



TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

A PROJECT PROPOSAL ON
OBJECT ORIENTED PROGRAMMING WITH C++

ECHO

SUBMITTED BY:

Bibek Dhungana (078BCT029)

Bishal Panta (078BCT036)

Braj Mohan Neupane (078BCT037)

SUBMITTED TO:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
PULCHOWK CAMPUS

SUBMITTED ON:

JULY 8, 2023

Acknowledgement

We would like to express our sincerest gratitude towards the individuals who have supported us in completing this project related to object oriented programming in C++. This project would not have been possible without the support and motivation they provided.

First and foremost, we would like to thank Er. Daya Sagar Baral for his invaluable guidance, expertise, and continuous support throughout this project. His deep understanding of OOP concepts and their willingness to provide guidance and feedback have been instrumental in giving shape to our ideas. His teachings, lectures, and assignments have equipped us with the necessary skills and knowledge to undertake this project.

Furthermore, we would like to thank the Department of Electronics and Computer Engineering for granting us this wonderful opportunity to develop a project that has helped us build a strong foundation in programming and OOP principles.

We would also like to acknowledge the contributions and support of our classmates and friends who have provided valuable insights, suggestions, and discussions during the development process. Their collaboration and brainstorming sessions have significantly enriched the design and implementation of the software.

Thank you all for your invaluable assistance and for being a part of this journey.

Sincerely,

Bishal Panta - 078BCT036

Braj Mohan Neupane - 078BCT037

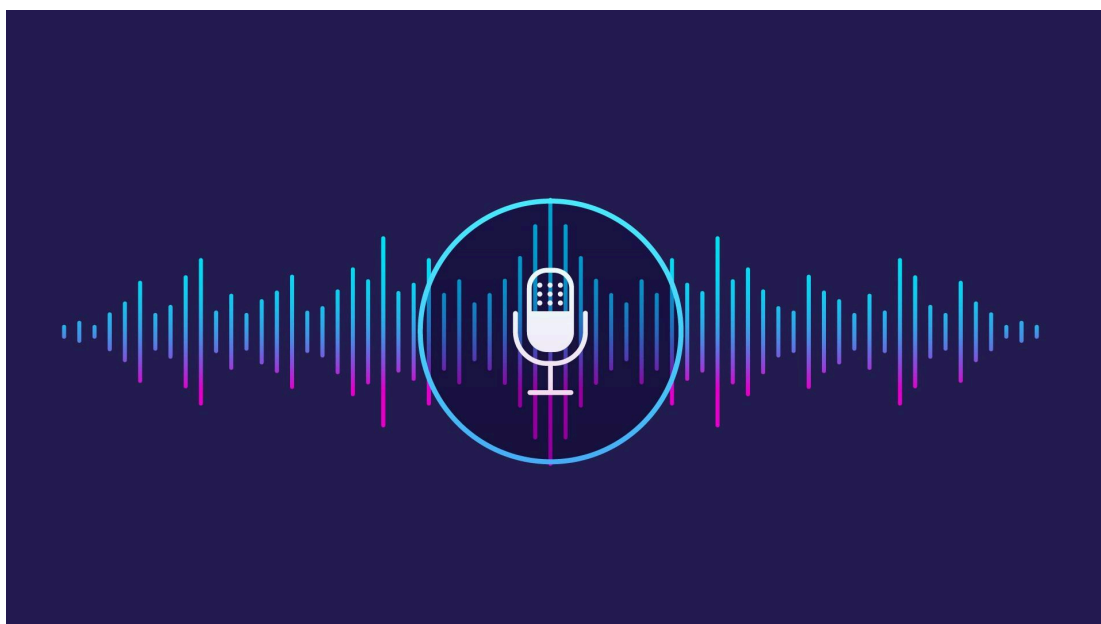
Bibek Dhungana - 078BCT029

Table of Contents

Introduction.....	3
Objectives.....	4
Proposed System.....	5
i. Description.....	5
ii. System Block Diagram.....	6
Project Scope.....	7
Methodology.....	8
Project Schedule.....	9

Introduction

Echo is an innovative application designed to revolutionize the way we interact with audio content. Echo is an advanced speech-to-text converter that seamlessly converts any form of audio/speech into a written text format, making it accessible, searchable, and analyzable. By harnessing the power of Language Model, Echo offers a plethora of features to enhance user experience, ranging from generating lyrics files and subtitles to providing real-time speech transcription and many more.



