**Title: DebateHub**

**A platform for the Nuanced Discussion**

DebateHub is a web application where users engage in nuanced discussions on various topics. Users can post debate questions with multiple viewpoints, vote on them, and interact with other users in a dynamic, communitydriven environment.

**TRAINER: By**

**Shilpa Pingali,Infosys Sanjana Mittapally**

**1. Problem Statement**

Modern social platforms often lack the depth and structure for meaningful discussions. DebateHub solves this by creating a platform for structured debates and discussions where users can post questions (debate topics) with 2 to 7 viewpoints (options). These debates are community-driven, allowing users to vote on options, express preferences, and engage meaningfully.

Key Features:

* Create debates with multiple viewpoints.
* Vote distribution visualized graphically.
* Search, filter, and sort debates.
* Social features like liking/disliking debates.
* Admin moderation for debates and user activity.

Outcome: Users experience a community-oriented space for nuanced debates, and admins ensure the platform remains well-regulated and engaging.

**2. Tech Stack**

**Frontend**: ReactJS (with Vite for enhanced development performance).  
**Backend**: Node.js with Express.js.  
**Database**: MongoDB for scalable and efficient data storage.  
**Template Engine**: EJS for dynamic rendering of pages (if necessary).

**3. External Libraries**

1. **bcrypt** (^5.1.1): For secure password hashing.
2. **cors** (^2.8.5): To handle Cross-Origin Resource Sharing, enabling the frontend to interact with the backend.
3. **dotenv** (^16.4.5): Manages environment variables securely.
4. **ejs** (^3.1.10): Template engine for rendering HTML with embedded JavaScript (optional for server-side rendering).
5. **express** (^4.21.2): Web framework for building RESTful APIs.
6. **express-validator** (^7.2.0): Middleware for input validation and sanitization.
7. **jsonwebtoken** (^9.0.2): For generating and verifying secure JSON Web Tokens (JWT).
8. **mongoose** (^8.8.1): ODM library for interacting with MongoDB.
9. **nodemailer** (^6.9.16): For sending emails, such as account confirmations or notifications.
10. **Nodemon** (^3.1.7): Development tool for auto-restarting the server on code changes.

**MODULES**

**USER MODULE:**

* **Create Debates:** Users can create new debates by specifying a question and 2-7 viewpoints (options).
* **Vote on Debates**: Each user can cast up to 10 votes across debate options, with graphical displays of vote distributions.
* **Search Debates**: Users can search for existing debates and view questions, options, and current voting stats.
* **Like/Dislike Debates**: Users can like or dislike a debate question to express their opinion.

