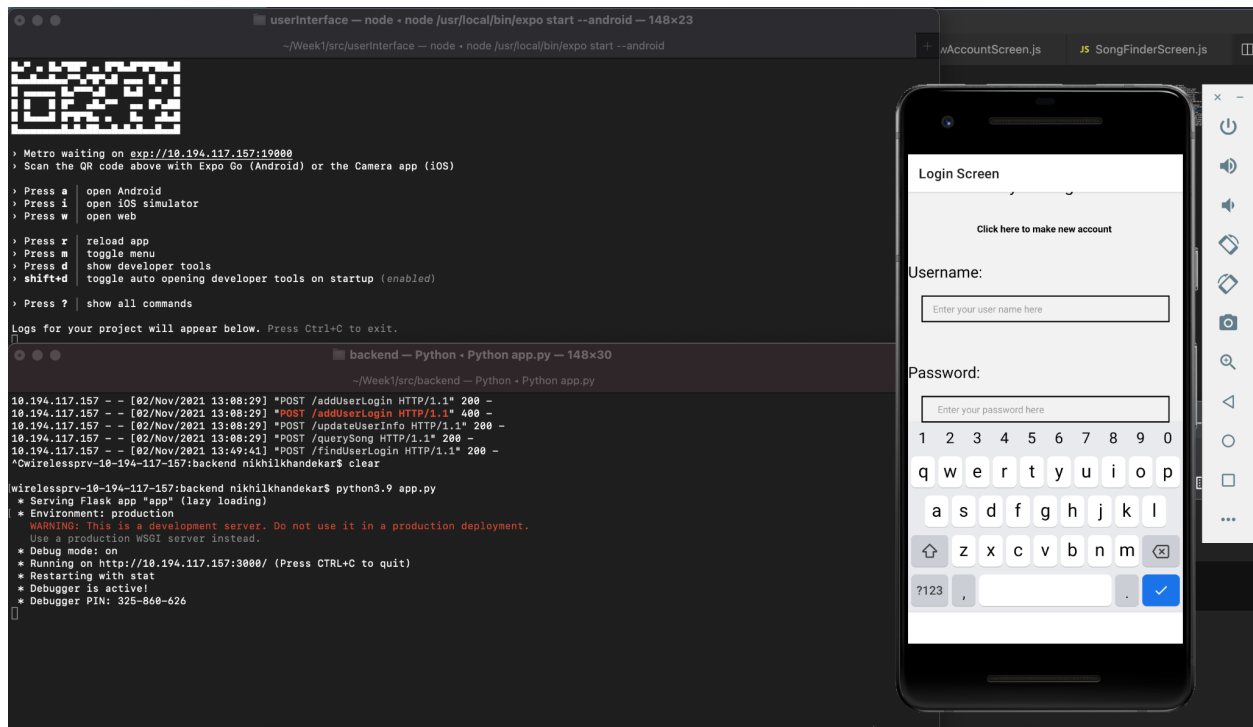


Week 1 Manual Test Plan

Installation Requirements:

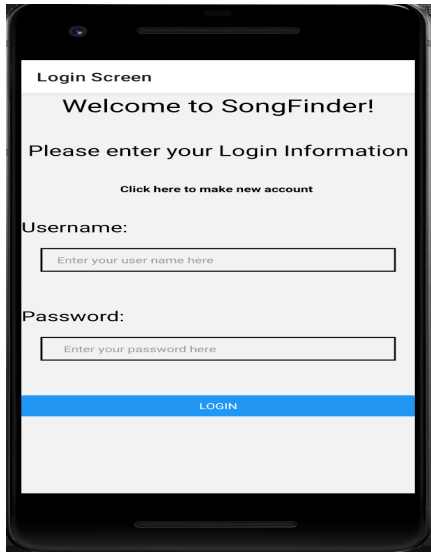
- Android Studio and the installation for the emulator to be able to see the application work on a mobile
- Python3.9
- Flask
- React Native
- Requests library
- Axios Library from React
- React Navigation from React Native

To run the application, open two terminals. cd one terminal into the backend directory and type in 'python3.9 app.py'. In another terminal, cd one terminal into userInterface directory and type in 'expo start --android'. In doing this, one will see the emulator open up the android page:

