Developer - Youtube Video Sharing App

We are seeking a developer to build a web app for sharing YouTube videos.

- **For Backend and Fullstack applicants:** using any languages/frameworks (Ruby on Rails is preferred). The application should include a real-time notification system to showcase your backend development skills.
- **For Frontend applicants:** Feel free to use any backend tech of your choice.

Key Features

FE applicants should accomplish the first 3 features. While BE and Fullstack applicants must complete features 4.

1. User registration and login
2. Sharing YouTube videos
3. Viewing a list of shared videos (no need to display up/down votes)
4. Real-time notifications for new video shares: When a user shares a new video, other logged-in users should receive a real-time notification about the newly shared video. This notification can be displayed as a pop-up or a banner in the application, and it should contain the video title and the name of the user who shared it.

Instructions

1. Develop a simple UI to demonstrate your backend functionality.
2. Implement the real-time notifications feature using WebSockets and background jobs.
3. Create a detailed README.md file with clear instructions for setting up the project on the reviewer's machine (see below for details).
4. Deploy the application and include the link to the site when submitting the project.
5. Plus point for full Docker configuration to run locally.
6. Submit the links to https://remi.group/project-for-remitano-developer-submission

Technical Requirements

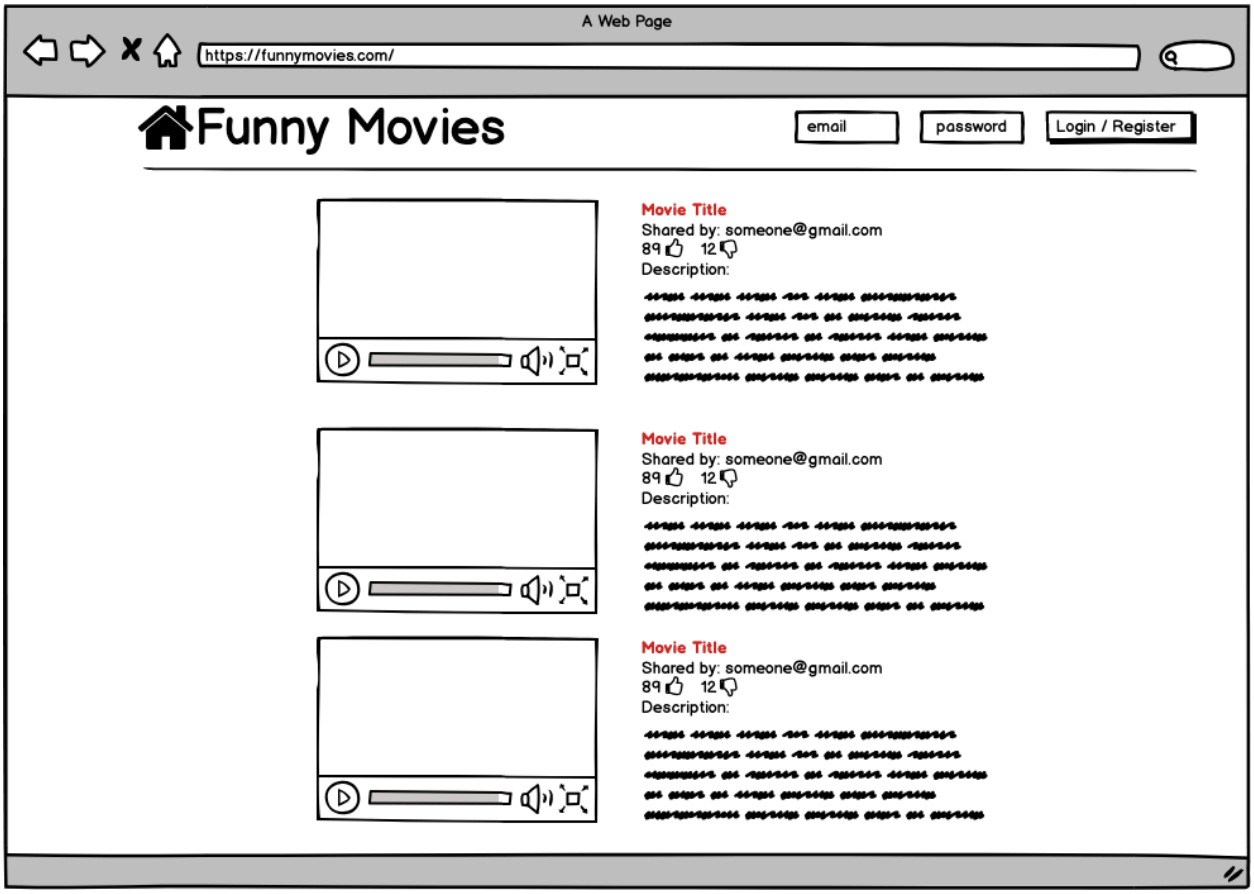
- Use Git with frequent commits
- If you're applying for the FS/FE developer position, **using React for frontend is a must**.
- For FE applicants: must use Typescript/React and use responsive design for the frontend.
- Include unit tests
- Include unit tests (again)
- Include integration test

