

**Chapter 1 SRS**

**TABLE OF CONTENTS**

**1**. **Introduction**

1.1 Purpose …………………………………………………… Page No 4

1.2 Scope……………………………………………………… Page No 4

1.3 Definitions, Acronyms and Abbreviations……………….. Page No 5

1.4 Overview………………………………………………….. Page No 5

**2**. **General Description**

2.1 Product Perspective……………………………………….. Page No 6

2.1.1 Product Function …………………………………. Page No 7

2.1.2 Hardware Interface………………………………… Page No 8

2.1.3 Software Interface…………………………………..Page No 9

2.1.4 Communication Interface…………………………...Page No 10

2.2 User Characteristics…………………………………………Page No 10

2.3 General Constraints………………………………………. ...Page No 12

2.4 Technologies used………………………………………...... Page No 13

**3. Specific Requirements**

3.1 Functional Requirements……………………………………Page No 14

3.2 Non-Functional Requirements ……………………………..Page No 16

3.2.1 AVAILABILITY……………………………………..Page No17

3.2.2 SECURITY…………………………………………...Page No17

3.2.3 RELIABILITY……….……………………………..Page No17

3.2.4 PORTABILITY……………………………………. Page No17

3.2.5 MAINTAINABILITY…………………………….... Page No17

#### 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the scope, functionality, and requirements of the YouTube Transcript Summarizer Chrome Extension. This extension is designed to enhance the user experience by providing automated transcription summarization, translation to multiple languages, and speech conversion features for YouTube video transcripts.

## 1.2 Scope

In this executive-level summary:

1. **Software Product(s):** The software product to be produced is a YouTube Transcript Summarizer Chrome Extension.
2. **Functionality:**
   * The extension will extract subtitles from YouTube videos.
   * It will generate concise summaries of the transcriptions.
   * Users can choose to translate the summaries into Hindi.
   * The extension will provide an option to convert the summarized content into speech.
3. **Application:**
   * The extension aims to enhance user experience on YouTube by providing automated transcription summarization and translation features.
   * Benefits include improved content comprehension, language diversity, and accessibility.
   * Objectives involve catering to a diverse user base and facilitating easy consumption of video content.
4. **Consistency:** The scope aligns with similar statements in higher-level specifications, maintaining coherence throughout the document.

## 1.3 Definitions, Acronyms, and Abbreviations.

## · *Transcription Summarization:* The process of condensing the content of video transcriptions while retaining key information.

## · *Speech Conversion:* The feature that converts summarized content into audible speech.

## 

## 1.4 Overview

# *This section serves as a guide to navigate through the Software Requirements Specification (SRS) document, providing a brief glimpse into its contents and organization.*

# 

# (1) Content Description:

# 

# The SRS document is structured to address the distinct needs of various stakeholders involved in the development and utilization of the YouTube Transcript Summarizer Chrome Extension.

# 

# *Section 1* introduces the document's purpose, scope, definitions, and a high-level overview, catering to a broad audience interested in understanding the project's goals and objectives.

# *Section 2,* "General Description," offers a detailed perspective on the extension, covering its context, functionalities, user characteristics, constraints, and the technologies employed. This section is particularly relevant for customers and potential users seeking a comprehensive understanding of the extension's scope and features.

# *Section 3,* "Specific Requirements," is the heart of the document. It provides in-depth insights for developers, detailing both functional and non-functional requirements. This section is crucial for those involved in the technical implementation and quality assurance aspects of the project.

# (2) Organizational Structure:

# 

# For Customers/Potential Users: Start with Section 1 to gain a holistic understanding of the extension's purpose and scope. Focus on Section 2 for a deeper dive into the extension's functionalities and constraints.

# 

# For Developers: Direct your attention to Section 3, which outlines specific technical requirements for the development and implementation of the YouTube Transcript Summarizer Chrome Extension. Section 2 provides additional context that may assist in aligning technical decisions with the overall project goals.

# 

# This organized structure ensures that readers can efficiently locate and focus on the sections most relevant to their roles and interests, promoting clarity and facilitating a smoother comprehension of the document.2. The Overall Description

#### 2. The Overall Description

The YouTube Transcript Summarizer is a Chrome Extension. It will be used for writing the transcript of any YouTube video (which has subtitles). It will allow the user to summarize the transcript of the video in a simple and crisp form.

