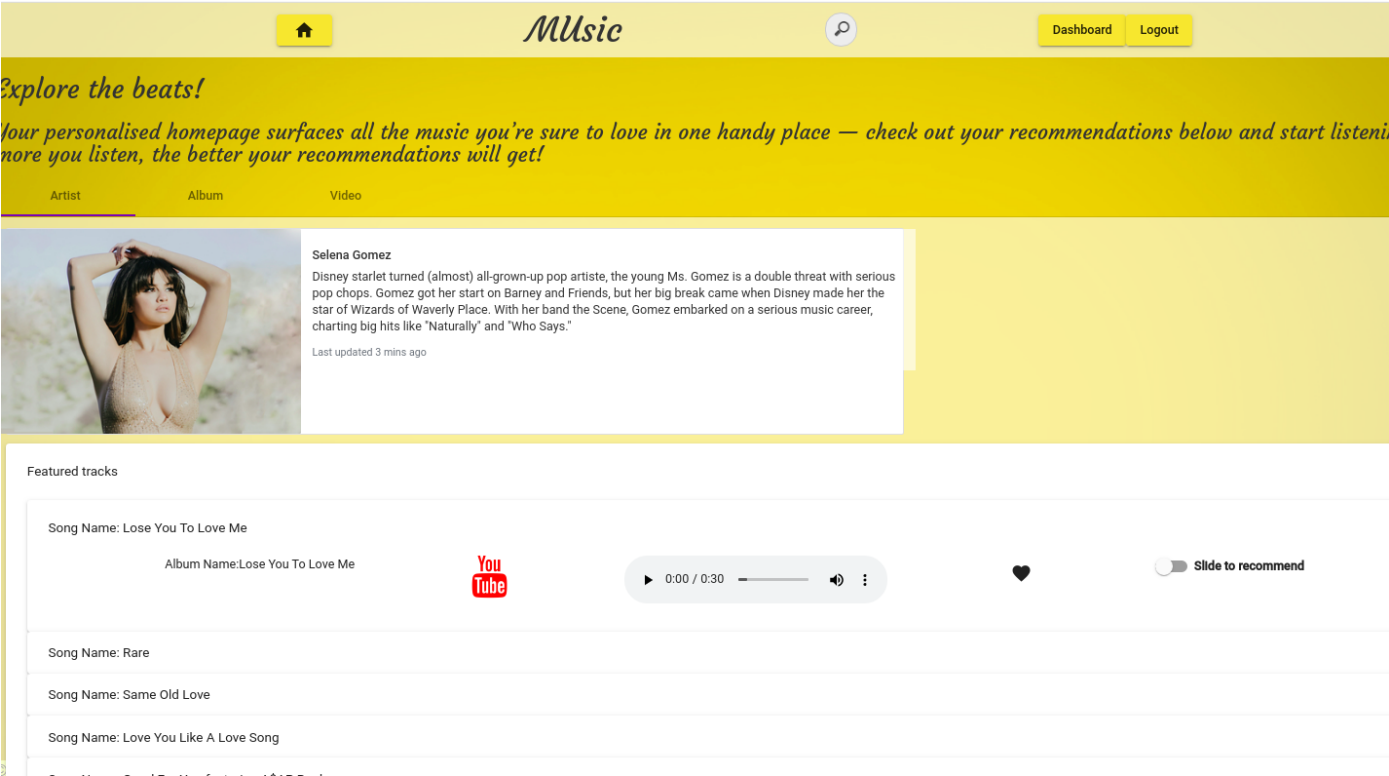📖 **README.md**

Music App

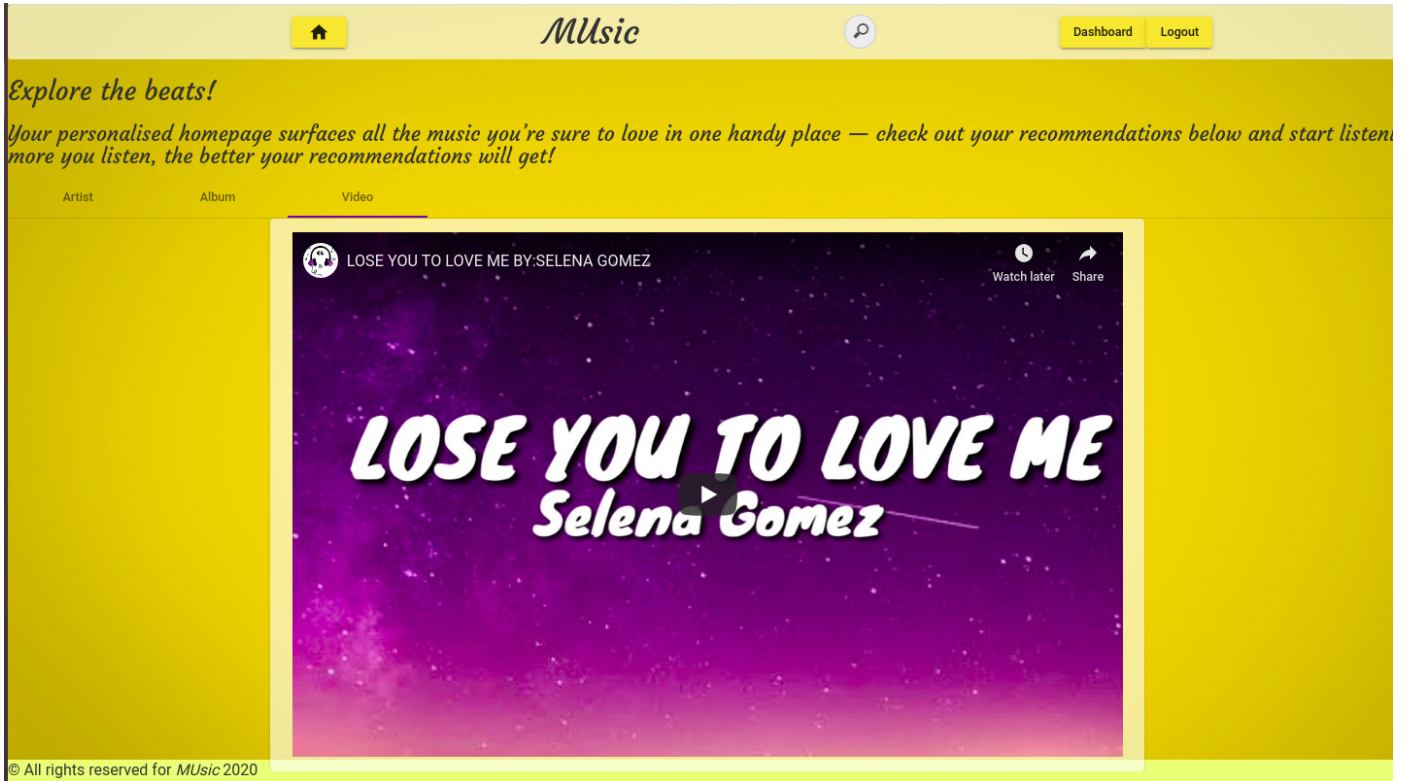Building a full-stack Music App