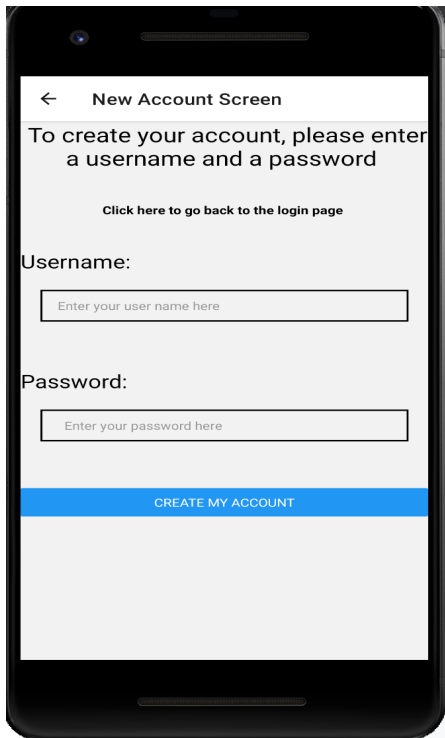


Manual Test Plan for Create Account Page

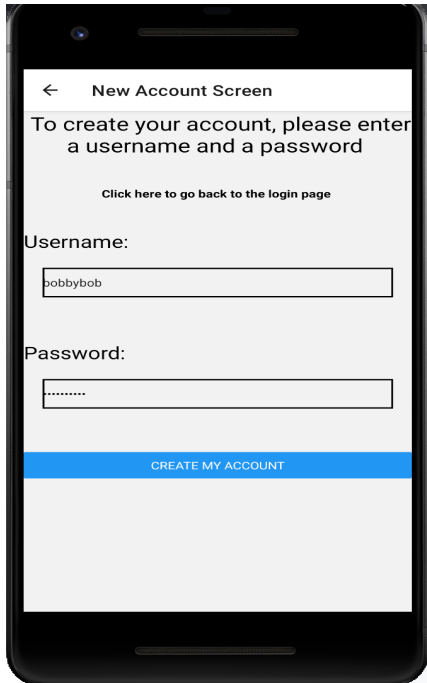
Suppose a user does not have an account and wants to create one. This can be done by first clicking the text “Click here to make a new account”.



From here, the user will get prompted to a new page:

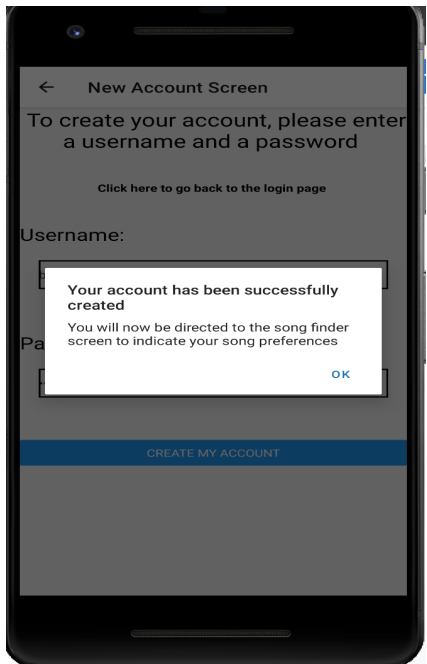


User will then enter their information onto the page and click create my account:



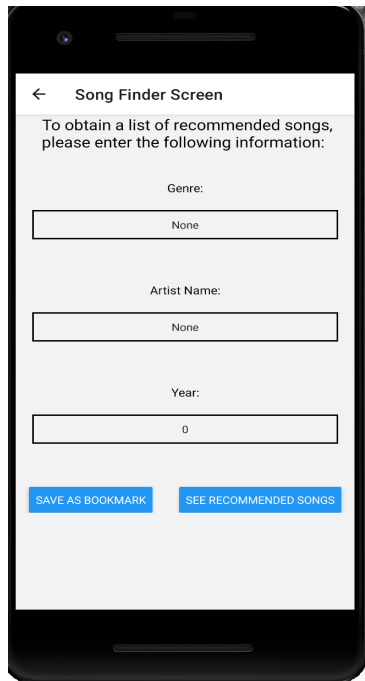
A mobile app screen titled "New Account Screen". At the top, there is a back arrow and the title. Below the title, a message says "To create your account, please enter a username and a password". Underneath, a link reads "Click here to go back to the login page". The form has two sections: "Username:" with a text input field containing "bobbybob", and "Password:" with a password input field showing eight dots. At the bottom, there is a blue button labeled "CREATE MY ACCOUNT".

