AI-Powered Tool for generating blogs (Vidlexify)

Abdul Rehman-SP21307
Touseer Amir-SP21312

Supervised By

Dr. Naveed Ahmad

Submitted for the partial fulfillment of BS Software Engineering degree to the

Faculty of Engineering & Computing

NATIONAL UNIVERSITY OF MODERN LANGUAGES
ISLAMABAD
November, 2024

ABSTRACT

The AI-Powered Tool for Generating Blogs from YouTube Links is an innovative tool designed based on AI, which will generate blogs from video links of YouTube with the help of machine learning algorithms. Writing good quality content from video sources in the digital world is a time-consuming activity and requires laborious effort. This project aims to automate it by developing a blog or process for content generation through the extraction, transcription, and summarizing of content from videos.

The tool accepts a URL of a YouTube video as input. The video gets preprocessed to download the video first in (.webm) extension then download its audio followed by conversion into text. After receiving this transcription, then the ML model may be applied to study and summarize content, allowing for key ideas, themes, and messages to be caught and therefore translated into well-structured and readable blog posts.

Customization features are also part of this system; this implies the tone, length, and style of the blog can be customized according to what a user prefers. With this tool, content marketing will change as bloggers, marketers, and content creators will have ease at transforming video content into written articles within a short period of time.

The effective use of AI to automate content generation, lessen human interaction, and still hold the final output within high accuracy and coherence is, in fact the main AI-Powered Tool for Generating Blogs from YouTube Links. The impact this may bring to the workflow of digital content creators can be huge, as well as streamline their productivity, and ultimately facilitate a practical solution to repurpose content fast and reliably in a demanding environment.

CERTIFICATE

Final Approval

It is certified that project report titled "**AI-Powered Tool for Generating Blog from YouTube Links**" submitted by **Abdul Rehman** and **Touseer Amir** for the partial fulfillment of the requirement of "**Bachelor's Degree in Software Engineering**" is approved.

COMMITTEE

Teacher Name
   Committee Member 1   **Signature:**   _____

Teacher Name
   Committee Member 2   **Signature:**   _____

Teacher Name
   Head Project Committee   **Signature:**   _____

Dr. Naveed Ahmad
Supervisor   **Signature:**   _____

DECLARATION

We hereby declare that our dissertation is entirely our work and genuine / original. We understand that in case of discovery of any PLAGIARISM at any stage, our group will be assigned an F (FAIL) grade and it may result in withdrawal of our Bachelor's degree.

Group Members   **Signature**

1. Touseer Amir   _____

2. Abdul Rehman   _____

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE	CAPTION	PAGE

LIST OF TABLES

1. Introduction

Videos have now become the most amazing source of distributing information, ideas, and stories to the digital world. However, making video content transform into text content, like blogs or articles, can enhance the reach and accessibility of video content. This project is aimed at developing a workflow to convert YouTube videos to some quality blog posts.

The process begins with the input of YouTube Video Link. Utilizing that link, the video is downloaded from YouTube into .webm format through Pytube, which is a Python wrapper for a YouTube media downloader. Once a video file is downloaded, then audio is extracted from it to separate the audio and visual inputs in the video. This audio file is therefore the base for producing text.

After all that, the video extraction will lead to the project's transcription phase where words spoken are translated into a text transcript. The transcript, then, will be the very outline of content from the video and all its strong points and nuances as presented by the speaker. Finally, an AI model receives the transcript as input to produce a coherent blog that is readable. This model takes the layout, formatting, and tone of a transcript and modifies it to that of a written blog-style article so that a raw transcript can be shaped into a polished article that appeals to a wider group of people.

The project thus enables content creators and businesses to extend their video content to where viewers like reading or need accessible formats, while making it easier to search and engage with across various platforms.

2. Existing System:

Existing systems in the field of video-to-text conversion and content repurposing are primarily focused on transcription and editing functionalities. However, they often lack end-to-end solutions that include all essential steps—such as downloading, audio extraction, transcription, and content generation—as a single streamlined workflow. Some systems include basic transcription and editing features, but they lack AI-driven capabilities for transforming transcripts into well-structured blog posts.

Additionally, these systems may not be financially feasible for individual users or small creators, as many require subscriptions or usage fees that can be high for longer or frequent content. Here is a comparison of some of the existing systems, their strengths, weaknesses, and limitations, followed by a comparative analysis highlighting the advantages of our proposed system, which integrates video download, audio extraction, transcription, and AI-based blog generation in a single workflow. This comprehensive approach aims to be both cost-effective and user-friendly for a wide audience.

2.1. Descript:

Descript is an AI tool which is used transcripting which is developed by the Andrew Mason and Publish by Descript Inc. in the 2017.

