

SRS DOCUMENT

Project Name: Building Bridges and Inclusive Virtual Meeting

Group Number: 07

Group Member Details:

a) **Ritik Kumar** (21mcme15)

b) **Anuj Taparia** (21mcme09)

Table Of Content:

Topic	Page Number
Introduction	2
1.1 Purpose	2
1.2 Scope	2
1.3 Definitions, Acronyms, Abbreviations	3
1.4 References	3
1.5 Overall Description	3
Overall Description	3
2.1 Product Perspective	3
2.2 Product Functions	3
2.3 User Characteristics	4
2.4 User Constraints	4
2.5 Assumptions and Dependencies	4
2.6 Apportioning Requirements	4
Specific Requirements	4

3.1 Interface Requirements	4-6
3.2 Functional Requirements	6-20
3.3 Performance Requirements	20
3.4 Logical Database Requirements	21
3.5 Design Constraints	21
3.6 Software System Attributes	21
Supporting Information	21-22

1. INTRODUCTION

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a comprehensive overview of the requirements for the development of the system aimed at facilitating collaboration through virtual meetings, including speech-to-text, text-to-speech, sign language recognition, and various configuration and management functionalities.

1.2 Scope

The system aims to provide a user-friendly interface for participants to collaborate effectively in virtual meetings by integrating features such as real-time translation, accessibility settings, agenda management, participant role assignment, and more. The scope of this document encompasses the functional and non-functional requirements, interface specifications, and constraints of the system.

1.3 Definitions, Acronyms, Abbreviations

- SRS: Software Requirements Specification
- UI: User Interface
- API: Application Programming Interface

- ASL: American Sign Language
- CNN: Convolution Neural Network

1.4 References

- <https://www.sciencedirect.com/science/article/pii/S2667305321000454>
- A generous amount of AI Tools

1.5 Overall Description

The system will be a web-based application accessible through standard web browsers. It will utilize modern web technologies, python libraries like OpenCV, Numpy, and TensorFlow along with various CNN architectures and models and APIs to provide real-time collaboration features, language translation, accessibility settings, and meeting management functionalities.

2. OVERALL DESCRIPTION

2.1 Product Perspective

The system will serve as a standalone platform for virtual meetings, providing essential features for collaboration and communication. It may integrate with existing virtual meeting platforms or services through APIs for enhanced functionality.

2.2 Product Functions

- Sign language recognition.
- Create, Join, and Leave meeting functions
- Meeting configuration and customization.

2.3 User Characteristics

The system is designed for users who want to participate in virtual meetings, including individuals with varying technical proficiency levels. Users may include professionals, educators, students, and individuals having physical disabilities (dumb people).

2.4 User Constraints

- Users must have access to a compatible device with internet connectivity.
- Users should have basic knowledge of virtual meeting etiquette and features.
- Users should know ASL.

2.5 Assumptions & Dependencies

- Assumption: Users will have access to stable internet connections.
- Dependency: Integration with third-party APIs for gesture recognition, translation, and accessibility features.