#### 2.1 Product Perspective

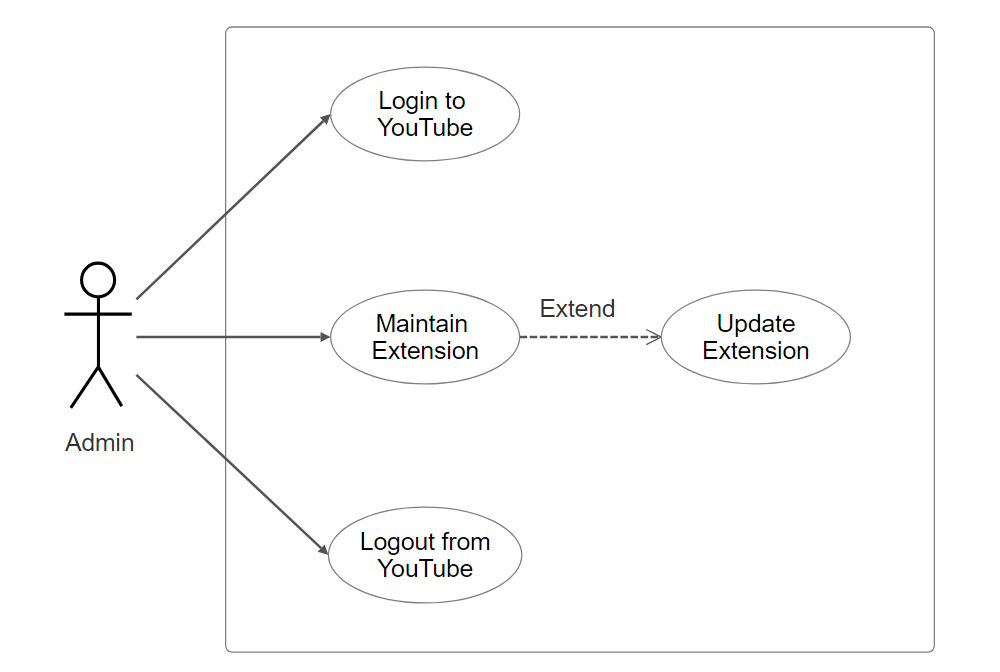
This YouTube Transcript summarizer will be used as a Chrome Extension that operates as an add-on to the Google Chrome web browser.

The core operation of the extension is independent within Chrome, but it will interact with the YouTube API to retrieve video transcripts, which is a web service provided by YouTube. While it relies on this API for data, it doesn't require external software installations or complex server infrastructure.

In summary, a Chrome extension, such as a YouTube transcript summarizer, is designed to operate independently within the browser and It doesn't rely on external software or systems outside of the browser for its core functionality, but it will have dependency on the YouTube API.

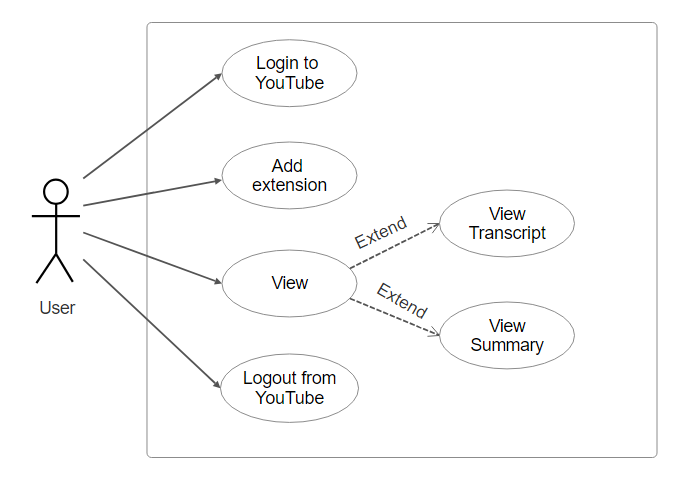
#### 2.1.1 Product Function

#### ADMINISTRATOR:



### Fig: Use Case Diagram for Administrator

#### USER:



**Fig: Use Case Diagram for User**

##### 2.1.2 Hardware Interfaces

### YouTube transcript summarizer Chrome extension will have minimal hardware interface requirements.

The following hardware requirements are considered:

**1. Web Browser:**

The Chrome extension will run within the Chrome web browser, so it's essential that the user's computer has the Google Chrome browser installed and up to date. Chrome extensions are specific to the Chrome browser and don't require specialized hardware interfaces.

**2. Internet Connection:**

A stable internet connection is necessary for the Chrome extension to interact with the YouTube API and fetch video data, including transcripts. The extension relies on network communication, so an internet connection is a hardware requirement.

**3. User Input Devices:**

Users will interact with the extension using standard input devices like a keyboard and mouse. There are no specific hardware interfaces beyond these standard input methods.

##### 2.1.3 Software Interfaces

This section outlines how the software interacts with other software components or systems. Here are the software interfaces that will be considered:

*1. Chrome Browser Interface*

The software is designed to function as a Chrome Extension, and it will interact with the Chrome browser.

*2.YouTube API Interface*

The YouTube Transcript Summarizer Chrome extension will communicate the YouTube API by making HTTP requests to the endpoints provided by YouTube. These requests will be made using JavaScript, which is the primary language for this extension development.

*3. Web Technologies utilized*

* HTML (Hypertext Markup Language):

HTML will be used to structure and present the content of the extension's user interface. HTML is a standard language for creating web pages and is used to define the structure of your extension's popup, settings pages, or any other web-based user interfaces.

* Python:

The python libraries used will be hugging face transformer. It provides pre-trained libraries and models. It will be used for summarizing the transcript.

##### 2.1.4 Communications Interfaces

User Interface (UI): A button will be visible to the user while playing a YouTube video. On clicking the button, the user will be able to see the transcript (if available) of the video. Further, another button will be visible for summarization and the user will get a summary of the transcript.

