

FULL-STACK

.....

Summary:

This project is to close full circle on full-stack with a focus on backend development. Students will complete all the necessary development to complete a login page. A boilerplate will be provided to help students get started with this project and they are welcome to modify it anyway they like. The boilerplate will include a frontend with Vue for the framework and Vuetify for the materials library. For the backend, it will include Node, Express, Axios, and MongoDB.

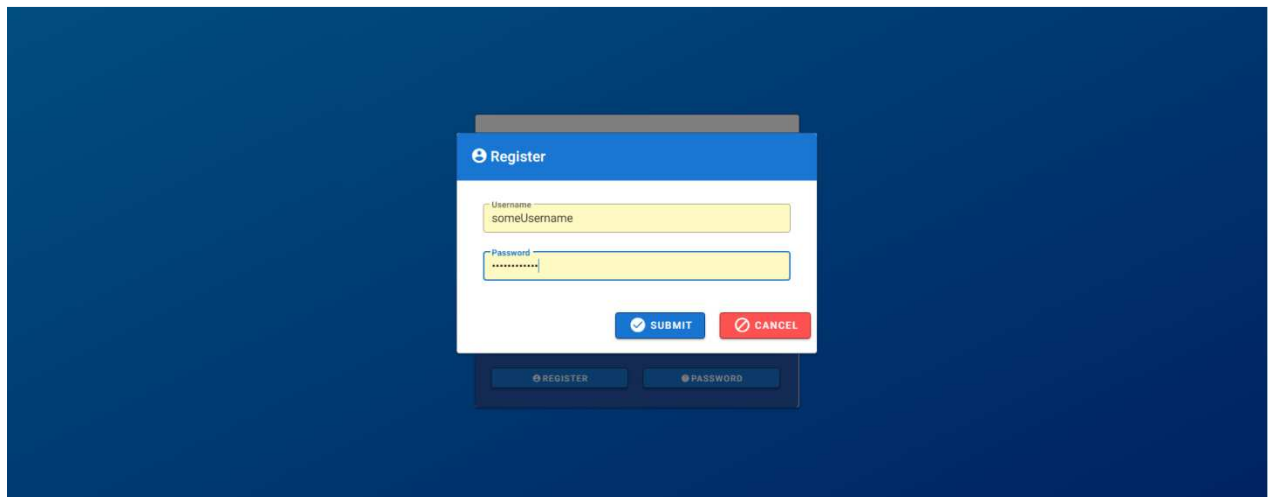
Requirements:

The app will be graded based on the use cases listed on the grade sheet. The application will be tested on chrome only, no other browser. Remember to review the proper documentation and slide decks for reference.

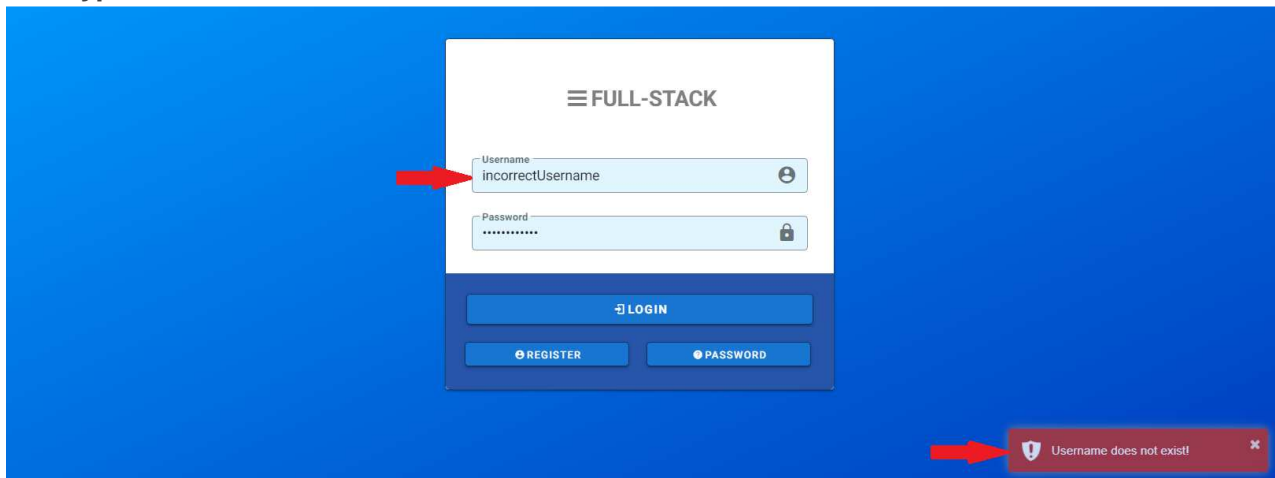
Preview:

The app must perform similar to the sequence of screenshots shown below but more detailed use cases are described on the grade sheet. The more similar the submission looks to these previews, the less chances of losing points.