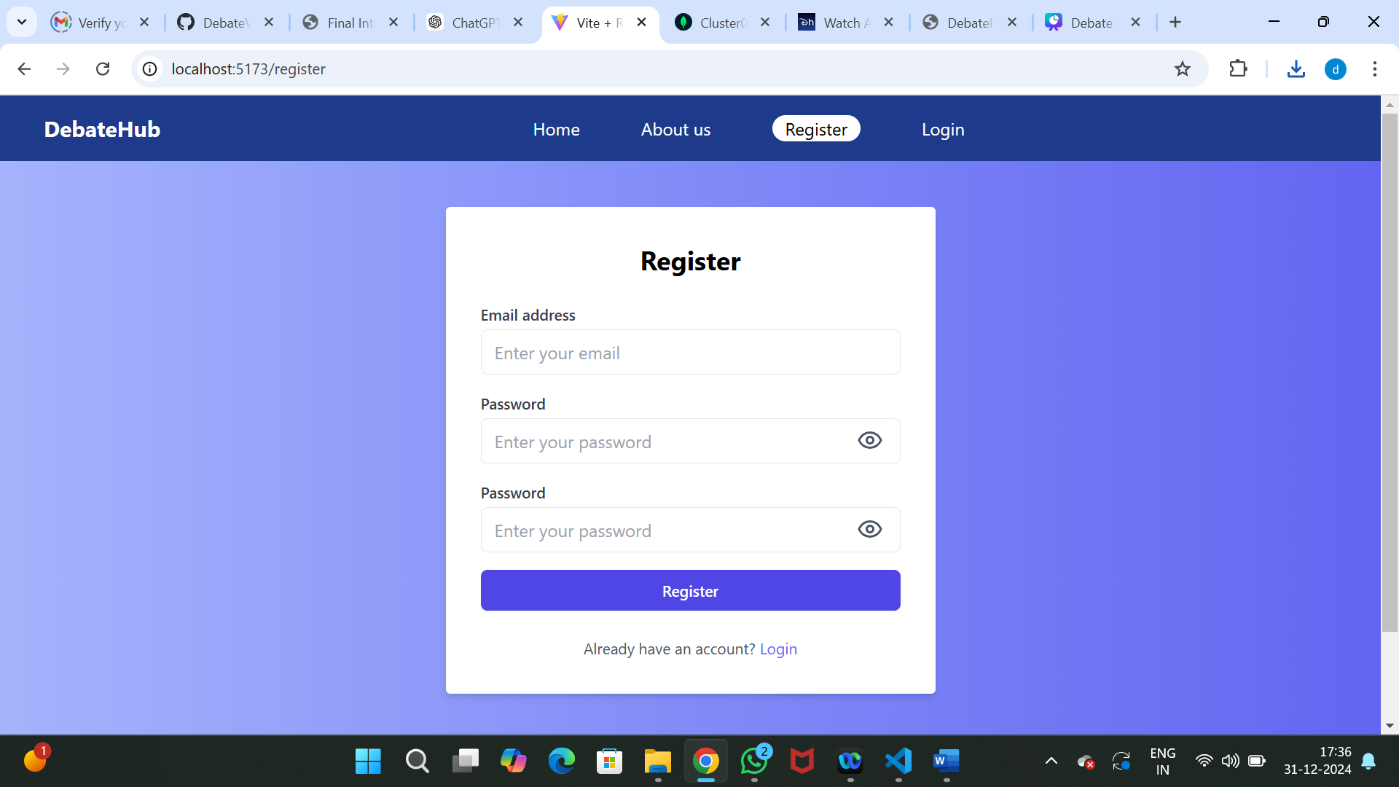
**ADMIN MODULE**:

* **Debate Moderation:**

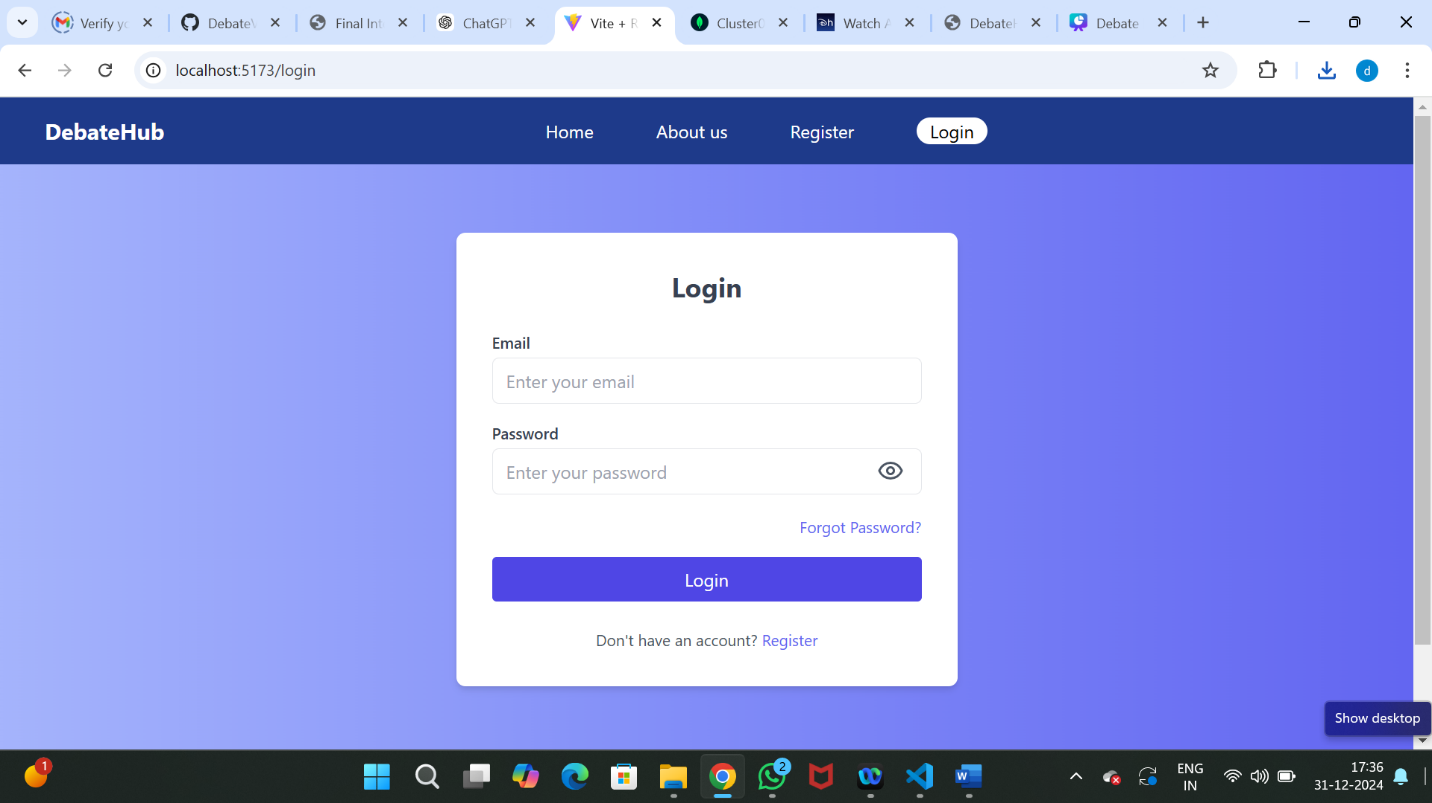
1. Admins have the authority to close debates, which halts any further participation or modifications.
2. Admins can edit or remove debate options or even delete entire debates when necessary.