2.1.1. Features:

- Transcription and editing in one platform.
- Supports video editing, subtitle creation, and collaboration tools.
- Integrates with Adobe Premiere Pro and other video tools.

2.2.2. Limitations:

- Limited free plan with restricted transcription hours.
- Overdub feature requires high-quality audio to perform well.
- Learning curve for advanced editing features.

2.2. REV:

Rev is a transcription service known for offering both AI and human-based transcription, ensuring high accuracy for users who need precise text outputs which is developed by Jason Chicola and Mark Chen and publish by Rev.com Inc. in 2010.

2.2.1. Features:

- Human and AI transcription options.
- High accuracy with human transcription for premium users.
- Captioning, subtitling, and translation services.
- API integration for automated workflows.

2.2.2. Limitations:

- Higher cost for human transcription services.
- Slower turnaround times for human-generated transcripts.
- Limited customization options in transcription format.

2.3. Trint:

Trint is a transcription tool developed by Jeff Kofman and launched by Trint, Ltd. in 2016, offering multi-language transcription with advanced editing and collaboration features, designed especially for journalists and content teams.

2.3.1. Features:

- High-accuracy transcription with multi-language support.
- Advanced editing tools and formatting options.
- Collaboration features for team editing.

2.3.2. Limitations:

- Subscription-based pricing, with no free version.
- Limited integrations with third-party platforms.
- Some languages have lower transcription accuracy.

2.4. Assembly:

AssemblyAI is an API-based transcription service that provides high-accuracy audio-to-text conversion, along with advanced features like sentiment analysis and entity recognition, tailored for developers building custom applications. It was developed by Dylan Fox, founded under AssemblyAI, Inc., and launched in 2017.

2.4.1. Features:

- API-based transcription with high accuracy.
- Supports additional features like sentiment analysis and entity recognition.
- Integrates easily into custom applications.
- Scalable for high-volume transcription needs.

2.4.2. Limitations:

- Requires programming knowledge to implement.
- Limited editing features compared to full applications.
- No free tier for API usage beyond trial limits.

2.5. Comparison between Existing Systems and Proposed System

Table 1: Comparison between Existing Systems and Proposed System

Sr.	Criteria	Descript	Rev	Trint	Assembly	Propose System
1.	Cost(Expensive)	Yes	Yes	Yes	No	Yes
2.	Media Output	Limited	Limited	Limited	Yes	Yes
3.	Accuracy	High	Very High	Medium	High	High
4.	Budget friendly	No	No	No	Yes	Yes
5.	Availability(Public)	Paid	Paid	Paid	Easily Available	Easily Available
6	Content Generation	No	No	No	Yes	Yes

3. Problem Description

3.1. Manual Blog Creation is Time-Consuming and Labor-Intensive

On the other hand, manually, creating a blog from video content is very time and labor-consuming as the audio has to be transcribed and the outcome edited and formatted in to a coherent article.

3.2. Existing Systems Are Limited to Transcription Only

While many current systems allow for automatic transcription by extracting audio from videos, they usually do not go beyond the transcription since editing and formatting of the text into a blog is something else that requires extra efforts to be performed manually. That's why the process is inefficient and unproductive for the content creators.

3.3. Dependency on Subtitles for Blog Generation

Others auto-generate blogs, but they process subtitles instead of processing audio itself. These systems can only generate content if subtitles are available. When videos are not subtitled, such platforms do not even produce a blog which reduces their utility.

4. Project Objective

Our project's core focus revolves around the question of how one can create an automated and efficient system to simplify the generation of blog content from video material. Our system will follow three core aspects.

4.1. Audio-to-Text Transcription: Converting audio recordings from YouTube videos in to an accurate text.

4.2. Content creation: Write good, structured blogs or articles from transcriptions using advanced models for automatically writing it.

4.3. Subtitle Independence: It will ensure that the system works even without subtitles, therefore providing an even more flexible solution to content providers. The goal is to provide a budget-friendly, user-friendly tool that enhances the productivity and efficiency of content creators by eliminating manual transcription and blog writing, while producing high-quality, ready-to-publish content.

5. Proposed System

5.1. Aims and purpose:

We aim to develop an automation system transforming video material into high-quality ready-for-publish blog posts.

5.2. Core aspects of the system would include:

Transcription of the material, generation of content, and independence in subtitles. The workflow of a user will be intuitive for inputting a link for a YouTube video so that the system automatically changes video to mp3 and transcribes audio; then it uses the same trained model to generate an article or blog. That is to say, the system will turn out to be very user-friendly, effective, and inexpensive.

