VCU College of Engineering

346 AI/Linguistics
Project Proposal

Prepared for

Clinton Farrell

DoD

By

Nathan Devore

Allen Lee

Connor Kohout

Nate Eldering

Under the supervision of

Tamer Nadeem

18 October 2024

Executive Summary

The Army needs to have clear communication in high risk environments that often are surrounded by heavy background noise, multiple speakers, and disruptions. The solution to ensuring that clear communication is a speech to text software that utilizes artificial intelligence to display in text what is being spoken in real time. The end goal for this project is to have a user-friendly interface that uses the software we design to in real time transcribe disruptive audio and provide an accurate transcription with low latency. With the support of OpenAI's whisper model our team plans to install the model on VCU's HPRC nodes and train it on a dataset we will create that will optimize the model to support the background noise, vocabulary, and accents encountered in Army environments. After doing the necessary research on what is available in today's market our team has decided that this is the best solution for us as it is reliable and the most cost effective plan. We plan to by the end of the first semester, have the whisper model installed and running on the schools athena cluster which will allow us to remotely access this service and collaboratively work on the project together through software development tools like VSCode and GitHub. Additionally by the end of the first semester we plan to obtain a set of training data from our project sponsor and begin training our installed model with this data. We will need to ensure that the data is correctly labeled and contains sufficient noise for the model to be able to accurately optimize to the project specific environments we want to account for. If this is not the case naturally we will design and document a process to manipulate the audio so it functions the same as naturally noisy audio. In the second semester we plan to begin testing our trained model, we will focus on ensuring it transcribes audio effectively (noise levels up to 100 db and accuracy threshold of 82%). We will also test the model for its ability to transcribe military specific vocabulary and various english accents with a maximum word error rate of 18%. Through the development of our model we will follow the appropriate codes and standard to ensure that our software provides the text transcription securely and only to authorized persons and that it is rigorously tested and a representative variety of conditions.

Table of Contents

Section A. Problem Statement

Automatic caption generation has become increasingly essential due to today's current landscape of remote work and online communications, the technology to support this has also made great advancements in the recent years since COVID-19. However, existing solutions still face significant shortcomings, especially in challenging environments characterized by disruptive noise such as ones seen in military environments. Clear and precise communication is essential in high risk environments and this can be aided with the help of speech to text technology.

The technologies that exist today use a variety of algorithms to enhance audio clarity and minimize background noise, some work in the time domain, treating the audio as a waveform and performing operations to optimize the quality of the desired sound by isolating it from other background sounds. Others function in the frequency domain, identifying which frequencies are present in the audio and focusing on the primary sources while attempting to enhance their quality by reducing unwanted frequencies. Although these algorithms have made significant strides they still fall short in providing optimal performance in noisy environments. Furthermore, the effectiveness of models using these algorithms heavily relies on the quality of their training datasets. Acquiring well-labeled, large-scale datasets that accurately represent various types of noisy audio can be a substantial challenge. This scarcity of high-quality training data complicates the training process, hindering the model's ability to generalize and perform reliably in real-world applications.

The client for this project created it with the idea of it being used in military aviation settings, for example a helicopter in which communication is through radio, it is important for that radio transmission to be clear and the help of automatically generated captions can aid this process. Additionally, in the setting of military air travel, the vocabulary and dialect differs from that of everyday conversation enough to where it is necessary to take into account when designing a software to turn this text into speech. For example the NATO phonetic alphabet is commonly used, these words when used out of context could create confusion for a model that is expecting something else but with the correct architecture and training it is possible to learn what these words mean and when they should be used. Additionally, accents are very prevalent in the army population and this also should be accounted for. A study done by Qxf2 (2023) to test the accuracy of OpenAI's Whisper model on detecting words with various English accents was done and found that for many different accents like Arabic, French, and German there was a low word error rate. For accents like Polish, Turkish, and Russian the word error rate was high. While there was no correlation drawn between which accents performed better than others these results are indicative that with the correct training the model has the ability to understand the English language through heavy accents.

