

Text based social media (Mid).

Description: The goal of this project is to create a text-based social media API that allows users to register, login, post text, comment on posts, follow other users, and view different types of posts and comments. The API will be used by a front-end application, such as a single-page app.

The API is expected to have up to 500 active users and will have two types of users: Free and Premium. During registration, users can select their plan type, and the username will be their email address. Payment involvement will be ignored at this stage. Posts are expected to be text only.

Requirements:

- 1) User Registration, Login and User Roles
 - Implement a secure registration system using email as the username and a password (bonus points: implement a password policy and an email validation system).
 - Implement a login system to authenticate users.
 - Support two user roles: Free and Premium (Allow users to select their role during signup).
 - Store user and role information in the database.
- 2) Posts and Comments
 - Free users can post small text with a limit of 1000 characters.
 - Premium users can post text with a limit of 3000 characters.
 - Free users can comment up to 5 times per post.
 - Premium users can comment unlimited times.
 - All responses will be flat under the original post (no replies or thread chaining).
 - Store the posts and comments in the database.
- 3) Following and followers
 - Store follower relationships: Maintain a list of followers for each user.
 - Add and remove followers.
- 4) Views
 - View all original posts by the people they follow, ordered by reverse chronological order.
 - View their own posts, including the latest 100 comments, sorted by reverse chronological order.
 - View all comments on their own posts.
 - View the latest comments on all posts they have posted or any posts by the people they follow.
 - Retrieve followers and following: View the list of followers for a user.
 - Search for a user to follow.
- 5) Shareable Link (Bonus): Users can create a shareable link that anyone can see containing only the original post and the latest 100 comments (at the time of view).

Include unit and/or integration tests to ensure functionality and security as a bonus task.

Additional Requirements:

- Ensure robust security mechanisms to prevent unauthorized access.

- Include a script with schema and data for database creation. Bonus points: Implement automated database schema generation.
- Document the REST endpoints for ease of use as a bonus task.

Tech Stack:

- Java 11 or 17.
- Maven as the build tool.
- Any Java web framework of your choice.
- PostgreSQL for data storage. You can use any desirable data access or ORM framework or library.

Submission:

- Share the project's GitHub repository.
- Include a README file detailing implemented features, tech stack, and usage instructions.

Deadline: Deliver the project within one week from the day you received it. If you need extra time, please let us know. You can have pre-registered users if you need to save time.

Notes: Approach the project innovatively while adhering to the specified functionality.