

Project Research Document
Music Critique Website
X00164430 Jakub Wysocki

The research document should provide detailed information on the project, its scope and aims, prospective users, and possible technologies that will be used for implementation.

You should carefully evaluate APIs, data sources, key libraries or other resources you need that are critical to the project success.

More specifically the research document should provide information on the following Topics:

Section 1: Detailed Discussion x 1 Page

Take the idea you have selected and develop it providing a detailed discussion on the system functionality and what you propose it should do. Who are the users?

Section 2: Existing Applications in this domain x 1/2 Page

In your research have you found anything close to this idea? List these in table form and identify similarities and differences

Section 3: Platform, Technologies and Libraries x 1/2 Page

What platform, technologies and libraries are you planning to use?

Section 4: The risks x 1 Page

What are the main risks to the project? (i.e., are you depending on 1 library to provide key functionality?)

Section 1:

My project idea is a Music critique website. On this website users will be able to view, review and comment on the newest or trending songs. The general idea is to pull down songs from Spotify using their API and display them on the website. The website should be intuitive, minimalistic and simple as the target audience is anyone who is interested in music whether, they're 10 years old or 100

For this project the focus is on the top trending songs. Top 50 songs from different regions will be used and put into separate categories according to the region.

There will be a register and login functionality. Registered users will be able to go into each song and leave a comment reviewing it or give it a score out of 10 which will then be displayed with the song for other people to see. They will also be able to use a chat box where they can communicate with other users in real time. A user who is the website admin may be able to delete or modify comments or messages in the chat box filtering inappropriate messages.

In addition to this user will be able to generate a report based on their actual Spotify profile. This will be achieved by users authorizing log in through Spotify and pulling their Spotify profile information using the API once again. Then they could choose whether or not to display it on their own profile on the website.

There will be a night mode button that users can access. I'm hoping to implement a cookie system which remembers the setting the user chose before leaving the website.

I would also like to implement like a "facts" tab on the website in which statistics about songs, artists and albums will be shown. For example, how many times a song/album/artist have been played or just random fun facts about music in general.

Another function that I'm planning to add is for the users to be able to change the language of menus into the top 3 commonly used – English/Spanish/French

Section 2:

Website	Similarities	Differences
https://www.metacritic.com/music	<ul style="list-style-type: none">• Shows new album releases• Has filters to view music by score or popularity	<ul style="list-style-type: none">• Music isn't the main focus. It also focuses on games, movies, tv• Only critics can leave a score and opinion• No chat boxes• Can't make summary of Spotify account
https://rateyourmusic.com/	<ul style="list-style-type: none">• Shows new album releases, charts• Rate songs• User profiles	<ul style="list-style-type: none">• No chat box• Has forums• Can't make summary of Spotify account
https://www.allmusic.com/?1665340539935	<ul style="list-style-type: none">• New releases, charts• User profiles• User reviews• User comments	<ul style="list-style-type: none">• Includes articles related to music.• Can't make summary of Spotify account

Section 3:

I'm planning to use visual studio code for my project. The framework of my choice is python using the technology of Django. The website will be created with the help of CSS, bootstrap, html and JavaScript. I'm planning to host the website on AZURE services. Everything related to the project will be maintained on GitHub

Section 4:

The following are some risks associated with the project:

- As time goes on, it constantly needs to be updated to each top 50 songs in each region which will be time consuming doing it manually
- Having too many users may slow down the run time of the chat box
- Implementing cookies may cause risks as a website cannot distinguish whether action is initiated by the user or not. If the cookie is present, it searches for it and then purposefully carries out the action as though the user had initiated it
- May have conflicts in the project if not resolved in a timely manner
- Lack of communication with supervisor may lead to lack of clarity and confusion
- Scheduling and/or estimation mistakes
- At any time, Spotify may change their API framework removing the functionality of the website.