




SEARCH	
RESOURCES	
CONCEPTS	

- ✓ 1. Project Overview
- ✓ 2. Project Instructions & Rubric
- ✓ 3. Approaching a Project
- ✓ 4. Your Project Workspace
- ✓ 5. Step by Step Guide - React & Redux ...
- ✓ 6. Would You Rather Workspace
- ★ 7. Project: Would You Rather...?

Knowledge

Get learning questions answered

Student Hub

Chat with peers and mentors

## App Functionality

The person using your application should have a way of impersonating/logging in (This could be as simple as having a login box that appears at the root of the app where the user select a name from the list of existing users. Alternatively, you could create a user creation process to allow a user to sign up for an account.) Your application should allow the user to log out and log back in, regardless of which user is selected. Once the user logs in, the home page should

We always want to make sure we know who the logged in user is, so information about the user should appear on the page. If someone tries to navigate anywhere by entering a URL in the address bar, the user is asked to sign in and then the requested page is shown. The user should be able to log out and log back in.

Once the user logs in, the user should be able to toggle between his/her answered and unanswered polls on the home page, which is located at the root. The polls in both categories should be ordered from most recently created (top) to the least recently created (bottom). The unanswered polls should be shown by default, and the name of the logged in user should be visible on the page.

What would be the point of seeing answered and unanswered polling questions if the user can't vote or see the results? Each polling question should link to the details of that poll. The details page for each poll should be available at `questions/:question_id`.

When a poll is clicked on the home page, the following is shown:

1. Text "Would You Rather";
2. Avatar of the user who posted the polling question; and
3. Two options.

For answered polls, each of the two options contains the following:

1. Text of the option;
2. Number of people who voted for that option; and
3. Percentage of people who voted for that option.

The option selected by the logged-in user should be clearly marked.

Since we want to make sure our application creates a good user experience, the application should show a 404 page if the user is trying to access a poll that does not exist. (Please keep in mind that newly created polls will not be accessible at their url because of the way the backend is implemented in the application.) It should also display a navigation bar so that the user can easily navigate between the home page and the application.

So what happens when someone votes in a poll? Upon voting in a poll, all of the information about the poll should be displayed. The user's response should be recorded and displayed on the poll details page. Users can only vote once per poll; they shouldn't be allowed to vote again after they've voted -- no cheating allowed! When the user comes back to the home page, the poll question should appear in the "Answered" column.

It would be no fun to vote in polls if we couldn't post our own questions! The form for creating new polling questions should be available at the `/add` route. The application should have a "Would You Rather" page and have a form for creating two options. Upon submitting the form, the user should be taken to the home page, and the new polling question should appear in the correct category on the home page.