Spotify PULSE

Personalized User-Learned Social Experience
Sprint 3 Planning

Team 18

Rheaa Sharma

Noah Stern

Dane Shaffer

Alex Polivka

Jason Zheng

Collin "Bodhi" Scott

Sprint Overview:

In this third sprint, we will focus mainly on ensuring full functionality of our server to any users on the internet. This includes proper error handling, better user interfaces and experience, data encryption, faster load times, more intuitive overall website design, and lastly a proper server that is up and running. We will also be implementing new features that are unique to our website and help us stand out among competitors. This includes a matchmaking tab where users can match with songs that are in line with their preferences, and update their preferences based on which songs they 'swipe left' and 'swipe right' on. Furthermore users will also be able to match with other users who share a similar interest in songs. Another thing that makes our website stand out is a feature that will allow users to merge and fuse playlists together. Merging will allow users to easily concatenate playlists, while fuse will get recommendations from two playlists, populate a new playlist with those songs, and add in some of their old songs into that playlist. Lastly, users will be able to easily view detailed information about artists, staying in line with our goal of being an 'all-in-one' stop for Spotify.

Scrum Master: Alex Polivka

Meeting Plan:
- Mondays 6:00 PM
- Thursday 8:30 PM with Project Coordinator

Risks and Challenges:

The biggest challenge by far this sprint is going to be setting up the server and all of the necessary work that comes with it. Refactoring the backend to reflect our new server capability is going to be a tough and repetitive process, where one mistake could cause a massive server issue. Additionally, refactoring the database calls to be inline with the server is going to pose the same issue.

Another big challenge is going to be keeping our Github clean, *especially* because of the creation and implementation of our server. Last sprint, our Github became very messy near the end, and we hope to do better this sprint. Our plan is to have branches for each feature (usually encapsulating 1 or more user stories) that are kept up to date with main. This should mitigate the challenge as much as possible.

Additionally, starting this sprint, our team plans to take feedback from outside users (mainly friends and family) on the state of Spotify PULSE. Taking such feedback in stride, and deciding what feedback is most important or relevant is going to be a new experience for many members of the team and may prove to be challenging.

Because this is our last sprint and we have final presentations coming up, our last challenge will simply be making sure everything from all sprints are up to par with not only our standards, but the Professors standards too. All old bugs and incomplete user stories that our team has been putting off fixing must be fixed this sprint.

Current Sprint Details:

User Story #1h:

Authentication and bad routing errors are handled gracefully and direct the user to an error page

#	Description	Time	Owner
1	Refactor entire backend to be able to handle errors and be consistent with routing	5h	Noah
2	Create exception handling class with custom messages and behaviors	2h	Noah
3	Implement error handling everywhere on the backend to account for all possible errors	3h	Noah
4	Create error page and redirect user to it if there was an error	1h	Rheaa
5	Add retries for authentication token before throwing error	2h	Noah
6	Manual testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is not registered with PULSE, then when the user encounters an authentication error they are redirected to the error page.
2. Given the user is registered with PULSE, then when the user encounters a routing error due to improper access or the page not existing, they are redirected to the error page.
3. Given the user is on the error page, they are given a custom code and information about their error.
4. Given the user is on the error page, they can click a redirect link to go back to the login page (which will redirect to the dashboard if they are already logged in).
5. Given the user is registered with PULSE, their token expires, and the token is able to be refreshed within the proper amount of retries due to there being no other errors, then their token is properly refreshed without further action from the user.

User Story #1i:

As a user I would like to be able to actually visit the public website, and I would like the load times to be as quick as possible.

#	Description	Time	Owner
1	Research and implement server and set up everything short of updating previous code to be inline with server framework	8h	Bodhi
2	Build framework to ensure that server size can not grow exponentially due to any input, at great monetary costs	3h	Bodhi
3	Test every facet of the framework to ensure that server size can not grow exponentially at great monetary costs.	2h	Bodhi
4	Refactor database for use in tandem with the server. Make current database calls more modular, decreasing black box effect with regards to the backend and server.	4h	Bodhi
5	Refactor backend routes to be inline with server and updated database framework	4h	Noah
6	Save previous chatbot information about current discussion on the server so that chatbot memory is extended and conversations are more fluid	1h	Rheaa

Acceptance Criteria:
1. Given the user has no technical experience, when they wish to access the Spotify PULSE services for any reason, they are able to navigate to a URL and get started with PULSE because it is hosted on a server.
2. Given the user access the PULSE webpage, when they are logging in and navigating between pages, they are able to do so and each web page or site service loads in a timely manner
3. Given that a user has free access to all aspects of the website, when they do any action that pertains to a server's function (i.e. uploading data, uploading themes etc…), the database will not grow exponentially in size regardless of if the user's input should cause it to.
4. Given that a user wants to access a specific page of the website, when they go to do so, they are able to do so by typing in the specific route's url instead of navigating there through the UI.