If the username is unique, then an alert will be shown showing that the account has been successfully created.



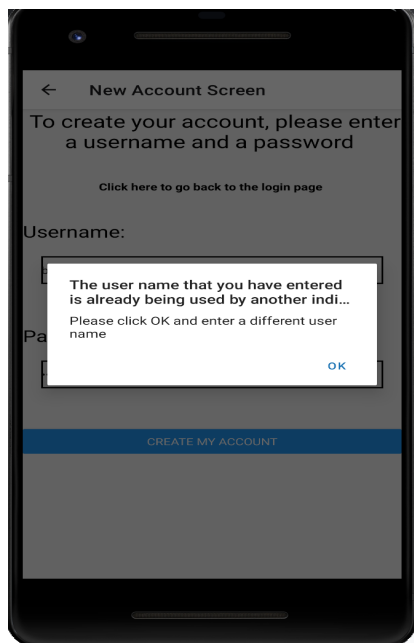
The same "New Account Screen" as before, but with a white alert dialog box in the center. The dialog box contains the text "Your account has been successfully created" and "You will now be directed to the song finder screen to indicate your song preferences". There is an "OK" button in the bottom right corner of the dialog box. The background of the screen is dimmed.

Clicking ‘OK’ will direct them to the song page for entering in their song information:



The image shows a mobile application screen titled "Song Finder Screen". At the top, there is a back arrow and the title. Below the title, a message states: "To obtain a list of recommended songs, please enter the following information:". There are three input fields: "Genre:" with a dropdown menu showing "None", "Artist Name:" with a dropdown menu showing "None", and "Year:" with a text input field showing "0". At the bottom, there are two blue buttons: "SAVE AS BOOKMARK" and "SEE RECOMMENDED SONGS".

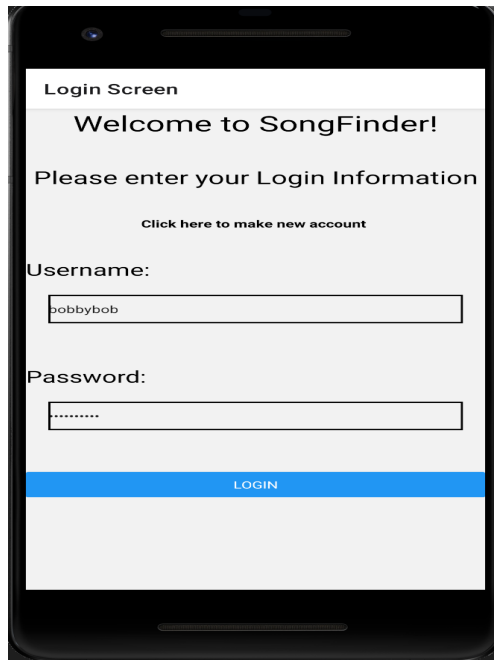
However, if a username is already registered inside the MongoDB database, then an alert will show saying that the user needs to choose a new username once a user clicks on the “Create my Account Button”



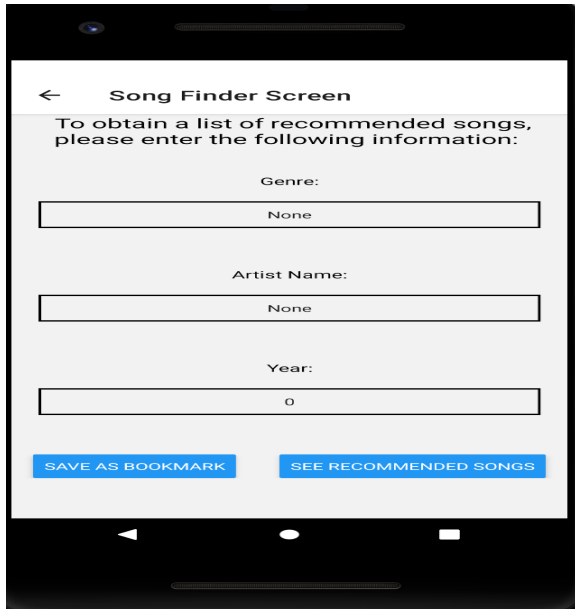
The image shows a mobile application screen titled "New Account Screen". At the top, there is a back arrow and the title. Below the title, a message states: "To create your account, please enter a username and a password". There is a link that says "Click here to go back to the login page". Below this, there are input fields for "Username:" and "Password:". A white alert box is displayed over the form, containing the text: "The user name that you have entered is already being used by another indi... Please click OK and enter a different user name". There is an "OK" button in the alert box. At the bottom, there is a blue button that says "CREATE MY ACCOUNT".