Collecting valuable labeled data for testing is the main cost of this project, ensuring that the audio is distorted in an accurate way, whether that be collected in real-world scenarios or modified after collection to appear more distorted. OpenAI's Whisper model, which was trained on a diverse range of languages and types of audio interference, has proven to be more robust without the need for dataset-specific fine-tuning to achieve high-quality results (Radford et al., 2023). This is relevant to our project, as we plan to use dataset-specific fine-tuning with the goal of our system being specialized for military environments and the paper supports the idea that this approach will be effective. The whisper model will be the main one used in this project as it is open source and provides a range of sizes, the smaller models perform very quickly while taking in a smaller number of parameters and producing output with less accuracy than the larger models which can take more parameters and perform slower. This range will be helpful as we gauge which features are more important in the context of this project, whether that be a faster or more accurate transcription.

Services like Microsoft and Google offer paid options that support fast live transcription in a large variety of languages. Google Cloud Speech-to-Text is customizable and allows users to adjust the domain of their speech recognition to specific modes like phone calls or video calls. Additionally this model offers the option to adjust the expected vocabulary so in specific settings similar words will be given higher priority. Google's model is good at handling pre-processed audio as well as live transcription in a variety of languages. This model has been used extensively across industries for applications such as transcription, voice commands, and customer service automation. However, it has limitations in extremely noisy environments, such as those found in military or industrial settings, where background noise can significantly degrade performance. Although the service includes features like noise-canceling algorithms and speaker diarization, achieving robust performance in environments with intense noise, such as helicopters or heavy machinery, remains a challenge. Microsoft's service offers similar customization options while also suffering from similar shortcomings such as their dependence on an internet connection and their service fee. Due to these limitations we decided they were not the best direction to move forward with for this project.

According to Keller (2010), the DARPA Robust Automatic Transcription of Speech (RATS) development program was launched with the goal of determining speech activity, identifying speakers, and spotting keywords in highly degraded audio environments. The system created through the RATS program took communication channels and put them into a system of four sub systems differing in their voice activation detection (VAD) algorithm (Thomas, Saon, Van Segbroeck, & Narayanan, 2015). The combination of these VADs and their respective focuses resulted in a robust model that was very good at handling communication channels filled with both stationary and dynamic noise. In the general field of automatic speech recognition, the RATS program is significant for its emphasis on robustness in extreme noise conditions, an area where traditional ASR systems struggle. The program's innovations, including the use of diverse acoustic features and deep neural network models, have laid a foundation for further advancements in speech recognition under adverse conditions.

