

Quack

Fontaine Coutino Villarreal

Overview

Quack will be an app similar to twitter, but improved. This social media implements posting, called quacking. Each quack can have up to 400 characters as well as 1 image. Each quack can have quomments which are comments. Quacks can also be quiked, similar to twitter likes. Users can be created and have the ability to be followed and follow other quackers.

Some features that the user can interact include a home page, explore page, and a ranking page. Home page includes quacks from the quackers they follow. Explore page shows quacks from other users in Quack. Lastly, the improvement over twitter. Quack allows a ranking page which include three types of ranking, quiques, followers, and both. This ranking page allows users to compare themselves to other quackers! This allows for better, high quality quacks!

After the application is fully working I'm planning on using a python script to populate the site. The script will add ~1,000 user . Then have each account follow others and like other posts. Just to have the ranking system be working.,

Frontend

The user will be able to interact with quack through two main html files. If user is not logged in and not in a current session, the user will always be redirected to login page. When logged in, the user is taken to main page. Instead of changing pages with each action, the right side of page will reflect what the user is trying to see, similar to ostaa. The sketches for these are on the last page. Home page will also include a help button to explain how website works.

Backend

The server for this project will be structured similar to ostaa's server. The modules being used will be express, mongoose, body-parser, cookie-parser,

muller (images), and the salting & hashing modules. The main difference will be that most routes will be POST requests. Mainly for sending sensitive information as well as submitting quacks, comments, etc. For retrieving quacks and rankings, it will mainly be GET requests. For the database. It will have 3 schemas; user, post, and comment.

Timeline

- 7/31

Have most html - css files done. Have calls to functions and some js implementation.

- 8/3

Implement most requests and start working on AJAX.

- 8/7

Having client and server working. Then add security to user login

- 8/10

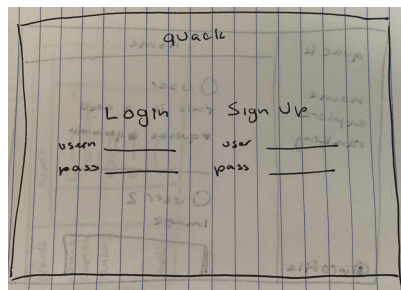
Finishing details and minor bugs. Record video.

- 8/11

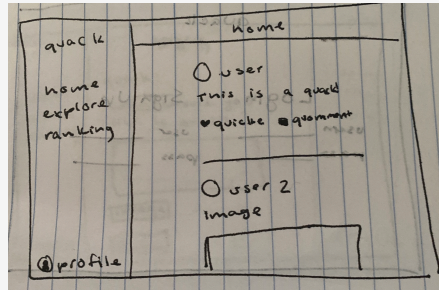
Populating app with users, comments, and likes.

Sketches

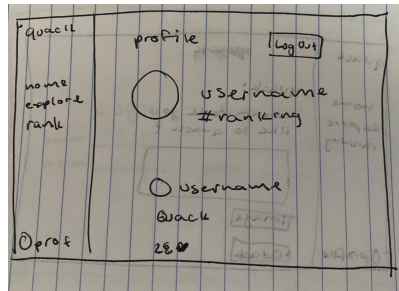
Login / Sign up



Home page



Profile



New Quack