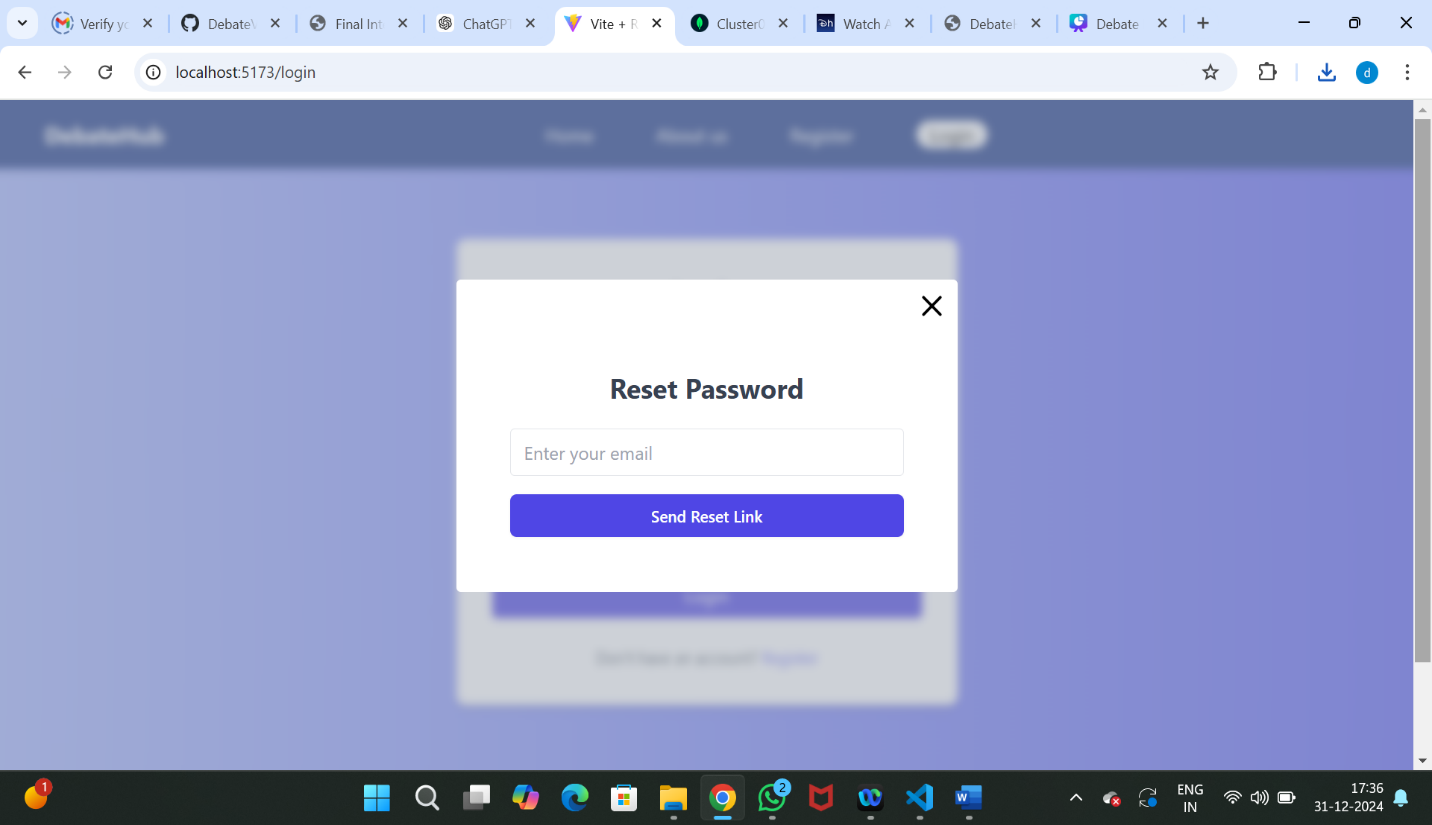
* **User Management:** Admins can control user accounts by activating or suspending users, allowing them to manage roles and access.
* **Moderation and Oversight:** Admins are empowered to filter, search, and review user profiles to ensure proper moderation and offer support when needed.
* **Analytics & Statistics:** Admins can access detailed analytics on debates, voting patterns, user behavior, and social engagement, helping to monitor and evaluate platform activity.

