**Musicboxd Requirements Review Package**

**Executive Summary**

Inspired by the popular film rating app Letterboxd, our team’s project is to create a music rating app called Musicboxd. Users can create an account with a unique username and password. After an account is made, users will be able to showcase their music taste by rating albums and songs they’ve listened to on a 5-star rating scale. As part of their profile, users will be able to list their current top four favorite albums as well as their current top five favorite songs for all other users to see. Users can edit their favorite albums and songs whenever they’d like to. Profiles will be customizable, users will be able to upload a profile picture and edit the colors of their profile to make their profile feel more personal. In addition to rating music albums, users will also be able to leave a short review to tell everyone their thoughts on the album or song. Other users can then like the review if they agree with it. Every review posted by a user will remain on their profile for them to have a collection of all the albums and songs they have listened to. If a user wishes to, a review can be deleted from their profile. If we have time, our group would also like to add lists as a feature to our app. This feature would allow users to create a list titled anything, and then users can add any albums or songs to it. For example, a user can create a list on their profile titled “Best albums released in 2025,” and then the user would be able to add any albums they’d like to the list. For cycle one, we plan on finishing our user interface including the user’s profile and the search page. Our goal is to have all the profile features previously mentioned complete. Additionally, our other delivery goal is to create and maintain databases for the users and reviews. If we have time, we would also like to add the list feature to Cycle 1; however, this feature is not essential and can be pushed to Cycle 2 or scrapped completely. Our ultimate vision is to create an app where users can express their thoughts about music and to connect with others with similar music taste. Since the demographic of people who like music is huge, we narrowed down our target persona to artsy Gen Z high school and college students. Older generations may find our app appealing, but when thinking of Letterboxd’s main demographic, we settled on our target persona being in the 16-26 age range. Specifically, our target persona is a young adult who wants to showcase their music which covers people who love mainstream music to people with a more esoteric music taste.

**Requirements**

Functional Requirements

The system shall allow users to sign up by creating a username and password.

The system shall allow users to sign in with a username and password.

Assuming the user is already logged in, the system shall allow the user to add the names and artists of their top four favorite albums to their profile when they click the add album button on their profile.

Assuming the user is already logged in, the system shall allow the user to add the names and artists of their top five songs to their profile when they click the add song button on their profile.

Assuming the user is already logged in, the system shall allow the user to rate an album or song using the five-star scale when they click the add review button on their profile.

Assuming the user is already logged in, the system shall allow the user to change the background color of their profile when they click the modify button on their profile.

Assuming the user is already logged in, the system shall allow the user to view the history of all their reviews when they click the view reviews button on their profile.

Assuming the user is already logged in, the system shall allow the user to delete a review of theirs when they click the delete button on a review on their profile.

Assuming the user is already logged in and has searched for another user, the system shall allow the user to view the other user’s profile when the view profile button is clicked.

Assuming the user is already logged in and is viewing another user’s profile, the system shall allow the user to like the other user’s reviews.

Non-Functional Requirements

The system shall return search results when users search for other users’ profiles in under 1 second 90% of the time.

The system shall add changes the user makes to their profile in under 1 second 90% of the time.

The system shall validate the user's credentials and log them in under 3 seconds 90% of the time.

Use Cases

Title: user logs into the system

Actor: registered user

Goal: access user profile

Preconditions: user has a valid account

Main Flow:

1. User enters username and password
2. System validates credentials
3. System displays user profile

Alternative Flow: invalid credentials – system displays error

Title: user creates an account

Actor: unregistered user

Goal: create an account

Preconditions: user does not have a valid account

Main Flow:

1. User clicks create account button
2. User enters username and password
3. System displays blank profile for user to customize

Alternative Flow: user already has an account – system displays error

Title: user enters their top 4 albums

Actor: registered user

Goal: user inputs the name and artist of their top 4 albums

Preconditions: user has an account and a profile set up

Main Flow:

1. User clicks add album
2. User enters name and artist
3. System displays user input
4. Repeat for each album

Alternative Flow: user already has albums listed – user deletes previous name and artist and replaces it with new name and artist

Title: user enters their top 5 songs

Actor: registered user

Goal: user inputs the name and artist of their top 5 songs

Preconditions: user has an account and a profile set up

Main Flow:

1. User clicks add song
2. User enters name and artist
3. System displays user input
4. Repeat for each song

Alternative Flow: user already has songs listed – user deletes previous name and artist and replaces it with a new song name and artist

Title: user writes a review

Actor: registered user

Goal: user rates and writes a review for an album or song

Preconditions: user has an account

Main Flow:

1. User clicks write review
2. User enters name and artist
3. User enters the number of stars they want to rate it
4. User enters a brief description of their thoughts

Alternative Flow: user changes their mind about writing a review – user presses exit button on review, and no review is created

Title: user designs profile

Actor: registered user

Goal: user can change the color of their profile

Preconditions: user has an account

Main Flow:

1. User clicks edit profile
2. Color options are presented to user
3. User clicked which option they want
4. System updates user’ profile appearance

Alternative Flow: user changes their mind about changing profile appearance – user clicks exit button on edit profile, and no update is made

Title: user searches for another user

Actor: registered user

Goal: user will input another user’s name and search results will appear

Preconditions: user has an account

Main Flow:

1. User navigates to the search bar
2. User inputs another user’s name
3. Search results appear
4. User clicks on the profile they want to view
5. The system brings the user to the profile they want to view

Alternative Flow: the user that was searched does not exist – error message will appear

Title: user likes a review

Actor: registered user

Goal: user will like a review they’ve read

Preconditions: user has an account, user is viewing another user’s profile

Main Flow:

1. User one navigates to user two’s reviews
2. User one clicks the like button on a review
3. System updates the like count on the review
4. System displays the number of likes on a review

Alternative Flow: user can unlike a review – system updates the like count, and system subtracts a like from the display