Feature 1: User Authentication

- Login:
    - User logs in successfully and gets redirected to home page
    - Can't find username/password, sends error message saying "Incorrect username/password"
    - Easy access to register page (link)
- Register:
    - Able to select universities supported (CU, CSU) from dropdown
    - User registers successfully and gets redirected to login page
    - Username already found in database, sends error message saying "User already registered"
    - Easy access to login page (link)
- The testing environment will be in staging. In this stage, we are focused on Quality Assurance.
- Testing data
    - Correct usernames & passwords
        - Should log in correctly
    - Incorrect usernames & passwords
        - Should display incorrect username and/or incorrect password
    - Just a username, no password
        - Should prompt a password
    - Just a password, no username
        - Should prompt a username
- User acceptance testers
    - To test this feature, we will test ourselves, including all corner cases. We will have 2-3 peers in the class try to log in to see if they can find additional problems. We will give them our documentation of the user acceptance criteria and see if the peers have any ideas of criteria we are missing.
- Results: This feature presented us with the most difficulty, especially with respect to password management. The feature now passes all of these test cases.

Feature 2: Presentation of data

- Specific Test cases:
    - 1. Top 10 songs
    - 3. Search feature
    - 4. Trends page
    - 5. Clicking on tabs in header / on home page
        - Sidebar, nav bar
- Test data
    - University of Colorado Boulder

- The testing environment will be in staging. In this stage, we are focused on Quality Assurance.
- Test results
    - Top 10: 10 songs should show up ranked by popularity, user should be able to select a timeframe and the songs should change accordingly
    - Search: should display songs with titles similar to the one searched for
    - Trends: should display the trends for the song selected, or an error message if no one has listened to that song yet
    - Navigation Bar: If clicked on the current page, should reload to the same page. Otherwise, take to the associated page
- User acceptance testers
    - Each member of the team will check the functionality of these pages.
    - We will have a new user test these pages for intuitive design.
- Results: Our application passes all of these tests. We got good feedback from new users about our UI design – the most major thing we fixed was making it more clear whether or not something was a button.

Feature 3: Acquisition of data

- The user can search for songs and add the songs to their favorites, which will then store that song and appropriately store the data in our database. Based on the timeframe and the number of people that like certain songs, we will log the information necessary in the databases.
- Testing/Results:
    - **Test: If a user looks up a song that exists within the database (Yellow Submarine)**
        - Result: Results should be displayed to them in a proper fashion.
    - **Test: If a user looks up a song that does not exist (nonexistent song)**
        - Result: Prompt them with “does not exist message”
    - **Test: If a user likes a song (clicking the card)**
        - Result: update the transaction
    - **If there are multiple songs/albums/playlists/artists with the same name,**
        - It will display all the data, not just one
- The testing environment will be in staging. In this stage, we are focused on Quality Assurance.
- To test this feature, we will test ourselves, including all corner cases, as well as finding 2-3 peers in the class to run through our application and test all functionality to see if they can find problems. We will give them our documentation of the user acceptance criteria and see if the peers have any ideas of criteria we are missing.

- Results: We extensively tested the search feature, and our application passes these tests. Users found the design intuitive and liked that several different versions of a song popped up.

New User Testing Observations

Case 1:

- User Successfully logged in
- Attempted to select Utah University, then confused about why nothing appeared. It was explained that there was not data for that school yet.

Case 2:

- User was able to sign up
- Log in with the same credentials as used in signing up
- User liked the layout of all the pages
- Use liked the overall concept of the problem that it was solving
- User wished they were able connect external music apps, so they don't have to use two different platforms

Case 3:

- Successful log in
- Searched for an artist in the song search, mentioned that there should be more results. I explained that we were limited to the number of results by the third party API we were using.
- User liked the profile page

Case 4:

- User was able to register/login with the same credentials
- User searched for artist "Lil Durk" which pulled up five songs, I explained that he should search for specific songs in the search function
- User liked the data trends page, but wished there was more instruction on how to use it.

Case 5:

- User attempted to register without an email, and the register page correctly did not allow that user to register
- User successfully logged out

Case 6:

- User registered and logged in successfully
- User expected clicking on a card on the top 10 page to pop up with more information rather than like the song

- User recommended a popup message instead of a separate page for liking a song
- Users wished that the trends page did not require entering a specific song name, and instead search with general terms and choose from a dropdown.

Case 7:

- User thought that the tiles in the profile page were hyperlinks (tried to click them). User explained that they thought that since the tiles turned green when you hover over them. This hover feature was changed after this user test.
- User liked the data trends page. They mentioned having a longer timeframe, but this is out of scope for our project.