

# Project Proposal

- **Basic Information:** The project title, your name, and e-mail address

**Title:** Guess the artist's top song

**Name:** Leisha Figuerres

**Email:** [lrfiguerres@dons.usfca.edu](mailto:lrfiguerres@dons.usfca.edu)

- **Background and Motivation:** Discuss your motivations and reasons for choosing this project, especially any background or interests that may have influenced your decision.

I chose a project that uses the spotify api because one of my most dearest interests is music. I am also a dedicated spotify user, who uses it every day. I believe creating a game where people can test how well they think they know their favorite artist's top songs will be a fun and interesting way to learn about them.

- **Project Objectives:** Provide the primary questions you are trying to answer with your project. What would a user learn and accomplish? Identify about 2/3 objectives and then address how these objectives are being met in the Features sections below.

In my web app users will input the name of an artist. If that artist does not exist in the spotify database, it will let them know and they will have the option to enter another artist. If that artist does exist, the program will ask them for their guess on what that artist's top song is. If that guess is correct, a function will be carried out. That function will display that top song's track information, the album it came from, and the artist's top ten songs. If the guess isn't correct they will be informed and asked if they'd like to guess again. If they don't want to guess the function that displays information will be carried out. If they do want to continue, they can continue to guess until they decide they no longer want to. The app will have a restart feature after each cycle has ended.

\*Additional **possible** feature: Playing the artist's top song along with the function that displays information. (Not guaranteed)

Objectives summarized:

- User is trying to accomplish guessing their artist's top song
- User will learn what the top song is
- User will learn what album the top song is from
- User will learn the top ten songs of the inputted artist
- User will ultimately test their knowledge on the artist of their choice

- **API:** If you are using an API, list the name of the API, add a link to the homepage of the API, and provide a short description about the API.

<https://developer.spotify.com/documentation/web-api>

<https://github.com/spotipy-dev/spotipy>

- **Features:** List the features without which you would consider your project to be a failure.
  - If any information regarding the top ten songs, track information, and album the top track is from is displayed incorrectly or fails to display to the user
  - If an artist the user enters is in the spotify database yet the program says it isn't
  - If the restart feature didn't work
- **Project Schedule:** Make sure that you plan your work so that you can avoid a big rush right before the final project deadline. Write this in terms of weekly deadlines and add in the Alpha Status Update, Beta Release, and Final submission dates. I am expecting the schedule to be fairly detailed.

Week 1 Nov 26th - Dec 3rd:

By Dec 1st:

- Have Alpha Status update complete
- Successfully request an artists top ten tracks, a songs track information, and an albums information

By Dec 3rd:

- Start creating/refining user input and conditional statements

Week 2 Dec 4th-8th:

- Have the structure of every scenario based on user's actions

By Dec 5th Tuesday:

- Have Beta Status completed and turned in

Dec 5th-8th:

- Stylize the web app
- Work hard towards completing it by debugging, getting help, and testing

By Dec 8th:

- Have project thoroughly tested, debugged, and finished (hopefully)
- Introduction

# Final Project Report

- **Introduction:**

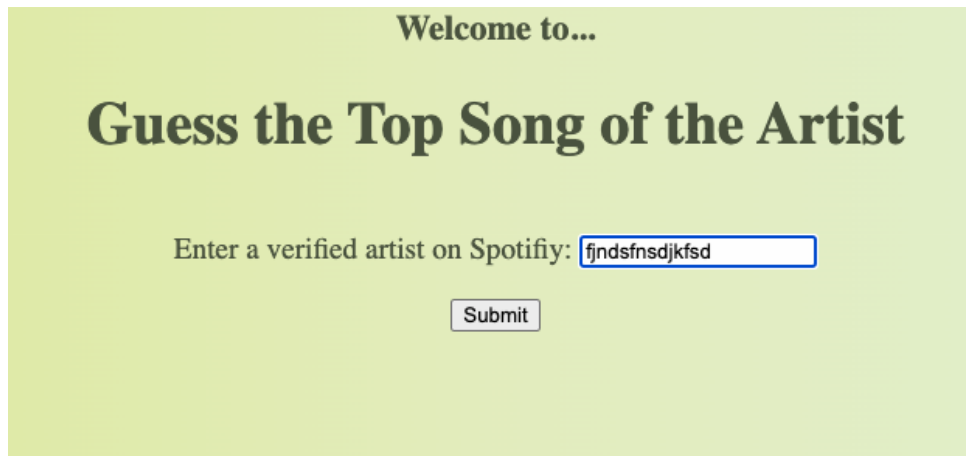
While brainstorming possible ideas for my final project. One particular topic kept popping up in my mind, music. I knew I wanted to incorporate fun, creativity, and an interest/passion of mine into my final project. That is when the idea for a game related to testing yourself on your favorite artist's music came into play. I began to structure the features of my project and was able to incorporate all the original components that I'd hoped to have. My web app is a game where users can input any spotify artist of their choice and try to guess what their top song is. User's are able to guess as many times as they want, restart completely, or reveal the answer. If the user guesses correctly, they are rewarded with information about the top track, the top ten songs of the artist, and the album the top track comes from! My web app tests your knowledge of your beloved music artists and measures how well you think you know them.