**OUTPUT SCREENSHOTS**

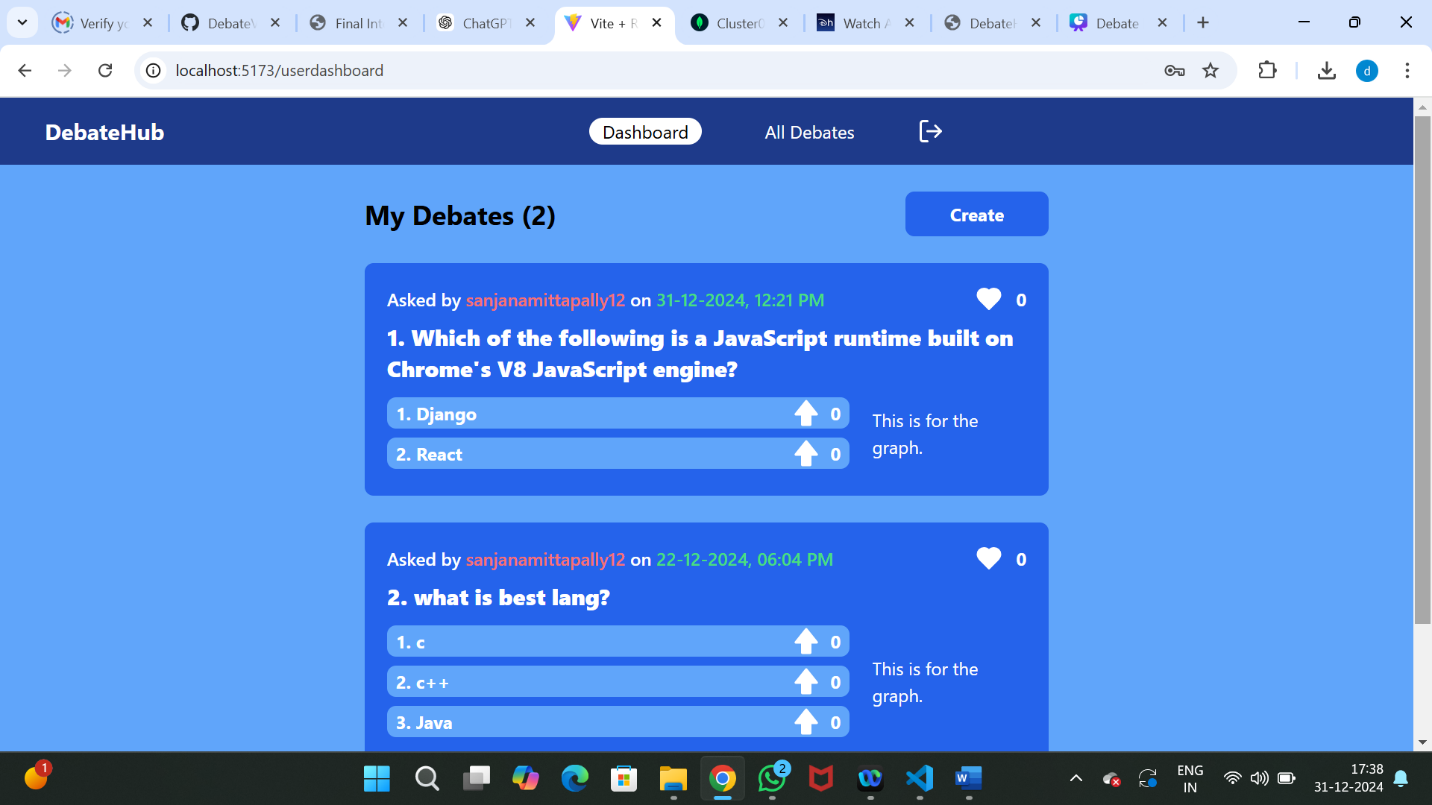
**Register**

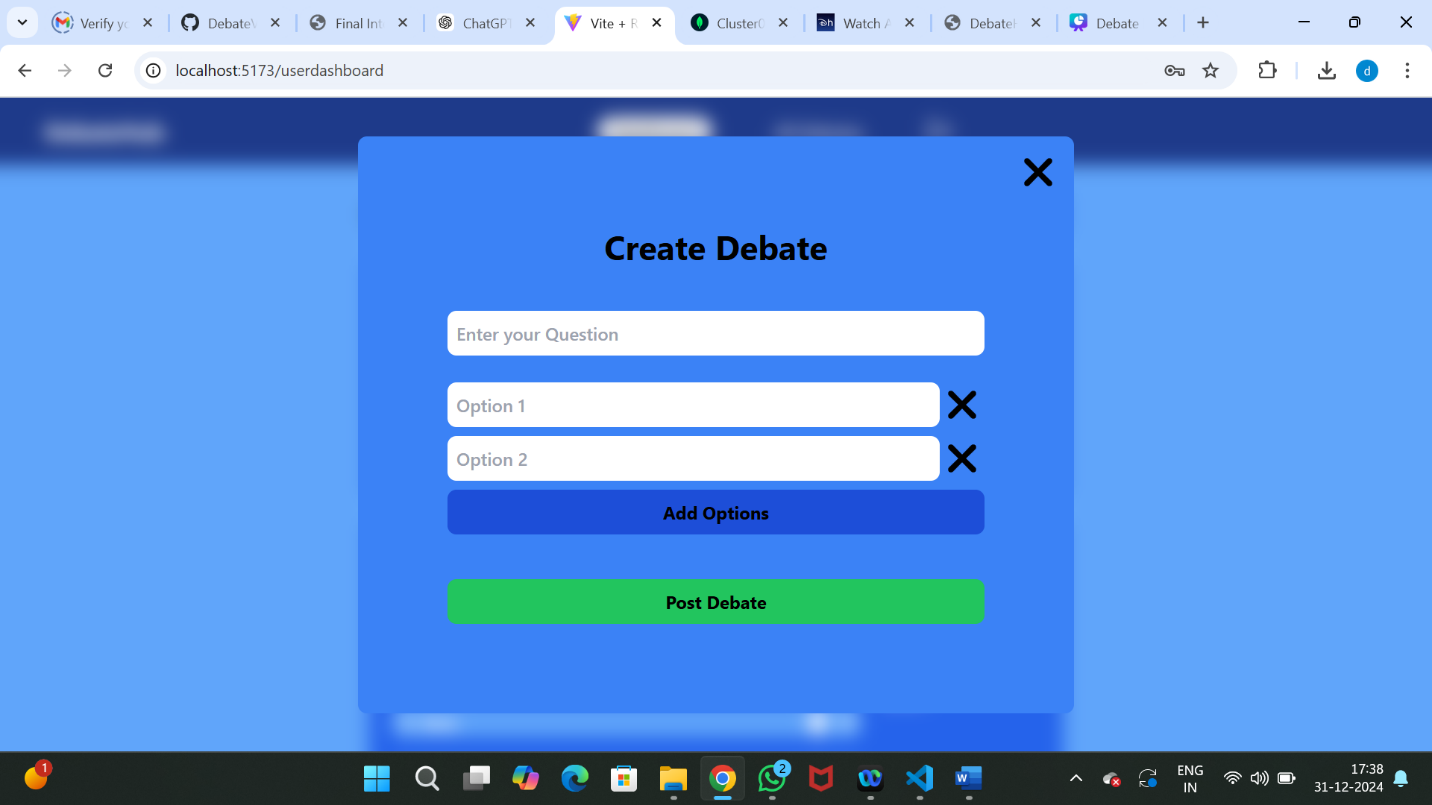