User Story #1j:

As a user, I would like my data to be secure in all ways, at all times.

#	Description	Time	Owner
1	Research possible encryption methods and decide which to use for database/server	1h	Bodhi
2	Ensure Spotify information, analytics, and personal information in the database is securely encrypted	1.5h	Bodhi
3	Ensure Spotify information, analytics, and personal information on the server is securely encrypted	1.5h	Bodhi
4	Use privilege controls and sanitation to ensure the database is not susceptible to sql injections, refactoring where necessary.	3h	Bodhi
5	Protect against other common exploits such as cross-site-scripting and buffer overflow	1.5h	Bodhi

Acceptance Criteria:
1. Given the user has data in the database, when a second malicious user attempts an sql injection, then the injection is stopped and the first user's data is safe.
2. **(new)** Given that the user logs in, when their data is stored in the database, then no sensitive information is stored unencrypted and therefore accessible to an attacker.
3. **(new)** Given that the user logs in, when their sensitive information (i.e. Spotify token) is stored in the database, then it is encrypted with a key that is not stored in environmental variables and therefore not accessible to an attacker.
4. **(new)** Given that a user is using Spotify PULSE services, when an attacker attempts to obtain database access from credentials hosted in the GitHub Repo, then they are unable to do so because such credentials are hosted with heroku's environment variables.
5. **(new)** Given that a user is using Spotify PULSE services, when an attacker attempts to use an exploit such as cross-site scripting, then the first user's experience is safe as the attacker is not able to perform such exploits due to strict https cookie handling

User Story #1k:

As a user, I would like for songs, albums, and artists, shows, and episodes to be displayed consistently across the website.

#	Description	Time	Owner
1	Create UI for individual song, album, artist, show, and episode displays	2h	Alex
2	Create UI for lists	1h	Alex
3	Implement display onto previously existing pages	2h	Alex
4	Create backend for retrieving songs, albums, and artists needed for display	1h	Dane
6	Manual Testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is viewing a list of songs, when the user clicks the song, the song will play on the user's Spotify.
2. Given a UI list of songs containing songs, when the user looks at the list, then the album art, song name, and artist name are all displayed.
3. Given a UI list of albums containing albums, when the user looks at the list, then the album art, album name, and artist name are all displayed.
4. Given a UI list of episodes containing episodes, when the user looks at the list, then the art, episode name, and show name are all displayed.
5. Given the UI list of shows containing shows, when the user looks at the list, then the art and show name are both displayed.

User Story #2h:
As a user, I would like to be able to customize my public profile.

#	Description	Time	Owner
1	Add UI on profile page to edit profile page	1h	Alex
2	Create modular profile display component	1h	Alex
3	Add new information fields in database	0.5h	Bodhi
4	Create backend to store and retrieve new information fields	1h	Dane
5	Manual Testing	1h	Dane

Acceptance Criteria:
1. Given the user is logged into PULSE, when the user is on the profile page, they can preview what their public friend's card will look like.

2. Given the user is on the profile page, when the user wants to pick their public profile colors, then they can do so on the profile page.
3. Given the user is logged into PULSE, when the user views a public profile, the card matches the other user's public profile color setting.
4. Given the user is on the profile page, when the user wants to pick their description, then they can do so on the profile page.
5. Given the user is logged into PULSE, when the user views a public profile, the card matches the other user's public description setting.

User Story #2i:

As a user, I would like backgrounds to be saved along with themes.