- **Approach**

## **The features I implemented:**

- 1) Input validation to test if the artist the user inputs is spotify verified or not

Scenario 1: (Doesn't exist)



Welcome to...

## Guess the Top Song of the Artist

Enter a verified artist on Spotify:

**The artist you searched for is not in the spotify database.**

[Back to Play Again!](#)

Scenario 2: (Does exist)

Welcome to...

## Guess the Top Song of the Artist

Enter a verified artist on Spotify:

## Guess the most popular song of the artist, SZA

Enter guess:

If guess is not working, try including **artists that might be featured** on the top track or any **additional information** about the track in parenthesis or brackets included in the song title!

[Back](#)

2) Input validation to check if a user's guess of an artist's top song is incorrect

Scenario 1: Incorrect

## Your guess was incorrect...

[Back to Play Again/Restart!](#)  
[Guess Again](#)  
[Reveal the answer... :\(](#)

Scenario 2: Correct

**The top song of SZA is: Kill Bill**

**You Guessed Correctly!**

**Top Ten Track List:**

1. Kill Bill
2. Snooze
3. Rich Baby Daddy (feat. Sexyy Red & SZA)
4. TELEKINESIS (feat. SZA & Future)
5. Slime You Out (feat. SZA)
6. All The Stars (with SZA)
7. Good Days
8. Kiss Me More (feat. SZA)
9. Open Arms (feat. Travis Scott)
10. Low

**Information about SZAs top track: Kill Bill**

Track url: [Play\\_Top\\_Track](#)  
Track ID: 1Qrg8KqiBpW07V7PNxwwL  
Track Popularity: 92  
Track Number on Album: 2  
Track Type: track

**Information about album that the top track, Kill Bill, comes from:**

Album Name: SOS  
Album Number of Tracks: 23  
Album ID: 1nrVofqDRs7cpWXJ49qTnP  
Album Release Date: 2022-12-08

[Back to Play Again!](#)

3) Multiple choices when user's guess is incorrect

Route 1 (Takes you back to homepage)

**Your guess was incorrect...**

[Back to Play Again/Restart!](#)

[Guess Again](#)

[Reveal the answer... :\(](#)

Welcome to...

**Guess the Top Song of the Artist**

Enter a verified artist on Spotify:

Route 2: (Lets you guess again, takes you back a page)

**Your guess was incorrect...**

[Back to Play Again/Restart!](#)

[Guess Again](#)

[Reveal the answer... :\(](#)

**Guess the most popular song of the artist, SZA**

Enter guess:

If guess is not working, try including **artists that might be featured** on the top track or any **additional information** about the track in parenthesis or brackets included in the song title!

[Back](#)

Route 3: (Reveals the answer along with information)

**Your guess was incorrect...**

[Back to Play Again/Restart!](#)

[Guess Again](#)

[Reveal the answer... :\(](#)

## The top song of SZA is: Kill Bill

Were you even close?...

1. Kill Bill
2. Snooze
3. Rich Baby Daddy (feat. Sexyy Red & SZA)
4. TELEKINESIS (feat. SZA & Future)
5. Slime You Out (feat. SZA)
6. All The Stars (with SZA)
7. Good Days
8. Kiss Me More (feat. SZA)
9. Open Arms (feat. Travis Scott)
10. Low

### Information about SZAs top track: Kill Bill

Track urls: [Play Top Track](#)

Track ID: 1Qrg8KqiBpW07V7PNxwwwL

Track Popularity: 92

Track Number on Album: 2

Track Type: track

### Information about the album that the top track, Kill Bill, comes from:

Album Name: SOS

Album Number of Tracks: 23

Album ID: 1nrVofqDRs7cpWXJ49qTnP

Album Release Date: 2022-12-08

[Back to Play Again!](#)

4) Displaying information about top track, top ten songs, and album of top track

Scenario 1: When guess is correct

## The top song of SZA is: Kill Bill

You Guessed Correctly!