#### 2.2 User Characteristics

**User Characteristics**

In developing the YouTube transcript summarizer, understanding the general characteristics of the intended users is pivotal in shaping the subsequent design and functionality of the system. The primary user types have been identified:

1. Casual YouTube Viewers: Casual users represent individuals who seek a simple and efficient means of summarizing YouTube video transcripts. They typically possess a varied educational background, ranging from high school to college levels. Their experience with technology may vary, necessitating an intuitive user interface that requires minimal effort to navigate. The design for casual users should emphasize ease of use, intuitive controls, and a visually appealing interface.
2. Researchers or Students: Students and academic users are a crucial user segment, often leveraging the YouTube transcript summarizer for educational purposes. They possess a higher level of education, ranging from undergraduate to postgraduate levels. Their experience with technology is moderate to high, and they require a detailed summary to aid in research and studies. The user interface should provide functionalities to export summarized content.

#### User Class 1: Casual YouTube Viewers

*Educational Level*: Varied, from high school to college education.

*Experience:* Moderate to extensive experience with using YouTube for entertainment and educational purposes.

*Technical Expertise*: Basic to intermediate familiarity with web browsers and common extensions.

##### Reasoning for Impact on Design:

Casual users require an intuitive and uncomplicated user interface (UI) to quickly access summarized video content. The design should prioritize simplicity and ease of use.

The extension should provide a clear interface to guide users through the summarization process, considering varying levels of technical expertise.

#### User Class 2: Researchers or Students

*Educational Level:* College to postgraduate education.

Experience: Extensive experience in academic research, studying, or professional tasks.

*Technical Expertise*: Proficient with web technologies and applications, including browser extensions.

Reasoning for Impact on Design:

Researchers and students seek accuracy and customization in the summary generation process.

#### Design Implications

User Experience (UX) Design:

Simplicity and Intuitiveness: The UI should be straightforward and easy to navigate for casual users, ensuring they can quickly utilize the extension.

System Architecture and Performance:

Efficiency: The summarization algorithm needs to be efficient to cater to both casual users who seek speed and researchers who prioritize accuracy and detail.

#### 2.3 Constraints

1. Regulatory Policies

Compliance with legal and regulatory requirements related to data privacy, user consent, and any other relevant policies.

##### *2. Hardware Limitations*

Adherence to the capabilities and limitations of various hardware configurations, ensuring the extension can function optimally across a range of devices.

3. Interface to Other Applications

Compatibility and seamless integration with the Chrome browser and YouTube API, conforming to the specifications and requirements of these platforms.

4. Audit Functions

Implementation of auditing capabilities to track user interactions, extension usage, and summarization processes for evaluation and improvement purposes.

5. Higher-Order Language Requirements

Development in a specified programming language or framework to align with existing technologies and maintain consistency in the development environment.

6. Signal Handshake Protocols

Adherence to established protocols for data communication and synchronization to ensure smooth interaction between the extension and external systems.

7. Reliability Requirements

Assurance of high reliability and availability to meet user expectations, minimizing downtime and ensuring consistent functionality.

8. Criticality of the Application

Awareness of the importance of the extension's functionality to users, guiding development priorities and quality assurance efforts.

9. Safety and Security Considerations

Integration of security measures to safeguard user data, prevent unauthorized access, and ensure a secure user experience.

These constraints help set the boundaries and guidelines for the development process, ensuring that the extension complies with legal, technical, and security aspects while meeting the expectations and needs of the users.

#### 2.4 Technologies Used

The technologies used to develop the YouTube Transcript Summarizer Chrome Extension involve a combination of web technologies, APIs, and natural language processing (NLP) algorithms. Here's a breakdown of the key technologies:

* *HTML, CSS, and JavaScript:*

HTML (HyperText Markup Language): Used for structuring the web page and defining its content.

CSS (Cascading Style Sheets): Used for styling and layout to enhance the UI/UX of the extension.

JavaScript: Used for dynamic interactions, handling user events, and managing the extension's behavior.

* *YouTube API:*

YouTube Data API: Utilized to access YouTube video data, including video transcripts, which is essential for summarization.

* *Natural Language Processing (NLP) Libraries and Algorithms:*

NLP Algorithms: NLP algorithms are used for text summarization, which is a crucial part of the extension's functionality. These algorithms could include techniques like Extractive Summarization or Abstractive Summarization.

Python (potentially): Often used for implementing NLP algorithms due to the availability of powerful NLP libraries like NLTK (Natural Language Toolkit), SpaCy, or Gensim.

* *Chrome Extension APIs:*

Chrome Extension API: Used to interact with the Chrome browser and enable the extension's functionalities within the browser.

