EECE 2140 — Speech-to-Text Notes App

Fall 2025

Iteration #03

Prepared for: Dr. Fatema Nafa

Team: Cassidy Sakamoto | Justin Glabicki | Nathan Tan

Due Date: November 28, 2025

Summary of Team Progress & Development Updates

Goal. Build an intuitive notes app that captures microphone input, transcribes to text in real time (with an offline fallback), and lets users edit, search, filter, and export notes.

Milestones

Target	Planned Scope and Status
First Draft (Oct 26)	Basic UI and speech-to-text recognition; GitHub repo and CI set up; contributions from all members; consultation with Dr. Fatema Nafa; working *record* → *transcribe* → *display* pipeline.
Second Draft (Nov 9)	Accuracy improvements; robust error handling (no input / no internet); note storage and UI recording controls; noise control and mic visualization (bar + "Too loud/Too quiet"); storage logic (timestamps, metadata, local save, export); in-app editing; search enhancements (keyword/phrase); filtering by date/tags.
Final Draft (Nov 24)	Accurate, low-latency transcription; clear recording indicator and partial text during capture; comprehensive error handling and mock tests; editable, easily retrieved notes in UI. Deliverables: README, demo video, architecture diagram, GitHub submission.

Description of Implemented Core Features

General

- Microphone capture via `sounddevice`; real-time callback pipeline.
- Interactive UI for recording, accessing, and editing notes.

Specific

- Offline transcription (fallback model).
- Start/Stop recording; save notes with timestamps & metadata.
- Keyword/phrase search across notes.
- Filter by date or topic tags.
- Export to `.txt` or `.md` with metadata headers.
- Noise gate and mic-level meter (real-time visualization).

- Settings for language, model, hotkeys.
- Tested file I/O and search functionality.
- Optimized for real-time feedback (chunked pipeline).
- Unit tests, error handling, and mock tests.
- **Stretch:** topic tagging via TF–IDF.

Leadership Rotation & Individual Contributions

Leadership Rotation

- Week 1 (Starting after 9/22): Justin Glabicki
- Week 2: Cassidy Sakamoto
- Week 3: Nathan Tan
- Rotation continues changing weekly in the same order.

Team Objectives

- Build an accurate, usable speech-to-notes application.
- Provide editing, saving, exporting, and reliable retrieval.
- Ensure the app is intuitive and fully operable via the UI.

Roles and Focus Areas

Member	Primary Contributions
Justin Glabicki	*UI Recording:* Start/Stop controls; errors (permissions, missing input, no internet); offline STT enablement. *UI Editing & Search:* Keyword/phrase/date/tag search; editing of notes; autosave (500 ms idle). *Saving & Export:* Export to `.txt`/`.md` with metadata; filename from note title.
Cassidy Sakamoto	*Audio Input:* Microphone capture and buffering using `sounddevice.InputStream`. Callback handles noise gate and mic-level meter; forwards chunks to transcriber. Chunked processing (10-chunk batches) with an `AudioBuffer`.
Nathan Tan	*Real-Time Processing:* 20 ms frames aggregated into 2 s chunks; partial transcripts per chunk; timing tracked. *Reliability:* 16 kHz mono PCM standard; input/output checks; three retries per chunk; domain dictionary replacements; lightweight punctuation.