### Top Ten Track List:

1. Kill Bill
2. Snooze
3. Rich Baby Daddy (feat. Sexyy Red & SZA)
4. TELEKINESIS (feat. SZA & Future)
5. Slime You Out (feat. SZA)
6. All The Stars (with SZA)
7. Good Days
8. Kiss Me More (feat. SZA)
9. Open Arms (feat. Travis Scott)
10. Low

### Information about SZAs top track: Kill Bill

Track urls: [Play Top Track](#)

Track ID: 1Qrg8KqiBpW07V7PNxwwwL

Track Popularity: 92

Track Number on Album: 2

Track Type: track

### Information about album that the top track, Kill Bill, comes from:

Album Name: SOS

Album Number of Tracks: 23

Album ID: 1nrVofqDRs7cpWXJ49qTnP

Album Release Date: 2022-12-08

[Back to Play Again!](#)

Scenario 2: When guess is incorrect and user chooses to reveal answer

## The top song of SZA is: Kill Bill

Were you even close?...

1. Kill Bill
2. Snooze
3. Rich Baby Daddy (feat. Sexyy Red & SZA)
4. TELEKINESIS (feat. SZA & Future)
5. Slime You Out (feat. SZA)
6. All The Stars (with SZA)
7. Good Days
8. Kiss Me More (feat. SZA)
9. Open Arms (feat. Travis Scott)
10. Low

### Information about SZAs top track: Kill Bill

Track url: [Play Top Track](#)

Track ID: 1Qrg8KqiBpW07V7PNxwwwL

Track Popularity: 92

Track Number on Album: 2

Track Type: track

### Information about the album that the top track, Kill Bill, comes from:

Album Name: SOS

Album Number of Tracks: 23

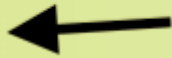
Album ID: 1nrVofqDRs7cpWXJ49qTnP

Album Release Date: 2022-12-08

[Back to Play Again!](#)

- 5) Restart feature **present on all pages** (EX: [“Back”](#), [“Back to play again!”](#), [“Back to Play Again/Restart!”](#))

[Back to Play Again!](#)



[Back](#)

[Back to Play Again/Restart!](#)

### - Roadblocks faced

Some roadblocks I faced was redirecting my users to different pages based on user input. I worked with Kelsey in class and we could not find a full on solution right then and there. However, we did find a roadway with the url for function. Kelsey ended up finding a way to redirect users to pages based on their input. However I wanted to see if I could do it with the url for function and still keep my code with minimal changes. At first nothing was working and I wasn't sure what's wrong. It took some research online to figure out that having a secret token as well as using sessions to transfer information to my other app routes/functions would help my functionality work the way I wanted. I started making headway when I would first get redirected to a page that wasn't the typical error page. I kept testing things out until I eventually got the redirect to work perfectly. It displayed all the information I wanted based on whether a user's guess was correct or not. If the guess wasn't , I made sure to add in the options I wanted like restarting from the beginning, guessing again, or revealing the answer. I did this by adding href links with the url for the method once again. My other major roadblock ended up being using css to style my website. For some reason I wasn't able to use style sheets. Kelsey and I tried to do it



in class subsequent to me trying on my own, and it wasn't appearing with my glitch file. Due to this I was reduced to using style tags. However, I still styled my website as I'd hoped, so it ended up being a success.

- **Instructions to run your program**

- 1) **Input a Spotify verified artist you hope to test your knowledge on**
- 2) **Enter your guess on what their top song is.** If your guess didn't work. consider featured artists that might be on the top track or any extra information that would be included about the track usually in parenthesis or brackets. Spotify often includes featured artists and additional information as a part of the title. Ex) Enter "ft \_\_\_\_" if there is a featured artist.
- 3) **Choose as you please!** Choose to play again entirely if your guess is correct. Or guess again, restart from the beginning, or reveal the correct answer if your guess is incorrect. You can play as many times as you'd like, make a game with friends, and however you decide!

- **Python packages used**

- from spotipy.oauth2 import SpotifyClientCredentials
- from fuzzywuzzy import fuzz
- import spotipy
- import sys
- from flask import Flask, session, request, redirect, url\_for
- import secrets

- **How can the program be executed?**

The program can be executed online by using this link. <https://ring-hyper-field.glitch.me/>

- **Conclusion**

I have had my ups and downs, however I am very proud of how my project turned out. I am especially happy that I was able to have all the original components I initially envisioned. Working to make all those specific components come to life was both invigorating and challenging. While working on my project I was tested on my time management, grit, and problem solving abilities. Overall, after completing it I can say I feel I have improved a bit in each of those factors and have learned a lot.

