Assignment 8 Write-Up

Part 1: Describe your website

1. What is the purpose of your website?
    a. The purpose of my website is to help curate a playlist based on the weather of a user's city.
2. What information do you convey with your website?
    a. When using my website, users are able to find out the weather forecast for the near future (few hours from city search time).
3. How is it interesting and engaging?
    a. The playlists generated are dynamic so even if it's the same weather for several days in a row there will be variation in songs and song ordering in the playlists.
4. Who is the target audience?
    a. The aesthetics of the website targets a younger audience with vibrant colors and pop art font style.

Part 2: How does a user interact with your website?

1. Connect to Spotify account and grant website access to playlist read, write capabilities
    a. Authorization input from user. User clicks through auth screen and is redirected to website
    b. Authorization times out, must reauth.
2. User inputs their city (any correctly spelled city in the world)
    a. If no value detected in input field when user clicks submit an alert will pop up to prompt city input
    b. User's screen will auto scroll to view the weather icon
3. User clicks on *Create Playlist, Get Recs*, and then *Add Tracks* in that order to build out their playlist.
    a. Clicking the buttons in any other order will break functionality
    b. More details written in Section 5
4. User can listen to their new weather curated playlist in the browser.
    a. After the Add Tracks button is clicked screen will auto scroll to view the playlist

Part 3: Describe what external tool you used (JavaScript library, Web API, animations, or other).

Name of tool: Accuweather API

1. Why you chose to use it?
    a. Free to use with account
    b. Had clear documentation of endpoints better than WeatherNetwork
    c. Had easy to parse responses
2. How you used it?
    a. I used the following API functionalities
        i. Get locationkey from city search
        ii. Get hourly forecast 1 hour
    b. Response from location key endpoint is needed for forecast api call
3. What it adds to your website?
    a. Getting accurate weather forecast is the first step to building out a playlist of weather based songs

Name of tool: Spotify API

1. Why you chose to use it?
    a. Widely considered the most popular music streaming platform aside from Youtube
    b. Is a popular API so many devs have played with it and I thought any question I had would likely have already been answered on stackoverflow
2. How you used it?
    a. I used the following API functionalities
        i. Connecting Spotify account through browser client
        ii. Creating a new playlist in Spotify account
        iii. Getting recommendations with params seed_genres, valance (float), danceability (float), energy (float)
            1. I A/B tested playlists generated with different ranges for valance, danceability, and energy.
        iv. Adding recommended tracks to playlist
        v. Displaying embedded music player in website after tracks added
4. What it adds to your website?
    a. Spotify API contributes the bulk of the functionality of the site.

Part 4: Describe how you iterated on your HW7 mockups, if at all, including any changes you made to your original design while you were implementing your website. (2-4 sentences max)

Not many changes were made. My mockups included one version which was a one screen site and another version that scrolled. My final followed the scrolling version.

Part 5: What challenges did you experience in implementing your website? (2-4 sentences max)

1. Making async functions execute in the order I want
    a. Connecting the different functionalities (creating playlist, getting recommendations, adding tracks) into one function was difficult. Could not figure out how to make the async API calls execute in the order needed.
    b. Tried async - await, tried promises, ultimately was not able to achieve a one button solution
2. No context sharing between https and http
    a. Accuweather API uses http urls and github pages is https
    b. While testing on my local machine (localhost) all accuweather api calls executed without error, but live pages causes errors unless secure browsing is disabled
    c. I learned that http and https cannot share content