5.3. Dataset

5.3.1. General information

Common Voice is a corpus of speech data read by users on the Common Voice website (http://voice.mozilla.org/), and based upon text from a number of public domain sources like user submitted blog posts, old books, movies, and other public speech corpora. Its primary purpose is to enable the training and testing of speech recognition (SR) systems, but we encourage its use for other purposes as well.

5.3.2. Dataset Structure

The corpus is split into several parts for your convenience. The subsets with "valid" in their name are audio clips that have had at least 2 people listen to them, and the majority of those listeners say the audio matches the text. The subsets with "invalid" in their name are clips that have had at least 2 listeners, and the majority say the audio does not match the clip. All other clips, i.e. those with fewer than 2 votes, or those that have equal valid and invalid votes, have "other" in their name.

The "valid" and "other" subsets are further divided into 3 groups:

- dev - for development and experimentation
- train - for use in speech recognition training
- test - for testing word error rate

5.3.3. Organization and Conventions

Each subset of data has a corresponding csv file with the following naming convention:

"cv-{type}-{group}.csv".

Here "type" can be one of {valid, invalid, other}, and "group" can be one of {dev, train, test}. Note, the invalid set is not divided into groups.

Each row of a csv file represents a single audio clip, and contains the following information:

- Filename - relative path of the audio file
- Text - supposed transcription of the audio
- Age - age of the speaker, if the speaker reported it

  Teens: '< 19'

  Twenties: '19 - 29'

  Thirties: '30 - 39'

  Forties: '40 - 49'

  Fifties: '50 - 59'

  Sixties: '60 - 69'

  Seventies: '70 - 79'

  Eighties: '80 - 89'

  Nineties: '> 89'

- Gender - gender of the speaker, if the speaker reported it

  Male

  Female

  Other

- Size: 13GB
- Accent - accent of the speaker, if the speaker reported it

  US: 'United States English'

  Australia: 'Australian English'

  England: 'England English'

  Canada: 'Canadian English'

  Philippines: 'Filipino'

  Hongkong: 'Hong Kong English'

  Indian: 'India and South Asia (India, Pakistan, Sri Lanka)'

  Ireland: 'Irish English'

  Malaysia: 'Malaysian English'

  New-Zealand: 'New Zealand English'

  Scotland: 'Scottish English'

  Singapore: 'Singaporean English'

South-Atlantic: 'South Atlantic (Falkland Islands, Saint Helena)'

African: 'Southern African (South Africa, Zimbabwe, Namibia)'

Wales: 'Welsh English'

Bermuda: 'West Indies and Bermuda (Bahamas, Bermuda, Jamaica, Trinidad)'

The audio clips for each subset are stored as mp3 files in folders with the same naming conventions as it is corresponding csv file. So, for instance, all audio data from the valid train set will be kept in the folder "cv-valid-train" alongside the "cv-valid-train.csv" metadata file.

5.4. Features

- Input YouTube Video Link
- Audio-to-Text Transcription
- Automatic Blog/Article Generation
- Subtitle-Free Content Creation
- AI-Based Content Structuring
- User-Friendly Interface
- Cost-Effective Solution

5.5. Expected Result

- Efficient and smooth video-to-blog conversion.
- Rich transcriptions and adequately contextual blog generation.
- Reducing manual transcription and editing time.
- Budget-friendly tool for diverse users.
- Reduced time from generating high-quality content.

5.6. Possible Extensions

The following features can be added to the project if time and resources permit:

Multi-language support for transcription and content generation.

6. Modules

This section outlines the key components of the project, detailing how each module functions in transforming video content into a written blog. The system operates through several stages, starting from link input to the final content generation. Each module plays a critical role in ensuring the seamless flow of data from video to text, enabling the automation of blog creation using AI.

