

Assignment 2 Report: Media Library Application

Pair: Megan Hayes (3996884) and Sweta Kumari (1265161)

Git usernames: meganeashayes and Sweta7

Git repository: assignment-two-media-manager-gui-swift-assignment-2sweta

Object Oriented Concepts and Design Patterns

Model View Controller pattern

The models we used were the MediaLibraryManager collection of files and the ImportFiles.swift file; these are called through the controller.

The view (contained the XIB files) can be changed; different outlets can be enabled or disabled or removed completely and the functionality associated with each outlet can be altered without changing things in the model; e.g. we can call the ImportFiles importMediaFiles function on window load, or via a button, or via a menu item; using these different calls doesn't change the functionality, and we don't have to use different methods to import the files.

Bindings

The TableView is bound to the add feature of the Array Controller to enable the content of the files encoded in the JSON to be shown in the GUI. The import function loads newly created File instances into the Array Controller's NSMutableArray, and instances of these files are bound to the TableView's cells; these can then be added to the table when the "Add Files" button (connected to the Array Controller's :add function) is clicked.

Features Implemented

Keyboard Controls

- Zoom in: ⌘+ or ⇧⌘= (Command+Shift+= or Command++)
- Zoom out: ⌘- (Command+-)
- These can be used instead of clicking the zoom in/out buttons

Main Menu

Under the File tab, there are options to call the Import and Export functions, to manually import files or to export them to an external JSON file. The About page can be accessed under the Assignment2 -> About Media Library App tab.

Main Window

Files are imported automatically on load, and clicking on the Add Files button underneath the table on the left side of the view will load the files. Clicking on a file entry in the table will open it in the main view, and display its metadata and an area where it is possible to record notes on the file.

On the right side of the document is the search field, where typing in a search term will narrow down the scope of files with filenames that match the term; this includes extensions of files.

About Window

We created a new controller and xib file to separately the functionality of the about window. The about window can be open from the “About Media Library App” in the main menu item.

Resources Used

AVKit and AVFoundation: to use functions associated with playing video and audio files

Aspects Not Working

- The text file content is currently not displaying in the outletTextFileView TextView; this is probably due to the content not being extracted correctly.
- When the application resizes, the box on the right and the controls on the far right side resize much more than the contents of the rest of the application, and the central content-displaying part of the window does not resize horizontally at all.
- Our app’s name in the XIB file display is “Media Library App”; however, when the application is run, the menu bar continues to display “Assignment2” even though we changed the name in the XIB file. We think this is because our project name is “Assignment2” and that the application displays this name in the menu bar regardless of changes in the XIB file.

Pair Work

Sweta: about page, zoom in/out, import/export functions and binding, GUI design.

Megan: report, implementing media content display (including earlier versions of display not included in final version), import/export functions, GUI design.