

Caroline Chen yc3756

Cherry Chu ccc2207

Adam Carpentieri ac4409

Haomin Gui hg2533

Part 1

Link: https://github.com/Adam7288/Autumn_Audio

Part 2

User Stories

1. As a chatroom user, I want to see other participants to see what they are listening to or have previously listened to.

My conditions of satisfaction are: The UI will show the participants current and recently played music.

2. As a chatroom user, I want to be able to talk to other chatroom users about the music we are listening to and the genre that the chatroom is about.

My conditions of satisfaction are: the user can send messages into the chatroom and see the messages that other users type and send.

3. As an active participant I want to share new music with other users and also listen to shared music from them.

My conditions of satisfaction are: 1. When you share the music, it should appear on other listener's playing queue in their Spotify app. 2. The UI will reflect that a song has been shared in the chat window.

acceptance tests

1. When a user logs on to a room with other participants, they will see **only** the recently and currently playing songs of the **chatroom participants**. A failure condition would be if the user does not see the data at all, or that they see data from other chatrooms. Another way to test is for another user to change their song, and for the first participant to verify they see the change in 30 seconds or less.

Test:

To test, we run the server and join as different users by opening multiple web pages (and register through Spotify with different user accounts) in browsers. Each user is able to send messages to the chat and receive messages from other users. Ideally, the message is only sent to the user in a specific chatroom (a specific genre). But the message is sent to all participants in all chat rooms. We are not able to fix it in this iteration. Then we waited for 30 seconds and joined by opening another page in the

browser, the message sent 30 seconds ago was not shown to the current user, which is consistent with what we want.

2. The first user can send a message to the chatroom, and another user can verify that they see the corresponding message. And vice versa. After this we can scale to three users to make sure it works for all participants. Lastly, we will share screenshots to ensure the messages arrive in the correct order. A failure condition would be when the screenshots do not match.

Test:

We join from two browsers as two users (with two Spotify account). User 1 sent something in the chat and user 2 was able to see it. We then opened a third page and joined as another user (with another Spotify account). User3 is able to see the message from user 1 and 2. We take screenshots and verify that the messages are arriving in the correct order.

3. When a user shares a song, the song should appear at the end of the queue of the other participant(s). A failure condition would be when their app does not display the song to be played. The chatroom should have a log for the shared song. The failure condition would be when the chatroom does not display this information.

Test:

We test by opening two browsers and letting the two users join the same chat room. We then have one user share a song and the second user share the song afterwards. The second song is at the end of the queue (after user 1's song). The chatroom logs the information of a shared song. Then we open the third browser as the third user. User 3 is able to see the first two songs and it shares a song. The song is at the end of the song queue and there is another log in the chatroom for this third song.

Part 3

Unit test: https://github.com/Adam7288/Autumn_Audio/tree/master/src/test/java

Automated unit test: https://github.com/Adam7288/Autumn_Audio/tree/master/src/test/java

Front end tests are ran when building the react frontend:

https://github.com/Adam7288/Autumn_Audio/tree/master/react

Front end tests are included in the following folders:

https://github.com/Adam7288/Autumn_Audio/tree/master/react/src/Component/Chatroom

https://github.com/Adam7288/Autumn_Audio/tree/master/react/src/Component/Lobby

Part 4

Reports from your automated style checker and automated bug finder:

Spotbugs:

https://github.com/Adam7288/Autumn_Audio/blob/master/documents/autumn%E2%80%93%20SpotBugs%20Bug%20Detector%20Report.pdf

Checkstyle: ("Missing package-info.java file" not fixed)

Package-info.java is used to generate javadoc for our package. Since we do not need package documentation for this MVP. We ignored this error.

https://github.com/Adam7288/Autumn_Audio/blob/master/documents/Checkstyle%20Results.pdf