6.1. Link _ Input

This module accepts the link of the YouTube video and submits it to the system so that video can be accessed, and from this point, the content processing workflow is triggered. This module provides the first step in the process using video to be converted into blog content.
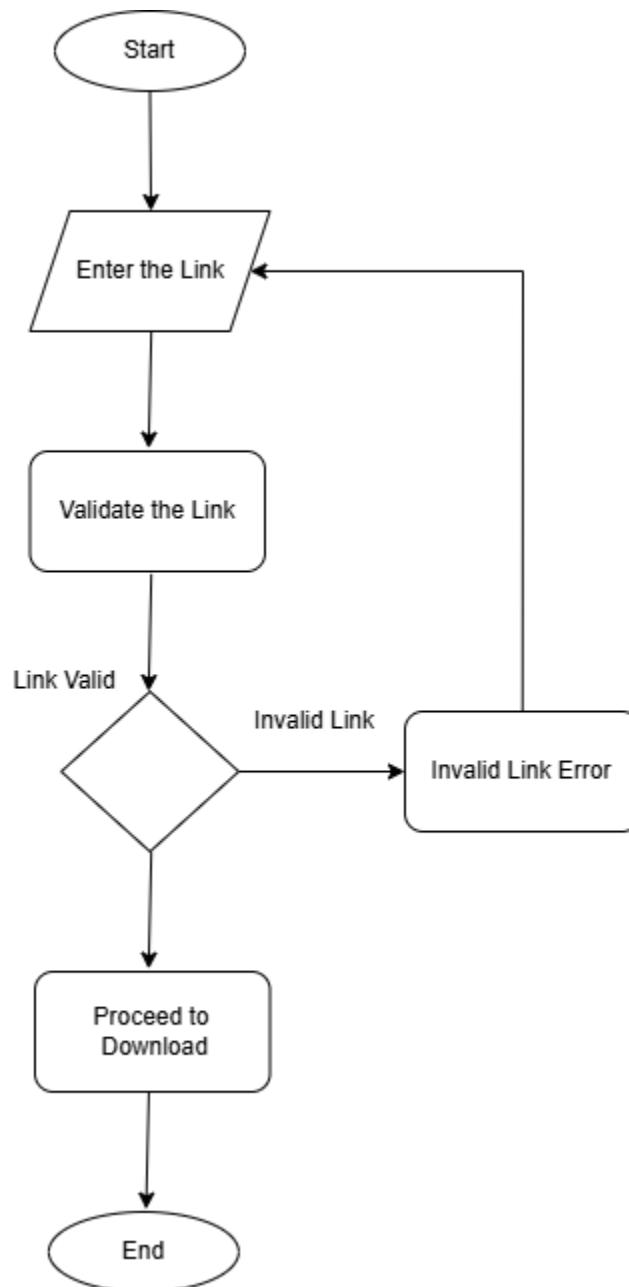

Figure 1: Flow Chart of Link Input

11

6.2 Flow chart of Download Video

The video is then downloaded from the link given into the system for further processing. This download ensures that the video file is available locally for the extraction of audio and transcription.
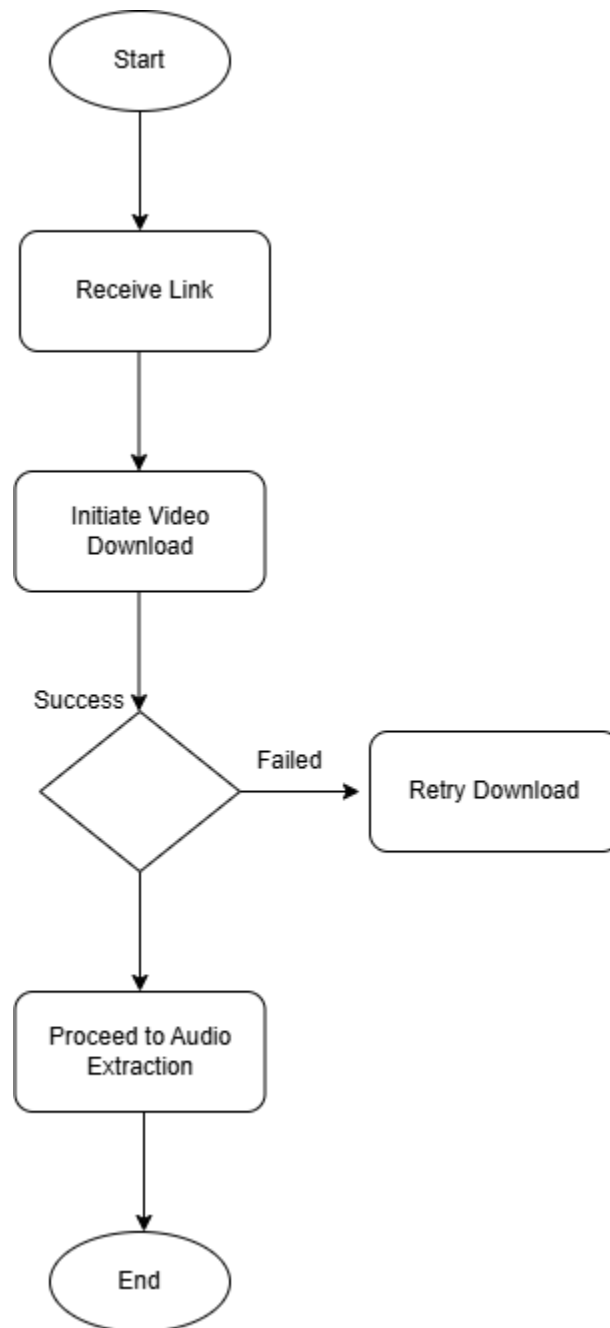