#	Description	Time	Owner
1	Create UI to combine background and themes	2h	Alex
2	Edit database structure to store backgrounds with themes	1h	Bodhi
3	Edit backend functions to store backgrounds with themes	1h	Dane
4	Edit frontend to retrieve backgrounds and themes together	1h	Alex
5	Manual testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is on the settings page, when the user retrieves a preset theme, then a background will be loaded with it.
2. Given the user is on the settings page, when the user saves a new theme, then the background will be stored with it.
3. Given the user has a saved theme, when the user retrieves that saved theme, then the stored background will be loaded with it.
4. Given the user has a saved theme, when the user wants to overwrite an old theme, they are given an intuitive method of doing so.
5. Given the user has a saved theme, when the user wants to overwrite that saved theme, then their selected background will be saved into that theme.

User Story #3k:

As a user, I would like to be able to specify what time period I want my stats to be from.

#	Description	Time	Owner

1	Allow advanced stats to be displayed in a time range	2h	Jason
2	Modify popup for above	1h	Jason
3	Allow followers to be displayed in a time range	1h	Jason
4	Ensure advanced statistics are stored correctly	1h	Bodhi

(Testing merged w/ 3L since it's tied to 3L)

Acceptance Criteria:
1. Given the user selects data as a advanced stat (# minutes, % times played, etc.), when the user generates a graph, then the graph displays data between a certain time range.
2. Given the user selects data as followers, when the user generates graph, then the graph displays data between a certain time
3. Given the user is on the "add graph" menu, when user selects any data type (that have time-dated data), the user can select time ranges
4. Given the user selected an invalid time range of data, when the user generates a graph, the graph handles the data gracefully & alerts the user to the fact that their time range results in invalid data.
5. Given the user does not specify a time range, when the user generates a graph, then the graph displays all the data.

User Story #3I:

As a user, I would like to be able to view statistics in multiple different graphs or charts

#	Description	Time	Owner
1	Separate data formatting functions into a separate file that can be called for in a easy manner	3h	Jason
2	Create emotion radar graph for top songs	1.5h	Jason
3	Create pie graph formatting for data	1h	Jason
4	Format data for bar graphs	2h	Jason
5	Add option to resize grid layout	1h	Jason
6	Edit page layout to match rest of website	1h	Alex
7	Fix filtering of song data for their filter menu with sorted by top song metric & an option to display	1.5h	Jason

#	Description	Time	Owner
	certain indexes (ie top 10-20 songs)		
8	Fix bug for graph state not being saved on "save graphs" and not being loaded	3h	Jason
9	Add/fix handling for exceptional cases such as no friends, no friend's data, no advanced data, etc.	2h	Jason
10	Test graph creation for all variations of popup & debug remaining bugs	5h	Jason
11	Redesign UI to be more user-friendly and visually pleasing	3h	Rheaa

Acceptance Criteria:

1. Given the user has selected a radar graph, when the user selects data dropdown, then they should see top songs and can generate said graph.
2. Given the user has selected a pie graph, when the user selects data dropdown, then they should see multiple options (other than the sample data) and can generate said graph.
3. Given the user is on the stats page, when the user tries to change the grid layout grid cell numbers, then the user can do so.
4. Given the user is on the filter menu, when the user tries to change songs selected, then they can select a range of songs sorted by top song metric to display
5. Given the user wants to save a layout, when the user goes back to the page, then the graph displays the songs selected when saved.

User Story #3r:

As a user, I would like to be able to view my friends' statistics as per previously mentioned user tracked statistics

#	Description	Time	Owner
1	Fix race condition w/ friends data bug	1h	Jason
2	Allow for display of friends' data on graph	4h	Jason
3	Edit page layout to match rest of website	1h	Alex
4	Test display of friend data	1h	Jason

Acceptance Criteria:

1. Given that the user is on the stats page, when the user opens the "Add Graph" menu, they see a prompt to add a friend's data.
2. Given that the user has unfriended a user, when the user tries to display a graph with that friend's data, then that data is not displayed
3. Given that the user has no friends, when the user tries to add a graph, they will not be able to add friend data to a graph
4. Given that the user has opened "Add Graph" menu, when a graph that can have multiple data, user can add friend data to display
5. Given that the user has opened "Add Graph" menu, when user has selected a data type that can not have friend's data, user can not add friend data to display

User Story #3x:

As a user, I would like to be able to preview graphs before creating them

#	Description	Time	Owner
1	Reformat popup to display in a better user-friendly format	2h	Jason
2	Allow for preview of graphs on the popup	1.5h	Jason
3	Stop display on popup for elements that cannot be modified	1h	Jason

