Feature Request: Update playlist & add region selection

**2023**

# OBJECTIVE

To give users the ability to update all of their Spotify playlists and save them to Spotify.

# BACKGROUND

Currently, Jammming supports the ability to create one new playlist at a time and save it to Spotify. However, updating an existing playlist is not supported. With over 5 million playlists created or edited daily on Spotify, this is key functionality to support. The search feature has a determined region to search, adding feature as the to select region could narrow and provide better client experience

These features accomplishes the following:

* Displays a list of the current user's playlists
* Allows a user to select one of their existing playlists. The application will load the name of that playlist and the tracks of that playlist in the playlist panel.
* The user may update the name and/or the tracks of the playlist and click "SAVE PLAYLIST". When the user selects "SAVE PLAYLIST", Jammming will save the current version of the playlist to the user's Spotify account.
* If the user selects a different playlist while the current playlist has unsaved changes, Jammming will load and display the new playlist, and will not save the changes to the old playlist. If the user returns to the old playlist, Jammming will display Spotify's version of the playlist, not the edited version.
* Fill up the navigation bar with region selection
* User can select multiple regions
* User can reset the selection and it will set to default search setting.

# TECHNICAL DESIGN

# Retrieve and Display Playlists and its tracks

A new component, **PlaylistList**, should be created. This component, on render, will retrieve a list of the current user's playlists.

We will need to initialize *state* for *PlaylistList* to contain a key for *playlists* that defaults to an empty array.

To retrieve playlists, we will create a new method, **getUserPlaylists()**, that hits the [https://api.spotify.com/v1/users/{user\_id}/playlists](https://developer.spotify.com/web-api/get-list-users-playlists/) endpoint. As we can see, this endpoint requires the user ID which we can retrieve with getUserProfile which return user ID.

In *getUserPlaylist()*, once the user ID has been retrieved, we can make our call to the */users/{user\_id}/playlists* endpoint. Upon completion of this request, we will update the current playlist's state to an array of the returned playlists. Rather than storing the entire playlists, we should create and store objects for each playlist that contain the **playlistId** and **name** of each playlist (the API will not return that playlist's tracks, thus we will need to use the playlist ID later to retrieve those tracks).

Add additional buttons above Tracklist component which will give a choice to user to create or update. Upon click of update button the the Track list should be removed from dom and PlaylistList component should render. After mounting process of component *playlistList*, call *.getUserPlaylists()* during mounted lifecycle with useEffect(). Upon success of api call the update button should removed from DOM and render instead drop down list of existing playlists.

Upon selection of user’s playlist the component playlistList should render Trakc component populated with selecttListItems (create separate api call function to get selectListItems as the getUserPlaylist will get only the name and id’s of playlist but not the tracks. Using returned value from last use these data to fetch and return an array of object from selectListItems. it should retrieve the playlist with the provided ID from getUserPlaylist. This method should call the [https://api.spotify.com/v1/users/{user\_id}/playlists/{playlist\_id}/tracks](https://developer.spotify.com/web-api/get-playlists-tracks/) endpoint and return a promise that will resolve to the retrieved tracks.

NOTE: the retrieved playlist track should inherit the use of customList use if the playlist track is retrieved it has to be passed to setCustomList state to modify it. And should also inherit the behavior of saving the playlist name

# Save Playlists

Saving a playlist should inherit behavior of trackList component save behaviour with addition to it it modify pushList.js function in order to check whether the playlist id exist if not it is a Tracklist behavior of and has Playlist ID it should behave according the Playlist component logic.

If Id exist it should follow the idea as per documantion of Spotify API to rename the playlist name <https://developer.spotify.com/documentation/web-api/reference/change-playlist-details> the initial state of name which will provided to the playlist name state should be same as retrieved from getPLaylist

It in order to update the the playlist items follow this documentation <https://developer.spotify.com/documentation/web-api/reference/reorder-or-replace-playlists-tracks>

Change name of playlist and update the playlist items has to be scoped into one function if needed separate this logic from pushList.js and the end logic of updating playlist has to set the customList state to ampty array;

Currently clear the *playlistName* and *playlistTracks* when a playlist has finished saving. We need to additionally clear the value of *playlistId* by setting it back to *null*.