Manual Test Plan for Login Page

For a user to be able to login, they need to go to the login page and enter in their username and password. For this application, I will be using the account that I created from the Manual Test Plan for Create New Account Page. The username was bobbybob and the password was nikhil7153:



Upon clicking the login, if the login was successful, then the page will re-direct to the page for a user to give their preference:



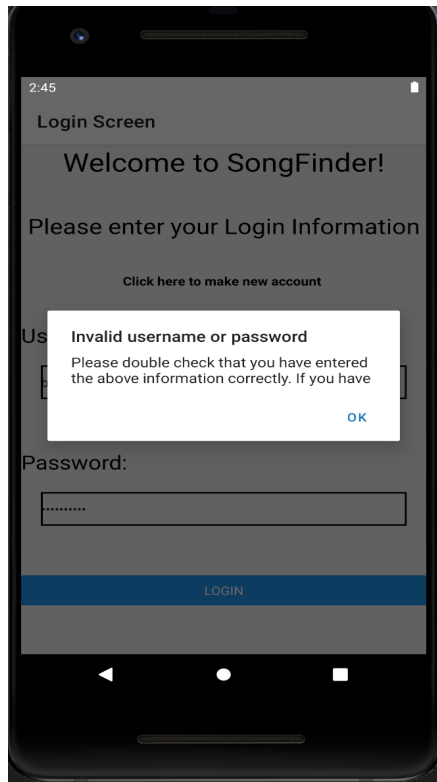
The image shows a mobile application screen titled "Song Finder Screen". At the top, there is a back arrow icon. Below the title, a message states: "To obtain a list of recommended songs, please enter the following information:". There are three input fields: "Genre:" with a dropdown menu showing "None", "Artist Name:" with a dropdown menu showing "None", and "Year:" with a text input field showing "0". At the bottom, there are two blue buttons: "SAVE AS BOOKMARK" and "SEE RECOMMENDED SONGS". The screen is displayed on a black smartphone frame.

However, if a user's login is not successful, because the username or password is not correct, an alert will be given as follows. The username will be wrong in this input as the username 'bobbyb' is non-existent in the database of user logins:



The image shows a mobile application screen titled "Login Screen". At the top, there is a back arrow icon. Below the title, a message states: "Welcome to SongFinder!". Below this, it says "Please enter your Login Information". There is a link that says "Click here to make new account". Below the link, there are two input fields: "Username:" with a text input field containing "bobbyb", and "Password:" with a text input field containing "*****". At the bottom, there is a blue button labeled "LOGIN". The screen is displayed on a black smartphone frame.

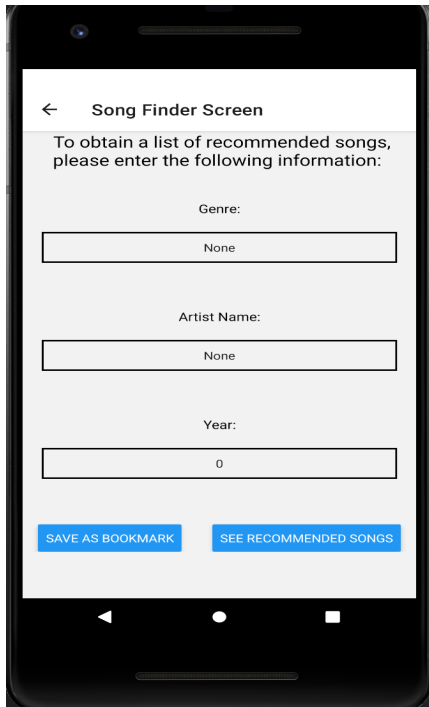
After clicking “Login,” the following alert will be shown:



