Callahan Hirrel

Taylor Barker

Jack Liu

Michael Spainhour

CSCI 250: Scalable Software

Project 2 Requirements

**PURPOSE AND SCOPE**

**Overall Goal:** To allow musicians to collaborate on projects without actually having to be in close physical proximity.

**Overall Scope:** Allows users to upload, share, and receive audio files with collaborating musicians (NOT used for the actual creation of audio files). Users may also view a history of their shares, which shows them each file they have sent and the username they sent it to, and each file they have received and the username they received it from.

**Stakeholders and Interests:**

User––Wants to upload, send, receive, and download audio files in a quick, safe, and intuitive manner.

**GLOSSARY**

*Software* and *program* are used interchangeably to refer to the proposed peer-to-peer network application.

*Audio file* refers to a file which only contains audio data (*e.g.* any files with the extensions .wav, .aiff, .mp3, etc…).

**USE CASES**

**Use Case 1: Create An Account**

**Primary Actor:** User

**Stakeholders and Interests:**

User––Wants to be able to create an account so that they are recognizable by username over the network, and so that their history is saved and only viewable by them.

**Precondition:** Software is running.

**Minimal Guarantee:** Software doesn’t crash, displays appropriate error message(s).

**Success Guarantee:** Account is created, account information stored on computer for later retrieval.

**Trigger:** User clicks "Create Account" button.

**Main Success Scenario:**

1. User enters username/password.
2. User confirms password.
3. Program saves username/password.
4. Program displays main window.

**Extensions:**

2a. "Password" and "Confirm password" contain different text.

2a1. Program displays error message to user.

2a2. Program clears both "Password" and "Confirm password" fields.

**Use Case 2: Adding an Audio File**

**Primary Actor:** User

**Stakeholders and Interests:**

User––Wants to add an audio file to program.

**Precondition:** Software is running and user is logged in.

**Minimal Guarantee:** Software doesn’t crash, displays appropriate error message(s).

**Success Guarantee:** User adds file, program displays filename under "Added Files" heading.

**Trigger:** User clicks "Add a File" button.

**Main Success Scenario:**

1. Program opens system browser.
2. User selects file from browser.
3. Program copies file to program directory.
4. Program displays file name under "Added Files" heading.

**Extensions:**

1a. Program cannot locate file

1a1. Program displays error message.

**Use Case 3: Sending an Audio File**

**Primary Actor:** Program

**Stakeholders and Interests:**

User––Wants file sent quickly and safely across network to another user.

**Precondition:** Software is running, user is logged in and connected to network.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** Program sends file across network to a different copy of the program, displays confirmation message to user.

**Main Success Scenario:**

1. User selects a file.
2. User clicks "Send File" button.
3. Program prompts user to select a destination.
4. User selects destination user.
5. Program sends file across network to designated user’s machine.
6. Program waits for confirmation that file was received.
7. Program displays confirmation.
8. Program logs sent filename and receiver’s username to share history.

**Extensions:**

2a. User has no file selected.

2a1. Program displays error message.

6a. Program receives no confirmation

6a1. After 10 seconds of waiting, program informs user the file was not received.

6a2. Program gives user option to resend.

6a3. If user resends, program repeats step 5.

**Use Case 4: Receiving an Audio File**

**Primary Actor:** Program

**Stakeholders and Interests:**

User––Wants to receive file and see filename.

**Precondition:** Software is running, user is logged in and connected to network.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** Program saves received file to program directory, displays message to user that file was received, displays filename under "Received Files" heading, sends confirmation message back across network.

**Main Success Scenario:**

1. Program receives incoming file.
2. Program sends confirmation message across network to sender.
3. Program saves file in program directory.
4. Program alerts user that new file was received.
5. Program displays filename under "Received Files" heading.
6. Program logs received filename and sender’s username to share history.

**Use Case 5: Saving an Audio File**

**Primary Actor:** User

**Stakeholders and Interests:**

User––Wants file saved to specific location

**Precondition:** Software is running and user is logged in.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** Program copies file to specified location, displays confirmation message to user.

**Main Success Scenario:**

1. User selects file from "Received Files" heading.
2. User clicks "Save File" button.
3. Program opens system browser.
4. User selects target directory from browser.
5. Program copies file to directory.
6. Program displays confirmation message.

**Extensions:**

2a. User has no file selected.

2a1. Program displays error message.

5a. Program cannot locate directory.

5a1. Program displays error message.

**Use Case 6: Connecting with Another User**

**Primary Actor:** User

**Stakeholders and Interests:**

User––Wants to successfully connect with another user over network.

**Precondition:** Software is running and user is logged in.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** Program connects to different copy of program across network, displays confirmation message.

**Main Success Scenario:**

1. User inputs IP address of different machine.
2. Program locates new machine over network.
3. Program sends username to connected program.
4. Program confirms connection established to user.
5. Program displays received username under "Connected with" heading.

**Extensions:**

1a. User inputs invalid IP address

1a1. Program displays error message, repeat step 1.

2a. Program cannot locate new IP address over network

2a1. Program displays error message, repeat step 1.

5a. Program receives no username

5a1. Program displays error message.

5a2. Program gives user option to disconnect.

5a3. If user remains connected, username under "Connected with" heading will read "No username."

**Use Case 7: Creating a Project:**

**Primary Actor:** User

**Stakeholders and Interests:**

User––Wants to successfully create a new blank project.

**Precondition:** Software is running and user is logged in.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** A new tab is created with the blank project GUI displayed.

**Main Success Scenario:**

1. User types a project name.
2. User initiates **Use Case 6**.
3. User clicks to create project.
4. Program creates new tab with blank project GUI displayed within.

**Extensions:**

3a. Project name field is blank

3a1. Alert message displayed, user instructed to name project, repeat step 1.

**Use Case 8: Audio Playback**

**Primary Actor:** Program

**Stakeholders and Interests:**

User––Wants to be able to playback audio files directly from the software.

**Precondition:** Software is running, user is logged in and viewing project, at least one audio filename displayed in project.

**Minimal Guarantee:** Software doesn't crash, displays appropriate error message(s).

**Success Guarantee:** OS’s default media player opened, audio file begins to play.

**Trigger:** User clicks an audio filename.

**Main Success Scenario:**

1. Program locates audio file using stored filename.
2. Program opens file via default media player.

**Extensions:**

1a. Program cannot locate file

1a1. Program displays error message, terminate use case.