With its diverse range of features and functionalities, Echo aims to provide a comprehensive solution for converting audio into text, catering to various needs and use cases.

Objectives

This project aims to develop a speech-to-text audio transcriber using OpenAI's Whisper model, focusing on implementing object-oriented programming principles and algorithms to build a robust and user-friendly application. The main objectives of this project can be summarized as follows:

1. To gain practical experience in object-oriented principles
2. To implement key OOP concepts such as encapsulation, inheritance, and polymorphism to create a well-structured and maintainable codebase
3. To apply design patterns and architectural principles to develop a scalable and extensible speech-to-text transcriber solution
4. To gain knowledge and experience in utilizing AI models for natural language processing tasks, such as speech transcription
5. To explore and experiment with different techniques for fine-tuning and optimizing AI models to enhance transcription accuracy
6. To collaborate and learn from team members' implementations of OOP concepts, fostering a deeper understanding and mastery of OOP techniques

Proposed System

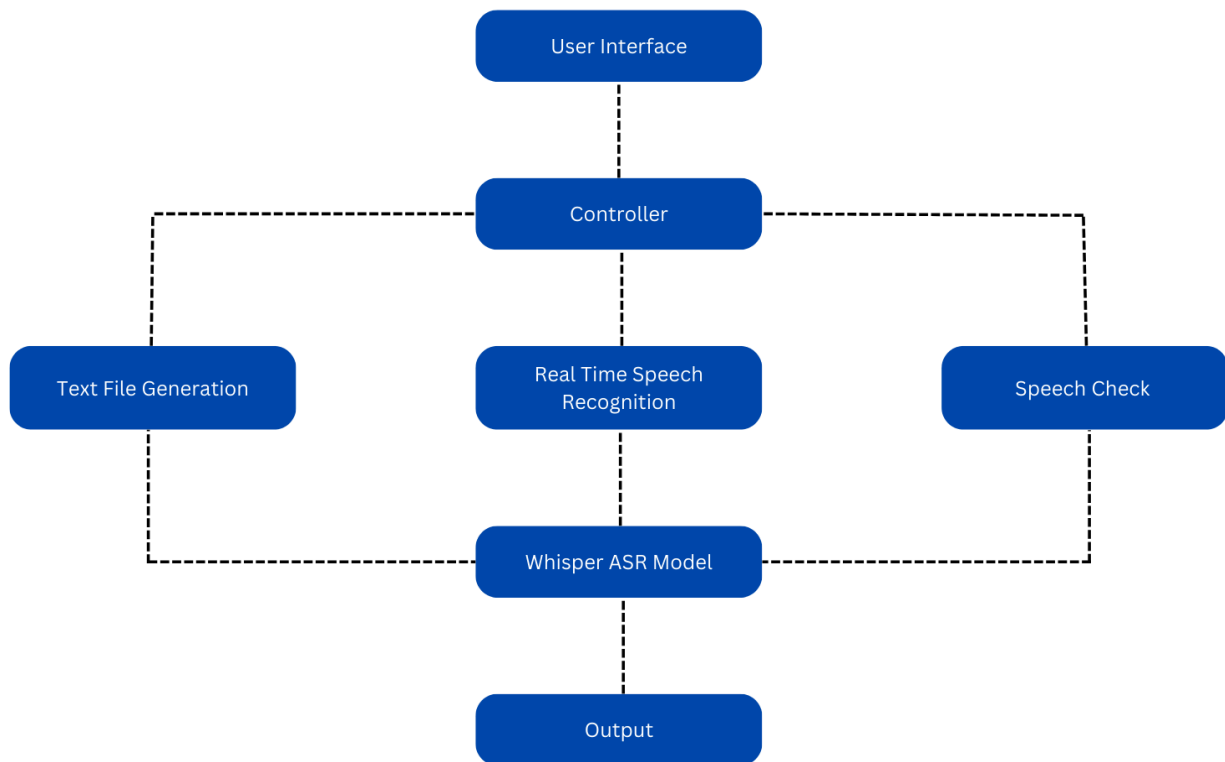
i. Description

The project aims to develop a speech-to-text transcriber powered by a state-of-the-art large language model, incorporating object-oriented programming principles and algorithms to create a robust and user-friendly application that revolutionizes the accuracy and efficiency of speech transcription. The system will leverage the advanced capabilities of the OpenAI's Whisper model to accurately transcribe spoken language into written text.