(Testing merged w/ 3L since tests would be 3L but instead of 'generate graph', you press 'preview graph')
Acceptance Criteria:
1. Given that the user is on the stats page, when the user opens the "add graph" menu, then the data is displayed in a user-friendly fashion
2. Given that the user is on the "add graph" menu, when the user presses "preview graph", then a graph preview is shown
3. Given that the user is on the "add graph" menu, when the user selects any combination of values, then invalid options should not appear.
4. Given that the user has not filled out all the fields correctly, when the user tries to create a graph preview, then an alert should appear.
5. Given that the user has generated a graph preview, when the user generates the graph, then the graph should look similar to the desired graph.

User Story #4j & #4i:

As a user, I would like to choose two playlists and have a new playlist made from the Spotify-recommended songs of both playlists or merge them.

#	Description	Time	Owner
1	Add frontend on playlist manager to search playlists and select 2 playlists to either merge or combine recommendations	3h	Alex
2	Add backend to search playlists	1h	Dane
3	Add backend to pull recommendations from two playlists and create a playlist from both	2h	Dane
4	Create backend to merge two playlists by inserting tracks in alternating order	2h	Dane
5	Manual testing	1h	Dane

Acceptance Criteria:
1. Given a user is logged in, when they navigate to the playlist manager page, they can choose to merge to playlists.
2. Given a user is logged in, when they navigate to the playlist manager page, they can choose to create a new playlist from recommendations based on two old playlists.
3. Given a user is on the playlist manager page, when they select to merge a playlist or create a fusion of recommendations, they are prompted with two scroll boxes of playlists populated with owned playlists.
4. Given a user selects to combine, when they see the two scroll boxes of playlists, there is a search bar to select unowned playlists and an input to name the new playlist.
5. Given a user selects two playlists, when they confirm their selection, a new playlist is generated on their account with the selected name and songs.

User Story #4n:
As a user, I would like my emotion presets to persist between logins

#	Description	Time	Owner
1	Edit emotion preset frontend to reflect storage changes	1h	Alex
2	Update database to store emotion presets and pull emotion presets	2h	Bodhi
3	Edit backend to allow for storage and retrieval	2h	Dane

| 4 | Manual testing | 1h | Dane |

Acceptance Criteria:
1. Given the user is on the parameter page, when they create a new emotion, then the name reflects it is a new emotion.
2. Given the user is on the parameter page, when they create a new emotion and leave and return to the page, then the new emotion and the parameters are saved.
3. Given the user is on the parameter page, when they create a new emotion from deriving an emotion from the playlist, then the new emotion is named accordingly and saved.
4. Given the user is on the parameter page, when they edit an old emotion and leave and return to the page, then the edited parameters are saved.
5. Given the user is on the parameter page, when the user tries to delete the preset emotions, then it is indicated that they cannot delete it.

User Story #4o:
As a user, I would like my playlist manager to dynamically display my playlist data after it is changed

#	Description	Time	Owner
1	Edit frontend to call for updates upon actions	1h	Alex
2	Edit backend to support simple updates edit	1h	Dane
3	Manual testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is on the playlist manager, when they add a song to a playlist, then the playlist song list updates to reflect that.
2. Given the user is on the playlist manager, when they remove a song from a playlist, then the playlist song list updates to reflect that.
3. Given the user is on the playlist manager, when they generate a new playlist, then the playlist list updates to reflect that.
4. Given the user is on the playlist manager, when they follow a playlist, then the playlist list updates to reflect that.
5. Given the user is on the playlist manager, when they unfollow a playlist, then the playlist list updates to reflect that.

User Story #4p:
As a user, I would like to choose at what portion of my playlist is reordered

#	Description	Time	Owner
1	Add frontend index selection	2h	Alex
2	Edit backend to allow index to be dynamic	2h	Dane
3	Manual testing	1h	Dane

Acceptance Criteria:
1. Given the user is on the playlist management page, when the user selects the reorder button, the selected playlist's songs are shown in an indexed manner in order.
2. Given the user has selected a playlist to reorder, when they are choosing indexes, then they are notified to select the starting song, the ending song, and where to insert the songs.
3. Given the user is on the playlist management page, when the user is reordering their playlist, then they can cancel the reorder selection.
4. Given the user is on the playlist manager page, when the user clicks the reorder button, their Spotify playlist is reordered according to the chosen indexes.
5. Given the user is on the playlist manager page, when the user reorders their playlist, the playlist song list updates to reflect that.

