



Strider Web Back-end Assessment - 2.3

Briefing

View test instructions, submission instructions, and evaluation criteria in the briefing:

 [Strider Technical Assessment Briefing](#)

▼ Test-specific information

- Test version: 2.3
- Test domain type: Web back-end
- Suggested time: 8 hours (not including time to set up environment)

Project Description

Overview

The Project Manager you work with wants to build a new product, a new social media application called Posterr. Posterr is very similar to Twitter, but it has far fewer features.

Posterr only has two pages, the homepage and the user profile page, which are described below. Other data and actions are also detailed below.

Pages

Homepage

- The homepage, by default, will show a feed of posts (including reposts and quote posts), starting with the latest 10 posts. Older posts are loaded on-demand on chunks of 10 posts whenever the user scrolling reaches the bottom of the page.
- There is a toggle switch "All / following" that allows you to switch between seeing all posts and just posts by those you follow.
- New posts can be written from this page.

User profile page

- Shows data about the user:
 - Username
 - Date joined Posterr, formatted as such: "March 25, 2021"
 - Number of followers
 - Number following
 - Count of number of posts the user has made (including reposts and quote posts)
- Shows a feed of the posts the user has made (including reposts and quote posts), starting with the latest 5 posts. Older posts are loaded on-demand when the user clicks on a button at the bottom of the page labeled "show more".
- Shows whether you follow the user or not
- Follow/unfollow actions:
 - You can follow the user by clicking "Follow" on their profile
 - You can unfollow the user by clicking "Unfollow" on their profile
- New posts can be written from this page

More Details

Users

- Only alphanumeric characters can be used for username
- Maximum 14 characters for username
- Do not build authentication
- Do not build CRUD for users

- When/if necessary to make your application function, you may hard-code the user. For example, you may need to do this to implement creating new posts, following, etc

Posts

Posts are the equivalent of Twitter's tweets. They are text only, user generated content. Users can write original posts and interact with other users' posts by reposting or quote-posting. For this project, you should implement all three — original posts, reposts, and quote-posting

- A user is not allowed to post more than 5 posts in one day (including reposts and quote posts)
- Posts can have a maximum of 777 characters
- Users cannot update or delete their posts
- Reposting: Users can repost other users' posts (like Twitter Retweet)
- Quote-post: Users can repost other user's posts and leave a comment along with it (like Twitter Quote Tweet)

Following

- Users should be able to follow other users
- Users cannot follow themselves
- Following and unfollowing will be done only on the user profile page

Extra feature: Search

Only work on this extra feature if you have enough time to complete the required features and get through all three phases of the interview.

- Implement a search feature that allows users to efficiently search through all posts
- This search feature should not return reposts
- This search feature should return quote posts, but only if the search matches the additional text added (do **not** return matches from the original post that was quoted on top of)
- (For Phase 2) This search feature should return reply-to-posts

Phase 1, coding

Estimated time: 7 hours

- Build out a RESTful API and corresponding backend system to handle the features detailed above. This RESTful API would communicate with single page JS app. This API you build should enable all the features on both of the pages.
- You should implement a real, production-ready database, and queries should be performant.
- Do not implement additional features beyond what is explained in the overview.
- Write some automated tests for this project.
- Do not build a front-end.

Phase 2, planning

Estimated time: 30 minutes

The Product Manager wants to implement a new feature called "reply-to-post" (it's a lot like Twitter's). These are regular posts that use "@ mentioning" at the beginning to indicate that it is a reply directly to a post. Reply posts should only be shown in a new, secondary feed on the user profile called "Posts and Replies" where all original posts and reply posts are shown. They should not be shown in the homepage feed.

What you need to do:

- Write down questions you have for the Product Manager about implementation.
- Write about how you would solve this problem in as much detail as possible. Write about all of the changes to database/front-end/api/etc that you expect. You should write down any assumptions you are making from any questions for the Product Manager that you previously mentioned.
- **Be thorough!**

This should be added as a section called "Planning" in the README.

Phase 3, self-critique & scaling

Estimated time: 30 minutes

In any project, it is always a challenge to get the code perfectly how you'd want it. Here is what you need to do for this section:

- Reflect on this project, and write what you would improve if you had more time.

- Write about scaling. If this project were to grow and have many users and posts, which parts do you think would fail first? In a real-life situation, what steps would you take to scale this product? What other types of technology and infrastructure might you need to use?
- **Be thorough!**

This should be added as a section called "Critique" in the README.