* *Version Control:*

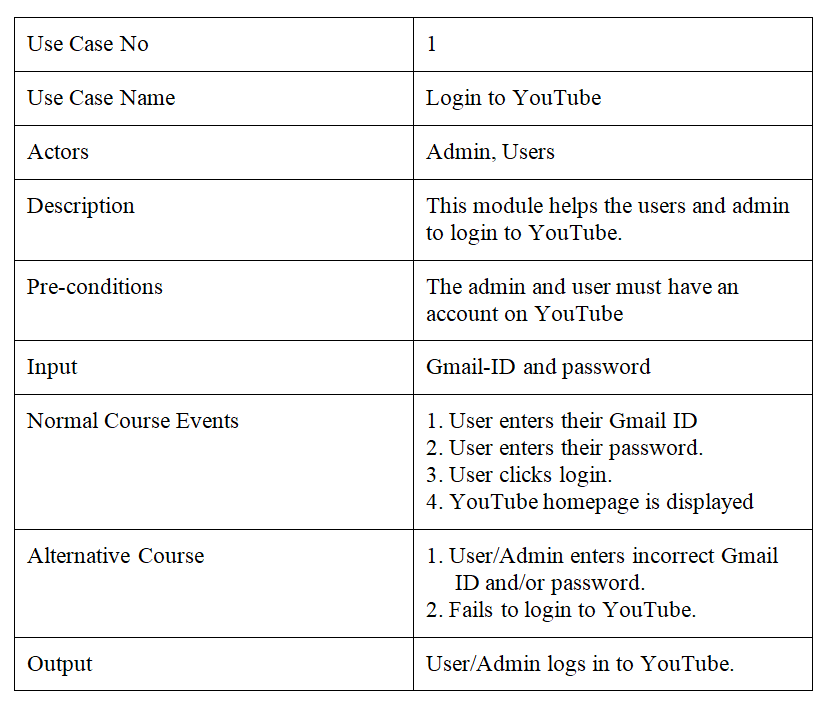
Git and GitHub: Used for version control, collaboration, and tracking changes during the development process.

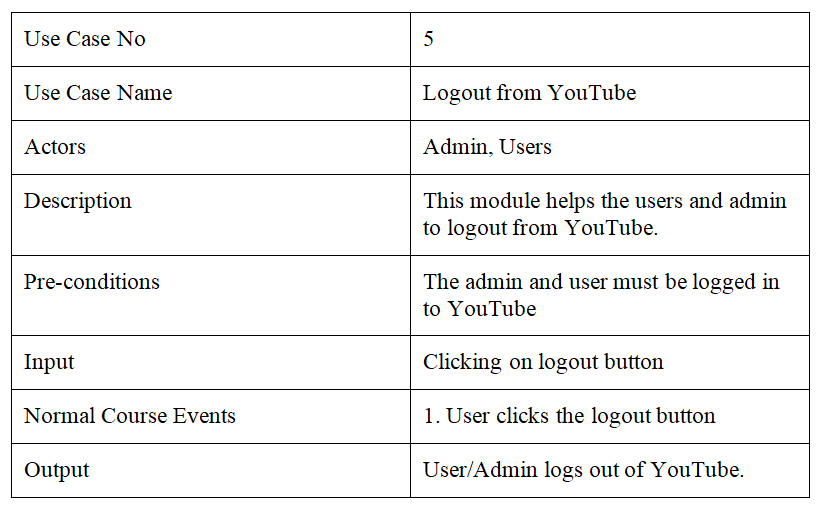
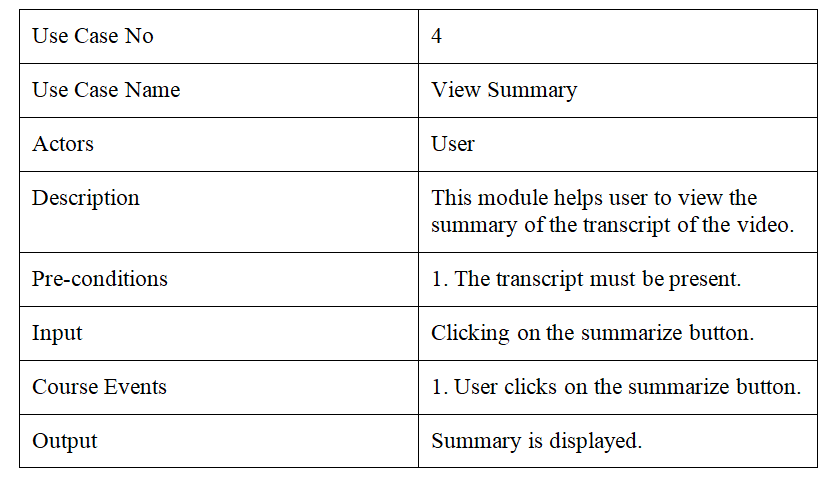
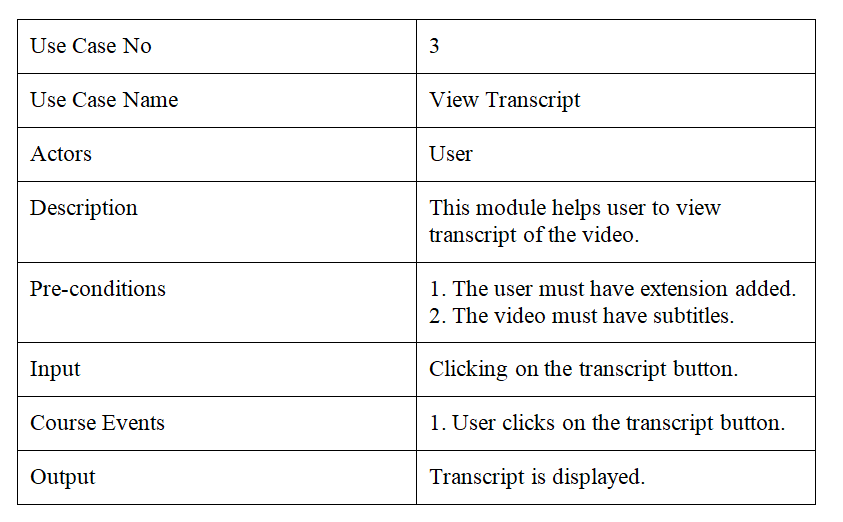
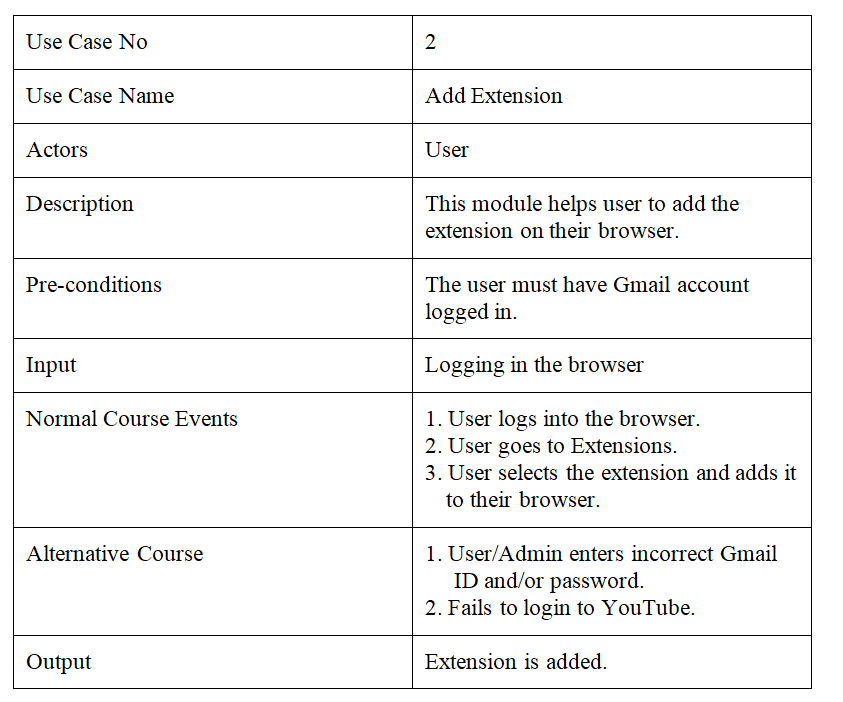
* *Text Editor or Integrated Development Environment (IDE):*