Figure 2: Flow Chart on Download Video

6.3. Flow chart of Audio Extraction

This module is about extracting audio from the downloaded video. The audio file is then used for transcription; actually, turning the speech into text for further content generation.
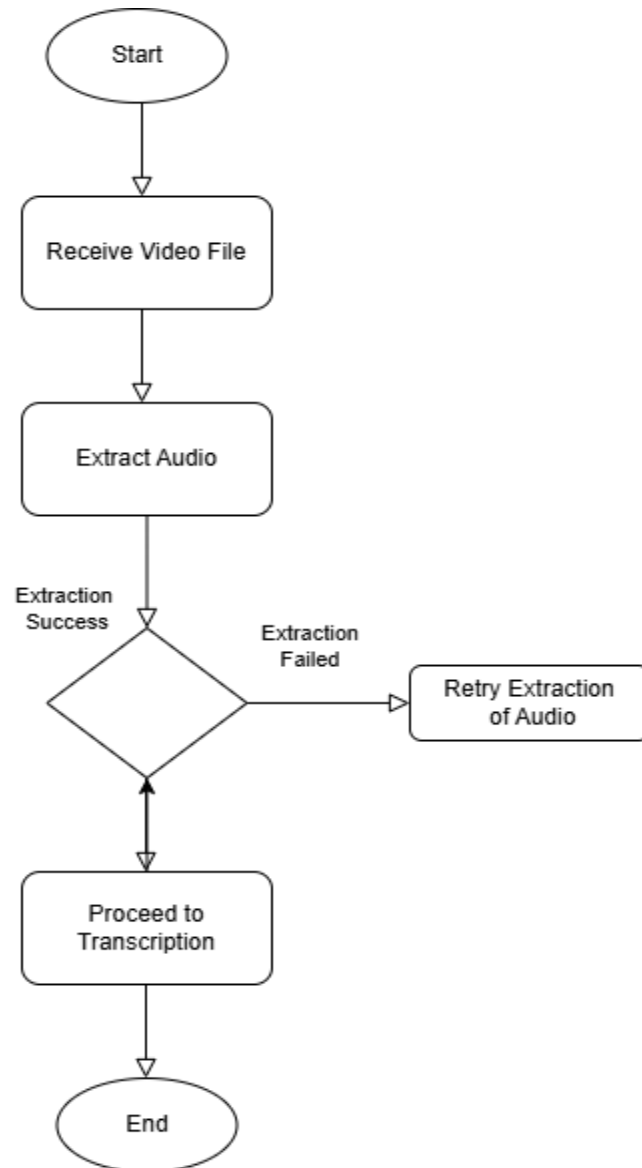

Figure 3: Flow Chart of Audio Extraction

6.4. Flow Chart of Transcription

Audio extracted through this tool is transcribed by AI speech-to-text models. This transcription work helps to develop blog material.
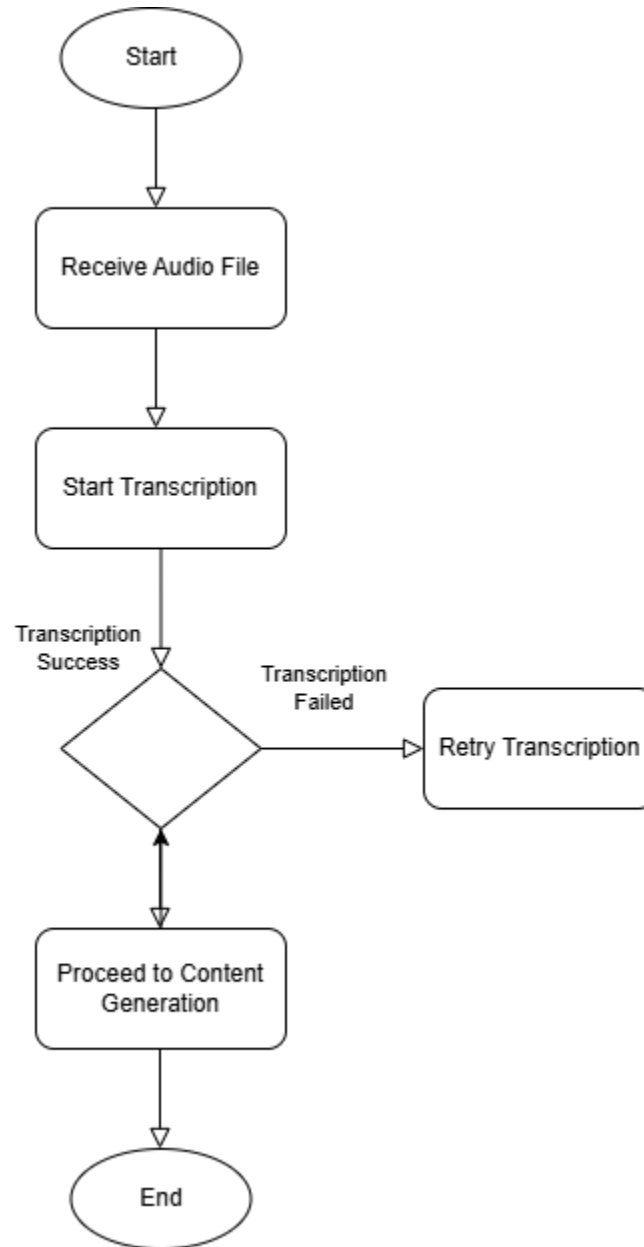

