

# CS 4530 Final Project - NUStack

Group 501 - Jason Mai, Ziyou Zhang, Brighton Le, Wa Fan

**Description of the project:** This project aims to transform Stack Overflow from a simple Q&A platform into a personalized learning environment by adding several core features. This includes a search history structure, which automatically saves users' last 10 searches with filters, allowing quick access through a dropdown menu. The bookmarking and notifications system enables users to save and organize valuable questions in a "My Bookmarks" page, while a real time notification system keeps them updated on any activity on their bookmarked content through a bell icon dropdown. The voting and leaderboard system allows users to upvote/downvote questions and answers, with a leaderboard displaying top-ranked questions by vote count. User authentication was implemented using Firebase authentication, connecting everything together, ensuring personalized data persists across devices with proper security controls. The project overall emphasizes personalization and engagement while respecting user privacy through clear data controls.

**Deployment Link:** <https://cs4530-f25-501.onrender.com/home>

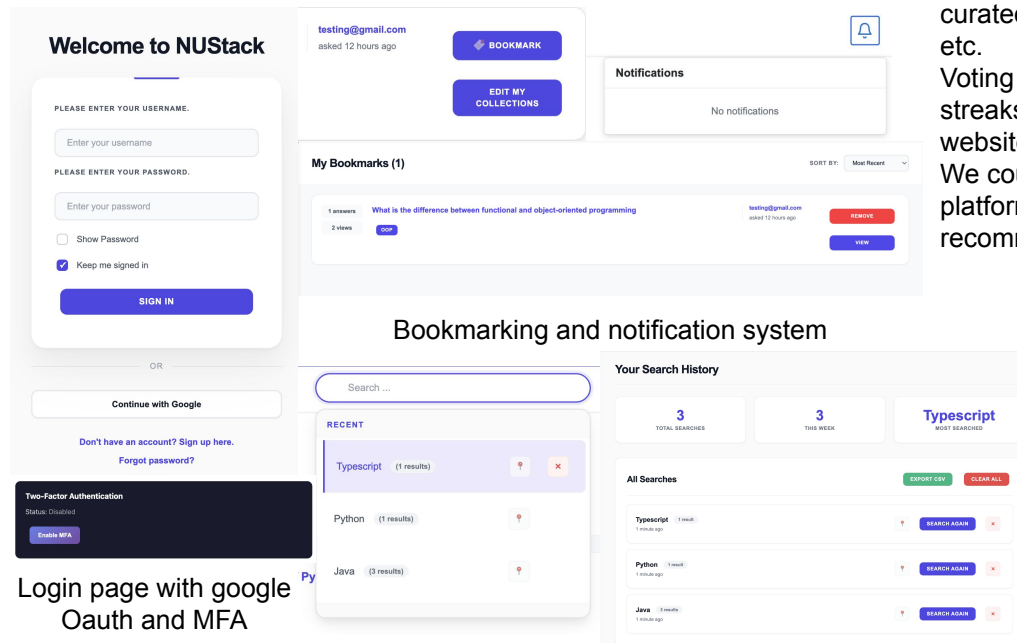
**Github Link:**

<https://github.com/neu-cs4530/fall25-project-fall25-project-group-501.git>

**Technology Stack and Design Choice:** The frontend is built with React and TypeScript, using Vite for fast development and React Router for navigation. Axios handles API requests, while Socket.IO provides live updates for notifications and activity events. Firebase Auth manages sign-in, sign-out, and session persistence, and the UI updates automatically using React hooks and shared context. The backend uses Node.js, Express, and MongoDB to store user data, questions, answers, and bookmarks. We follow a layered structure with controllers, services, and models. Firebase Admin middleware secures all write operations, and rate limiting is applied to prevent API abuse. Testing for this project includes Jest for unit tests, Supertest for API routes, Cypress for end-to-end flows, and Stryker for mutation testing. Overall, the system combines a React frontend with an Express backend to deliver personalized, real-time features in a secure full-stack architecture.

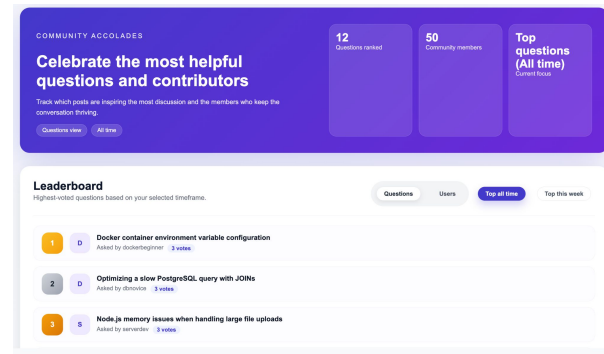
**Future Work:** In the future we could improve bookmarks to support collections, so that it is not limited to just questions. Users would be able to browse other people's collections if they choose to make them public, and bookmark these collections if they wish to. This feature allow users to find curated set of questions that might be helpful to them for studying a test and etc.

Voting and reputation could be extended with achievement badges or streaks to award active contributors. This could boost engagement of the website and encourage users to answer more questions. We could also incorporate machine learning to produce a more adaptive platform, such as auto generated tags and personalized question recommendations.



Bookmarking and notification system

Search bar and history page



Leaderboard for voted questions and users