CVWO Final Writeup for Gossip on Rails

Han Shangru, Stanley (A0179442X)

Reflection

Even though a basic forum web app seemed like a simple endeavour, Gossip on Rails turned out to be a great learning experience. Prior to this project, I was familiar with building React apps and Node.js backends, so a backend written in Rails had a steep learning curve, especially with its opinionated nature. Nevertheless, most of my previous knowledge was still applicable, allowing me to create a scalable platform for users to discuss and share information.

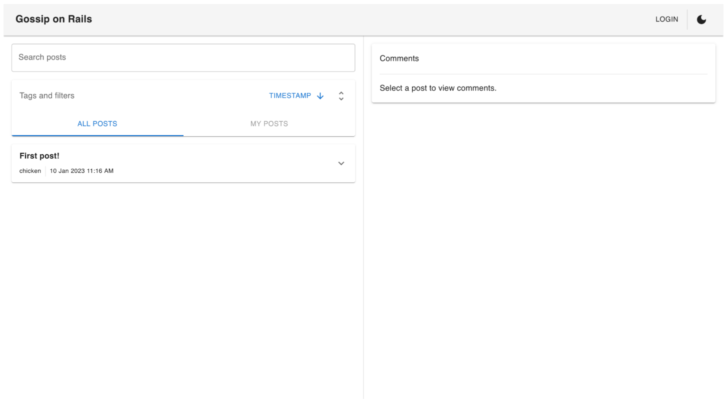
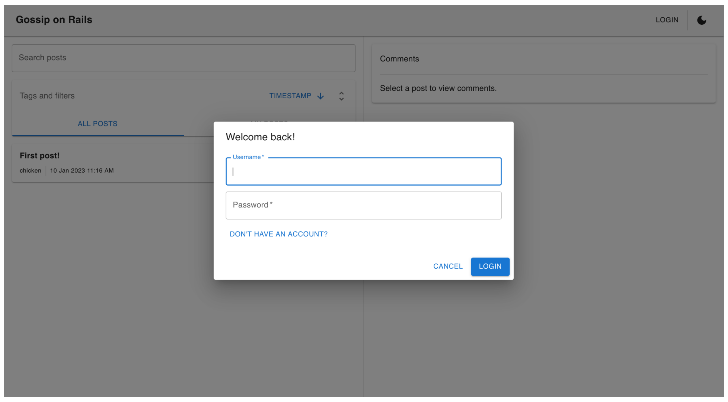
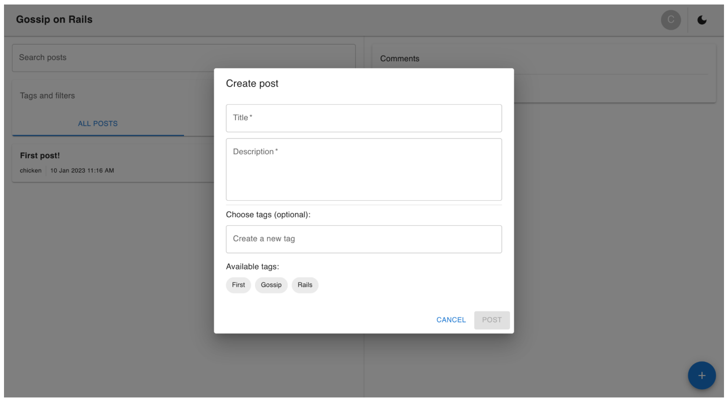
An interesting part of this project was designing the comments data structure to support nested comments, while preserving the ability to edit and delete them. Modelling Reddit's comment threads, I wanted deleted comments to retain the comments nested under them. Thus, I added a flag to the comments structure in the database that indicates whether a comment has been deleted, preserving the structure of the tree while allowing the frontend to display them accordingly. In addition, the comments table in the database also references itself to complete the tree (or, more accurately, forest) structure.