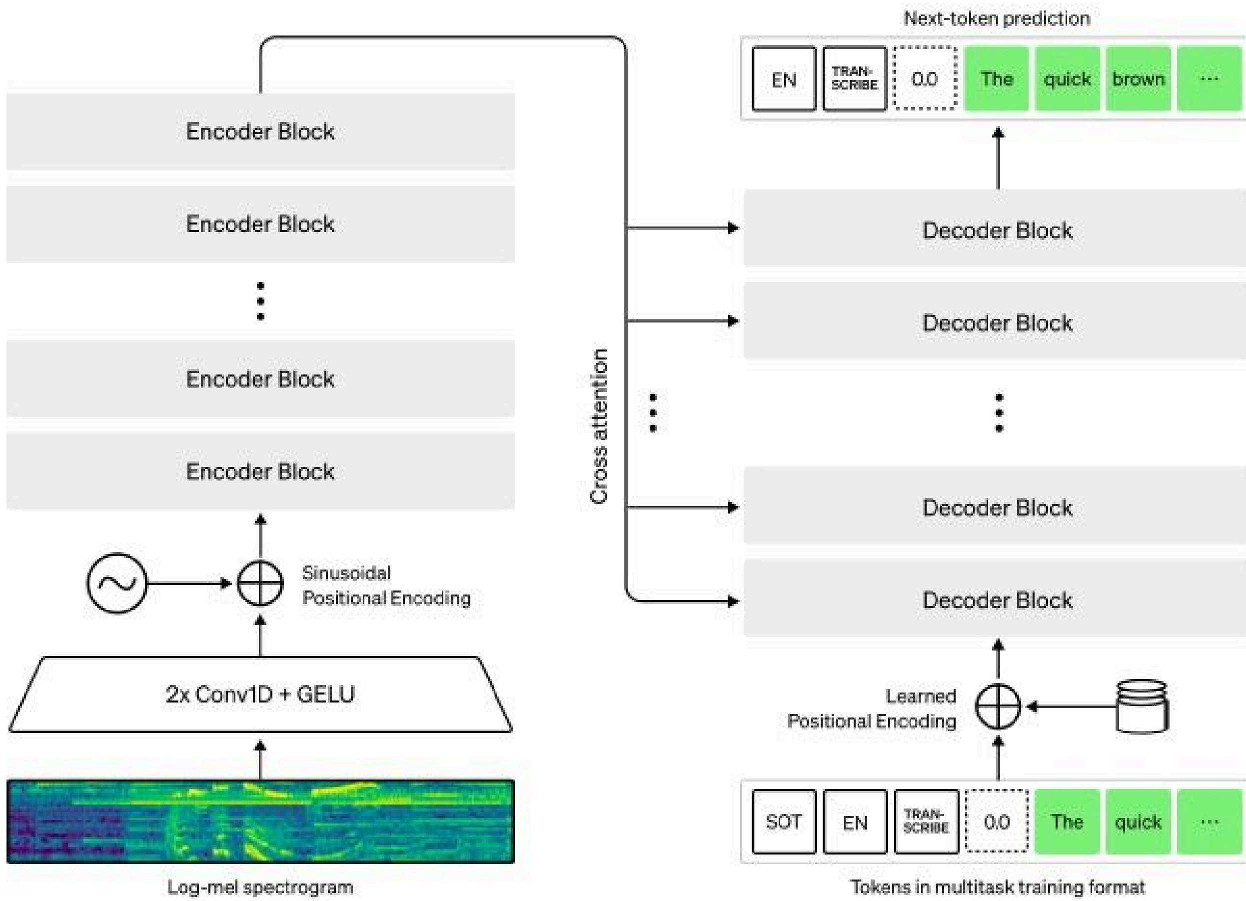


Figure 1. Open AI's Whisper model architecture

Section B. Engineering Design Requirements

B.1 Project Goals (i.e. Client Needs)

Create an automated solution to accurately caption live audio in military scenarios. Current systems for speech recognition do not effectively address the unique challenges presented by the military environment, such as background noise, diverse accents, and specialized vocabulary. Therefore, the main goal is to create a proof of concept that shows it is possible to tailor existing speech recognition models for this application.

- Develop a system to automatically caption live audio in military settings.
- Ensure the system is robust against various types of background noise (e.g., helicopter sounds).
- Address challenges such as regional accents and English as a second language for a large portion of personnel.
- Incorporate military-specific vocabulary and protocols (e.g., phonetic alphabet and repeated communications).
- Achieve accuracy comparable to current systems used in civilian applications, tailored for military use.

B.2 Design Objectives

The design will focus on specific measurable objectives that align with the needs of the military environment. These objectives ensure the model developed can function under different conditions and accommodate the unique needs of the military.

- The design will transcribe audio with at least 82% accuracy in noisy environments.
- The design will correctly interpret and transcribe military-specific vocabulary.
- The design will process speech with varying accents and dialects common among military personnel
- The design will allow for real-time transcription with latency
- The design will support transcription accuracy tests with audio from various noise levels and conditions.

B.3 Design Specifications and Constraints

The system will be required to meet certain technical constraints relating to noise levels, processing power, and speech recognition accuracy. These constraints are essential for ensuring that the system can work under real-world military conditions and handle the varied demands of this environment.

The design will focus on testing and optimizing the model under the following constraints:

- **Noise environment constraints**: The system must function in environments with noise levels up to 100 dB (helicopter noise) and still maintain an accuracy threshold of 82%.
- **Accent and dialect constraints**: The model must be able to transcribe speech from personnel with southern U.S. accents, as well as non-native English speakers, with at least 82% accuracy.
- **Vocabulary constraints**: The system must recognize and transcribe military language, phonetic alphabets, and repeated communications without exceeding a word error rate of 18%.
- **Latency constraint**: Transcription of live audio must have a latency of no more than 2-3 seconds.

B.4 Codes and Standards

There are no specific codes that need to be followed, however there are standards.

- ISO/IEC 27001 - Captions should only be accessible to authorized persons and systems.
- ITU-T P.800 - The accuracy of the system should be tested under various conditions.