Figure 4: Flow Chart of Transcription

6.5. Flow Chart of Content Generation

Transcription is done when the system produces a structured article or blog post in return, using the text transcribed. It uses AI to make it more readable and SEO-friendly content.
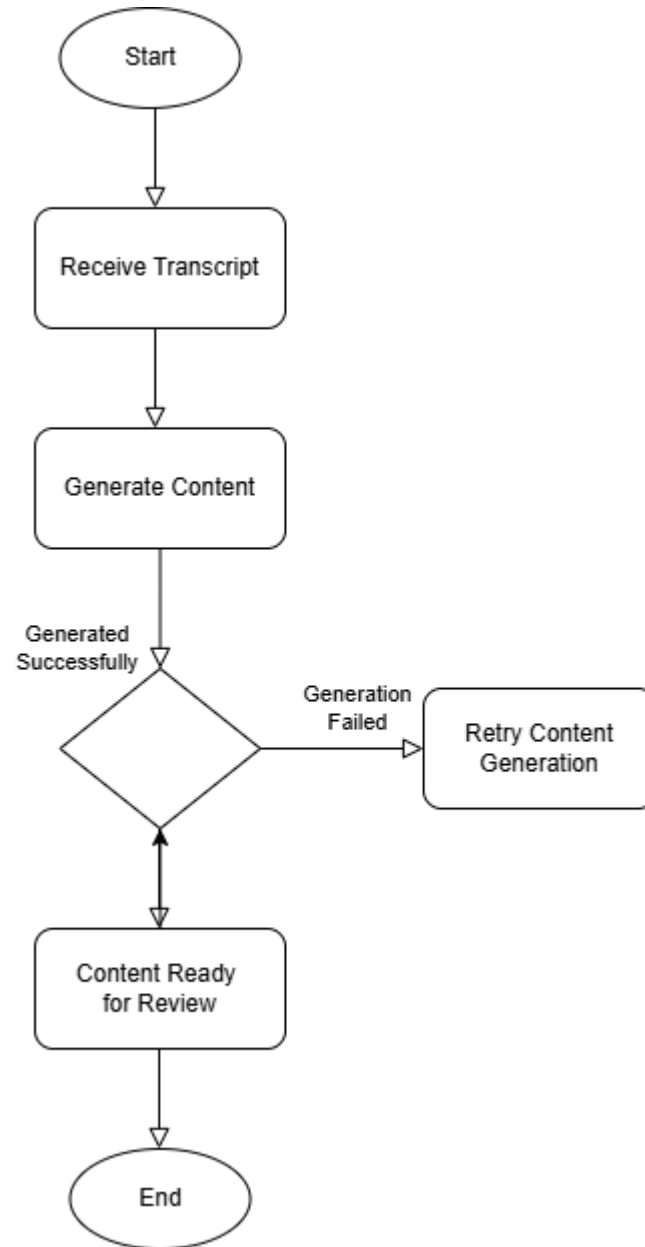

