This assignment builds up on the prior one, we need to complete the back end for our application in Node / Express.
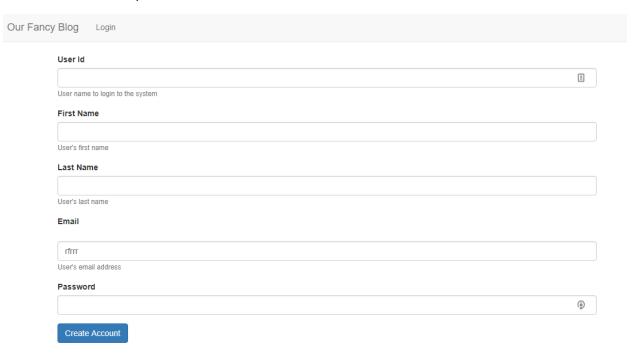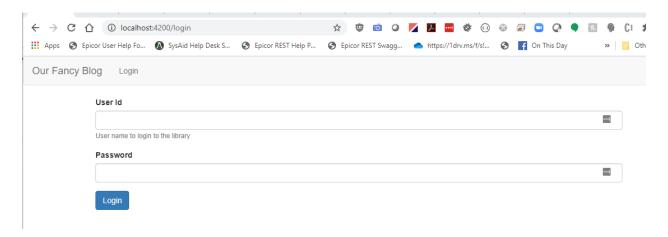
The goal of this assignment is to start writing our front end to our Blogging Application. The first step is to create the basic framework, UI and login screen

1. Create a new Angular Application
2. Create a basic navigation bar component for future use
    a. You may use bootstrap for this
    b. The navigation bar should be at the top or the left or right side of the screen
3. Create a "New User" component
    a. This component should allow you to enter the required information to successfully create an account in the back-end system (see Assignment 3)
    b. It should handle any errors returned from the API
    c. When an account is successfully created it should re-direct user to the login page (see below)
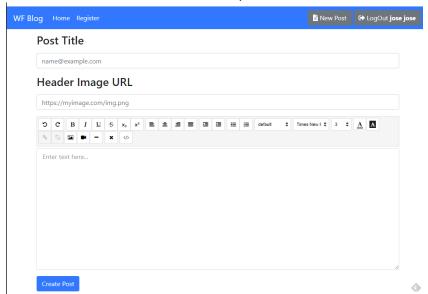

4. Create a simple login page
    a. This login page should ask you for your username and password
    b. It should then perform authentication against our back-end API (see assignment 3)

c. When the token is received it should store that token for later use
d. If an error is received it should report said error to the user
e. Once the user is authenticated it should redirect to the "Inner Sanctum" (a secure area) in your application. A simple hello world component would suffice for this
f. The authentication should happen by making a login query call against either my api at https://unf.josecgomez.dev/ or your own implementation of assignment 3
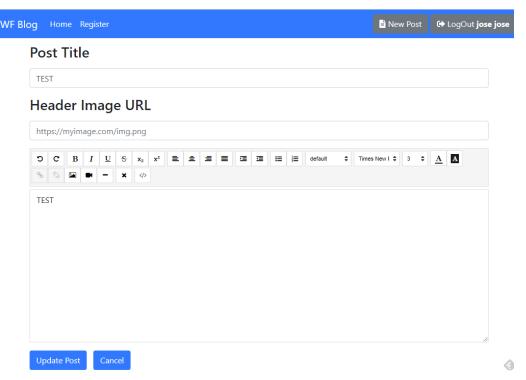g. It should work regardless since the APIs are to be identical

5. Implement LogOut Functionality which would remove the token from storage and re-direct the user to the login page.
6. Create Post Component
   a. Create Post Component will allow an Authenticated Valid user to create a blog post and submit it to the API
   b. It will handle all errors returned by the API Appropriately
   c. Non-Authenticated user cannot create posts or even reach the Create Post Page

7. Edit Post Component
    a. Any Authenticated user can edit their own posts, if a post belongs to the current user the user should the option to Edit that post and submit those changes to the API
    b. Other user cannot edit posts that aren't theirs
    c. Non-Authenticated users cannot edit ANY posts

8. View Posts Component
    a. This component should use the API to query (all posts) available and display them in Chronological Order (most recent post first)
        i. If there are no posts available the component should just display no posts found and provide the option to create a new one (if the user is logged in)
    b. Each post should list their title and body along with the Author and the Creation Date
    c. These posts are ALL visible to everyone authenticated or not and should be the first page that shows up when you visit the site ("/") Route
    d. If a Post belongs to the current user there should be an Edit and a Delete button which would allow the user to Delete the post and or navigate to the Edit Post component and make changes
    e. When a user chooses to Delete a post, they should be asked for confirmation before proceeding

f. All Errors must be handled appropriately, and the user must be notified.

Extra Credit

Implement an Edit Account feature which allows a logged in user to update it's name, email address and such. Only the current logged in user should be able to edit their own account.

Add a WYSIWYG editor to the Post to allow you to submit a post which contains HTML and render it back on the screen.

Implement comment feature that allows you to comment on your or someone else's blog post entry.