including,
        - all the requirement details (functional/ non-functional)
        - the design details for front-end and backend with reasons for the design decisions as outlined in tasks 1 to 4 in "The tasks in the assignment" section.
        - A detailed record of usage of AI agents, including the submitted prompts and received responses.
        - The report's length is limited to a maximum of 12 pages including the designs.
    - All the above should be compressed as a .zip folder and uploaded to the Courseweb submission link by 5.00pm, 3rd May 2024 (GMT +5.30). Name the .zip file with your group ID.