Section C. Scope of Work

Project Scope:

The primary objective of this project is to develop an audio transcription system capable of accurately transcribing audio files, even in challenging environments with significant background noise. The system will leverage the Whisper model, running on a cluster, to process and analyze audio files efficiently. The project aims to develop solutions that enhance the model's performance under conditions such as loud sounds, white noise, or other disruptive environments.

Key Objectives:

1. Implement the Whisper model on the school's cluster.
2. Process audio files and transcribe them with a focus on noisy data.
3. Train the model to handle diverse audio conditions and voice distinctions.
4. Ensure the system can recognize and differentiate between multiple speakers in an audio clip.
5. Develop a user-friendly interface or tool for testing and validating transcriptions.

Timeline & Milestones:

- **Phase 1:** Initial setup of the Whisper model on the cluster. Completion of setup, including all dependencies such as ffmpeg and Rust (End of Month 1).
- **Phase 2:** Gathering and preparing audio data for model training, potentially synthesizing additional data (Month 2).
- **Phase 3:** Testing transcription accuracy under varied conditions, including environments with high noise levels (Month 3).
- **Phase 4:** Model optimization and refinement based on test results. Implementation of voice distinction capabilities (Month 4).
- **Phase 5:** Final testing and deployment of the transcription tool for user interaction (Month 5).

Responsibility of the Team:

- Set up and maintain the Whisper model on the cluster, ensuring smooth operation.
- Gather and preprocess the required audio data.
- Train, test, and refine the model.
- Collaborate with the project sponsor and faculty advisor to ensure timely progress and meet deliverables.
- Maintain regular communication and provide updates on the project's status.
- Develop user documentation and a final report summarizing project outcomes.

Exclusions:

- This project does not involve developing audio recording hardware or other tools outside the transcription scope.
- The system is not responsible for handling real-time audio processing or live transcription beyond testing datasets.

C.1 Deliverables

In order to mitigate risks associated with the completion and delivery of the project deliverables, provide an outline of the most potentially disruptive, foreseeable obstacles. Some important issues to discuss with the design team, sponsor, and faculty advisor include the following:

- What deliverables require access to campus? Which/how many students regularly access campus and are physically available to complete tasks?
- What work can be done remotely? What resources might be needed in order to ensure that remote work can be completed effectively (e.g. software licenses, shared drives/folders, etc.)?
- What deliverables require ordering from third-party vendors? Will any components potentially require extended lead times? What can the team do in order to mitigate potential supply chain disruptions?

The following is a list of all agreed-upon project deliverables for the audio transcription capstone project, applying the Whisper model on the cluster:

- **Whisper Model Setup:** Successful setup of the Whisper model on the school's cluster, including all dependencies (e.g., ffmpeg, Rust).
- **Data Collection & Preprocessing:** A dataset of audio files, including both clean and noisy environments, to train and test the transcription model. This may involve synthesizing additional data if necessary.
- **Trained Whisper Model:** A fully trained Whisper model, optimized to handle noisy environments in an audio file for radio voices.
- **Transcription Tool:** A functioning transcription tool with an interface for users to upload audio files and receive

- **Project Documentation:** User documentation, detailing how to use the transcription tool, as well as a technical report summarizing the model setup, data preprocessing, training process, and final performance metrics.
- **Academic Deliverables:**
    - Team Contract
    - Project Proposal
    - Preliminary Design Report
    - Fall Poster and Presentation
    - Final Design Report
    - Capstone EXPO Poster and Presentation

Risk Mitigation and Obstacles

- **Access to Campus:** The model is being run on the school's cluster, which requires access to campus resources. Regular campus access will be necessary to manage the model's training and maintenance, as the cluster resources are hosted by the school.
    - **Mitigation:** Team members who need physical access to campus resources have been identified and will coordinate their access. Remote access to the cluster is available to mitigate the need for constant campus presence.
- **Remote Work:** Most work, including data preprocessing, model testing, and report writing, can be done remotely. Shared drives and collaboration tools like GitHub will be used to manage code and documentation across the team.
    - **Mitigation:** Ensuring that all necessary software (e.g., Python, Whisper model dependencies) is installed and configured for remote access. Regular virtual check-ins will be held to maintain progress.
- **Third-Party Dependencies:** There are no components requiring third-party ordering. However, the project does rely on open-source software (Whisper, ffmpeg, Rust), which may present compatibility challenges or delays if updates or changes occur during the project.
    - **Mitigation:** The project team will monitor updates to dependencies and ensure compatibility through testing.