- Search, Recommendation, Adding To favourites, Adding Comments, Country Special
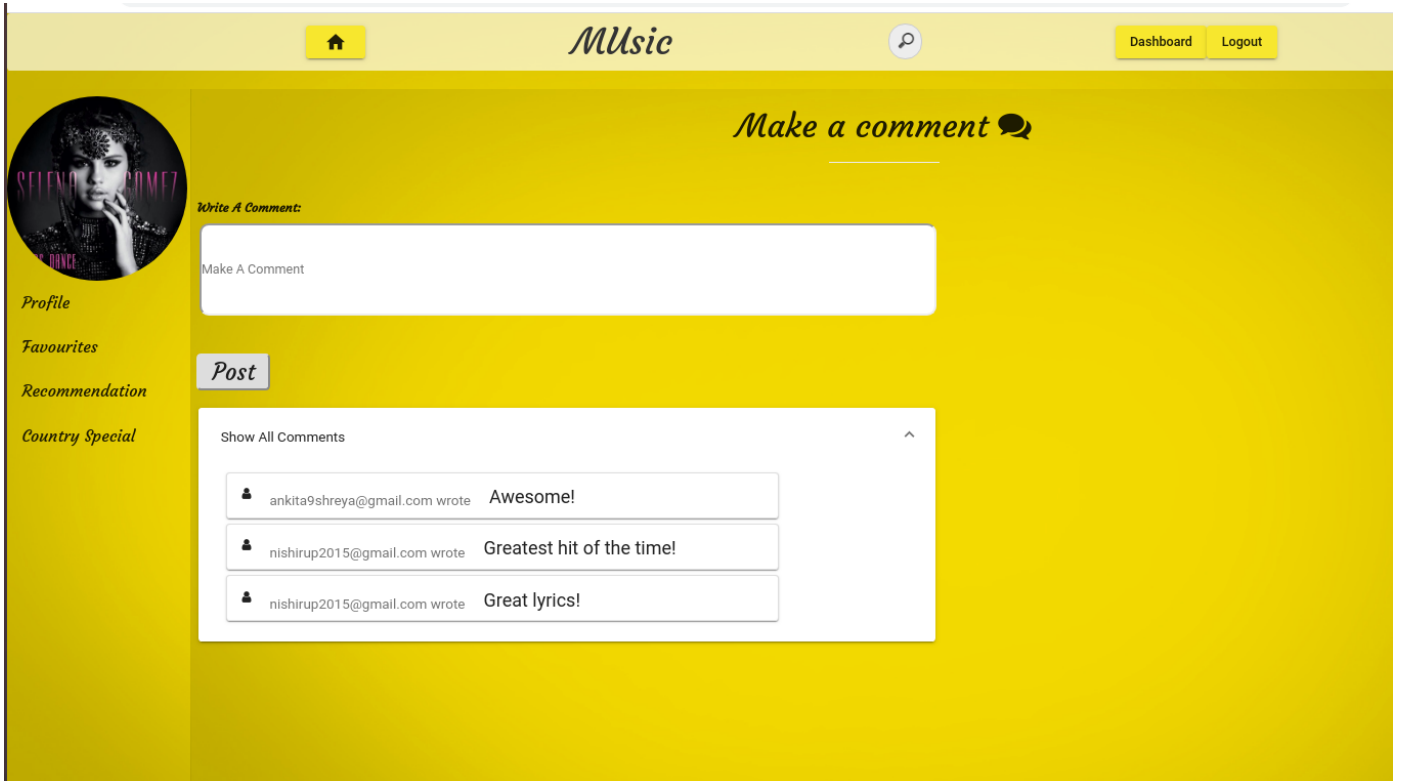- User friendly UI

