

Requirements

- Better CLI which presents the users with all the available commands
- Listen back to a single or multiple sounds
- Enhanced ways to listen back to a sound
 - Layered audio files on top of each other
 - Reversed
 - Random segment
- Sorting files based on
 - Name
 - Size
 - Date created
 - Date modified
- Grouping audio files in the archive
 - List available files and directories
 - Create directories
 - Add files to directories
 - Rename directories
 - Rename files
 - Remove directories
 - Remove files from directories
- Export audio files feature
 - ZIP files and directories and save them into the archive memory
- Error handling for all implemented functions
- Update testing of the features

Use Case Name	UC-1: Export Audio Files
Summary	A selected audio file and/or a directory from the audio archive is exported into a ZIP file.
Rationale	With a growing audio archive, users might be interested in sharing some of the audio files with other people. Rather than searching for the file in the computer memory using a file manager or the command line, users are able to natively select the files and export them into a ZIP file, which can be sent to another user.
Users	All users
Preconditions	A file/directory needs to be uploaded into the audio archive
Course of events	<ol style="list-style-type: none"> 1. The user uses the Export Files command in the interface of the audio archive. 2. The software prompts the user to select a file and input the name for the ZIP file that will be created. 3. The software exports the selected item into a single zip file which is saved in the export directory of the audio archive. 4. The software informs the user that the files were successfully exported.
Exceptions	<ol style="list-style-type: none"> 1. In Step 2, if the audio archive does not have any audio files uploaded, the use case is aborted, and the user is informed by the software that there are no files available. 2. In Step 4, if the zipping of the files fails for any other reason, the software informs the user about an unexpected problem and prompts the user to try again. The use case resumes at Step 1.
Alternative paths	<ol style="list-style-type: none"> 1. In Step 3, the user selects more than one file to send. In this case, the software zips all the selected files into a single zip file. 2. In Steps 1, 2, 3, or 4, the user can abort the Send File feature. In this case, the software will not export any selected files.
Postconditions	The selected file has been exported into a zip file.

Use Case Name	UC-2: Play a segment of sound
Summary	A random segment of a single audio file is played. The segment is always the half length of the original audio file.
Rationale	On certain occasions, a user might be interested in listening back to a random segment of a sound rather than playing the whole audio. With the play a segment of sound feature, the user can easily achieve that without creating a segment of audio as a separate file and have a unique segment every time they use the feature.
Users	All users
Preconditions	An audio file needs to be uploaded into the audio archive
Course of events	<ol style="list-style-type: none"> 1. The user uses the Play Segment command in the interface of the audio archive. 2. The software prompts the user to select a file to listen to. 3. The software temporarily creates a randomly determined segment of the inputted sound. 4. The software plays the segment of the sound and deletes it afterward.
Exceptions	<ol style="list-style-type: none"> 1. In Step 2, if the file path is invalid, the use case is aborted, and the software informs the user that the path is invalid.
Alternative paths	<ol style="list-style-type: none"> 1. In Steps 1, 2, 3, or 4, the user can abort the Play Segment feature. In this case, the software will not create any segments of the audio files.
Postconditions	The randomly determined segment of a file is played and then deleted.

Use Case Name	UC-3: Organize sounds into folders
Summary	Organize sounds in folders by using functionalities such as creating a folder, moving sounds to a folder, deleting sounds from a folder, etc.
Rationale	Users might want to put certain sounds in one folder depending on their preference so that it is easier for them to find them and organize them. Instead of exiting the app and doing operations such as creating a folder and moving sounds to a folder using their operating system capabilities, they can use the built-in functionalities of the app which replicate the same commands without exiting the sound archiver app which makes it convenient for users.
Users	All users
Preconditions	Having sounds to move from one directory to another
Course of events	<ol style="list-style-type: none"> 1. The user creates a folder using the ‘-cr’ command. 2. The user adds the sound to the folder using the ‘-add’ command. 3. Display the results.
Exceptions	<ol style="list-style-type: none"> 1. The user types the name of the sound that has to be moved wrong in which case he will be prompted again and given feedback that the file was not found. 2. The user indicates the name of the folder to be moved to wrong in which case he will be prompted to try again.
Alternative paths	<ol style="list-style-type: none"> 1. The user creates a folder and decides to move files later. The user is not required to fill the folder with files immediately. 2. The user might want to delete a sound from a folder for which they can use ‘-rm’ command 3. The user might choose to rename the folder for which they can use ‘-rn dir’ command. 4. The user might want to delete the directory they created for which they can use the ‘-rm dir’ command
Postconditions	Files are organized in the folders that the user intends them to be

Use Case Name	UC-4: Sorting Audio based on meta
Summary	The sorting functionality allows users to organize and access audio files based on various attributes that are common to all audio files like date created, file size, etc.
Rationale	After adding a lot of sounds to their library, many users start finding it hard to find particular audio files and want some structure as to how the audios are displayed. It could be that the audio(s) they are looking for was created a long time ago, or they want to access audio files with a certain duration first or last. Whatever the specific case is, sorting the data ordinarily based on a given attribute makes it easier to navigate the library and find specific audios because it gives you an idea of where certain audios are in the list.
Users	All users
Preconditions	At least two audio files with associated metadata are present in the library
Course of events	<ol style="list-style-type: none"> 1. The user selects the sorting method from the system interface. They can choose to sort files based on file name, size, date created, or last date modified. 2. The user then enters the folder path of the folder they want to be sorted based on the criteria they chose in step 1. 3. The system then sorts the files in the given folder and prints out the sorted list without modifying the folder.
Exceptions	<ol style="list-style-type: none"> 1. If the user enters an invalid folder path, they will be notified through the interface that the specified path doesn't exist.
Alternative paths	-
Postconditions	<ol style="list-style-type: none"> 1. The sound library displays the sorted files according to the applied criteria 2. The user can navigate the sorted audio files and perform any task that was available to them before sorting.

Use Case Name	UC-5: Play sounds in reverse
Summary	Give users an option to manipulate audio playback by adding a reverse feature.
Rationale	Provide the user with different ways to explore their audio files.
Users	All users
Preconditions	Having a sound file to manipulate
Course of events	<ol style="list-style-type: none"> 1. The user finds the sound file they wish to play in reverse 2. Use the '-pr' command to play the sound 3. Listen to the sound played in reverse
Exceptions	<ol style="list-style-type: none"> 1. If the user enters an invalid path for the audio, they will be notified through the interface that the specified path doesn't exist.
Alternative paths	-
Postconditions	<ol style="list-style-type: none"> 1. The sound is played in reverse

Use Case Name	UC-6: List All files In Directory
Summary	Displays all the audio files in a given directory.

Rationale	After accumulating a significant number of audio files, users might lose track of what is in each folder and might want to see all the files in a particular folder.
Users	All users
Preconditions	The named directory should exist.
Course of events	<ol style="list-style-type: none"> 1. The user selects the list option through the system interface 2. The user provides the path of the folder with the files they want to see. 3. The system then prints/displays all the audio files in the given folder.
Exceptions	<ol style="list-style-type: none"> 1. If the user enters an invalid folder path, they will be notified through the interface that the specified path doesn't exist.
Alternative paths	-
Postconditions	<ol style="list-style-type: none"> 1. All the files in the given folder are displayed. 2. The user can navigate the displayed audio files and perform any task that was available to them before getting the displayed list.