Figure 5: Flow Chart of Content Generation

7. Methodology

The methodology opted for this project is INCREMENTAL MODEL, since there's no as fluctuation in requirements so we will work in increments and after each increment will proceed to next increment.
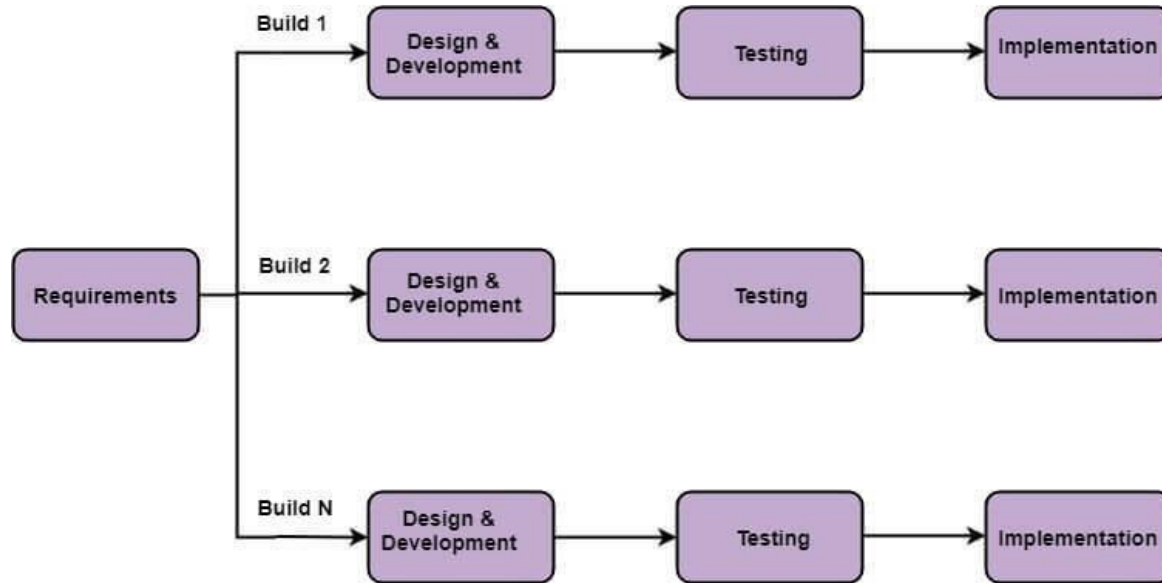

Figure 6: Incremental Model

8. Project Scope

Our final goal is to produce an automated system that transforms video content into high-quality ready-to-publish blogs. For this, we will use audio from a YouTube video, transcribe it, and then use an AI model to develop a well-structured blog post or article. This way, content creators won't waste their time doing manual transcription and formatting; it even works without subtitles, making it much more flexible and accessible.

- The system proposed will fall in the ranges of the following:
- The content generation for bloggers, journalists, and content creators
- Audio-to-text transcription with content generation from YouTube videos
- Cost-effective solution to any level of content creators
- Works well with videos that do not require subtitles to be dubbed
- Accessible to either individual users or businesses looking to repurpose video content efficiently.

9. Feasibility Study

9.1. Risk involved

The Proposed system is unlikely to fail. Still, some risks and possible challenges will naturally impact its development and functionality. The biggest risk would be the accuracy and consistency of the transcription process with handling complexity on the generation of quality blog content from transcribed audio files. It could face a limitation with user interfaces, particularly when dealing with large video files and where in the content is processed over long periods.

Table 2: Risks

No.	Key Risks	Hypothetical Statement	Resolution
1	Transcription Accuracy	The transcription m ight not be 100% accurate and thereby reduce the quality of the blog.	Implement the advanc ed transcription models and post-processing for high accuracy.
2	Subtitle-Free Content	The system fails to process videos that do not have subtitles, ther eby affecting the g eneration of the blog.	Design the system to work solely on audio to eliminate dependence on subtitles.
3	System Performance with Large Files	The System may encounter functionality issues dealing with long video files.	Optimize the backend and use efficient video processing libraries to handle large video files
4	User Interface Limitations	Small desktop interface may cause usability issues for the users.	Focus on developing a clean, simple and responsive interface that works through multipl e screen sizes.
5	Performance Impact Due to Modules	Making use of many modules with dependencies produce low system performance.	Carefully select and optimize the required modules to minimize overhead and ensure smooth operation.

10. Solution Application Areas

This system is for content creators, bloggers, journalists, and businesses that aim to repurpose video content into quality blogs or articles efficiently. The broad application of this system in its widest range is for users to immediately generate text-based content from YouTube videos or any other video platforms. It will automate the transcription, content structuring, and blog generation process, so it's a suitable choice not just for individual content creators but also for organizations looking to streamline their content creation process for better productivity.