User Story #6d:

As a user, I would like to see my game scores in a presentable UI

#	Description	Time	Owner
1	Verify game scores are still being stored correctly	1h	Rheaa
2	Show game scores in a coherent format	2h	Rheaa
3	Edit game page layout to match rest of website	1h	Alex
4	Manual Testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is logged in, when they go to the games page, they are able to see the scores of previous games below the links to all the games
2. Given the user is on the games page, when they have finished playing game 1, they can see their scores update in the table (or an alternative coherent formats) for game 1
3. Given the user is on the games page, when they have finished playing a game 2, they can see their scores update in the table (or an alternative coherent formats) for game 2

4. Given the user is on the games page, when they have finished playing a game 3, they can see their scores update in the table (or an alternative coherent formats) for game 3
5. Given the user is on the games page, when they have finished playing a game 4, they can see their scores update in the table (or an alternative coherent formats) for game 4
6. Given the user is on the games page, when they have finished playing a game 5, they can see their scores update in the table (or an alternative coherent formats) for game 5

User Story #9c & #9d & #9e:

As a user, I would like to transfer Spotify playback to a different Spotify device on PULSE, control the queue, and change my Spotify playback.

#	Description	Time	Owner
1	Create expanded playback page	2h	Alex
2	Display information about currently playing information	2h	Alex
3	Create backend function to pull spotify devices	1h	Dane
4	Create backend function to change spotify device	1h	Dane
5	Create backend function to pull queue	1h	Dane
6	Create backend function to update queue	1h	Dane
7	Create search bar for songs, artists, albums, playlists, episodes, or shows	1h	Alex
8	Create backend function to change spotify playback based off criteria	2h	Dane
9	Manual testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is logged in, when they click the button on the playback bar to go to the expanded playback page, then they are taken to the page
2. Given a user is on the expanded playback page, when they inspect the page, they are presented with buttons to view the queue, search for a new spotify item, and change spotify device.

3. Given a user is on the expanded playback page, when they want to search for a new spotify item, then they can select a criteria to search for.
4. Given a user is on the expanded playback page and their player is open, when they are presented with search results and they click on one, then their player starts playing from what was clicked
5. Given the user has searched for something, when they want to add an item to a queue, the search box has a button to add each item to a queue.
6. Given the user has searched for something, when they want to inspect their queue or devices, then there is a list to inspect them.
7. Given the user has searched for something, when they want to change devices, then there is a button next to each device to select it as your active device.

User Story #12a:

As a user, I would like to get 'matches' on songs in a user-friend dating-like manner

#	Description	Time	Owner
1	Create navigation to song matchmaking page	0.25h	Rheaa
2	Create basic card component for matchmaking page	2h	Rheaa
3	Create left and right buttons for swiping, create a button to view all the songs you swiped right on	1h	Rheaa
4	Create animation for left and right swiping and additional animations	2h	Rheaa
5	Getting the songs from the backend functions and loading the image, name and artist onto the card	2h	Rheaa
6	Edit page layout to match rest of website	1h	Alex
7	Create backend functions to get image, name, artist, and genre of a song	1h	Noah
8	Algorithmically recommend user a song based on previous selections by dynamically updating and storing their preferences	4h	Noah
9	Write route to display all of the user's liked songs and add them to playlists as well as other options	3h	Noah

10	Store and retrieve song match data and recommendation parameters from database	0.5h	Bodhi
11	Manual Testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is logged in, when they navigate to the matchmaker page, then the songs section of the matchmaker page loads up.
2. Given the user is on the matchmaker page, when they toggle to the user section and swipe right on a card, then the user gets added to the view section
3. Given the user is in the matchmaker page, when they toggle to the user section and swipe left on a card, then the user does not get added to the view section
4. Given the user is in the matchmaker page, when they are done matching songs, then there is a button that allows them to see all the songs they have matched with so far.
5. Given the user is on the songs page, when they toggle to user instead of song, the user matching page is loaded up.

User Story #12b:
As a user, I would like to get matched to other users based on their song preferences.