Manual Test Plan for Song Finder Page

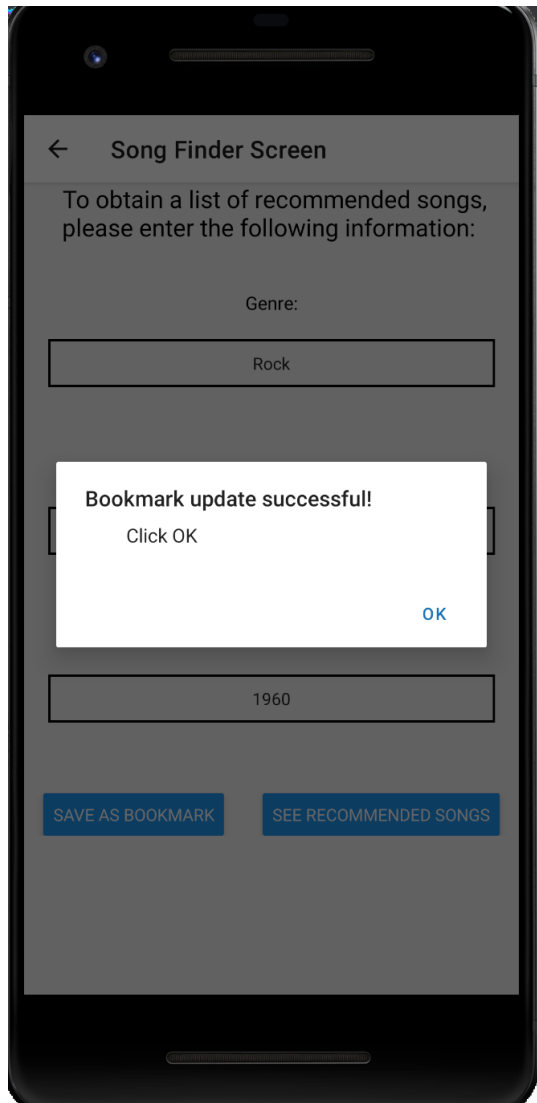
Upon successfully logging in or creating an account, a user will get directed to the Song Finder which will show the following form:

This user has never saved a bookmark which is why the following default values are shown:



The screenshot shows a mobile application interface for the 'Song Finder Screen'. At the top, there is a back arrow and the title 'Song Finder Screen'. Below the title, a message states: 'To obtain a list of recommended songs, please enter the following information:'. The form consists of three input fields, each with a label above it: 'Genre:' with a dropdown menu showing 'None', 'Artist Name:' with a dropdown menu showing 'None', and 'Year:' with a text input field showing '0'. At the bottom of the form, there are two blue buttons: 'SAVE AS BOOKMARK' and 'SEE RECOMMENDED SONGS'. The entire form is displayed on a black smartphone screen with a white background for the form itself.

However, suppose a user types in some input for each of the parameters. In this case, if a user, bobbybob, clicks on “Save As Bookmark,” after populating the fields, an alert message will be shown:



The page will then reload with the new user preferences:

← Song Finder Screen

To obtain a list of recommended songs,
please enter the following information:

Genre:

Rock

Artist Name:

Bob Dylan

Year:

1980

SAVE AS BOOKMARK SEE RECOMMENDED SONGS

Note that the application also checks user input for the year parameter and ensures that the user is inputting a year between 1950 and 2021 and that the year can be converted to an integer. In the cases that the user's input fails for the year param, an alert message will be shown:

The screenshot shows a mobile application interface titled "Song Finder Screen". At the top, there is a back arrow and the title. Below the title, a message states: "To obtain a list of recommended songs, please enter the following information:". There are three input fields: "Genre:" with the value "Rock", "Artist Name:" with the value "Bob Dylan", and "Year:" with the value "1910". At the bottom, there are two blue buttons: "SAVE AS BOOKMARK" and "SEE RECOMMENDED SONGS".

The screenshot shows the same "Song Finder Screen" as above, but with an alert message displayed. The alert message is titled "Year value is not valid!" and contains the text: "Please enter an integer value greater than 1949 and less than 2022 to get the correct year". There is an "OK" button in the bottom right corner of the alert. The input fields show "Genre: Rock", "Artist Name: Bob Dylan", and "Year: 1910". The "SAVE AS BOOKMARK" and "SEE RECOMMENDED SONGS" buttons are still visible at the bottom.

