Class Extractor

Functional Specification

Elliot Liskin

Last Updated: October 25, 2015

Overview

Class Extractor is a study tool for students that allows them to recall, prioritize, and organize a large amount of information gathered over the course of a semester in a class with the goal of enabling more effective and more efficient studying.

This functional specification is neither complete nor finished, as it is wholly expected that this document and the program’s requirements will change and evolve as this semester and the next semester progress. The purpose of this document is to lay out the specifications and requirements as understood now.

Scenario

Imagine a college student, Joe, who is in five classes this semester. Joe is a good student, attending lecture, diligently taking notes, and completing all assignments on time. However, as finals loom towards the end of the semester, he is overwhelmed with the sheer amount of information that must be recalled. Joe took notes, but it is possible he might have missed some important ideas discussed in class. The textbook is a dauntingly large resource to look through, and the lecture PowerPoints that were posted online are missing lots of information.

If Joe had used Class Extractor, he could have had all of the information of the course right in front of him. He could have recorded all of his lectures, synthesized that information with what he typed in his notes and what was mentioned in the PowerPoint slides, and had it presented nicely to him in an easy-to-read, easy-to-navigate format that would have facilitated the studying process.

Non Goals

This version will not support the following features.

* + - Offline audio-to-text transliteration (requests must be made to IBM’s servers for transliteration to work)
    - Track topics and compare topic frequencies and importance across lectures (Class Extractor will only be able to track topics within a single lecture, as opposed to over the course of all lectures in a class)
    - In-app creating, editing, or viewing of presentations (while audio recordings and textual notes can be created in this version, support of that for presentations will not be)
    - Meeting the Mac App Store submission requirements (meeting these requirements is often a focus of Mac app design, but it is unimportant for the sake of this version)
    - Support for platforms other than OS X

User Interaction Model

When the user opens the application, they are presented with a splash screen that displays the name of the app and gives the user the option to create a new lecture or open an old one. A lecture, in Class Extractor, is equivalent to a new document in Word or a new presentation in PowerPoint; it is the basic unit of work that the user interacts with. If the user selects to open an old lecture, they are presented with a FinderKit dropdown to navigate to their lecture file of choice (FinderKit is the dropdown pane that is seen when saving and opening documents in Mac apps). If the user selects “Create a New Lecture,” the splash screen fades out and a new window fades in that allows the user to add a new recording, add a text document, add a presentation, go back to the splash screen, or start extracting.

This window is rectangular, with its width greater than its height, has a title of “Add to Lecture,” and is laid out such that the “Add Recording” button is on the far left, the “Add Notes” button is in the center, and the “Add Presentation button” is on the right, with the back button placed in the bottom left corner of the window and the “Start Extracting” button in the bottom center. This window acts as the main hub for this lecture; Class Extractor works by using these three prongs (recordings, textual notes, and presentations) to calculate topic importance, so organizing the interface of the app in this way is only natural. If the user clicks on “Add Recording,” the “Add Notes” and “Add Presentation” buttons become translucent and slide over to the right to make room for two new buttons, “Record New Audio” and “Add Other Recording.” These buttons slide out to the right from the original “Add Recording” button, fading in as they do so. Clicking on “Record New Audio” allows the user to start a recording session, in which a professor’s lecture can be recorded in-app, while “Add Other Recording” opens a FinderKit dropdown that allows the user to select a recording to use from elsewhere in the filesystem. If the user clicks on “Record New Audio,” a new window fades in that allows the user to see the recording session. It is a simple window, with a big button on the left that has a microphone on it and says “Start Recording,” and an audio meter to the right of the button, allowing the user to gauge the sound level. When the user starts the recording, the application starts recording audio and the button switches to a “Pause Recording” button. On the first window, the “Record New Audio” and “Add Other Recordings” buttons fade out, the “Add Recording” button decreases its alpha (to indicate they can no longer be accessed), and its text changes to “Recording in Progress” (the alpha value of a user interface element is a spectrum of translucency; a value of zero means the element is completely transparent, whereas a value of one is completely opaque). The “Add Text” and “Add Presentation” buttons increase their alpha back to one and slide back to their original positions. If the user pauses the recording, the button changes its text to “Continue Recording,” and a new button fades in to the right of that button with the text “Done Recording.” Clicking “Continue Recording” makes the app start recording again and the “Done Recording” button fade out. If the user presses “Done Recording,” the recording window fades out and the “Recording in Progress” button on the original lecture window changes to “Recording Added.”

Clicking on “Add Text” on the “Add to Lecture” window leads to a similar series of options and interaction elements as adding a recording. “Add Recording” and “Add Presentation” decrease their alpha values and slide off to the left and to the right, respectively. To the left of the “Add Text” button fades in a “Create New Text” button, and to the right fades in a “Add Other Text” button. Clicking “Add Other Text” drops down a FinderKit pane, while clicking “Create New Text” opens a new text editor window that allows the student to type notes. Similar to what happened with the recording buttons, the text buttons on the original window change to “Text Editing in Progress” while the user is typing notes. At the top of the text editor, there is a toolbar that has a button on the far left that says “Done Typing.” When the user clicks this button, the two side “Add Text” buttons fade away and the middle button changes to say “Text Added.”

If the user clicks on “Add Presentation,” a FinderKit pane drops down and allows the user to select a presentation from their computer. Upon completion, the “Add Presentation” button text changes to say “Presentation Added.” As noted above in the Non Goals section, creating, editing, and viewing presentations will not be supported in this version of the app, so the only feature that needs support is adding a presentation.

Before adding a recording, text, or presentation, this “Extract Now” button has a lowered alpha value (such as around 0.5) to indicate that it cannot be pressed, but once one of those is added the button increases its alpha to one and can be pressed. Once the button is clicked, an indeterminate progress view (a circle that spins indicating that something is loading) appears over the button. It is at this point that Class Extractor sends the audio to IBM’s Watson for transliteration and categorizes topics and determines the importance of each one. When Class Extractor is finished processing all of this information, the “Add to Lecture” window fades out and a new window fades in with the title “Lecture Overview.”

The lecture overview uses a tab navigator (these are often what are found at the bottom of iPhone apps that allow the user to switch between various screens, such as in the Facebook app) to display two different interfaces: a word cloud and a timeline. The word cloud shows an overview of all of the major topics in the lecture, with the size of a word’s cloud being a function of how important it is in the lecture. Clicking on a cloud causes the interface to zoom into the cloud, with new clouds appearing for subtopics of the clicked topic. In each cloud is also other information about that topic, such as links to the exact spots in the recording, text, and presentation that it can be found.

If the user switches to the other interface, the timeline, using the tab bar, the app will present a timeline with a mapping of each topic, and each subtopic, to its corresponding place in the audio recording. A long bar with signify the audio recording at the bottom of the window, and each topic will be a bar that extends from when it is first discussed to when it is no longer being discussed in the lecture, allowing for topics to overlap. Each subtopic will be positioned vertically above its parent topic. The audio recording can be listened to right in this interface.