

CSC 436: Web Applications

Final

DUE DATE: November 21, 11:59PM

Goal: Update the Todo app to meet the following requirements:

1. Bootstrap changes for the front-end:
 - a. Make front-end React application reactive (scale according to browser size) utilizing **bootstrap** and **react-bootstrap**
 - b. Implement modals for login and registration using **bootstrap** and **react-bootstrap**
2. Add a “users” page accessible at /users/ which will list all registered users and provide links to their profile pages where their Todos can be viewed
3. Add a “profile” page accessible at /users/:userId which will display a particular user’s Todos
4. Implement proper authentication using **bCrypt** to store credentials securely and authorize user requests using **jsonwebtoken**.
 - a. When a user registers, passwords should be hashed using `bCrypt.hash()` before inserting into the database.
 - b. When a user logs in, their plaintext password should be compared to the stored hash using `bCrypt.compare()`, if comparison returns true, a body containing an `access_token` should be returned. The payload should be the user’s database ID.
5. Creating, toggling, or deleting a Todo require an Authorization token to be passed and a user should only be able to perform these operations on their own todos. Store the Authorization `access_token` in your React application and send it as a header when issuing these requests.
6. Deploy your React front-end and Node backend to Heroku.

Optional:

1. Extend the front-end and back-end components to include the ability to create TodoLists. A `TodoList` object consists of a title and description. Individual Todos will then be added to TodoLists. Update render logic throughout the front-end to handling rendering multiple lists of todos.

Submission

1. Create a .txt file named **FIRSTNAME_LASTNAME.txt** containing the following:
 - a. A link to your GitHub repo
 - b. Your MongoDB Atlas DB connection URI (which includes the DB username and password)
Do not upload your connection URI or DB credentials to GitHub.
 - c. The private key you used to hash bcrypt passwords.
Do not upload your private key to GitHub.
2. Upload the .txt file to the D2L dropbox for the final. This will indicate to me that your repo is ready for grading.

