

Stack Overflow Clone - DBMS Project

We have developed a CQA software and populated it with the data dump from stack exchange.

A user can perform the following anonymously:

- Search for posts sorted by score or activity date by:
- A user
- Tag(s)
- View posts

If the user signs up/signs in to the site, then they can perform the following:

- Ask questions(posts) and assign predefined tags
- Comment on posts
- Answer questions(posts)
- Edit posts
- Delete posts
- Edit their personal information

Contribution

- Harshit Pant (CS21BTECH11021): Frontend(API connection, Hooks, Home, Post, Profile Pages)
- Vishal Vijay Devadiga (CS21BTECH11061): Backend(Models, Controllers)
- Satpute Aniket Tukaram (CS21BTECH11056): Frontend(Signin/Signup pages, Cookies)
- Mahin Bansal (CS21BTECH11034): Backend(Database design, Data cleanup, Cookies)

Languages

Javascript, HTML/CSS, PostgreSQL

Libraries

Frontend: ReactJS, Axios, Phosphor Icons

Backend: Node.JS, ExpressJS, Sequelize, jsonwebtoken

Code Description

We have used vite and node.js to create the skeleton of frontend and backend respectively. Backend code is contained in the 'backend' folder. Frontend is contained in the 'src' folder with public folder in the main folder.

Backend provides a simple REST API to query from the database. It first connects with the database, then starts the server at the port 5172 locally. Models in 'models' folder specify how the tables of the database are represented. Controllers in 'controllers' folder use these models and the active connection to query all data required by the frontend and presents the results in a JSON format.

For the Frontend, all pages in the 'pages' folder use components from the 'components' folder to provide a graphical interface to interact with the backend. Frontend queries data from the

backend using hooks from the 'hooks' folder and displays the data in a user-centric way. It is defined to look good with CSS.

How to run?

DATABASE SETUP

Most of the setup files are located in 'config' folder of backend. Following are the steps to make the database.

- Follow instructions in 'Stepsdb.txt'
- Read 'cdb.sql' in postgresql after creating the database 'cqadb'.
- Read 'readdata.sql' in postgresql after placing the csv files in 'Data' folder of backend.
- Uncomment the lines in 'index.js' of backend and let it sync and populate the table(s) by running the backend code
- Comment out the lines again. The Database is now ready.

CODE SETUP

Open a terminal in the main folder:

```
npm install
npm run dev
```

Run the above code to start the frontend. Open a terminal in the main folder:

```
cd ./backend
npm install
npm start
```

Run the above code to start the backend.