Any preferred text editor or IDE, such as Visual Studio Code, Sublime Text, or Atom, for writing and managing code.

#### 3. Specific Requirements

3.1 Functional Requirements





* 1. Nonfunctional Requirements (Software System Attributes)

##### 3.2.1 AVAILABILITY

Our extension will be accessible 24/7, with minimal downtime during maintenance or updates. Users will be notified in advance of scheduled downtime. In case of unexpected outages, our extension will handle errors gracefully and provide clear error messages. Users should have uninterrupted access, even during updates or with intermittent internet connectivity.

##### 3.2.2 SECURITY

User data, including video transcripts and summarized content, will be encrypted and stored securely. Secure communication protocols, such as HTTPS, will be used for data fetching.

##### 3.2.3 RELIABILITY

Our extension will consistently provide accurate video transcript summaries with a low margin of error. It will feature robust error handling and comprehensive logging for a smooth user experience. Quality assurance will be in place to identify and address potential issues promptly.

##### 3.2.4 PORTABILITY

Our extension will be primarily compatible with the latest Google Chrome browser. It will adapt to various screen sizes and resolutions, ensuring a consistent user experience. While Chrome is our primary platform, we will consider potential compatibility with other popular web browsers in the future.

##### 3.2.5 MAINTAINABILITY

The source code will be well-documented, following coding standards and best practices for maintainability. We will use version control tools like Git to track code changes and encourage collaboration among developers. Updates, bug fixes, and feature enhancements will have a clear and straightforward deployment process. User documentation will be provided to assist users in effectively using our extension, enhancing overall maintainability.

***Chapter 2 SDS***

**TABLE OF CONTENTS**

1. **Introduction**

1.1 Purpose ……………………………………………………Page No 20

1.2 Scope………………………………………………………Page No 20

1.3 Terms, Definitions, Acronyms and Abbreviations………..Page No 21

1.4 Overview of Document ……..……………………………..Page No 21

2. **System Architectural Design**

2.1Two tier Architecture……………….Page No. 23

2.2 Detailed Description of Components……………….Page No. 24

3. **Data Design**

3.1 Database Description…………………………………Page No 28

4 **User Interface Design**

4.1Detailed Description of Components……………………… Page No 32

**5. TYPES OF TESTS (With Implementation)**................................................. Page No 31

## 7. References

#### 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the scope, functionality, and requirements of the YouTube Transcript Summarizer Chrome Extension. This extension is designed to enhance the user experience by providing automated transcription summarization, translation to multiple languages, and speech conversion features for YouTube video transcripts.

