CS4530 Final Project: KnowledgeOverflow
Group 107: Mohit Gupta, Aakarsh Kaushal, David Li

Our Features

Google-Auth and subscriptions

We added the ability to link your Google account to KnowledgeOverflow so users can be reassured that their account is secure. We also offer the ability to subscribe to question threads that seem interesting and offer email updates as well as real-time website notifications regarding new answers on subscribed questions.

Karma, Leaderboard, and Shop

We added Karma to KnowledgeOverflow. Now users can upvote and downvote questions and answers on the site to show how helpful they are. Karma is displayed publicly to show how reputable users are. Karma will also affect users' rankings on the public leaderboard. Accumulating a lot of karma allows users to buy exclusive items in the shop to customize their profile, such as new profile pictures, username colors and animations, and even a new theme for the website. Personalize your KnowledgeOverflow experience!

Stripe API

We added purchasable API keys so that users can access their own KnowledgeOverflow APIs and play with it as they like. The public endpoints available right now allow users to get all questions or filter the questions with their own custom filters to search for question content and metadata.

Demo and Source
Demo: https://cs4530-s25-107.onrender.com/
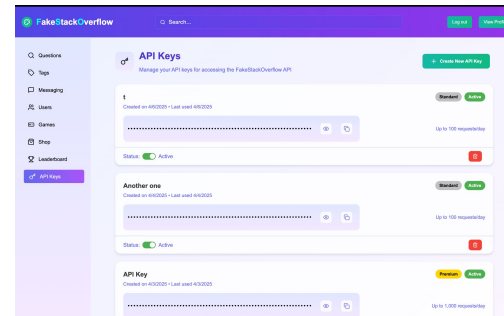Source: repo link

Our Technology and Stack Design

This implementation was built off the existing FakeStackOverflow codebase. We implemented Google-Auth with passport and subscriptions using SendGrid Mail. The karma and moderation were implemented by adding features to the User object. Leaderboard and Shop were created as additional pages accessible from the existing sidebar. The real-time notifications and leaderboard updates are socket events. Stripe API was created using Stripe and documentation created using Swagger. The frontend relies on several React hooks and the backend relies on the use of many socket events.

Our continuous integration pipeline runs an automated test suite on the frontend and backend components, and then deploys the site using Render.

Future Work

We managed to complete most of the features we initially wanted, but we had some difficulty making complete personalization. In the future we would like the user to be able to update their own profile picture and receive titles based on their activity in the community.

After purchasing API keys, the user is able to access all their available API keys on the API key page.