#	Description	Time	Owner
1	Build the toggle between user and songs with functionality	1h	Rheaa
2	Create new card component for user : with user image and favorite song (and maybe top artist and song)	1.5h	Rheaa
3	Animation for swiping	1h	Rheaa
4	Edit page layout to match rest of website	1h	Alex
5	Create a button to view all the people you swiped right on	0.25h	Rheaa
6	Connect with the backend functions	2h	Rheaa
7	Create algorithm to filter through users and display the ones with a closest 'match'	3h	Noah
8	Create criteria for a user to match with another	3h	Noah

	user with and without advanced stats		
9	Store and retrieve user match data from database	0.5h	Bodhi
10	Manual Testing	1h	Rheaa

Acceptance Criteria:

1. Given the user is in the matchmaker page, when they toggle to the user section, then they are able to see a card in the center with details such as the user's image and favorite song.
2. Given the user is in the matchmaker page, when they toggle to the user section and swipe right on a card, then the user gets added to the view section
3. Given the user is in the matchmaker page, when they toggle to the user section and swipe left on a card, then the user does not get added to the view section
4. Given the user is in the matchmaker page, when they are done matching for the moment, then there is a button that allows them to view all the people they have matched with so far
5. Given they user is on the matchmaker page, when they toggle to song from user, the song matching page is loaded up

User Story #13a-g:

As a user, I would like to search for new artists and follow them, unfollow artists, see an artist's related artists, see an artist's top songs by country, see what genres an artist is associated with, be able to inspect what artists I follow, see the audio features of a song

#	Description	Time	Owner
1	Add information page for artists	2h	Alex
2	Add information page for songs	1h	Alex
3	Add search bar for songs and artists	1h	Alex
5	Edit page layout to match rest of website	1h	Alex
6	Create backend to follow and unfollow artists	2h	Dane
7	Create backend to see an artists related artists	1h	Dane
8	Create backend to get top songs for an artist	1h	Dane
9	Create backend to pull genres	1h	Dane

10	Create backend to see audio features of a song	1h	Dane
11	Create backend to pull followed artists	1h	Dane
12	Manual Testing	1h	Rheaa

Acceptance Criteria:
1. Given the user is logged in, when they want to get info from their artists or songs, then they can navigate to the information page.
2. Given the user is logged in, when they click on a followed artist, then the user can see the artist's info.
3. Given the user is logged in, when they want to see an artist they have not followed yet, then they can search an artist and see their info and have the opportunity to follow them.
4. Given the user is logged in, when they want to see info for a related artist, then clicking on the artist repopulates the info box with the new artist's info
5. Given a user logged in, when they want to see the info of a song, then they can search a song and see its info.

Total Hours per Person:

Developer	Hours
Rheaa Sharma	30
Noah Stern	30
Dane Shaffer	30
Alex Polivka	32
Jason Zheng	34.5
Bodhi Scott	31
Total	**187.5**

Remaining Backlog:

Functional Requirements:

1. Basic Infrastructure
    a. As a user, I would like to be able to register for a PULSE account using my email (if time allows)
    b. As a user, I would like to be able to register for a PULSE account using OAuth such as Google (if time allows)
    c. ~~As a user, I would like to be able to register for a PULSE account using Spotify~~
    d. ~~As a user, I would like to be able to login and manage my PULSE account~~
    e. As a user, I would like to be able to reset my password if I forget it
    f. As a user, I would like to sync my account across devices/platforms
    g. ~~As a user, I would like to access all other pages from a home page~~
    h. ~~As a user, I would like authentication and bad routing errors to be handled gracefully and direct the user to an error page~~
    i. ~~As a user I would like to be able to actually visit the website, and I would like the load times to not be abysmal.~~
    j. ~~As a user, I would like my data to be secure in all ways, at all times.~~
    k. ~~As a user, I would like for songs, albums, and artists to be displayed consistently across the website.~~
2. Settings
    a. ~~As a user, I would like to have dark/light mode~~
    b. ~~As a user, I would like to be able to change text size~~
    c. ~~As a user, I would like to have custom backgrounds~~
    d. ~~As a user, I would like to have preset themes to change up the interface's look~~
    e. ~~As a user, I would like to be able to view my Pulse friends~~
    f. ~~As a user, I would like to edit my username~~
    g. ~~As a user, I would like to modify my profile information such as gender, profile picture and location.~~
    h. ~~As a user, I would like to be able to customize my public profile.~~
    i. ~~As a user, I would like backgrounds to be saved along with themes.~~
3. Statistics Dump
    a. ~~As a user, I would like for my top songs and artists for the last 4 weeks, 6 months and all time to be tracked~~
    b. ~~As a user, I would like to be able to export Spotify data and import it into PULSE to get the following statistics~~