New Features

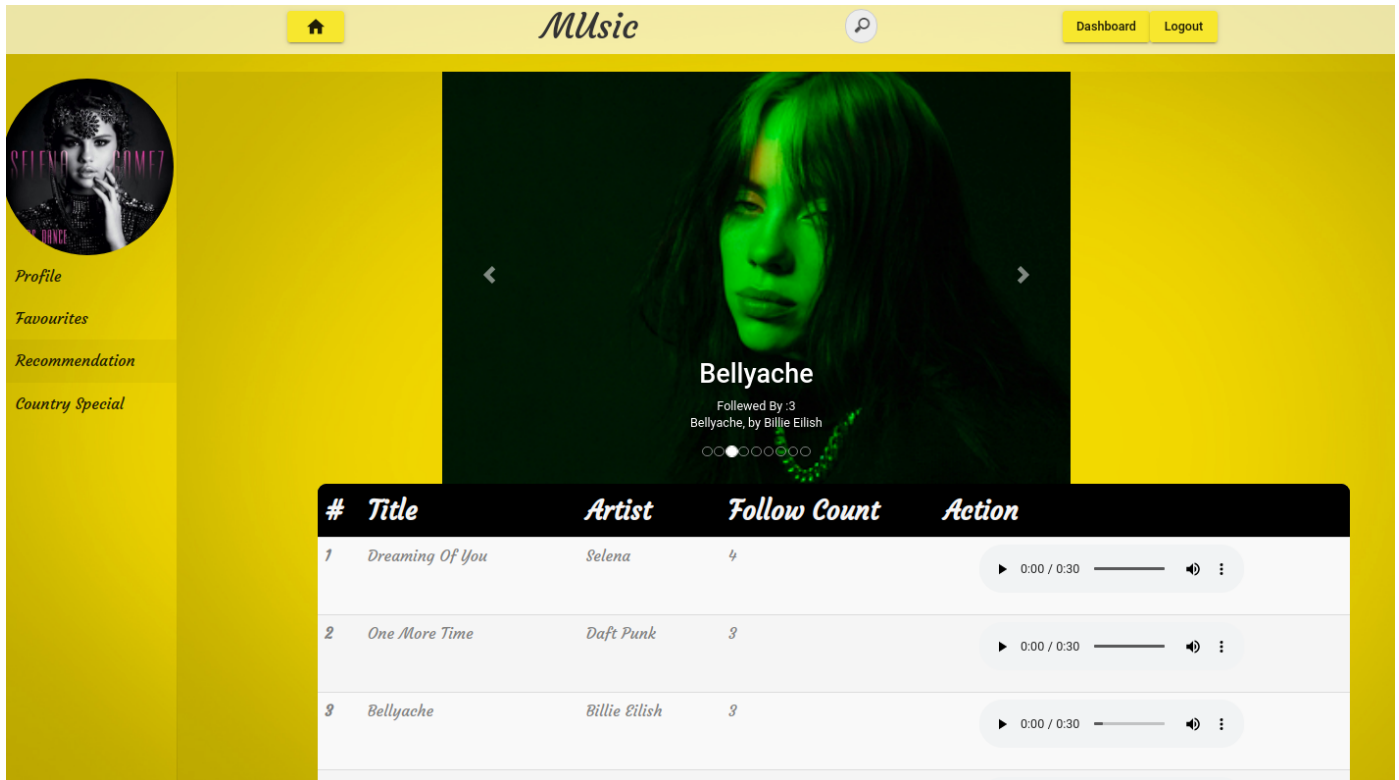- Search feature with multiple categories of artist, albums, Video

