CS 408 Team 18

Project Name: Pitch

Project Owners: Ryan DeSalvio, Keith Droll, Nicolas Bratton, Garrett Kizior

**Product Backlog**

**Problem Statement:**

There are currently many services available for sharing and finding new music. Since the eve of the music sharing industry people have been creating playlists. Mixtapes were the beginning of this idea of having a collection of music in one space. Today sophisticated playlists exist on services like iTunes and Spotify. We found there to be a need for a new way to simply and easily create music playlists, but with your friends. Our goal is to create a social network solution for collaboratively creating a playlist with your friends.

**Background Information:**

Music is undergoing a change in the way that people listen. Streaming services are becoming ever more popular with the advent of Spotify, Apple Music, Pandora, and Tidal. With music becoming more and more accessible, the industry has become even more influential and popular than ever before. This popularity birthed the idea of Pitch. With music being more accessible to listen to, how do you know what to listen to? Streaming services are focused on delivering tailor made experiences to individual users. The services do a very good job of this but do not do a good job showing what your friends are listening to or making music a social event. This is where Pitch comes in. Pitch is a social media network designed around the idea of sharing music that you’re listening to and allowing friends to come together and make playlists to share. Pitch allows the sharing of music by creating a feed of suggested songs and playlists your friends share. This allows you to go to your favorite music streaming service and listen to the songs and playlist your friends shared. Then, when you find new songs that you like, you can come together with a group of friends and collaboratively create a playlist to share on Pitch. The playlist creation is done by putting a song up for vote within a group on Pitch. The rest of the group votes on the suggested song and if the majority of the group approve the song gets added to the collaborated playlist. Pitch will also suggest songs to add to a groups playlist based on previous song choices in order to help add to the amazing playlists our users will create.

**Environment:**

Pitch will be designed using HTML5, Bootstrap, Java (or C#), and Angular JS. We will also take advantage of the Spotify API to fetch related artists or related songs to suggest songs to users. We will use MySQL for storing user information and posts on the website. The website will be hosted on a local server.

**Requirements:**

***Functional:***

|  |  |  |
| --- | --- | --- |
| Backlog Id | Functional Requirement | Hours |
| 1 | As a user, I would like to create a profile | 5 |
| 2 | As a user, I would like to send friend requests | 4 |
| 3 | As a user, I would like to create friend groups | 6 |
| 4 | As a user, I would like to view a feed on profile | 6 |
| 5 | As a user, I would like to post a suggested song on feed | 5 |
| 6 | As a user, I would like to post a suggested playlist on feed | 5 |
| 7 | As a user, I would like to put a song up for vote on a group playlist | 6 |
| 8 | As a user, I would like to vote on a song that is up for vote on a group playlist | 3 |
| 9 | As a user, I would like to see all new activity on my profile in a notification page | 8 |
| 10 | As a user, I would like to be able to vote on Pitch generated song suggestions | 4 |
| 11 | As a user, I would like to be able to upload a profile picture | 2 |
| 12 | As a user, I would like to integrate my Spotify account picture within the website | 5 |
| 13 | As a user, I would like to save playlists to Spotify account | 6 |
| 14 | As a user, I would like to have a tutorial to learn how to navigate Pitch | 3 |
| 15 | As a developer, I would like for usernames and passwords to be stored on a database | 3 |
| 16 | As a developer, I would like for user posts to be stored on a database | 3 |
| 17 | As a developer, I would like for friend groups to be stored on a database | 3 |
| 18 | As a developer, I would like for playlists to be stored on a database | 3 |
| 19 | As a developer, I would like to implement a real-time voting system | 3 |
| 20 | As a developer, I would like to for Pitch to generate recommended new songs for playlists based on songs already in the playlists | 7 |
| 21 | As a developer, I would like to implement a real-time feed | 6 |
| 22 | As a developer, I would like to implement a notification system | 10 |
|  | Total: | 106 |