c. ~~As a user, I would like for number of minutes listened to be tracked in my listening activity~~

d. ~~As a user, I would like for percentage of total music listened to be tracked in my listening activity~~

e. ~~As a user, I would like for times when music was listened to (morning, afternoon, night, etc.) in my listening activity~~

f. ~~As a user, I would like for times when songs were listened to fully or skipped or repeated in my listening activity~~

g. ~~As a user, I would like to to see songs, artists, genres and favorite era of music sorted by the metrics above~~

h. ~~As a user, I would like to track my follower numbers~~

i. ~~As a user, I would like for my most recently listened to songs to be displayed~~

j. ~~As a user, I would like for my songs, playlists, and albums that I've liked recently to be displayed~~

k. ~~As a user, I would like to be able to specify what time period I want my stats to be from.~~

l. ~~As a user, I would like to be able to view statistics in multiple different graphs or charts~~

m. ~~As a user, I would like to be able to create custom layouts for the statistics page~~

n. As a user, I would like to be able to download my and my friend's statistics in a CSV and text file

o. As a user, I would like to be able to download my and my friend's statistics in a pdf file

p. ~~As a user, I would like to be able to create custom graphs with specified data and formats~~

q. ~~As a user, I would like to be able to have many choices of graphs to display data~~

r. ~~As a user, I would like to be able to view my friends' statistics as per previously mentioned user tracked statistics~~

s. As a user, I would like to import my friends stats from a csv/txt file to compare

t. ~~As a user, I would like to be able to view my playlists on the website~~

u. ~~As a user, I would like to see what genres I've listened to percentage-wise~~

v. ~~As a user, I would like to see what emotions I experience when listening to Spotify percentage-wise~~

w. ~~As a user, I would like to see what emotions I experience when listening to a selected song percentage-wise~~

   x. As a user, I would like to be able to preview graphs before creating them (if time allows)

   y. As a user, I would like to be able to modify graphs after they have been created (if time allows)

 4. Playlist Page

   a. ~~As a user, I would like for recommendations for songs to add on the playlist based on Genre~~

   b. ~~As a user, I would like for recommendations for songs to add on the playlist based on Artist~~

   c. As a user, I would like for recommendations for songs to add on the playlist based on Song Writer

   d. As a user, I would like for recommendations for songs to add on the playlist based on Location

   e. ~~As a user, I would like for recommendations for songs to add on the playlist based on Album~~

   f. ~~As a user, I would like for recommendations for songs or playlists to based on User emotion presets (informed by Spotify API)~~

   g. As a user, I would like an AI to DJ for me based on BPM matching

   h. ~~As a user, I would like a playlist activity monitor that displays the amount of times the playlist was played in an user friendly manner on the playlist page (IMPOSSIBLE)~~

   i. ~~As a user, I would like to choose two playlists and have a new playlist made from the Spotify-recommended songs of both playlists~~

   j. ~~As a user, I would like to choose two playlists and have a new playlist to merge them~~

   k. ~~As a user, I would like to be able to set my emotion parameters using a user chosen playlist.~~

   l. ~~As a user, I would like to manage playlists through Pulse~~

   m. ~~As a user, I would like to auto generated playlists through Pulse based on genre~~

   n. ~~As a user, I would like my emotion presets to persist between logins~~

   o. ~~As a user, I would like my playlist manager to dynamically display my playlist data after it is changed~~

   p. ~~As a user, I would like to choose at what portion of my playlist is reordered~~

 5. Song Page

   a. ~~As a user, I would like to be able to find similar songs (based on Spotify API parameters such as loudness) to the current song~~

   b. ~~As a user, I would like to be able to find similar songs by specifying Spotify API target parameters such as loudness to a chosen song.~~

   c. As a user, I would like to be able to view lyrics to songs

     d. As a user, I would like for songs to be automatically captioned based on lyrics (If time allows)

     e. As a user, I would like for songs to be automatically removed from playlists after a certain threshold of skips set by the user

     f. As a user, I would like for the song to have music videos automatically linked if available on Youtube or other such platforms

     g. ~~As a user, I would like for a song activity monitor that displays the amount of times the song was played in an user-friendly manner on the song page~~

