BCDV1007 - Full Stack II
Music Sharing Interface

Due **Monday, Nov 9th, 1AM** – 25% of Final Grade

System Requirements

Contact: **Mike Denton**

Date: **Oct 26th 2020**

Version #: 1.0

1 Objective

This document contains a specification of the course assignment. It is a task where students practice skills to build a dynamic web interface using the HTML,CSS and JavaScript.

This task will also include, working together in a group developing the project, plan, manage and coordinate development activities, to be done effectively to a deadline.

2 Teams

Teams will consist of one to two students (with two being the target number).

3 User Interface

The UI design will be HTML5/CSS3 and JavaScript

4 Specification

The project will be based on a Music Download Application such as iTunes or Napster.

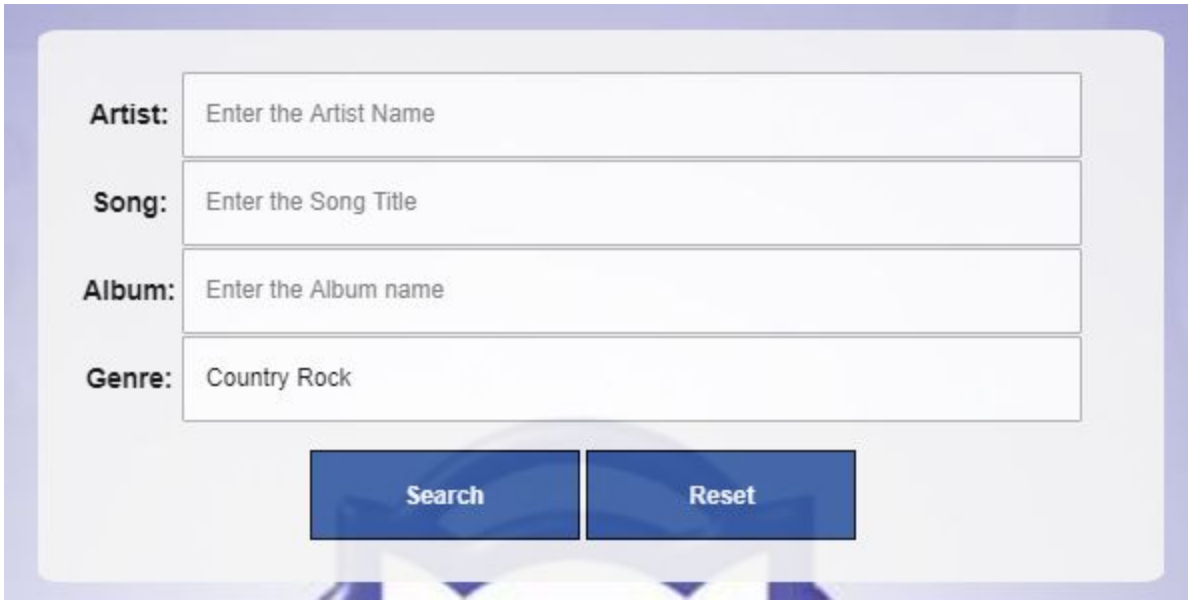
5 Requirements

The minimum viable product MVP for our interface is we need to have the following:

- A search form with inputs to search for an Artist, Song Title, Album and Genre
    - The search form will trigger a search and filter the music data and display the results in the search results container
- A table/container of a list of Favorite Album Cover images
- A container with that contains search results
    - The song information includes Artist, Song, Time, Album, Rating, Genre
    - The song data rows will have buttons Download and Favorite

5.1 Music Search Form

- The form should have the following text inputs Artist, Song, Album
  - Search criteria should be case insensitive
- The form should have one dropdown/select input for Genre
  - Genre should have at least three options for selection
- The form should submit a search without reloading the entire page
- Clicking the Search Button
  - Will trigger a search if there is search criteria entered
  - Will display a prompt 'Please enter search criteria'
- The form will trigger a search on the following user actions:
  - Click the Search button
  - Presses Enter key in any of the input text boxes
- The form will reset on the following user actions:
  - Click of the Reset button



5.2 Favorite Lists

- The favorites list is a dynamic list of Album cover images
    - Image will be added to this container on User click of the Favorites button in the search results table
    - Images will removed from the container when the user clicks on the image
    - The favorites list should remain on each new search submission

5.3 Music Search Results

- The search results is a dynamic container that displays the results of the search form criteria
- The search results has the following visibility:
    - It will not be visible on page load, as there is no criteria and no initial results
    - It will be visible when there is matching result are found after a search has been triggered by the search form
- Each data row will contain a column for Artist, Song, Time, Album, Rating and Genre
- Each data row will contain the following buttons
    - A Download button
        - When clicked it will display an alert message ‘Download started’;
    - A Favorite button
        - When clicked it will result in adding the Album cover image of the current song row to the Favorites list

Artist	Song	Time	Album	Rating	Genre					
Jason Aldean	Dirt Road Anthem	3:49	My Kind of Party	4	Country Rock		Favorite			Download
Jason Aldean	Tattoos an Tequila	4:44	We Back	4	Country Rock		Favorite			Download

5.4 Music Data

- The data used for this application is a static data structure of Song details within the page. There will be no external API or data retrieved from a database.
- There should be at minimum 15-20 songs in the data structure, with a mix of at least 3 genres. (ie. 5 Jazz, 8 Rock, 7 Classical)

6 Demo

- A demo of this application can be found at the following link:

  https://youtu.be/78atc-RtAfM

7 Submission

1. The project will uploaded and submitted to Black Board
2. Include a **README** file with the project that includes the following:
   - The names and student number of all the members of the team.
   - Instructions for installing or running the project.

Specification	Percentage
Search Form	25%
Search Results	35%
Favorites List	15%
UX Design	5%
Clean Code and Clarity	10%
* Demonstration of Programming Concepts	15%

** **Demonstration of Programming Concepts** includes logical approach and use of JavaScript best practices. Also, use of the DOM Level API (window and document) for event handler and DOM traversal and manipulation. Use of ES6 syntax where possible is also considered.

Bonus

Bonus marks will be awarded for the following completed items. Note: this is only applicable where the project mark is less than 100%

Specification	Percentage
Implementing Remembering Previous Favorites selection	3%
Deployed to Web Server/Hosting - Live Web Site	2%