

Feature Request

Application: Jammming
Date: August 11, 2020
Developer: Sunjay Armstead

Objective

To implement a conditional modal window to send a user to Spotify to connect their account before accessing the Jammming application.

Background

Currently, when a user first uses Jammming they enter a search term, are then sent to Spotify to log in to their account, and then are redirected to Jammming. Unfortunately, their search term is lost when they are redirected back to Jammming from Spotify.

In this feature request, I propose the following application updates:

- Create a full screen modal window that only appears if the user has not connected Jammming to their Spotify account.
 - This modal will have a button that says CONNECT TO SPOTIFY
 - The user is sent to Spotify to sign in, and then is redirected back to Jammming to create their playlists
- All Jammming users that authenticate with Spotify will have a check mark at the top of the app that says CONNECTED TO SPOTIFY.
- A new **Footer** component will also be created to protect the intellectual rights of both Sunjay Armstead and Codecademy.

Technical Design

The Spotify API requires the user to leave the client for Spotify account authorization and return when authorization is complete. In order for the **Modal** and **TopBar** components to render correctly, a "loggedIn=true" cookie will be installed each time `Spotify.getAccessToken()` is called. For security, this cookie's max-age will be set to 1 hour.

A new `componentDidMount()` method will be added to the **App** component. The method will check if the "loggedIn" cookie includes the Boolean "true." If this value is not present, the method will initialize the "loggedIn" cookie to the Boolean "false." In either case, the method will set the `loggedIn` **App** state to the value of the cookie. **App** will pass its `loggedIn` state as a `loggedInStatus` prop to child components in its `render()` method.

The new **Modal** component will contain all of the UI features of the modal window. Its render method will contain a JSX section container with a welcome message and button to connect to Spotify. Above the return statement, a conditional statement will return "null" if `this.props.loggedInStatus === true`. This will ensure that the user is only prompted to authorize with Spotify using the **Modal** on their initial log in.

Modal's button property `onClick` will call the `Spotify.getAccessToken()` function to authorize Jammming to access their Spotify account. After successful authorization, the user is sent back to Jammming to search songs and the Modal box will disappear when App's `loggedIn` state is set to "true."

The new **TopBar** component will contain new `componentDidMount()` method similar to the App component. This method will check the Boolean value of the "loggedIn" cookie. If "loggedIn" is true, the `loggedIn TopBar` state will be set to "true," else it will be set to "false. If TopBar's `loggedIn` state is true, it will have a checkmark with the message "CONNECTED TO SPOTIFY."

The new **Footer** component will contain logic for the current year for the copyright statement. Copyright will cover Sunjay Armstead and Codecademy. The privacy policy will link to <https://sunjayarmstead.com/privacy-policy>. This component will render at the very bottom of App's `render()` method.

Caveats

Currently, after a user types in a search query, they are sent to Spotify to authorize their account, and are then sent back to resubmit their query. The goal of this update, then, is to eliminate the need to type the same search query more than once. To achieve this goal, a modal blocks the user from searching until they have signed in.

A simpler solution would be to save the search query as a prop, state, or cookie and then return the value to the search field after a user returns to the Jammming application. However, the method proposed in this feature request creates a more intuitive authorization flow.

A **Modal** component provides space to explain the Jammming app and the steps a user should take to connect their Spotify account to the client. Further, a **TopBar** that verifies a user's connection to Spotify will help the user know when the Jammming app has access to their Spotify account. Both the new **Modal** and **TopBar**, then, will prevent search query resubmission and provide a more intuitive authorization flow.