6. Game Page

     a. ~~As a user, I would like to play the game "Guess the Song/Artist" as a quiz format and heads-up format~~

     b. ~~As a user, I would like to play games such as Guess who listens to this song as a quiz format and heads-up format~~

     c. ~~As a user, I would like to play the game "Guess the Next Lyric" as a quiz format and heads-up format~~

     d. ~~As a user, I would like to see my game scores in a presentable UI~~

7. Search (if time allows)

     a. ~~As a user, I would like to be able to search for songs, albums, artists, playlists.~~

     b. As a user, I would like to be able to filter my search with keywords such as year, genre, new songs and popularity.

8. Song Player (if time allows)

     a. ~~As a user, I would like to be able to listen to samples of songs when they are recommended to me~~

     b. ~~As a user, I would like to be able to control Spotify playback through a music player on PULSE~~

     c. ~~As a user, I would like to start Spotify playback through a Spotify device on PULSE after searching for a playlist or song~~

     d. ~~As a user, I would like to manage the Spotify queue by searching a song and adding or removing it~~

     e. ~~As a user, I would like to change my Spotify playback to a new song or playlist~~

9. Music Uploader (if time allows)

     a. As a user, I would like to be able to upload local files and add them to new/existing playlists

     b. As a user, I would like to be able to auto-fill metadata for these local files

10. TicketMaster API (if time allows)

     a. As a user, I would like to be able to view music show recommendations when viewing relevant songs/artists/albums

11. PULSE Chat Bot (if time allows)

      a. ~~As a user, I would like to interact with a PULSE chatbot through a textbox~~
      b. ~~As a user, I would like to be able to get song recommendations per Genre, Artist, Location, Album and User emotion through PULSE chatbot~~
      c. ~~As a user, I would like to be able to automatically populate playlists per Genre, Artist, Location, Album and User emotion through PULSE chatbot~~
      d. ~~As a user, I would like to be able to get song information by inputting a short phrase from the song into PULSE chatbot~~
      e. ~~As a user, I would like to get help with finding/doing a certain action on the website by asking PULSE chatbot~~
      f. ~~As a user, I would like to be able to give feedback to developers about requested features, bugs or general messages through PULSE chatbot~~

12. Dynamic Music Matching
      a. ~~As a user, I would like to get 'matches' on songs in a user-friend dating-like manner.~~
      b. ~~As a user, I would like to get matched to other users based on their song preferences.~~

13. Information Page
      a. As a user, I would like to be able to inspect what artists I follow
      b. As a user, I would like to see what genres an artist is associated with
      c. As a user, I would like to see an artist's top songs by country
      d. As a user, I would like to see an artist's related artists
      e. As a user, I would like to see the audio features of a song
      f. As a user, I would like to search for new artists and follow them
      g. As a user, I would like to be able to unfollow artists

Non-Functional Requirements:

I. Quality of Life
    A. As a user, I would like the website to be responsive with reasonable speed and at all times
      1. Reasonable speed: < 400ms for typical navigation inputs
      2. Be able to reasonably handle double-inputs

II. Accessibility
    A. As a user, I would like the website to be usable in multiple formats
      1. Support for landscape and portrait mode
      2. Support for desktop and mobile browser
      3. Support for scaled-up/down windows

III. Maintainability
    A. As a developer, I would like to be able to easily maintain features
    B. As a developer, I would like to be able to easily add new features

IV. Security

A. Personal information and passwords will be encrypted when transferred using an 18-bit encryption
B. Encrypt using a more-secure Caesar cipher (if time allows)
C. User information from Spotify should be stored in a secure manner to protect against common exploits such as cross-site-scripting, buffer overflow, and SQL injection using industry standard practices such as Microsoft's own SQL Server documentation on common security practices. This includes practices like input sanitization, privilege controls, and encryption

V. Architecture and Performance
A. As a developer, I want to run the frontend using ReactJS and the backend using Python
B. As a developer, I want to use SQL for storing data
C. As a developer, I want to connect the frontend and backend using Flask