## 1.2 Scope

In this executive-level summary:

1. **Software Product(s):** The software product to be produced is a YouTube Transcript Summarizer Chrome Extension.
2. **Functionality:**
   * The extension will extract subtitles from YouTube videos.
   * It will generate concise summaries of the transcriptions.
   * Users can choose to translate the summaries into Hindi.
   * The extension will provide an option to convert the summarized content into speech.
3. **Application:**
   * The extension aims to enhance user experience on YouTube by providing automated transcription summarization and translation features.
   * Benefits include improved content comprehension, language diversity, and accessibility.
   * Objectives involve catering to a diverse user base and facilitating easy consumption of video content.
4. **Consistency:** The scope aligns with similar statements in higher-level specifications, maintaining coherence throughout the document.

## 1.3 Definitions, Acronyms, and Abbreviations.

## · *Transcription Summarization:* The process of condensing the content of video transcriptions while retaining key information.

## · *Speech Conversion:* The feature that converts summarized content into audible speech.

## 

## 1.4 Overview

# *This section serves as a guide to navigate through the Software Requirements Specification (SRS) document, providing a brief glimpse into its contents and organization.*

# 

# (1) Content Description:

# 

# The SRS document is structured to address the distinct needs of various stakeholders involved in the development and utilization of the YouTube Transcript Summarizer Chrome Extension.

# 

# *Section 1* introduces the document's purpose, scope, definitions, and a high-level overview, catering to a broad audience interested in understanding the project's goals and objectives.

# *Section 2,* "General Description," offers a detailed perspective on the extension, covering its context, functionalities, user characteristics, constraints, and the technologies employed. This section is particularly relevant for customers and potential users seeking a comprehensive understanding of the extension's scope and features.

# *Section 3,* "Specific Requirements," is the heart of the document. It provides in-depth insights for developers, detailing both functional and non-functional requirements. This section is crucial for those involved in the technical implementation and quality assurance aspects of the project.

# (2) Organizational Structure:

# 

# For Customers/Potential Users: Start with Section 1 to gain a holistic understanding of the extension's purpose and scope. Focus on Section 2 for a deeper dive into the extension's functionalities and constraints.

# 

# For Developers: Direct your attention to Section 3, which outlines specific technical requirements for the development and implementation of the YouTube Transcript Summarizer Chrome Extension. Section 2 provides additional context that may assist in aligning technical decisions with the overall project goals.

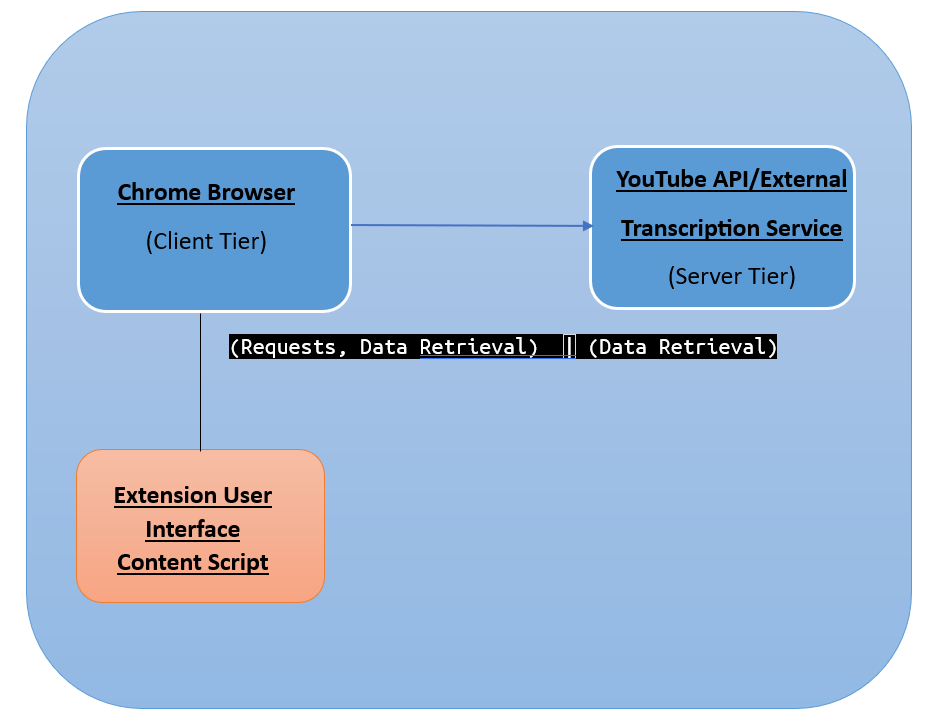
# 

# This organized structure ensures that readers can efficiently locate and focus on the sections most relevant to their roles and interests, promoting clarity and facilitating a smoother comprehension of the document.2. The Overall Description

#### 2. System Architectural Design

Pictorial representation of the system architecture is presented.

