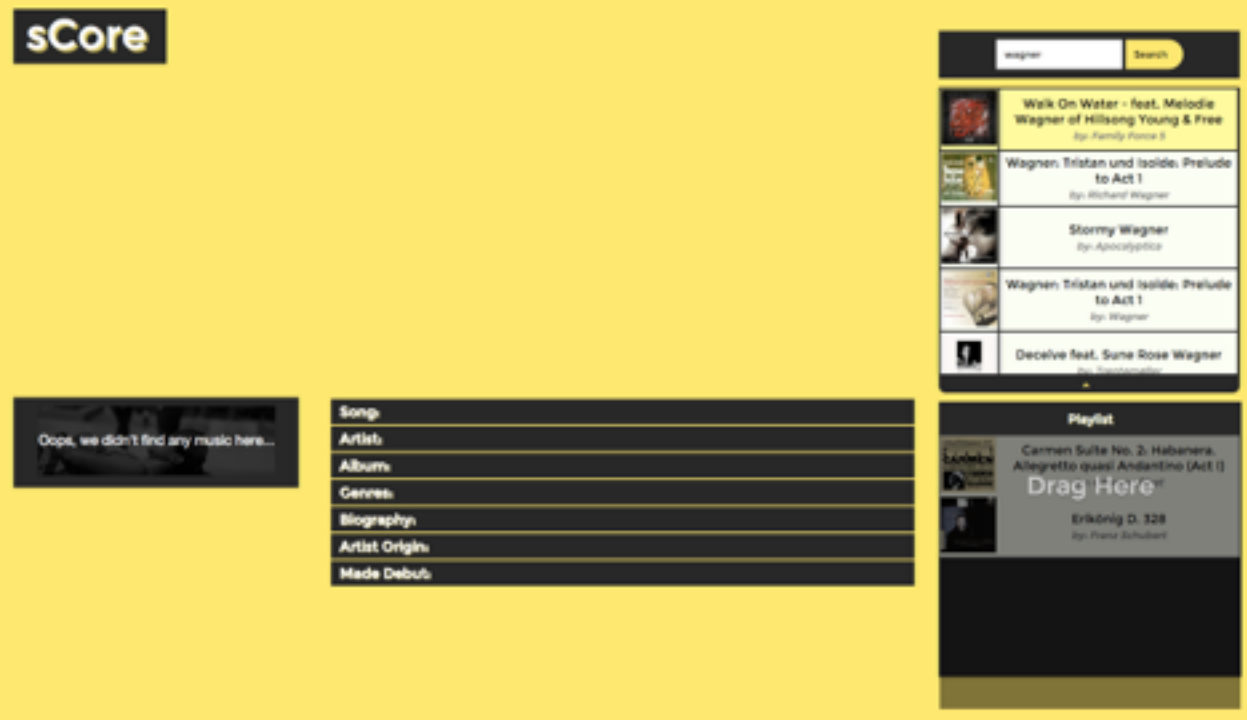
Kenneth Hungria kah2204
Justin Walters jw3043
Mikhail Klimentov idk2146
Team: Octothorpe

User Manuel

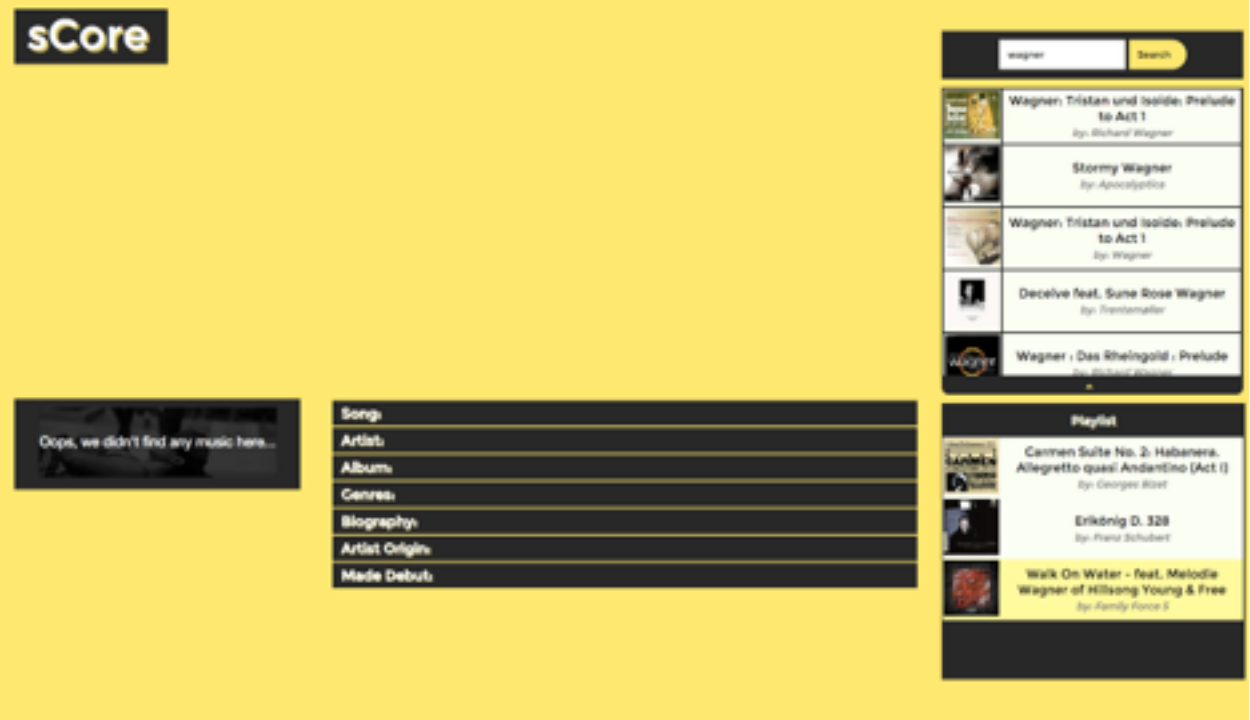
In order to run the web application, there are several things that must happen beforehand. In our testing and running of the application, we ran the application through a localhost server. If possible, open up a server on any port and run our index.html page. For example, Kenneth was able to run a localhost using python by going to his terminal, going to the directory in which the index page is held and typing “python -m SimpleHTTPServer 8000”. This would run a server on port 8000 that would allow the user to use index.html. We also recommend that the user have the spotify application on their computer as our widget seems to throw the person to another page in a new tab to play the song.

