Senior Design Engineering Notebook:

Speech and Language Transcriber (SaLT)

William Hundley

**3 October 2024**

Today we met with the main goal of getting everyone’s personal computer to run the current version of the project. My mac already has the project running successfully, so I was giving guidance to others and also testing out the progress on the different branches of the GitHub. Originally, I was using the main branch, but it turns out that the most recent developments are on the ‘lm-dev’ branch. Below is a screenshot of the project running on my computer.

**A screenshot of a video chat

Description automatically generated**

I spent the last 30 minutes of the period reading documentation given to us by Dr. Liu on Dash, which is a framework for building apps in Python. The document I started reading is titled ‘py-gui-dash-mg.md’.

**5 October 2024**

I spent about 1 hour reading more about Dash. I looked at the other resource given to us by Dr. Liu titled ‘py-gui-dash-mr.md’. I started to realize that the documentation given to us was a little dense and hard to understand so I did some outside research. I did like a 30-minute tutorial on the Dash website that was interactive and showed me some basics, but it wasn’t very relevant to the code for our project. Here’s a little snapshot from the tutorial:

A screen shot of a computer program

Description automatically generated

After getting a brief introduction to Dash, I moved onto tutorials on pydub, which the project using to manage and manipulate audio. I glanced over the notes that Dr. Liu sent but I learn much better through video/walkthrough tutorials. I watched two videos:

* https://www.youtube.com/watch?v=B31RiiRt\_TE
* https://www.youtube.com/watch?v=x19qoZ7OayA

Then I went back and looked at the current version of SaLED to see if I recognized anything I learned in my research. I definitely felt more comfortable, but at the same time I was lost in the code and started realizing that a lot of stuff was hardcoded into the GUI.

**7 October 2024**

I read the documentation on Whisper AI provided by Dr. Liu called ‘finetune\_whisper\_atc0\_all.md’. This was kind of interesting because it was notes related to our specific project’s usage of Whisper. It basically showed and explained how they trained the AI using an ATC dataset. I didn’t really learn that much because it was pretty dense, but it was a nice introduction. I also watched this quick 20-minute video on some guy building an audio transcription app using Whisper which was pretty helpful to see him code in real-time: https://www.youtube.com/watch?v=cNLXzXyuzUs.

**Hurricane Milton Evacuation October 8 – 12**

**13 October 2024**

I began writing requirements for the audio playback that Ziad was working on. Below are the 10 requirements I wrote:

* [REQ-1] The system shall allow the user to load a selected .mp3 file.
* [REQ-2] The system shall display a play button that, when clicked, begins playback of the .mp3 file from the start.
* [REQ-3] The system shall visually indicate that the audio file is playing, such as by changing the play button to be shaded a darker color.
* [REQ-4] The system shall allow the user to pause playback of the .mp3 file at any point by clicking the pause button.
* [REQ-5] When the .mp3 file is paused, the system shall store the current timestamp to allow resuming from that exact position.
* [REQ-6] The system shall allow the user to resume playback from the paused position by clicking the play button after a pause.
* [REQ-7] The system shall ensure that the audio playback resumes within 0.2 seconds of clicking the play button.
* [REQ-8] The system shall provide a restart button that allows the user to reset the .mp3 file to the beginning.
* [REQ-9] When the restart button is clicked, the system shall reset the playback to the 00:00 timestamp and play from the start.
* [REQ-10] The system shall allow the user to stop playback of the .mp3 file entirely by clicking a stop button.

I need to run these requirements by Ziad to see if he has any feedback or issues. The goal was just to get a jump start of the requirements document that we were warned about.

**14 October 2024**

Today we learned that we still need to submit an update presentation. We started ours before the hurricane but there was a miscommunication about it being canceled so we never finished. I worked on revising it. Originally, we just had a bunch of diagrams that were made without many words. Below are some of the slides I completed or contributed to.