11. Tools & Technology

Following tools and technology will be required in development of the Project

11.1. Software

- Python
- Youtube
- Transformers
- Django
- Jupyter Notebook
- Visual Studio Code
- React.js-Tailwindcss
- CICDPipelines
- Figma
- PosgreSQL

11.2. Hardware

- PC / Laptop with a powerful processor, ideally Core i5 or above, for better performance
- Storage Solution
- Containerized Server

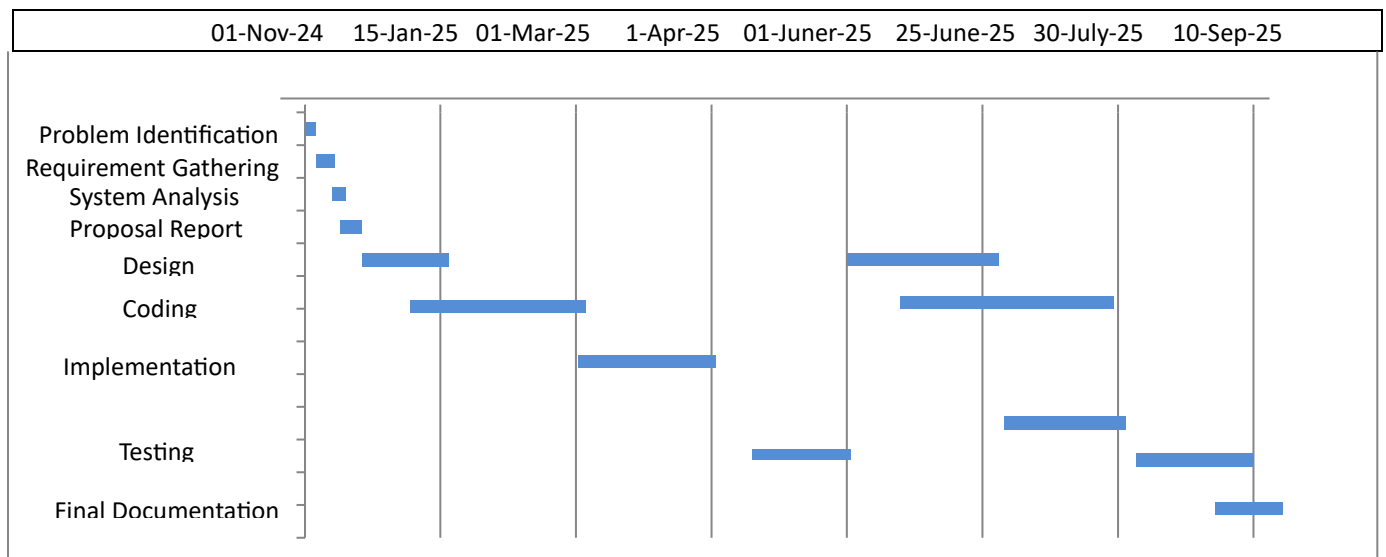| Sr.	Tools & Technology	Rationale
1	PC / Laptop (Core i5 or above).	Ensure that video files have smooth processing, transcribe, and generate content.
2	Storage Solution	SSD or Cloud Storage Storage for large video and audio files for efficient processing.
3	Server	Host the Back-end part of the System to Handle Requests on a Mass Scale to Huge Amounts of Users and Colossal Amount of Data.

12. Expertise of the Team Members

Our Team Member have prior knowledge and their domain of expertise

Table 4: Team Expertise

Name	Expertise
Abdul Rehman	Python development, Database, Automation, DevOps, Machine Learning, Design & Architecture Documentation
Touseer Amir	JavaScript, React.js, Node.js, UI/UX Designer ,Design and Architecture Documentation

13. Milestones

Table 5: Milestone

	01-Nov-24	15-Jan-25	01-Mar-25	1-Apr-25	01-Juner-25	25-June-25	30-July-25	10-Sep-25

Problem Identification
Requirement Gathering
System Analysis
Proposal Report
Design
Coding
Implementation
Testing
Final Documentation

14. References

1. Smith, J., 2020. Automated Video Transcription and Content Generation: Challenges and Solutions.[online] ResearchGate. Available at: <https://www.researchgate.net/publication/339792830_Automated_Video_Transcription_and_Content_Generation> [Accessed 1 November 2024].

2. Gupta, P., 2021. Exploring AI-Driven Blog Generation from Video Transcriptions. [online] Academia.edu. Available at: <https://www.academia.edu/38923456/Exploring_AI-Driven_Blog_Generation_from_Video_Transcriptions> [Accessed 3 November 2024].

3. OpenAI. 2023. GPT Models for Content Creation from Video Transcriptions. [online] Available at: <https://openai.com/research/gpt-content-creation> [Accessed 6 November 2024].

4. Google Cloud. 2024. Speech-to-Text API for Video Transcription. [online] Available at: <https://cloud.google.com/speech-to-text> [Accessed 7 November 2024].

5. **YouTube. 2024. YouTube Transcription and Captioning Services.** [online] Available at: <https://support.google.com/youtube/answer/2734796> [Accessed 10 November 2024].

6. Mozilla Common Voice Dataset. Available at: https://www.kaggle.com/datasets/mozillaorg/common-voice. Accessed on [Insert Date].