2.6 Apportioning Requirements

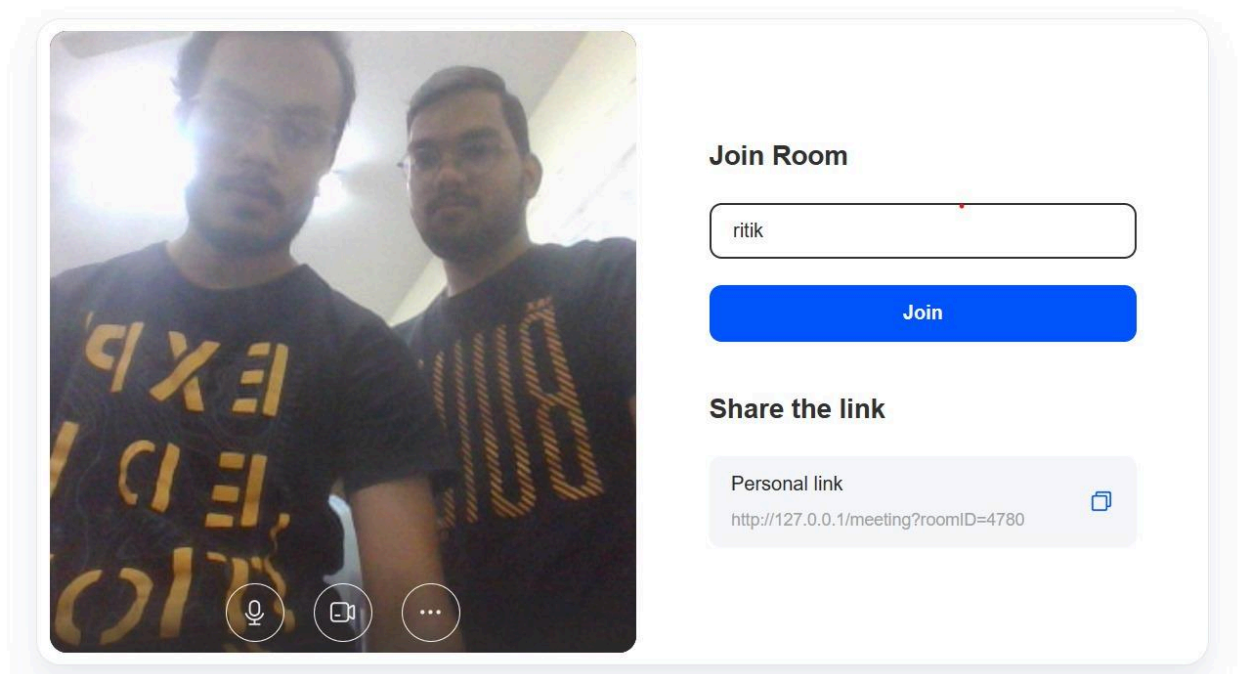
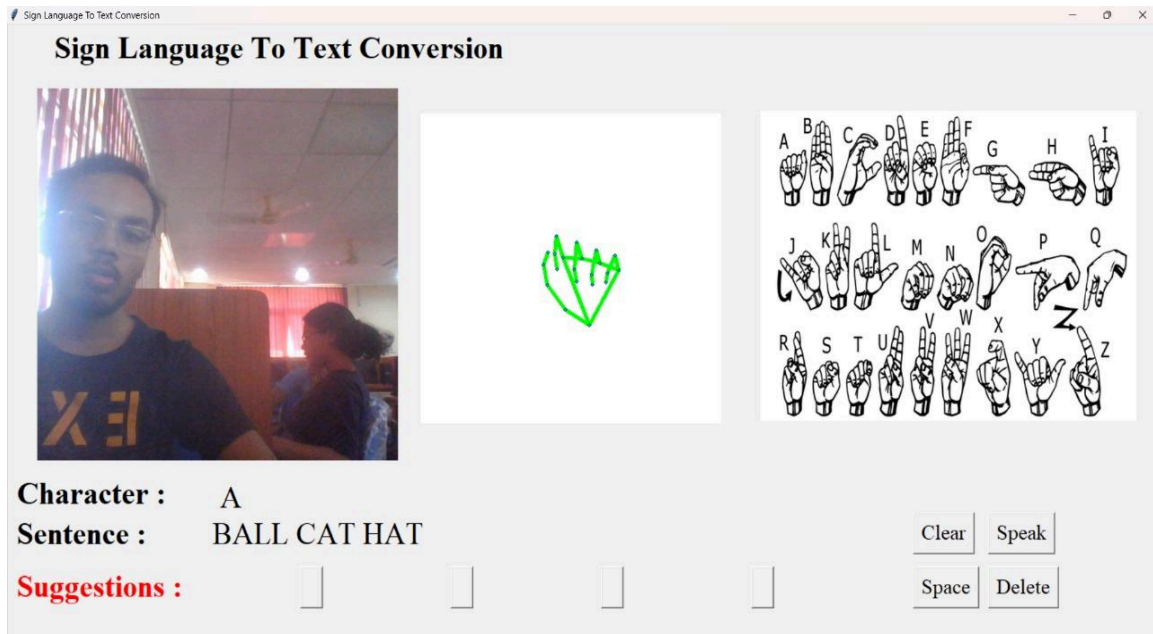
Future enhancements may include:

- Integration with additional virtual meeting platforms.
- Enhanced accessibility features.
- Advanced AI capabilities for speech and sign language recognition.

3. SPECIFIC REQUIREMENTS

3.1 Interface Requirements

- 3.1.1 User Interface: The system will have a user-friendly web-based UI accessible through standard web browsers.



- 3.1.2 Hardware Interface: No specific hardware requirements.
- 3.1.3 Software Interface: Integration with third-party APIs for gesture recognition, translation, and accessibility features.

- 3.1.4 Communication Interface: Real-time communication between meeting participants through text, audio, and video.