# Retrieve and Display Playlists

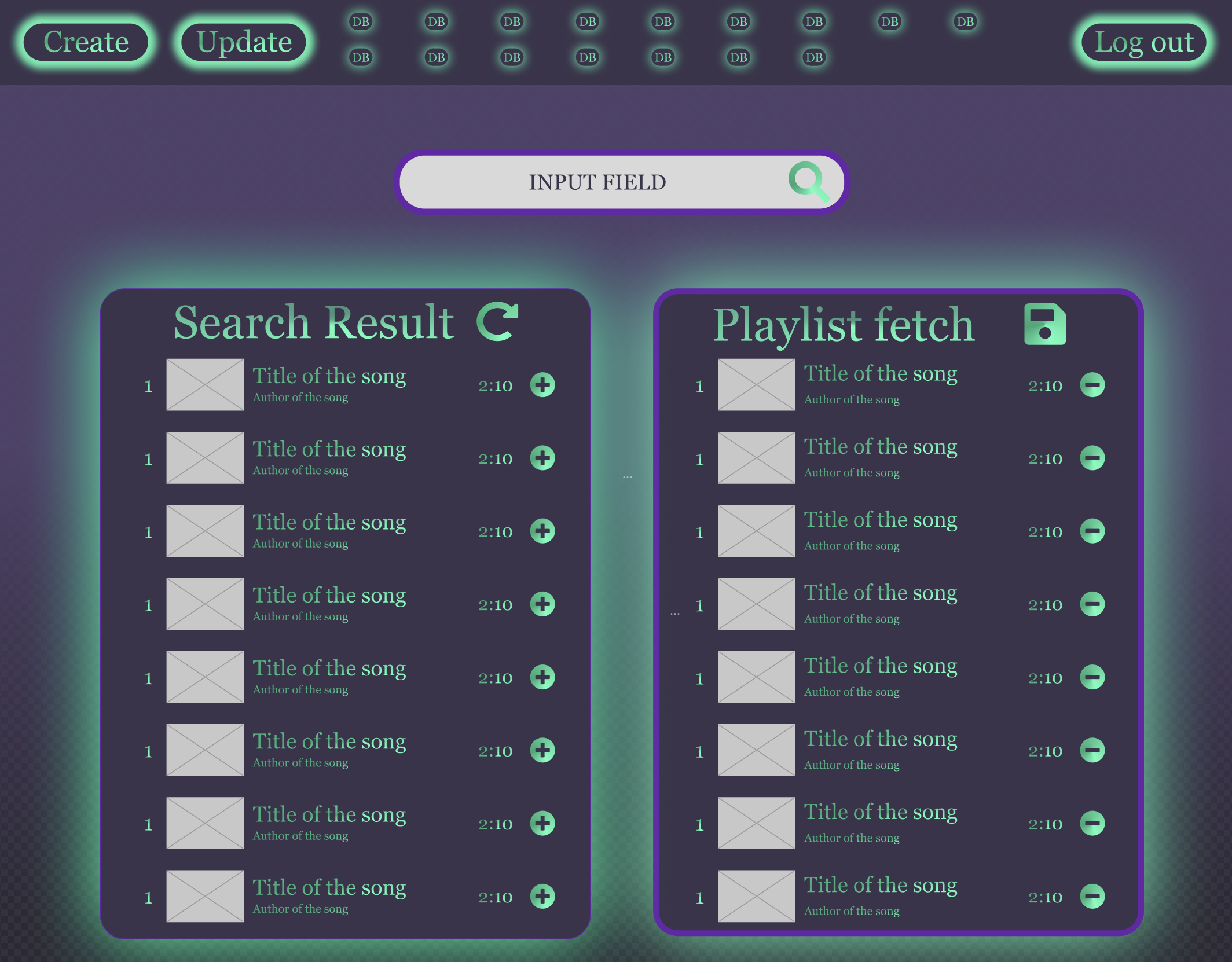
Create a component Regions container on menuNav component and take the entire space with flex control over layout of items. Each items should represent a region for narrowing search settings.

Create a dynamic customRegions state on Regions which holds an empty array. The state should be passed to spotifySearch module in order to set a query string “marke:” use a ternary operator to set default or provided array.

Create a reusable component Region which will be rendered by using map method inside of Regions component. In Regaions component create an object regions which hold a keys of regionValue and regionName and use it to map method and pass as a props (region) to a Reagion component. Pass setCustomeRegions function as props to Region component. use props to render input button should display props.region.regionName and hold a value props.region.regionValue

App should render all regions

App should look as follows upon implementations:



# CAVEATS

**Asynchronous Existing Playlist Save Requests**

In the initial implementation of Jammming, every playlist save necessitated a playlist to be created with the specified name and then the tracks to be saved to the playlist after that playlist creation completes. With the added functionality of saving a pre-existing playlist, we no longer need to create a new playlist as the first step every time. As a result, when saving a pre-existing playlist, we could send two simultaneous requests: one to save the updated name, the other to save the updated track list. This would result in a more complicated save playlist flow and would result in marginal returns in time efficiency when saving a playlist. Since the user is not prevented from continuing to interact with Jammming during playlist save, this efficiency would result in no change to user experience. Therefore we will not implement this functionality since the loss in code readability outweighs any gain in user experience. This would be a premature optimization.

**Excess Playlist Saves**

Currently, Jammming will save the displayed playlist name and tracks on every "SAVE PLAYLIST" click regardless of whether or not data has changed. This could result in unnecessary requests to the Spotify API (especially considering Playlist names are not likely to change often). As noted in the previous caveat, though, extra requests to the Spotify API will not result in any change to the user experience. If we find in the future that users are intentionally abusing this functionality and we run the risk of throttling the Spotify API, we will implement this functionality. However, at this point, this would be another premature optimization.