##### 2.1 Two- tier architecture :

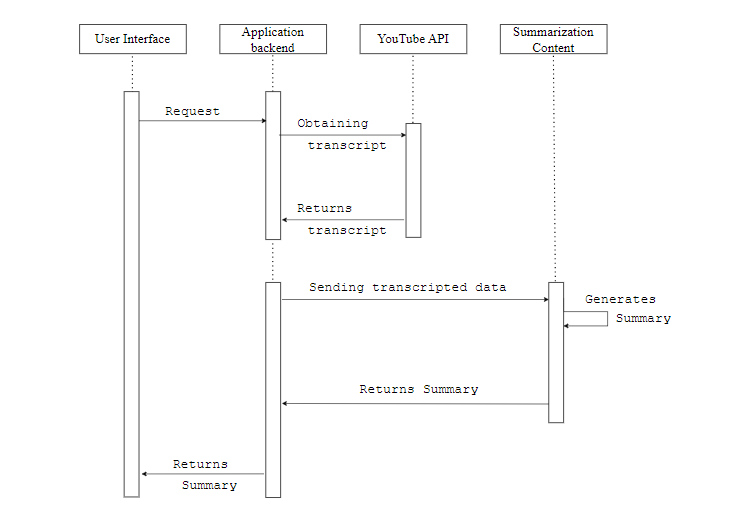


* **Client Tier:** This represents the user's web browser, where the Chrome extension runs. In your case, it's the primary interface through which users interact with your extension.
* **Server Tier:** This represents external services or APIs that your extension interacts with. This can include services like YouTube's API for video data and transcripts.
* **Extension User Interface Content Script:** Extension User Interface Content Script represents components within the client tier (the Chrome Browser) of your architecture.

#### 2.2) Detailed Description of Components

*NOTE: This section is the main focus of the technical design portion of the SDS, the detailed design. This section will provide most of the basis for implementing the product.*

Sequence Diagram



User Interaction:

User requests to summarize a YouTube video by entering its URL.

The user interface component captures the request.

Request Processing:

The user interface sends a request to the application backend, indicating the YouTube video URL.

The application backend component processes the request.

YouTube API Interaction:

The application backend component communicates with the YouTube API to obtain the transcript of the specified video.

The YouTube API returns the transcript data to the application backend.

Summarization Process:

The application backend sends the transcript data to the summarization component.

The summarization component processes the transcript and generates a summary.

Summary Return:

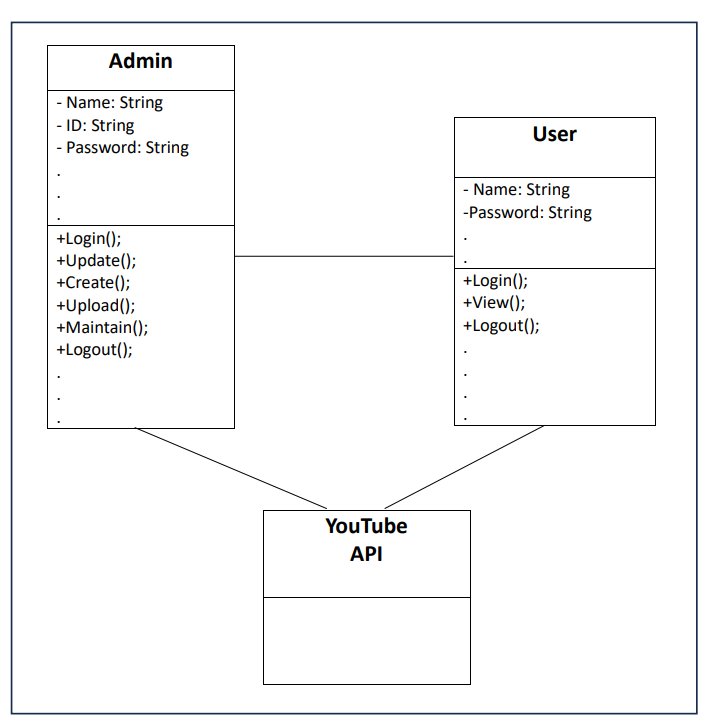
The summarization component provides the summary to the application backend.

The application backend returns the summary to the user interface.

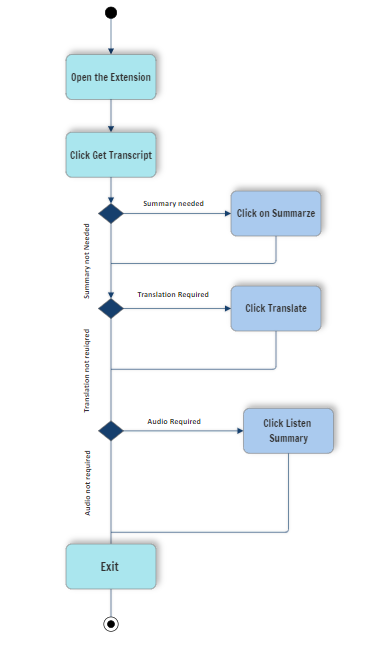
User Interface Update:

The user interface displays the summary to the user.