A screenshot of a computer

Description automatically generated

A screenshot of a graph

Description automatically generated

A screenshot of a white background

Description automatically generated

A white background with black text

Description automatically generated

**16 October 2024**

Most of the current sprint is working on the SRS document, so I wanted to write some more requirements. I added the following requirements to the document:

* [REQ-11] Upon stopping, the system shall reset the timestamp to 00:00 but not resume playback unless the play button is clicked again.
* [REQ-12] The system shall provide an option to load .mp3 files from the user's local storage.
* [REQ-13] Once an audio file is loaded, the system shall verify that it is in the correct format (.mp3) before allowing playback.
* [REQ-14] The system shall apply a consistent visual effect (e.g., color change or highlighting) when the user hovers over any playback control (play, pause, stop, restart) to indicate interactivity.
* [REQ-15] The system shall load the waveform of the selected .mp3 file once the user selects a valid audio file from their local storage.
* [REQ-16] The system shall verify that the selected file is in .mp3 format before attempting to load and display the waveform.
* [REQ-17]  If an invalid file format is selected, the system shall display an error message and prevent waveform loading.
* [REQ-18] The system shall ensure that the displayed waveform represents the entire duration of the audio file, including time markers or timestamps for key intervals.
* [REQ-19] If the system fails to load the waveform (e.g., due to a corrupt or invalid file), it shall display an error message and reset the interface to allow the user to select a new file.

**17 October 2024**

I continued to write more requirements for the SRS.

* [REQ-20] The system shall ensure that the waveform is fully cleared from the display when a new audio file is selected or when an error occurs during loading.
* [REQ-21] The system shall allow the user to select a section of the waveform by clicking and dragging over a portion of the waveform display.
* [REQ-22] The system shall visually highlight the selected section of the waveform (e.g., by shading the selected region or changing its color).
* [REQ-23] The system shall display a vertical spectrogram along the left side of the screen corresponding to the selected section of the waveform.
* [REQ-24] The system shall dynamically update the spectrogram in real time as the user modifies the selected waveform section (e.g., changing the start or end points).
* [REQ-25] The system shall provide an editable text area within the web app where users can view and modify the transcribed text.
* [REQ-26] The system shall provide a "Save" button that allows the user to save the edited transcription to a file on their local computer.
* [REQ-27] The system shall allow the user to select the file name and destination folder for the saved file using a standard file explorer interface.

**22 October 2024**

I worked on some of the introductory sections of the SRS. I wrote the below section for 1.1. Purpose and Scope.

A document with text on it

Description automatically generated

I also contributed to sections 1.4.1. Definitions and 1.4.2. Acronyms.

**23 October 2024**

I once again worked on the SRS, this time contributing to the introductory sections again. I wrote a draft of the section 1.2. Intended Audience as seen below:

A document with text on it

Description automatically generated

I also wrote a few more requirements as seen below:

* [REQ-30] The system shall display a red vertical line that moves across the waveform in real time as the audio plays.
* [REQ-31] The red line shall begin at the start of the waveform (timestamp 00:00) and move progressively toward the end, corresponding to the current playback position.
* [REQ-32] The system shall ensure that the red line moves smoothly and consistently, without stuttering, as the audio progresses.
* [REQ-33] When the user pauses the audio, the red line shall stop at the exact position corresponding to the timestamp of the paused audio.

I need to start considering breaking these requirements up into the different sections provided in the template for the SRS. Right now, I am just brainstorming them as I play around with the application and read the backlog items. I have also been referencing the project proposal.

**24 October 2024**

Today in class we worked on finishing up the sprint and building the sprint 4 backlog. I also spent time setting up the references folder on our GitHub. The reason for doing this is that many of the documents provided by Dr. Liu are not public, so we need a way to incorporate them into our SRS document under section 1.3. References. Putting them in the GitHub allows me to put links in the document and also maybe a section in the README. I also wrote a handful more requirements:

* [REQ-34] If the user restarts or seeks to a different position in the audio, the red line shall jump to the corresponding position in the waveform immediately.
* [REQ-35] The system shall display the current audio timestamp, showing the time in mmformat (e.g., 01:15 for 1 minute and 15 seconds).
* [REQ-36] When the user manually moves the red line by interacting with the waveform, the system shall display the updated timestamp immediately.
* [REQ-37] The system shall provide a "Preview" button that allows users to play only the selected portion of the audio waveform before saving it.
* [REQ-38] The system shall provide a "Save Clip" button that allows users to save the selected portion of the audio waveform as a new .mp3 file.
* [REQ-39] The system shall prompt the user to specify a file name and destination folder for the new .mp3 file using a file explorer interface.
* [REQ-40] If an error occurs during the clip saving process (e.g., insufficient storage space or file permission issues), the system shall display an error message and allow the user to retry saving the clip.

Today is also the first day that I uploaded this engineering notebook to the GitHub. We added to the sprint 4 backlog items tasks related to the timestamp lines (on main waveform and local spectrogram) and the fast forward and rewind buttons, so I am excited to hopefully start coding and spend a little less time playing catchup on documentation.