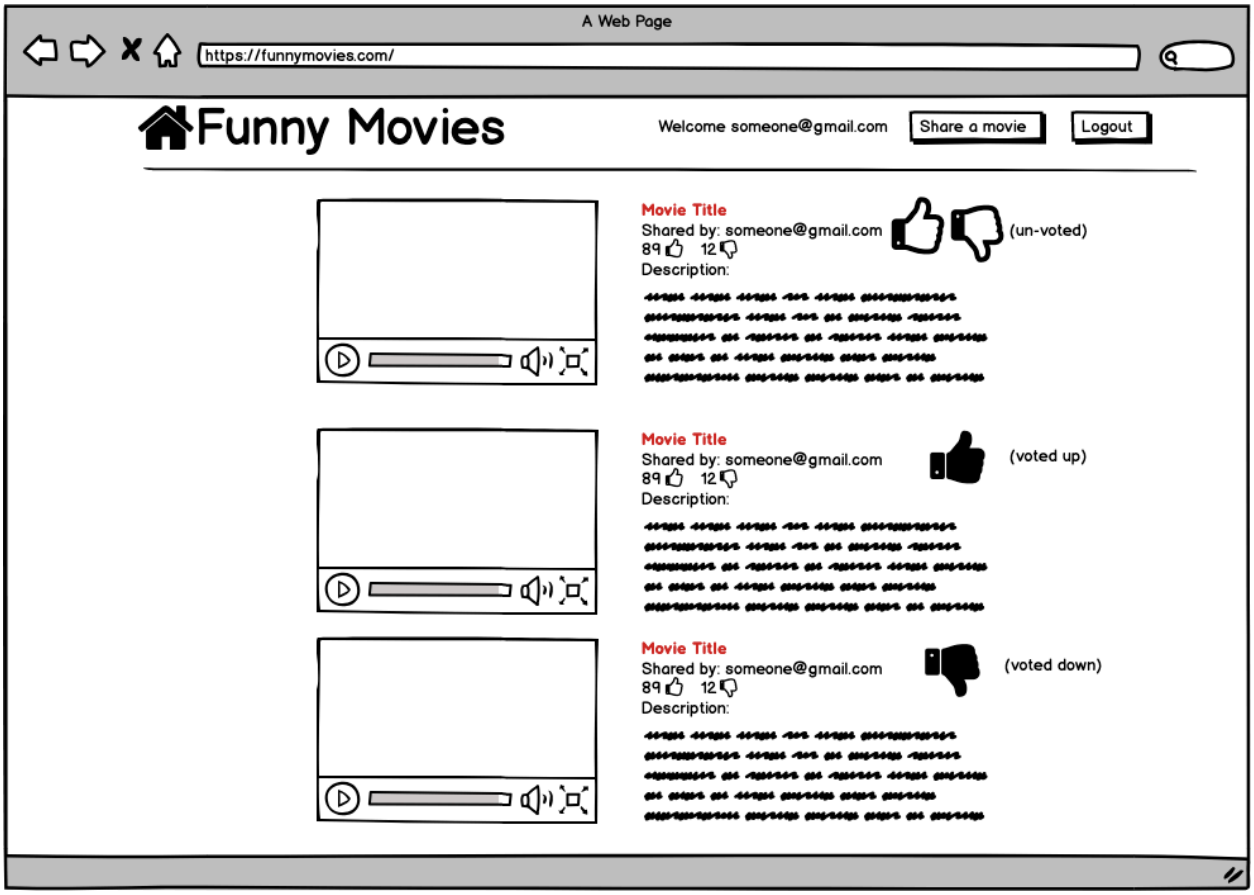
README.md Guidelines

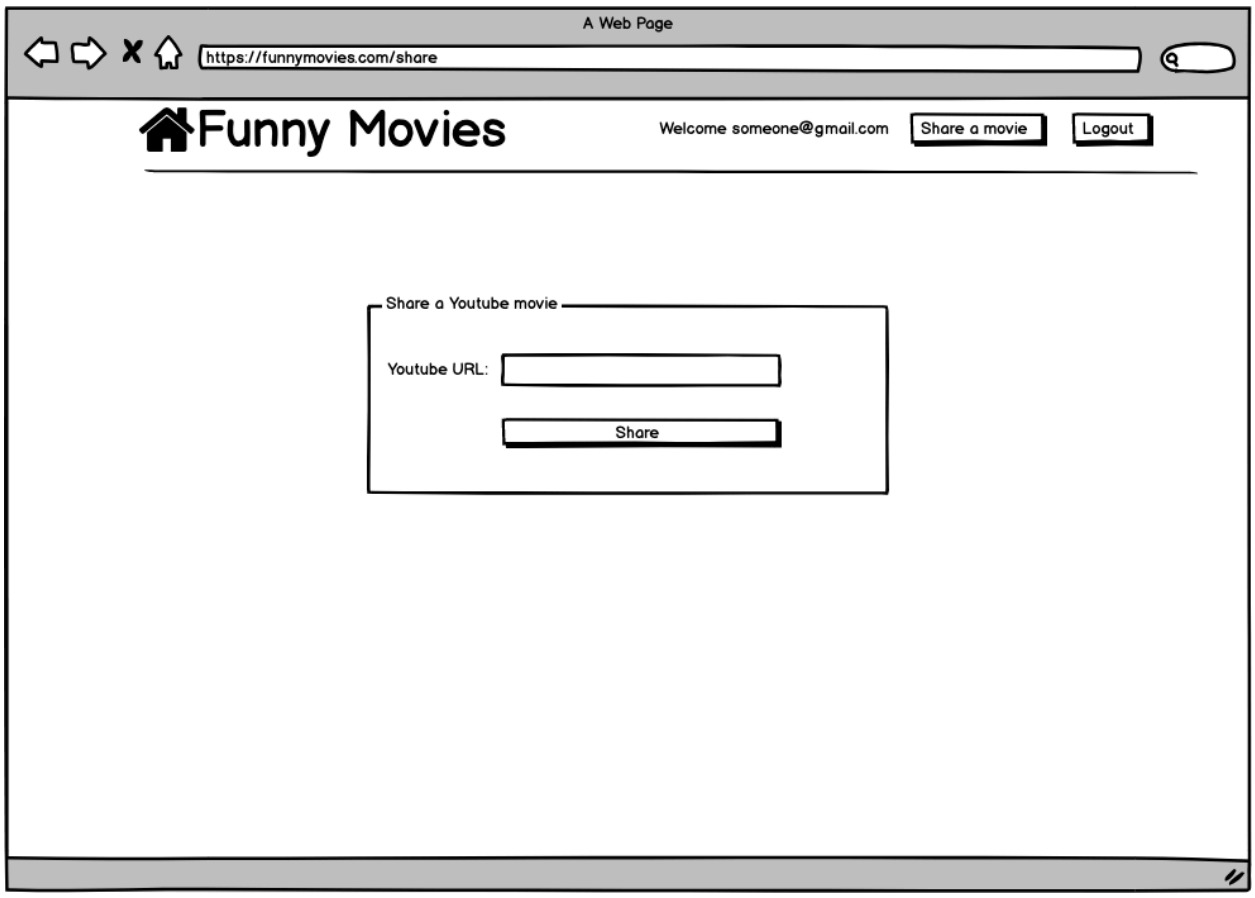
Please ensure your README.md file includes the following sections:

1. Introduction: A brief overview of the project, its purpose, and key features.
2. Prerequisites: List required software and tools, along with their appropriate versions.
3. Installation & Configuration: Step-by-step instructions for cloning the repository, installing dependencies, and configuring settings.
4. Database Setup: Instructions for setting up the database, running migrations, and seeding data if necessary.
5. Running the Application: How to start the development server, access the application in a web browser, and run the test suite.
6. (BE/FS) Docker Deployment: Instructions for deploying the application using Docker, including building the Docker image and running containers (optional for Backend developers)
7. Usage: A brief guide outlining how to use the application, including any specific features or functionality the reviewer should be aware of.
8. Troubleshooting: Common issues that may arise during setup and their potential solutions.

Please ensure **comprehensive** and **effective testing**, as we will closely examine the unit test section. Including a well-structured README.md and showcasing Docker deployment skills will demonstrate your ability to document your work effectively, create user-friendly guidelines, and deploy modern applications.
Sample Wireframes:


A Web Page
https://funnymovies.com/
Funny Movies
email
password
Login / Register
Movie Title
Shared by: someone@gmail.com
89 12
Description:
Movie Title
Shared by: someone@gmail.com
89 12
Description:
Movie Title
Shared by: someone@gmail.com
89 12
Description:


A Web Page
https://funnymovies.com/
Funny Movies
Welcome someone@gmail.com
Share a movie
Logout
Movie Title
Shared by: someone@gmail.com
89 12
Description:
(un-voted)
Movie Title
Shared by: someone@gmail.com
89 12
Description:
(voted up)
Movie Title
Shared by: someone@gmail.com
89 12
Description:
(voted down)


A Web Page
https://funnymovies.com/share
Funny Movies
Welcome someone@gmail.com
Share a movie
Logout
Share a Youtube movie
Youtube URL:
Share