Class Diagram



Activity Diagram



3 Data Design

3.1 Database description

For a YouTube transcript summarizer Chrome extension project that uses Firebase as the backend, we will use Firebase's real-time database and Firestore for different data storage needs. Here's what Firebase databases we will include in the backend:

##### Firebase Realtime Database:

Firebase Realtime Database is a NoSQL, cloud-hosted database that allows you to synchronize data in real-time.

Videos:

videos

├── videoID

│ ├── title

│ ├── URL

└── ...

Transcripts:

transcripts

├── videoID

│ ├── transcriptID

│ │ ├── text

│ └── ...

└── ...

Summaries:

summaries

├── summaryID

│ ├── text

│ ├── userAttribution

│ ├── videoID

***└── …***

User Data:

users

├── userID

│ ├── username

│ ├── email

│ ├── ...

└── …

#### Firebase Firestore:

Firestore is Firebase's document-oriented NoSQL database that offers more advanced querying and scalability features.

NLP Models:

nlpModels

├── modelID

│ ├── modelName

│ ├── embeddings

└── ...

API Keys:

apiKeys

├── apiKeyID

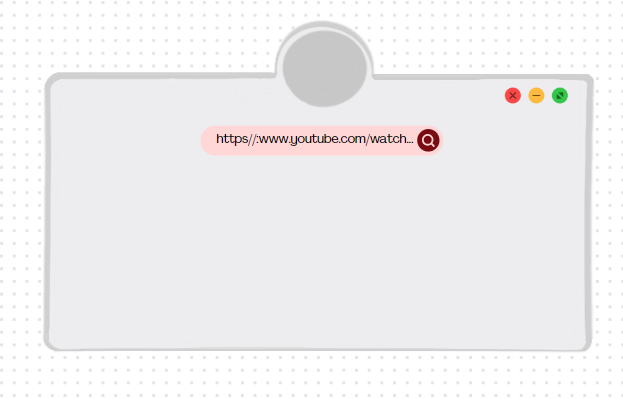
│ ├── key

│ ├── accessTokens

└── ...

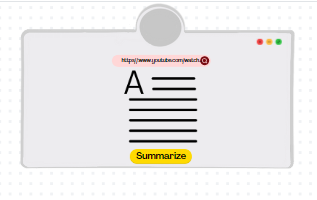
4) User Interface Design

* Home page of extension



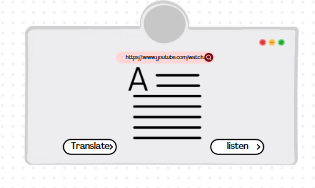
After The user clicks on the extension from the chrome extensions tab it will display an initial home page

* The Transcript page



The Transcript page will load with the Transcription of the video the user is playing or the URL he has selected for transcript. This will have a button of Summarize which will tell user that if he wants to summarize the transcript, he may click this button

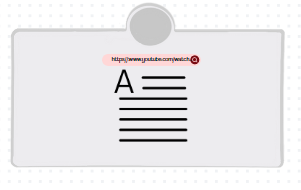
* The Summary page



The summary page will contain the summarized text and two more buttons which the user can chose according to his choice.

The translate button will translate the summary and the Listen button will play the audio of summary.

* The translation page



If the user selects to Translate the text the translated summary will be displayed.

NOTE: the above interface is a proposed description and representation of the user interface. The final interface may vary.

##### 5. TYPES OF TESTS (With Implementation)

Unit Tests

Unit testing for the "YouTube Transcript Summarizer" involves systematically assessing various aspects of the extension's functionality. This includes testing the user interface to ensure user interactions trigger the expected responses and testing each function to check the expected outcome.

System Tests

System testing is a crucial phase in software development that evaluates the entire software system, rather than individual components or units. System testing will be done to test the whole extension. It will include testing the transcript retrieval, summary retrieval and checking its working for chrome.

**6.References**

* G. Begum, N. Musrat Sultana, and D. Ashritha, “YOUTUBE TRANSCRIPT SUMMARIZER.” Accessed: Oct. 3, 2023. [Online]. Available: https://ijcrt.org/papers/IJCRT22A6393.pdf
* B. Saji, “Creating a YouTube Summarizer - Mini NLP Project,” *Analytics Vidhya*, Jan. 13, 2022. https://www.analyticsvidhya.com/blog/2022/01/youtube-summariser-mini-nlp-project/ (accessed Oct. 1 2023).
* “YouTube Video Transcript Summarization with Hugging Face Transformers | Python NLP Projects,” *www.youtube.com*. https://youtu.be/3V-MJhJvRWg?si=3UxVgSstYXbNXFqU (accessed Oct. 5, 2023).
* S. Jain, J. Jain, G. Kaur, Y. Rani, "YouTube Transcript Summarizer", \_International Journal of Science and Research\_, vol. 12 issue 2, February 2023