Since the user has some familiarity with the website, below are some of the examples of actions that can be done. These include things such as searching for a song, dragging a searched song to a playlist, saving the playlist and playing a selected song.

One of the features is the drag and drop. We added this feature to allow the user to be able to search for songs and drag them to a playlist to save for later. Every time the playlist is changed, then it is saved in a local Storage where if the page were to be refreshed, they would still be there. Meaning the only way to get rid of it is to drag it out of the box towards the middle.



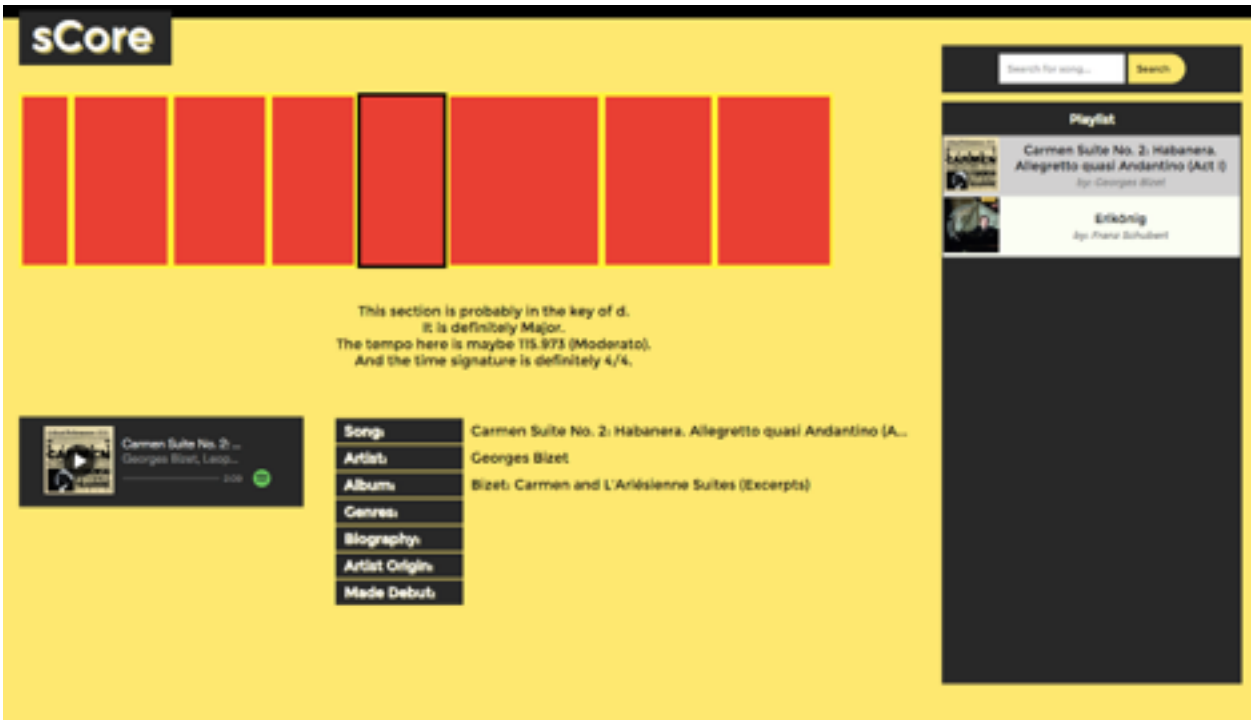
This shows up when the user searches for a song and wants to drag a song to the playlist. The cursor isn't showing up if you were to click and drag the first result, this would show up, where the drag here would tell the user they can drag the song to the playlist.



Here the user can drag the song on the playlist to the middle of the page in order to remove it, as a result, this would cause the playlist to update and the local storage as well. If the user were to refresh the page right now, their information would still be saved.

The other feature is the playing of a song. Either the playlist or the results can play the song that has been found. This way the user can try a song to see if its the right rendition before they add it to their playlist. There is a precaution however. The Echo Nest API only allows us to have 20 calls per minute and each play of the song counts as 3 calls. So there is a disadvantage that we ran into that a user can't go through more than 6 songs per minute. Below shows the result of playing a song.

This last picture shows how the information for each bar is portrayed. The user can click on each bar and a small bit of information will pop out. Now we see that the information is viewed per bar and the section is seen below.



Ideally we wanted multiple bars, to toggle between each information, but time constraint kept us from doing that. In the end the idea is that the user can look for songs, add and remove from the playlist, see general information about the song and their artist and see specific musical information on each sections.