Project Specification for Song Sharing web app. (Team 1)

Nathan Koch, Tony Padilla, Josh Broitman, Krish Vaswani

Product Owner for Sprint 1:

Product Backlog and Sprint Back Log

<https://docs.google.com/spreadsheets/d/1gTtBHbgwUq2K3MSA4mupKDdtNwQ0Vvrnh6d-pURSf6I/edit#gid=0>

Wireframes

<https://wireframepro.mockflow.com/view/M4c5608582151b5933fbc487bafa2d4871584903132740#/page/Dff4810bbb8fd5a3df027c2e901d1d6b3>

<https://wireframepro.mockflow.com/view/Md17b97b2b180f339acb12a4521ed42871584905912764>

Models/Forms:

SongShare

Stream:

**Proposal Review and Overall**

SongShare is a platform that makes listening to music easier when in a group. It will allow one person to be the “DJ”, and others can send them requests of songs on Spotify. The DJ can then choose to play the songs according to the preferences of the audience. This can be used at small get-togethers, where there are multiple people who want to play music. It can also be used at parties or radio stations, where there is one specific DJ playing music for multiple people.

SongShare allows for DJs to control their ‘Streams’ which act similar to Spotify’s playback queue. Streams can be monitored and managed by the DJ. They can adjust the stream moving songs up and down as well as removing songs from the stream altogether. They can also add songs to their own queue There are currently two different ways songs can be added to the Stream: Songs can be added by listeners at any point in time, or a voting system that democratically elects the next song. Users can pick and choose which songs they would like via SongShare using the Spotify api linked to SpotiPy. Users will be able to follow DJs, which will add a list of DJs that they are following to their profile page, giving easy access to their favorite DJs.

Django will be used for the backend, with REACT for the frontend as it will allow for the use of Semantics UI. This will make the website more beautiful and easier to read/follow. On the homepage, the user will see their favorite streams that are live, the most popular live streamers and the most recently viewed streams by the user. Each user will be given the option to become a DJ and link their account to Spotify in order to run their own stream.

Users will be able to search for their favorite DJs and select which DJ they are currently listening to. After selecting their DJ they can add songs to the stream by searching for their favorite songs. We will use some quick ways to determine possible suggestions to the song if the user enters an incorrect song title.

**This Sprint**

This Sprint we will be focusing on setting up the foundations for the website. This will include: basic frontend, access to spotify through Spotipy, and adding songs to the playlist through the DJ’s account. Our goal is to be able to setup an interaction between Song Share and Spotify’s playback queue while on our local machine. This will include making our own “Stream” on SongShare to feed into the playback queue on Spotify. This may require creating a playlist directly through our app, this will need to be done through spotipy.