The same result will apply for a string value for the Year parameter:

← Song Finder Screen

To obtain a list of recommended songs, please enter the following information:

Genre:

Rock

Artist Name:

Bob Dylan

Year:

sdhfhdsfjkjhds

SAVE AS BOOKMARK SEE RECOMMENDED SONGS

After clicking the “Save as Bookmark” button, the following alter will be shown:

← Song Finder Screen

To obtain a list of recommended songs, please enter the following information:

Genre:

Rock

Artist Name:

Bob Dylan

Year:

sdhfhdsfjkjhds

SAVE AS BOOKMARK SEE RECOMMENDED SONGS

Year value is not valid!

Please enter an integer value greater than 1949 and less than 2022 to get the correct year

OK

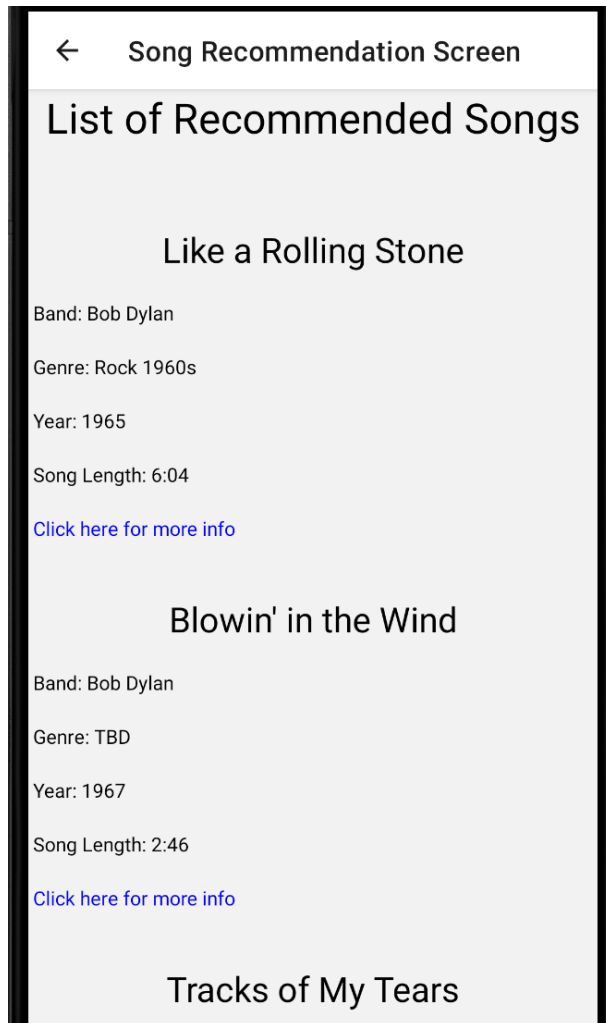
Manual Test Plan for Recommended Songs

To get to the recommended songs page, one must input valid information for Years parameter in the Song Finder form. If not, then the same alert messages will be shown. If not, then the user will get the same alert messages that they got for creating a bookmark.





Suppose the user chooses to click on the “See Recommended Songs” button upon entering this form information:

The screenshot displays a mobile application screen titled "Song Finder Screen". At the top left is a back arrow icon. Below the title, a grey instruction box contains the text: "To obtain a list of recommended songs, please enter the following information:". Below this, there are three input fields, each with a label above it: "Genre:" with the value "Rock", "Artist Name:" with the value "Bob Dylan", and "Year:" with the value "1965". At the bottom of the form, there are two blue buttons: "SAVE AS BOOKMARK" on the left and "SEE RECOMMENDED SONGS" on the right.

The following information will then get inputted from this information. Note that this query will output any song that is matching at least one of the parameter values:



If someone clicks on the blue text, it will go to a website from which all the data is fetched from:

 cs.ubc.ca/~davet/music/track  

Dave Tompkins :: Music Database

[INTRODUCTION](#) [DISCS](#) [COVERS](#) [SONGS](#) [ARTIST TAGS](#) [YEAR](#) [FORM](#)

[ARTISTS](#): [P](#) [A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

[SONG TITLES](#): [P](#) [A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

[LISTS](#): [BEST](#) [CHART](#) [COMMENT](#) [DJ](#) [NAMES](#) [KEYWORDS](#) [SERIES](#) [EVENTS](#)

[SEARCH](#): [SONG](#) [LYRICS](#) [ARTISTS](#)

Bob Dylan
Like a Rolling Stone

Song Information

Song Artist(s): [Bob Dylan](#)

Song Title: [Like a Rolling Stone](#)

Year: [1965](#)