- Videos section added in search(shows search result from youtube data)

- Emails the users at registration time

- Allows to add comment to your Favourite songs

- Shows Recommendation of users

Project Milestones

Milestone 1

- Create REST API to fetch data(refer Napster API)
- Create a search bar with a search button to search musix by artist/playlist.
- Search section will have text field with ID #search-text-field
- A button to submit search text field content with ID
- On submission of search text should query the results from last.fm or napster.com and display search results.
- Get the results displayed in search results section.
- Give an id #search-results to search result section.
- Search results should show a series of card like components and a card should have the following attributes.
- Assign .musix-card class to each playlist cards and all the playlist cards displayed in all sections should have the below details with attributes.
- assign .musix-track class to describe track name.
- assign .musix-artist class to describe artist.
- assign .musix-image class to show image.
- toggle .recommend & .unrecommend classes to recommend and unrecommend buttons.
- Create a recommend button attached with every playlist card. Give recommend button a class .recommend(toggle button)
- Click on Recommend button and the button should change to Unrecommend.
- The Unrecommend button should have a class .unrecommend and Recommend button should have class .recommend.
- View recommended palylist under my-recommendations section
- Display all recommended palylist in this section. Recommended playlist should be displayed under my-recommendations section.
- Unrecommend button, button should change to Recommend again, playlist should disappear from my-recommendations section

Milestone 2

- Create a Dashboard view (Angular Route /dashboard) with four sections Display Favorite Playlist, countries(playlist from different countries), recommendations for a playlist from napster.com/last.fm one under the other and User Profile Page view.
- This Dashboard is the default view to be shown.
- The 3 sections are:
- Favorite with Id #Favorite.

- Countries with ID #Countries.
- Recommended tracks with ID #recommended
- View all Favorite tracks/playlist cards under Favorite section
- Display all Countries tracks/playlist under Countries section
- View all playlist/tracks recommendations from 3rd party tracks service provider (NAPSTER API/last.fm) under recommendations section

Milestone 3

Pagination

- In case of multiple records page should have pagination option to display data in card layout

Authorization

- User should not be able to add anything in favorites & recommendation list until logged in.
- Create Login Page
- Create Register Page (User's email address should be a userid)
- Create Edit Profile / Change Password page (Email address cannot be changed)
- Upload profile image while register & displaying the same in toolbar after login
- Encrypt password using bcrypt while storing in database during registration

Milestone 4

- Implement Test Automation – write Unit Tests for Backend and FrontEnd.
- Add E2E Test Cases, Dockerize and Implement CI using Gitlab Runner.
- Create the **README.md** file with Steps to Execute.

Acceptance Criteria

The final evaluation of submissions if the following is achieved:

- Meeting the Core functional requirements as stated
- Write Test Cases with each functionality with all the features
- Follow the MVC model
-

Evaluation Rubrics (*On a Rating from 1 to 10 – wherein **1** is **Non Existent** to **10** being **Exceeded Expectations***)

1. Functional Requirements
2. Non-Functional Requirements
3. Code Quality
4. Standards, Styles and Guidelines
5. Aesthetics and Accessibility