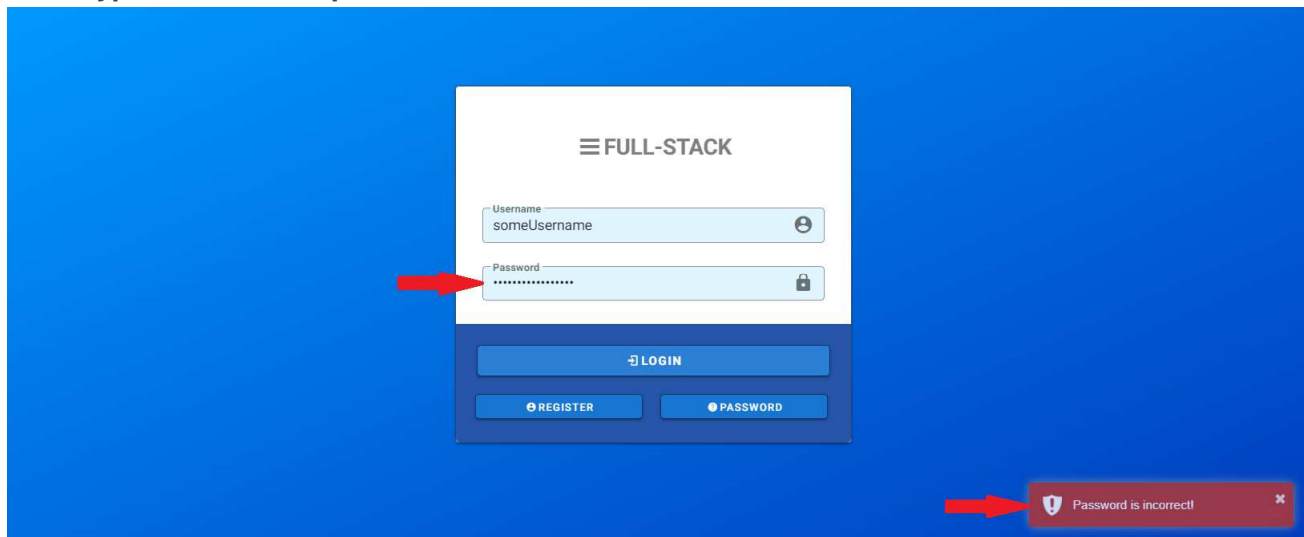
1. Create an account:



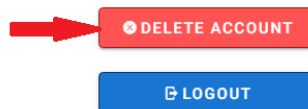
2. Type an incorrect username:



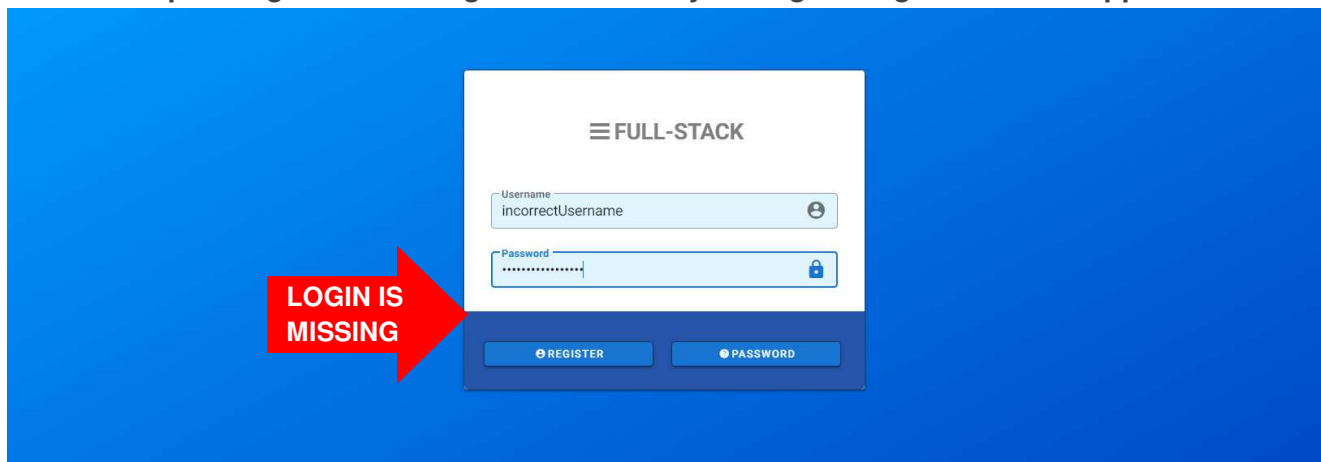
3. Type an incorrect password:



4. Successfully login and click the delete button to delete account:



5. Attempt to login 5 times to get locked out by having the login button disappear:



Run Instructions:

To run the app, follow the instructions below.

1. Install Node, MongoDB, and Robo3T on your computer
2. download app.zip boilerplate from 'start' folder of google drive
3. extract .zip to local folder
4. run **npm install** on root and client terminal
5. run **npm run dev** on root terminal
6. run **npm run serve** on client terminal
7. ctrl+click the localhost link provided to launch the site on chrome

Deadline:

The deadline is found on the row that contains the code 'hw06' under the homework column of the calendar table at the end of the syllabus.

Submission:

The submission will only contain the dashboard.vue, login.vue, and posts.js file. During grading, these 3 files will be pasted on the boilerplate and compiled accordingly. When sending the .js file, make sure to change the .js extension to a .txt extension because most email services do not allow sending .js files. The TA will convert the .txt file back to .js when grading.

The completed files must be sent to archive.master.apollo@gmail.com with the email subject **hw06-codeName-studentId-firstName-lastName**. The code name can contain any letter, number, or character but must not reveal your actual name or student Id. You will use the same code name for all your submissions. So an example subject line might be **hw06-grapefruit123!-john-smith-1234567**.