Album Information

Album: [Power To The People 195](#)

Track: [14](#)

Track Classification

Genre: [Rock 1960s](#)

MusicBrainz Genre: [Rock/Folk](#)

Audio

[20 sec. sample](#)

Lyrics

Original Source: [LyricWiki](#)

Once upon a time you dressed so fine
Threw the bums a dime in your prime, didn't you?
People call, say, "Beware doll, you're bound to fall"
You thought they were all kiddin' you
You used to laugh about
Everybody that was hangin' out
Now you don't talk so loud
Now you don't seem so proud
About having to be scrounging your next meal

How does it feel
How does it feel

If a user inputs values for all fields where the database does not have anything in common with any fields then the following output will be shown when the “See Recommended Songs” gets clicked:

[←](#) Song Finder Screen

To obtain a list of recommended songs,
please enter the following information:

Genre:

hip-hop

Artist Name:

Taylor Swift

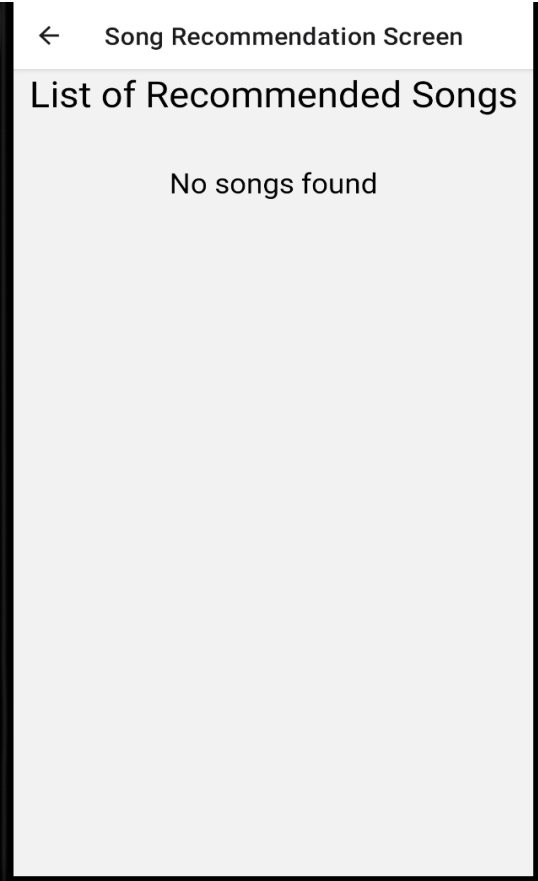
Year:

2016

SAVE AS BOOKMARK

SEE RECOMMENDED SONGS

Result:

