CVWO Assignment 2022

Github repository: https://github.com/lshaoqin/TagUp

Reflections

Overall, I have learnt a lot from creating TagUp, and I have no regrets for working on this despite an already busy winter break. Being off to a late start due to other commitments, there were points where I was discouraged by the steep learning curve and, looking at friends having lots of fun during their winter break, wanted to give up. I think what brought me through was my experience working on a couple other small projects in the past, and realising that the process of learning - new languages, new concepts, new architectures - is often the driest and most difficult part of the process. Indeed, after I got past the initial stages of struggle and trying to figure out what's going on, things were a lot smoother and felt more meaningful, as I was finally able to apply what I had learnt to build up my application piece by piece. It is always a great feeling to commit a change after adding some feature or improvement, and it is really satisfying to see the various pieces fit together to make a complete app.

An area for improvement, however, would be in terms of planning and perhaps scalability of the app. Having done the ideation and part of the coding for the app while I was still learning React and Rails, there are things that I would have done differently if I was given a second chance.

For example, the tag upvoting system turned out to be harder to implement than I had initially thought, since I had not considered the fact that I would have to store each individual vote to keep track of whether the current user has voted for the tag before. A feature I had initially wanted to implement was to generate a list of the most popular tags as suggested tags. However, in my current implementation, this would have meant counting the number of votes on every single tag, on every single post every time the database had to return this list, which entails significant overhead. A solution I considered was to add an extra table which simply stores each tag along with an int corresponding to the total number of votes, which updates whenever a vote is created or changed. However, I did not have enough time to implement this.

Additionally, there were some instances where I did not follow the Don't Repeat Yourself(DRY) principle, as I did not plan out extensively every aspect of the app in advance. For example, I found myself in the conundrum of copy-pasting code whenever I needed to authenticate the user token, yet deciding that since I likely wouldn't be using this code elsewhere, there was no point transforming it into a function(I would later then find various other places where it is required). When I was deploying the app, I then had to go through the troublesome process of manually editing the verification token on multiple points in the app. In short, this was a very good practical lesson on the importance of the DRY principle, and of planning ahead.

Finally, my rudimentary knowledge of CSS limited the overall aesthetics of the site and slowed me down considerably when attempting to figure out how to make certain things look the way I want them to. Although I don't think UI/UX design is an area that I want to focus hard on, I still believe I should work on building a stronger foundation in CSS for the future.

Accomplishments

Personal

- I now feel quite confident in using React and Ruby on Rails, which I had no experience in.
- Gained understanding of APIs, and one way in which the frontend and backend of an app can interact.
- Gained familiarity with the GitHub workflow.
- Got to deploy a website for the first time!

Technical features

- Login/sign up system, with encrypted user tokens for verification of actions
- Ability to perform CRUD for posts and comments
- Unique tagging system which allows users to add and vote for tags which they feel relevant
- Ability to filter and sort posts based on tag, tag votes, and date posted

Areas for improvement/possible future implementations

General

- 'Forgot password' feature (Requires user email)
- Password strength checks
- Support pictures and other multimedia in posts

UI/UX

- Optimise display for mobile devices
- Profile pictures
- General aesthetics and element scaling
- Themes/dark mode

Filtering

- Ability to filter by username(when clicked)
- Ability to filter by tag votes for all tags
- Ability to filter by multiple tags

Tagging

- More efficient vote counting in the backend
- Better interface for adding tags
- Autosuggest tags based on post contents

User Manual

TagUp is designed to be a simple, easy-to-use forum with the unique feature of community tagging, meaning the community can decide what tags should be attached to a post.

Users can view, filter, and open posts without logging in, but an account is required to post, comment, and add or vote on tags. To make an account, simply head to the signup page and enter a name and a password.

There are no fixed tags in TagUp, meaning users can tag a post with whatever they wish. To add a tag either when making or viewing a post, simply type the new tag into the provided text box and hit Enter. To upvote or revoke your upvote from a tag while viewing the post, press it once. To downvote a tag, give it a double-click.

For screenshots and more detailed instructions, please visit the readme at https://github.com/lshaoqin/CVWO_frontend.