**Login**

****

**Reset password**

**Admin Dashboard**

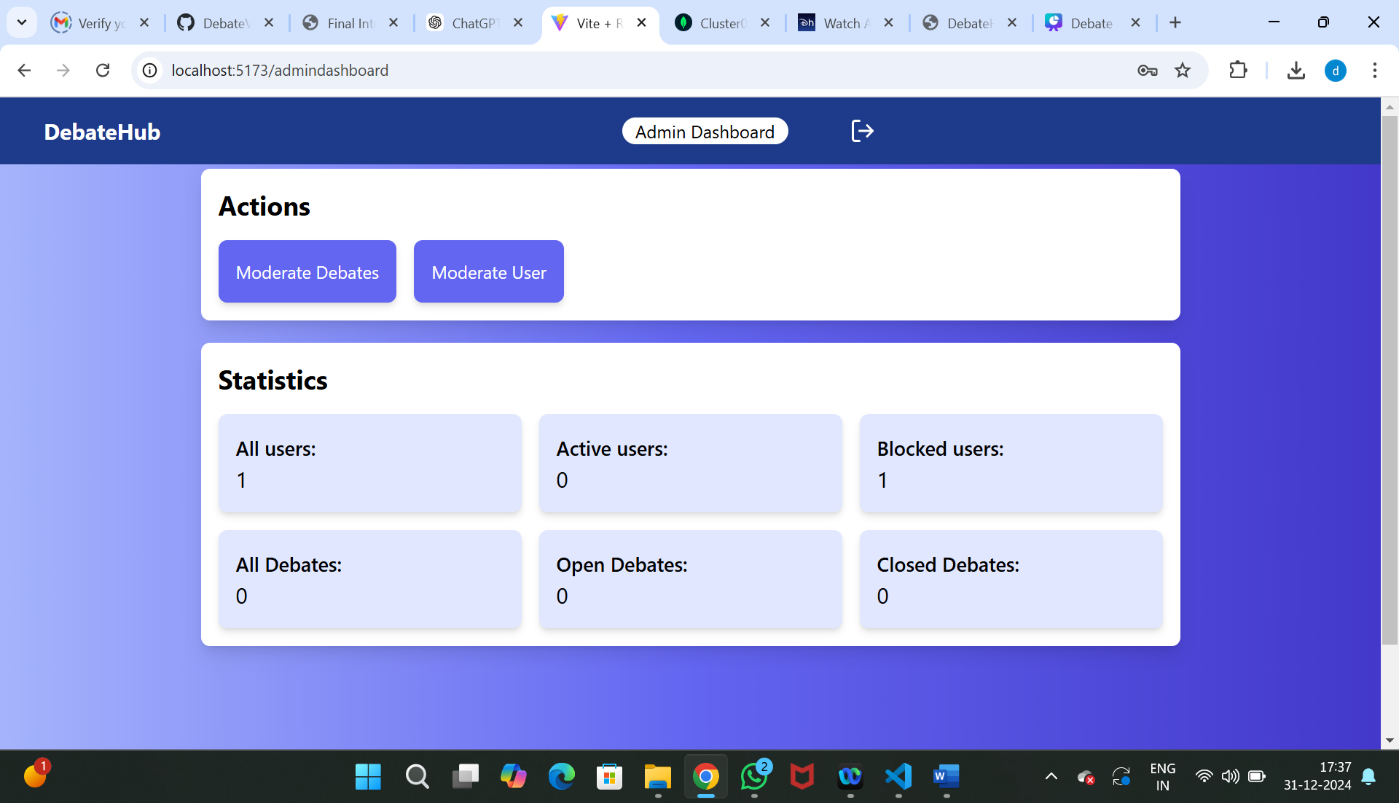