By identifying these potential risks and obstacles, the team is prepared to manage and mitigate disruptions to the project timeline.

C.2 Milestones

Milestone	Description	Estimated Time	Completion Date
Whisper Model Setup	Install and configure Whisper model on the school's cluster,	2 weeks	October 31, 2024

	including ffmpeg and Rust dependencies.		
Data Collection & Preprocessing	Collect and preprocess audio data, including noisy environments and synthesized data.	3 weeks	November 21, 2024
Initial Model Training	Train the Whisper model using the collected audio data to handle noisy environments.	3 weeks	December 12, 2024
Testing & Evaluation of Transcriptions	Evaluate model performance under various conditions, including multiple speakers and noisy environments.	3 weeks	January 2, 2025
Speaker Differentiation Implementation	Add functionality for recognizing and differentiating multiple speakers in an audio file.	3 weeks	January 23, 2025
Transcription Tool Interface	Develop user interface for uploading audio files and displaying transcriptions.	3 weeks	February 13, 2025
Final Model Optimization	Optimize the model for final testing and evaluation, ensuring all requirements are met.	2 weeks	March 5, 2025
Project Documentation & Reports	Complete all necessary documentation, user manuals, and final project reports.	3 weeks	March 26, 2025
Capstone EXPO Preparation	Prepare poster, presentation materials, and final demonstration for Capstone EXPO.	2 weeks	April 16, 2025

This table provides a breakdown of key milestones to guide the project. Each milestone is designed to ensure steady progress while allowing for iterative improvements, especially following the Agile approach.

C.3 Resources

The Whisper model training project primarily relies on access to the school's server cluster and the open-source Whisper software. However, certain software, hardware resources, and possible cloud services were considered in case additional computing power was required. This section outlines all project expenditures and, if relevant, any experimental setups or prototypes developed.

Since we are utilizing open-source software and libraries, the direct cost for software tools is minimal. However, potential costs for commercial or cloud-based services are also considered.

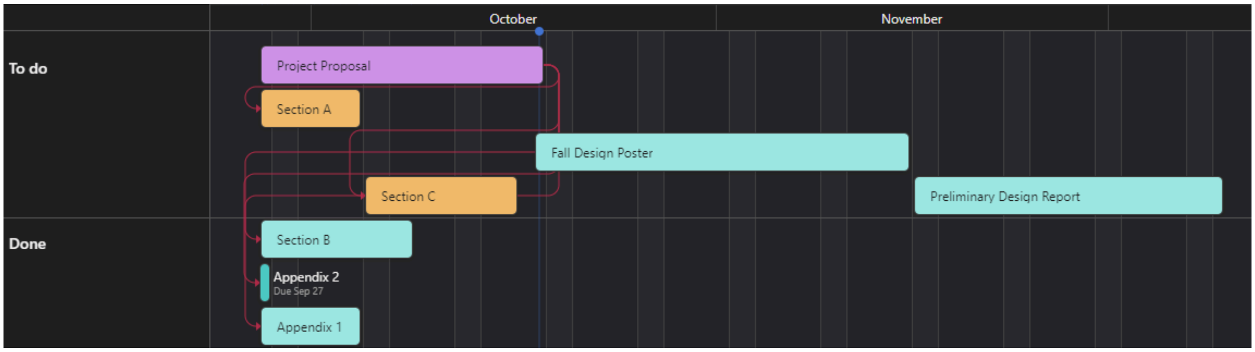
Software	Vendor	Cost
Whisper	OpenAI	$0
Python	Python.org	$0
FFmpeg	FFmpeg.org	$0
VSCode	Microsoft	$0
AWS Cloud Computing (Optional)	Amazon	$70

The project utilizes the school's server cluster for training, which incurs no direct hardware costs. However, for future scalability or more significant computation needs, we considered cloud computing resources.

We expect to receive some datasets from our advisor, which incur no direct cost. Synthesizing data may require additional storage and processing but has no direct monetary expense in the current setup.

Dataset	Vendor	Cost
LibriSpeechASR corpus	LibriSpeech	$0
Advisor Provided Data	DoD	$0
Synthesised Data	Self made	$0

Appendix 1: Project Timeline



Appendix 2: Team Contract

Step 1: Get to Know One Another. Gather Basic Information.