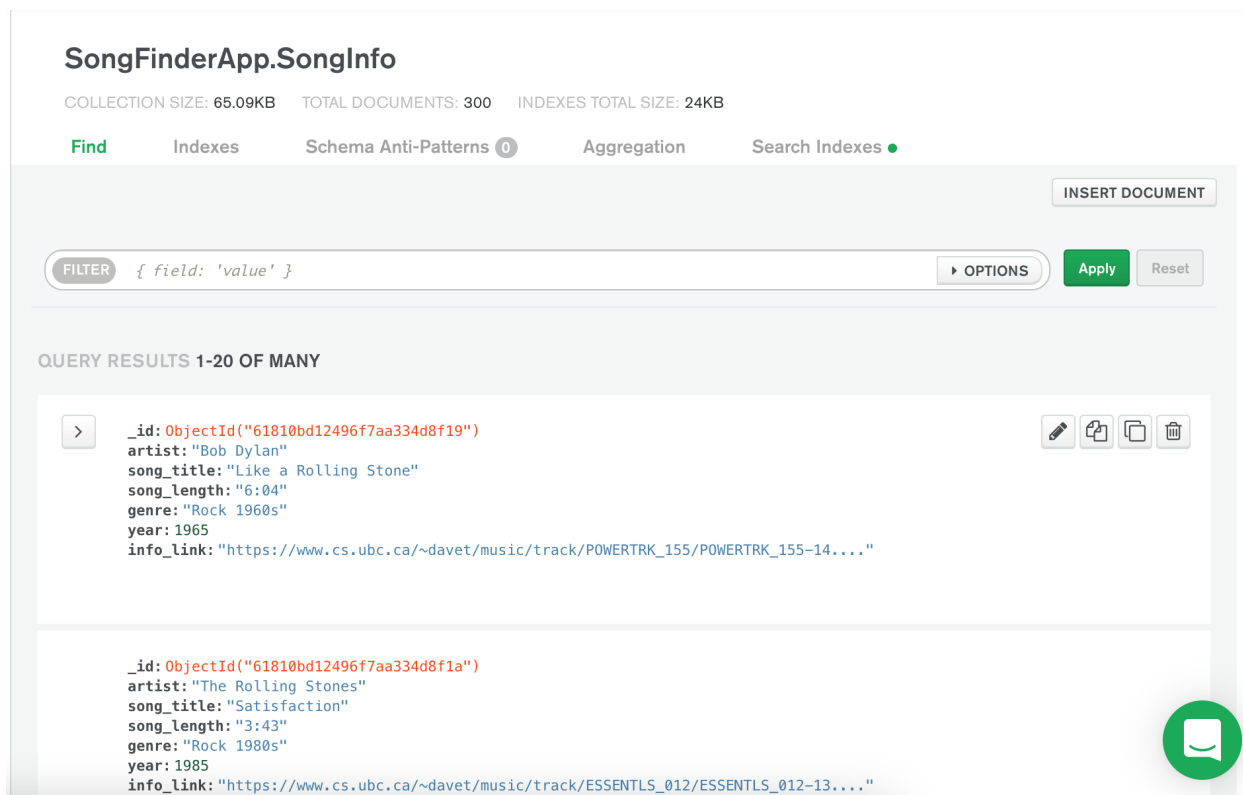


Manual Test Plan for Webscrapping

To webscrape data, cd into the webscraper directory and then run the command `python3.9 Driver.py -n <number of songs to be scrapped>`:

```
wirelessprv-10-194-117-157:backend nikhilkhandekar$ cd webscraper
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$ python3.9 Driver.py -n 300
```

The following will get outputted inside the MongoDB database:



The screenshot displays the MongoDB SongFinderApp interface. At the top, the title 'SongFinderApp.SongInfo' is shown. Below it, statistics indicate a collection size of 65.09KB, 300 total documents, and a total index size of 24KB. Navigation tabs include 'Find' (active), 'Indexes', 'Schema Anti-Patterns', 'Aggregation', and 'Search Indexes'. An 'INSERT DOCUMENT' button is located in the top right. A filter bar contains the text '{ field: 'value' }' and buttons for 'OPTIONS', 'Apply', and 'Reset'. The 'QUERY RESULTS 1-20 OF MANY' section shows two document entries. The first entry is for Bob Dylan's 'Like a Rolling Stone' (1965, Rock genre, 6:04 length). The second entry is for The Rolling Stones' 'Satisfaction' (1965, Rock genre, 3:43 length). Both entries include an '_id' field with an ObjectId and an 'info_link' field with a URL. A green chat bubble icon is visible in the bottom right corner.

This class can also handle bad output as well such as strings instead of ints and ints that are greater than 500 or less than 1 will output an error:

Error for string being given as an input:

```
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$ python3.9 Driver.py -n sdfkds
usage: Driver [-h] [-n NUMSONGS]
Driver: error: argument -n/--numSongs: invalid int value: 'sdfkds'
```

Error for value less than 0 being as an input for number of songs:

```
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$ python3.9 Driver.py -n -1
Traceback (most recent call last):
  File "/Users/nikhilkhandekar/Week1/src/backend/webscraper/Driver.py", line 14, in <module>
    scraper = SongInfo(numSongs)
  File "/Users/nikhilkhandekar/Week1/src/backend/webscraper/SongInfo.py", line 24, in __init__
    raise ValueError("Please enter a number between 0 and 500")
ValueError: Please enter a number between 0 and 500
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$
```

Error for value greater than 500 being as an input for number of songs:

```
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$ python3.9 Driver.py -n 600
Traceback (most recent call last):
  File "/Users/nikhilkhandekar/Week1/src/backend/webscraper/Driver.py", line 14, in <module>
    scraper = SongInfo(numSongs)
  File "/Users/nikhilkhandekar/Week1/src/backend/webscraper/SongInfo.py", line 24, in __init__
    raise ValueError("Please enter a number between 0 and 500")
ValueError: Please enter a number between 0 and 500
wirelessprv-10-194-117-157:webscraper nikhilkhandekar$
```