Milestone 2 Deliverable

Objectives

1. Render all the HTML pages from Module 1 using NodeJS and Handlebars.
2. Integrate the app with Instagram for authentication.
3. Store the "access_token" in the users session.
4. Log the user out if the token is expired or not found.
5. Update the "Dashboard" and "Search Results" pages to actually pull data from the Instagram API.

Requirements

1. Render all the HTML pages from Module 1 using NodeJS and Handlebars.
    - Each page needs an express route handler which calls "res.render()"
    - Each page needs a handlebars template which is passed into "res.render()"
    - Each page should inherit from a handlebars "layout"

2. Integrate the app with Instagram for authentication.
    - Set up an Instagram account and sign up as a developer at https://instagram.com/developer/register/
    - Create an Instagram "app" so you can authenticate to the Instagram API
    - From the homepage of your project create a "login" button which takes a user to the Instagram authentication page for your Instagram app you created above
    - If the user is already logged into Instagram in their browser they should be redirected to the Dashboard page
    - If the user is NOT already logged into Instagram it should allow them to enter their username and password and then if they do so successfully it should redirect to the Dashboard page

3. Store the "access_token" in the users session.
    - Once authenticated Instagram will give you an access_token which must be saved to the users session for future API calls on behalf of the user

4. Log the user out if the token is expired or not found.
    - The app should check for the existence and validity of the Instagram access_token at each request and redirect the user back to the home page if the access_token is missing or no longer works.
    - **HINT:** This would be a great place to utilize express middleware, e.g. `app.use()`

5. Update the "Dashboard" and "Search Results" pages to actually pull data from the Instagram API.
    - The "Dashboard" page should request the first page of the users feed and display the image, number of comments and number of likes for each post returned from the Instagram API
    - The "Search Results" page should send off a search query to Instagram's API based on what the user entered in the search input and then display the results which are returned from Instagram's API for that particular search term.
    - The "Search Results" page should display the same data for each post as the "Dashboard" page described above (image, number of comments, number of likes)