Task: This initial time together is important to form a strong team dynamic and get to know each other more as people outside of class time. Consider ways to develop positive working relationships with others, while remaining open and personal. Learn each other's strengths and discuss good/bad team experiences. This is also a good opportunity to start to better understand each other's communication and working styles.

Team Member Name	***Strengths each member brings to the group***	***Other Info***	***Contact Info***
Nathan DeVore	NLP, Python, Java, C, Engineering, problem solving		devoreni@vcu.edu (434) 282-8258
Allen Lee	Experience with Machine Learning, Java, C(++,#), Python in that order.	Currently in a few classes that may also help, like Artificial Intelligence and Databases. Have had some fairly unresponsive groups in the past.	leea18@vcu.edu (804) 683-8526
Nate Eldering	Communication, technical skills, problem solving, Python, Java, C	currently taking machine learning, artificial intelligence and databases.	elderingn@vcu.edu 703-935-6689
Connor Kohout	C, Java, Python, sql	currently in ML	kohoutck@vcu.edu 703-508-6386

Other Stakeholders	***Notes***	***Contact Info***
Tamer Nadeem	Faculty advisor.	tnadeem@vcu.edu
Nibir Dhar	Contact from project sponsor.	dharnk@vcu.edu

Step 2: Team Culture. Clarify the Group's Purpose and Culture Goals.

Task: Discuss how each team member wants to be treated to encourage them to make valuable contributions to the group and how each team member would like to feel recognized for their efforts. Discuss how the team will foster an environment where each team member feels they are accountable for their actions and the way they contribute to the project. These are your Culture Goals (left column). How do the students demonstrate these culture goals? These are your Actions (middle column). Finally, how do students deviate from the team's culture goals? What are ways that other team members can notice when that culture goal is no longer being honored in team dynamics? These are your Warning Signs (right column).

Resources: More information and an example Team Culture can be found in the Biodesign Student Guide "Intentional Teamwork" page (webpage | PDF)

Culture Goals	***Actions***	***Warning Signs***
Getting to each meeting on time.	- Create meetings with everyone's schedules in mind - Communicate in the Discord, posting reminders before each meeting	- If a student misses a meeting, they receive a warning - If a student continues to miss meetings, the issue will be brought to the faculty advisor
Making everyone aware of any delays in the schedule	- Keep each other informed about each person's portion of the project - Create achievable benchmarks, and communicate when they cannot be met	- Student has not contributed what they communicated they would in the weekly meeting
Keep work well documented	- maintain clean well documented code - keep work consistent with others and up to date on github	- pushes to github with no comments or explanation
Communication and understanding	- let others know if responsibilities are too much or not reasonable - adjust responsibilities if needed	- Student seems to never complete anything
consistent communication with sponsor	- meeting on zoom or live some other way	- skipping meetings or multitasking during meetings

Step 3: Time Commitments, Meeting Structure, and Communication

Task: Discuss the anticipated time commitments for the group project. Consider the following questions (don't answer these questions in the box below):

- What are reasonable time commitments for everyone to invest in this project?
- What other activities and commitments do group members have in their lives?
- How will we communicate with each other?
- When will we meet as a team? Where will we meet? How Often?
- Who will run the meetings? Will there be an assigned team leader or scribe? Does that position rotate or will the same person take on that role for the duration of the project?

Required: How often you will meet with your faculty advisor advisor, where you will meet, and how the meetings will be conducted. Who arranges these meetings?
See examples below.

Meeting Participants	***Frequency Dates and Times / Locations***	***Meeting Goals Responsible Party***
Students Only	As Needed, On Discord Voice Channel	Update group on day-to-day challenges and accomplishments
Students Only	Weekly, 6pm Thursday,	Actively work on project
Students + Faculty advisor	Once a week via Zoom, time to be determined	Update faculty advisor and get answers to our questions
Project Sponsor	Twice a month via Zoom, time to be determined	Update project sponsor and make sure we are on the right track

Step 4: Determine Individual Roles and Responsibilities

Task: As part of the Capstone Team experience, each member will take on a leadership role, ***in addition to*** contributing to the overall weekly action items for the project. Some common leadership roles for Capstone projects are listed below. Other roles may be assigned with approval of your faculty advisor as deemed fit for the project. For the entirety of the project, you should communicate progress to your advisor specifically with regard to your role.