****

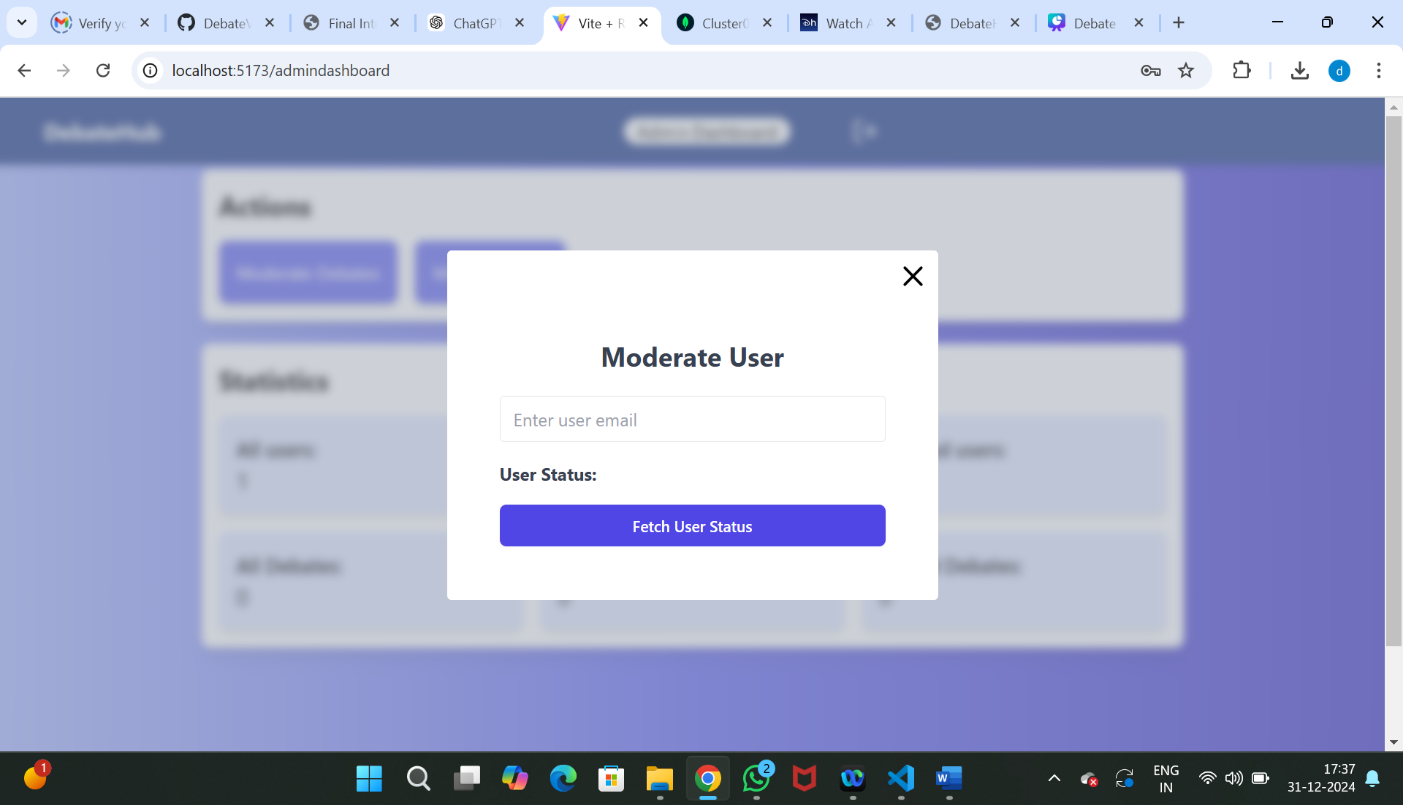
**Create Debate**

**Search and Sort debates**

