This Week's Project Update Team 3

This week we made progress laying the groundwork for our message board application:

- Divided responsibilities into four clear modules: Setup & Authentication; Boards & Threads; Posts, Comments & Likes; and Polish, Security & Testing.
- Initialized the GitHub repository and committed the initial project skeleton.
- Created and configured Docker Compose for the app, database, and Redis containers.
- Verified all containers start successfully and established the development environment.

Message Board Project Breakdown & Responsibilities

This document outlines how we will split the message board project into four equal parts. Each part includes a concise description of the responsibilities and tasks to be owned by each team member.

Chunk 1: Setup & Authentication

- Initialize the Git repository and Docker Compose setup (app, database, Redis).
- Create and configure environment variables and .gitignore.
- Design and implement the User model (id, email, password_hash, etc.).
- Develop registration and login API routes and associated UI forms.
- Integrate bcrypt (or Argon2) for secure password hashing.
- Set up session management using express-session with Redis or JWT with refresh tokens.

Chunk 2: Boards & Threads

- Define the Board model (id, name, description) and Thread model (id, board_id, user_id, title, created_at).
- Implement CRUD API endpoints for boards and threads.
- Create service layer functions and authentication middleware for these endpoints.
- Build basic UI pages: home page listing boards and board view showing threads.

Chunk 3: Posts, Comments & Likes

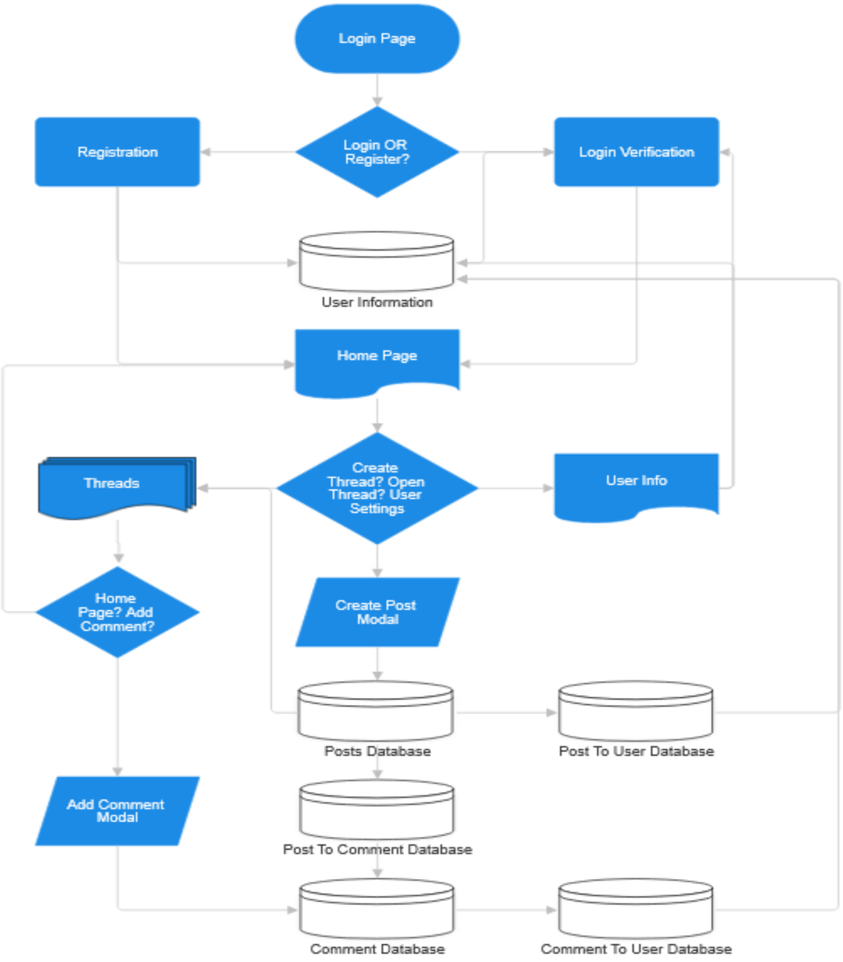
- Design the Post model (id, thread_id, user_id, body, created_at) and Comment model (id, post_id, user_id, body, created_at).

- Implement CRUD API endpoints for posts and comments.
- Create endpoints for like and unlike functionality and define the like data model.
- Develop UI components for the thread view: display posts, comments, reply form, and like buttons.

Chunk 4: Polish, Security & Testing

- Add input validation and sanitization middleware to prevent XSS and SQL injection.
- Implement rate-limiting using Redis and enforce role-based access control (user vs. admin).
- Finalize UI styling for responsive design and build an optional user profile page.
- Write unit and integration tests, perform bug fixes, and update project documentation.

Project Workflow Diagram


Login Page
Registration
Login OR Register?
Login Verification
User Information
Home Page
Threads
Create Thread? Open Thread? User Settings
User Info
Home Page? Add Comment?
Create Post Modal
Posts Database
Post To User Database
Add Comment Modal
Post To Comment Database
Comment Database
Comment To User Database