- **Before meeting with your team**, take some time to ask yourself: what is my "natural" role in this group (strengths)? How can I use this experience to help me grow and develop more?
- **As a group,** discuss the various tasks needed for the project and role preferences. Then assign roles in the table on the next page. Try to create a team dynamic that is fair and equitable, while promoting the strengths of each member.

Communication Leaders

Suggested: Assign a team member to be the primary contact for the client/sponsor. This person will schedule meetings, send updates, and ensure deliverables are met.

Suggested: Assign a team member to be the primary contact for faculty advisor. This person will schedule meetings, send updates, and ensure deliverables are met.

Common Leadership Roles for Capstone

1. **Project Manager:** Manages all tasks; develops overall schedule for project; writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas.
   Required: On Edusourced, under the Team tab, make sure that this student is assigned the Project Manager role. This is required so that Capstone program staff can easily identify a single contact person, especially for items like Purchasing and Receiving project supplies.
2. **Logistics Manager:** coordinates all internal and external interactions; lead in establishing contact within and outside of organization, following up on communication of commitments, obtaining information for the team; documents meeting minutes; manages facility and resource usage.
3. **Financial Manager:** researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out team purchase requests; monitors team budget.
4. **Systems Engineer:** analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces.
5. **Test Engineer:** oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules; oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations.
6. **Manufacturing Engineer:** coordinates all fabrication required to meet final prototype requirements; oversees that all engineering drawings meet the requirements of machine shop or vendor; reviews designs to ensure design for manufacturing; determines realistic timing for fabrication and quality; develops schedule for all manufacturing.

Team Member	***Role(s)***	***Responsibilities***

Nathan DeVore	System Engineer Test Engineer	- Keep track of all project specifications and make sure that all aspects of the prototype meet requirements - Test prototype to ensure quality
Nate Eldering	Project Manager	- Keep everyone working on a timely consistent schedule - Ensure everyone feels safe and open to share ideas
Allen Lee	Logistics Manager	- Communicate with members of the project to coordinate meeting times - Acquiring information the team requires to continue the project
Connor Kohout	Manufacturing engineer	- Ensure that the product is made to task

Step 5: Agree to the above team contract

Team Member: *Signature: Nathan DeVore*________________

Team Member: *Signature: Allen Lee*_____________________

Team Member: *Signature: Nate Eldering*_________________

Team Member: *Signature: Connor Kohout*________________

References

Provide a numbered list of all references in order of appearance using APA citation format. The reference page should begin on a new page as shown here.

[1] VCU Writing Center. (2021, September 8). *APA Citation: A guide to formatting in APA style.* Retrieved September 2, 2024. https://writing.vcu.edu/student-resources/apa-citations/

[2] Teach Engineering. *Engineering Design Process*. TeachEngineering.org. Retrieved September 2, 2024. https://www.teachengineering.org/populartopics/designprocess

[3] Keller, J. (2010). DARPA launches RATS program for advanced speech-recognition algorithms in noisy conditions. *Military & Aerospace Electronics, 21*(4), 10. Retrieved from http://proxy.library.vcu.edu/login?url=https://www.proquest.com/trade-journals/darpa-launches-rats-program-advanced-speech/docview/338452109/se-2

[4] Dudam, R. (2023, May 26). Testing OpenAI Whisper with different accents - Qxf2 BLOG. Retrieved October 18, 2024, from https://qxf2.com/blog/testing-openai-whisper-with-different-accents/

[5] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). *Robust speech recognition via large-scale weak supervision*. arXiv. https://arxiv.org/abs/2212.04356

[6] Thomas, S., Saon, G., Van Segbroeck, M., & Narayanan, S. S. (2015). Improvements to the IBM speech activity detection system for the DARPA RATS program. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4500–4504. https://doi.org/10.1109/ICASSP.2015.7178822

[7] OpenAI. (2022, September 21). *ASR summary of model architecture* [Image]. Retrieved October 18, 2024, from https://images.ctfassets.net/kftzwdyauwt9/d9c13138-366f-49d3-a1a563abddc1/8acfb590df46923b021026207ff1a438/asr-summary-of-model-architecture-desktop.svg?w=3840&q=90