Key features

1. **Real-time Transcription:** The system will provide real-time speech-to-text conversion, allowing users to instantly view transcriptions as they speak.
2. **Song to Lyrics Conversion:** Building upon the speech-to-text capability, the system will provide an innovative functionality to convert songs into lyrics. Users can upload audio files of songs, and the system will transcribe the lyrics, allowing for easy access and enjoyment of song lyrics.
3. **Subtitle Generation:** The system will offer an additional feature for automatically generating subtitles for videos. Users can upload video files, and the system will transcribe the audio content and synchronize it with the video, producing subtitles for improved accessibility and comprehension.
4. **Support for Multiple Audio Formats:** The system will be designed to handle various audio formats commonly used, ensuring compatibility and flexibility for users. It will support formats such as WAV, MP3, enabling seamless processing of different audio sources

ii. System Block Diagram



1. The User Interface interacts with the users, providing options for inputting audio files and selecting functionalities.
2. The Controller manages the flow of data and control between the User Interface and the different modules.
3. The Real Time Speech Recognition, Text File Generation, and Speech Check modules receive instructions from the Controller based on the selected functionality.
4. All three modules converge to the Whisper ASR Model, which processes the input audio and generates the desired output.
5. The Output/Result component displays the transcriptions, converted lyrics, or generated subtitles to the users.

Project Scope

We are aiming to create an interface for everything speech-to-text with the help of OpenAI's Whisper Automatic Speech Recognition (ASR) model. We used a Port of OpenAI's Whisper model in C/C++ for this project. The Echo Project will focus on all the basic use cases for speech-to-text conversion and implement it with minimal user experience. The main scope of this project is to successfully implement Real Time Speech Recognition.

The Scopes of this Project include:

- Real Time Speech Recognition.
- Subtitle File Generation (.srt)
- Lyrics File Generation (.lrc)
- Speech extraction from various audio formats (.mp3, .wav)
- Create a Minimal UI to go along with all these features.

Constraints

Our Project shall have some constraints that will have to be overcome. The major constraint is that there is no native support for audio in C++. Our Project relies on audio processing and to have no native library for audio was surely a handicap for us. Another constraint for us was setting up the SFML graphics library for each of our workspace. Working with libraries in C++ is always a hassle, but keeping these constraints in mind we understood that this will be a great learning opportunity and with that belief we will be giving our best to this project.

Methodology

The project will employ a combination of approaches and methodologies to achieve its objectives. The following outlines the key methodologies that will be utilized:

1. OpenAI's Whisper Model: The project will leverage OpenAI's Whisper model. By utilizing the Whisper model, the project aims to achieve accurate and efficient speech-to-text conversion, enabling users to interact with the software through voice commands.
2. Object-Oriented Programming (OOP) Principles: The software development process will adhere to OOP principles, which emphasize the organization of code into reusable and modular components known as objects.

The project will follow an iterative and incremental development approach. It will involve the following key steps:

1. Requirements Analysis: This phase will involve gathering and analyzing the requirements of the software, including the desired functionalities, user interactions, and performance expectations.
2. System Design: The software's design will be established, outlining the different components, their interactions, and the overall app structure.
3. Implementation: The software will be developed using C++ programming language, following OOP principles. The Whisper model will be integrated into the software to align it with the targeted speech recognition tasks.
4. Testing and Evaluation: Rigorous testing will be conducted to validate the software's functionalities, performance, and accuracy of speech recognition.

By combining the power of OpenAI's Whisper model, OOP principles, and AI model fine-tuning techniques, the project aims to deliver a robust, efficient, and accurate speech recognition software application.

Project Schedule

The schedule that we will have adopted for our project can be summarized below:

Topic	Days Required
Requirements Gathering and Analysis	2-3
System Design and Architecture	2-3
Implementation and Integration	7-10
Customization and Error Handling	2-3
User Interface Refinement and Testing	2-3
Documentation and Finalization	2-3

The above mentioned schedule is an approximation and might change as per requirement.