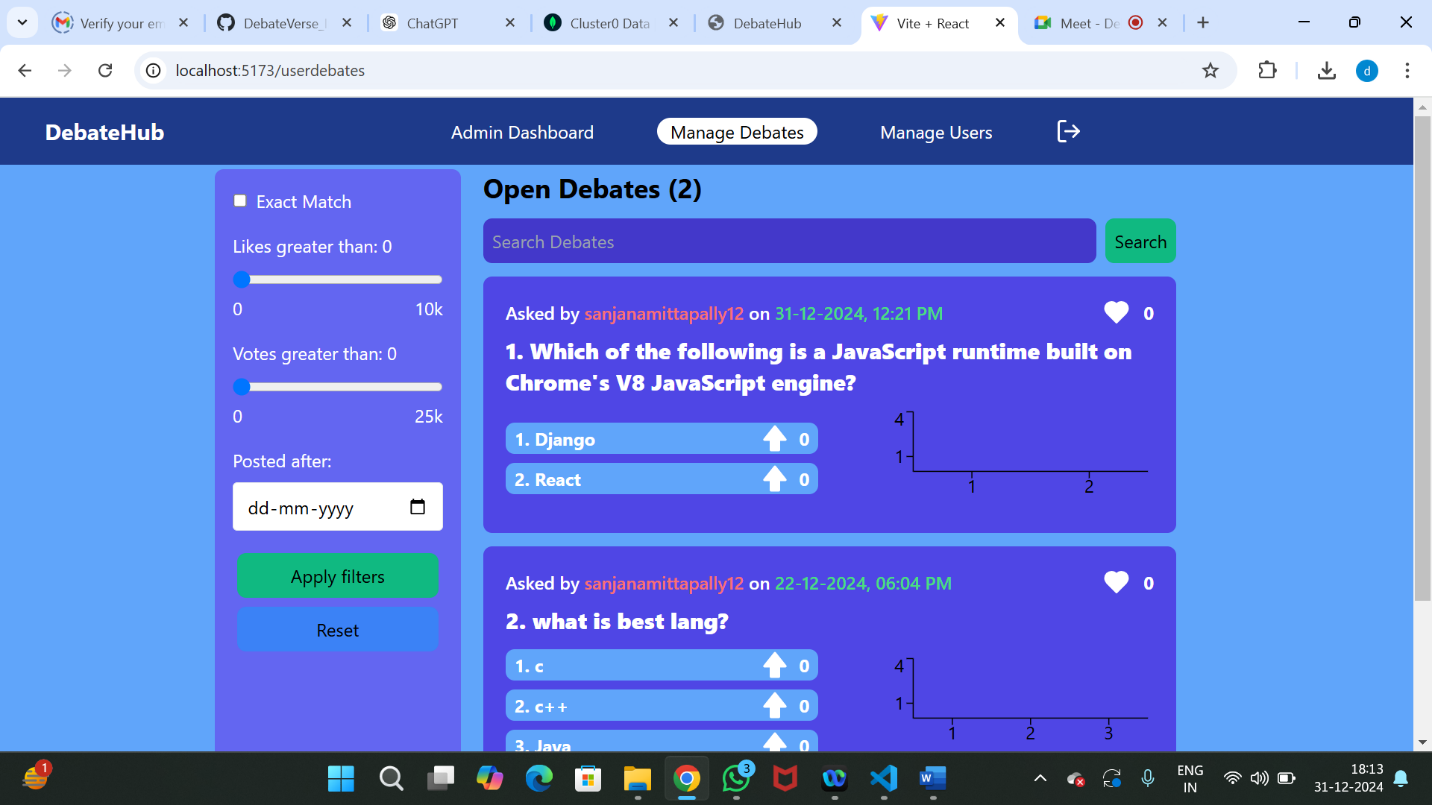
****

**Admin Dashboard**

****

**Moderate User**

**Manage Debate**

****