***Non-functional:***

* As a developer, I would like for all pages to have a meaningful URL
* As a user, I would like to have fast response times with user input
* As a user, I would like my profile to be password encrypted
* Pitch will be able to handle at least 100 users
* Website should be easily navigable
* Interface needs to be simple, intuitive and responsive
* We must have fast page loading and page refreshing, thus minimizing computer resources
* User interface must be attractive and simple

**Use Cases:**

*Case 1: Create a profile / Tutorial*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose ‘Create Account’ on landing page | 1. System loads ‘Create Account’ page |
| 1. Insert profile information and submit | 1. Profile information is stored on DB and ‘Tutorial’ page loads |
| 1. Click through Tutorial pictures | 1. Redirect to ‘Home” page |

*Case 2: Send/Receive friend requests*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose “Add friends” | 1. System loads “Add friends” page |
| 1. Type in friend’s name and submit | 1. Load friend’s user account |
| 1. Click on Add friend | 1. Friend request sent to friend and DB updated and return to ‘Home’ page |
|  |  |
| 1. Choose “Notifications” | 1. System loads “Notifications” page |
| 1. Choose whether to accept or deny incoming friend requests | 1. Response updates DB and a friend connection is either created or not and return to ‘Home’ page |

*Case 3: Create/Accept or Deny Friend groups*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose ‘Create Friend Group’ on landing page | 1. System loads ‘Create Friend Group’ page |
| 1. Insert names of all friends that should be in group | 1. Each friend receives a Friend group request on their ‘Notification Page’ and DB updates. Return to ‘Home” page |
|  |  |
| 1. Choose “Notifications” | 1. System loads “Notifications” page |
| 1. Choose whether to accept or deny incoming friend group requests | 1. Response updates DB and a friend group connection is either created or not and return to ‘Home’ page |

*Case 4: Suggest/ vote on Songs in Friend group*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose “My Friend Groups” | 1. System loads ‘My Friend Groups’ page |
| 1. Choose wanted friend group | 1. System loads specific friend group page |
| 1. Type in song to be added to playlist | 1. System sends suggested song to be voted on in each friends notifications and DB is updated. |
|  |  |
| 1. Choose “Notifications” | 1. System loads “Notifications” page |
| 1. Choose whether to accept or deny incoming song suggestions for friend group playlist | 1. Response updates DB and song is either added or not to friend group playlist. Friend group playlist page is loaded |

*Case 5: Show Feed*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Go to landing page | 1. System loads ‘Home’ page |
|  | 1. Feed information is received from DB and in order of post datetime |
| 1. View feed |  |

*Case 6: Add items to Feed*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Go to landing page | 1. System loads ‘Home’ page |
| 1. Select ‘add feed item’ | 1. Open ‘add feed item’ module |
| 1. Type a song or select playlist to share | 1. Updates DB and return to ‘Home’ page with new feed item visible to user and all user’s friends |

*Case 7: Suggested songs for playlist / voting on suggest songs for playlist*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose “My Friend Groups” | 1. System loads ‘My Friend Groups’ page |
| 1. Choose wanted friend group | 1. System loads specific friend group page |
| 1. On bottom of page vote for each suggested song you want | 1. A vote sends a song suggestion as same as Case 4 |

*Case 8: Update Profile Picture*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Click on profile picture | 1. System loads ‘Update Profile Picture’ page |
| 1. Upload new picture or select ‘Use my Spotify picture’ | 1. Picture is saved on DB and return to ‘Home’ page |

*Case 9: Export Playlist to Spotify*

|  |  |
| --- | --- |
| **Action** | **System Response** |
| 1. Choose “My Friend Groups” | 1. System loads ‘My Friend Groups’ page |
| 1. Choose wanted friend group | 1. System loads specific friend group page |
| 1. Select “Export Playlist to Spotify” | 1. System sends playlist to users Spotify account using Spotify API |