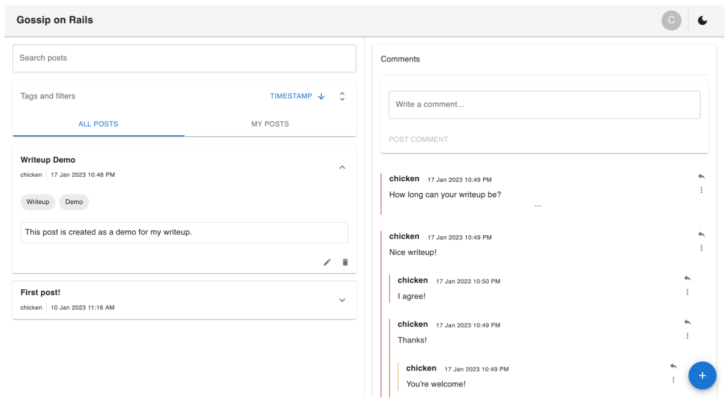
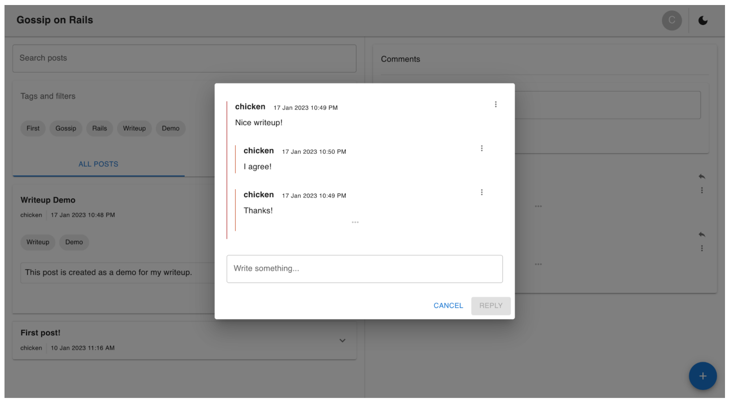
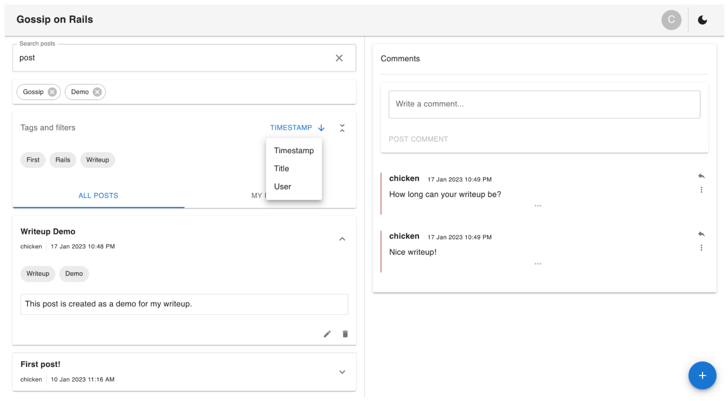
Originally, I had perceived integrating JWT authentication and implementing user management features such as login and registration to be the more challenging aspects of the project. In many Node.js frameworks, a basic username and password system requires specialised packages to handle token management and password hashing. However, with the help of Rails, many of these implementations have already been abstracted into the framework, making it relatively easier to set up the authentication system, which I can appreciate.

Despite having to learn a new language and understand the high level workings of the framework, Rails provided a solid structure for my application, allowing me to easily set up the database and corresponding CRUD capabilities.

Overall, this project allowed me to improve my skills and gain a deeper understanding of web development with React and Rails. Even though the Material UI library can look dated and consequently make the app less visually appealing, I hope you would enjoy using the core functions of the app!

User manual

 Gossip on Rails LOGIN Search posts Tags and filters TIMESTAMP ALL POSTS MY POSTS First post! chicken 10 Jan 2023 11:16 AM Comments Select a post to view comments.	The web app lands the user straight into the forum view, which is split into two main panels: the posts and search panel on the left, and the comments panel on the right. From here, the user can already see existing posts and their respective comments.
 Gossip on Rails LOGIN Welcome back! Username * Password * DON'T HAVE AN ACCOUNT? CANCEL LOGIN	To create posts and comments, the user can log into the forum with the Login button on the top right. The popup dialog also handles user registration, where a simple username and password will get the user a place in the discourse.
 Gossip on Rails Create post Title * Description * Choose tags (optional): Create a new tag Available tags: First Gossip Rails CANCEL POST	After logging in, the Create Post floating button will appear at the bottom right, clicking which will show a dialog where the user can input the title and description of their new forum post, together with any tags they wish to associate with their post.

 Gossip on Rails Search posts Tags and filters TIMESTAMP ALL POSTS MY POSTS Writeup Demo chicken | 17 Jan 2023 10:48 PM Writeup Demo This post is created as a demo for my writeup. First post! chicken | 10 Jan 2023 11:16 AM Comments Write a comment... POST COMMENT chicken 17 Jan 2023 10:49 PM How long can your writeup be? chicken 17 Jan 2023 10:49 PM Nice writeup! chicken 17 Jan 2023 10:50 PM I agree! chicken 17 Jan 2023 10:49 PM Thanks! chicken 17 Jan 2023 10:49 PM You're welcome!	Clicking on existing posts will expand them to reveal their details, together with buttons to edit and delete them (if the user is the creator of the post). The comments panel will also be populated with an input section to create new comments and a list of existing comments below. Comments have a tree structure and can be nested so readers know exactly which comment another is replying to.
 Gossip on Rails Search posts Tags and filters First Gossip Rails Writeup Demo ALL POSTS Writeup Demo chicken | 17 Jan 2023 10:48 PM Writeup Demo This post is created as a demo for my writeup. First post! chicken | 10 Jan 2023 11:16 AM Comments chicken 17 Jan 2023 10:49 PM Nice writeup! chicken 17 Jan 2023 10:50 PM I agree! chicken 17 Jan 2023 10:49 PM Thanks! Write something... CANCEL REPLY	If the user is the creator of a comment, they can also edit or delete the comment by clicking its three-dots icon. When replying to individual comments, its nested comments are also accessible for users to refer to when crafting their response.
 Gossip on Rails Search posts post Gossip Demo Tags and filters TIMESTAMP First Rails Writeup Timestamp Title User ALL POSTS MY Writeup Demo chicken | 17 Jan 2023 10:48 PM Writeup Demo This post is created as a demo for my writeup. First post! chicken | 10 Jan 2023 11:16 AM Comments Write a comment... POST COMMENT chicken 17 Jan 2023 10:49 PM How long can your writeup be? chicken 17 Jan 2023 10:49 PM Nice writeup!	Finally, there is a section where users can search for posts based on keywords and tags, together with a button that sorts the posts based on their timestamp, title or the username of the creator.