3.2 Functional Requirements

- 3.2.1 Use case model:

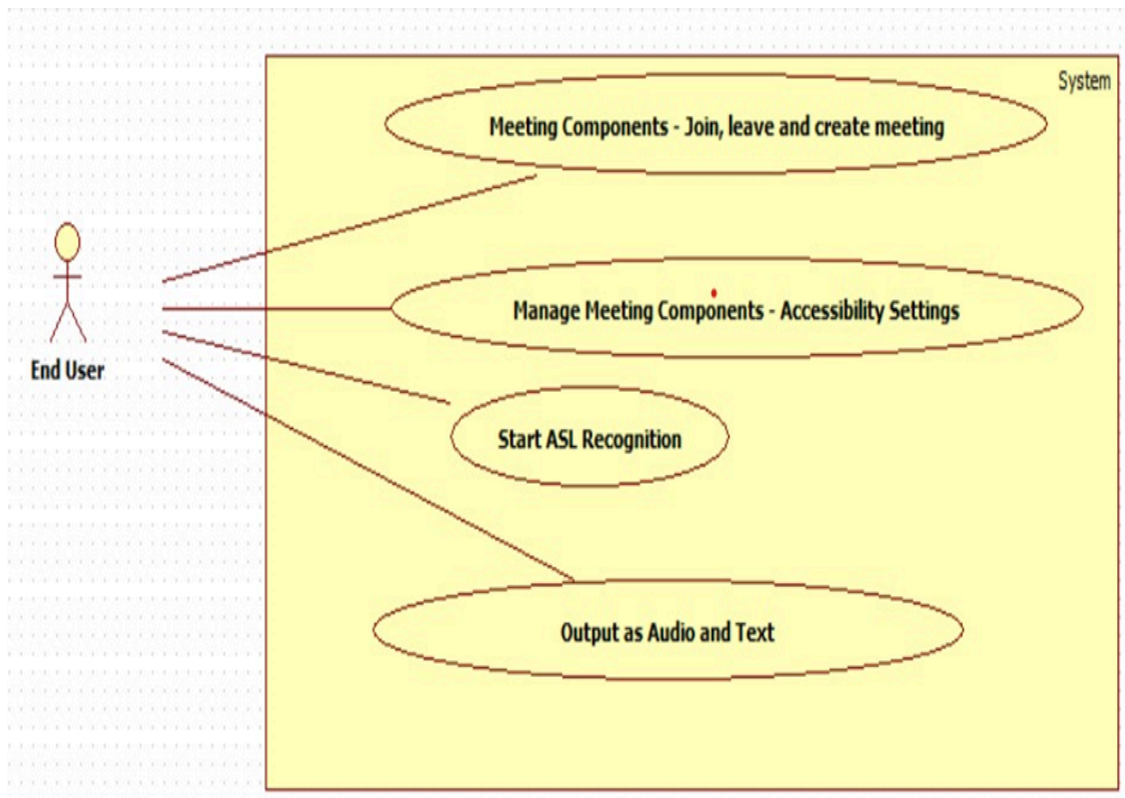


Fig: Use Case Diagram

- 3.2.2 Use Case Specifications
 - a) Use Case Specifications 1:

Use Case ID:	IVMPD1.0		
Use Case Name:	Meeting Management - Join, Leave, and Create		
Created By:	Anuj Taparia	Last Updated By:	Ritik Kumar
Date Created:	09-02-24	Date Last Updated:	04-04-24

Actors:	- End User
Description:	<p>This use case involves the management of virtual meetings by the end user, including joining existing meetings, leaving meetings, and creating new meetings.</p> <p>These functionalities facilitate user interaction and participation within the virtual meeting platform.</p>
Preconditions:	<ol style="list-style-type: none"> 1. The end user is logged into the virtual meeting platform. 2. The virtual meeting platform is accessible and functional.
Postconditions:	The end user successfully joins, leaves, or creates a virtual meeting.
Normal Flow:	<ol style="list-style-type: none"> 1. The end user accesses the meeting management interface within the virtual meeting platform. 2. End-user selects the option to join an existing meeting. 3. The platform verifies the end user's credentials and grants access to the selected meeting. 4. End user participates in the ongoing meeting activities. 5. The end user decides to leave the meeting. 6. The end user returns to the meeting management interface. 7. End-user selects the option to create a new meeting. 8. The platform validates the provided information and creates a new meeting. 9. The end user receives confirmation of the successful creation of the meeting.

Alternative Flows:	A1 - If the end user encounters issues while joining a meeting (e.g., invalid credentials), the platform notifies the end user and provides troubleshooting steps. A2 - If the end user encounters issues while creating a new meeting (e.g., conflicting schedules), the platform prompts the end user to try again.
Exceptions:	If the virtual meeting platform experiences technical difficulties, such as server downtime, the platform notifies end users and advises them to try again later or contact support.
Includes:	<ul style="list-style-type: none"> - Joining Existing Meetings - Leaving Meeting - Creating New Meeting
Priority:	High
Frequency of Use:	Regularly by end users to manage their participation in virtual meetings.
Business Rules:	-----
Assumptions:	<ul style="list-style-type: none"> - End users have stable internet connectivity to access the virtual meeting platform. - The virtual meeting platform supports concurrent meetings and user interactions.

b) Use Case Specifications 2:

Use Case ID:	IVMPD2.0		
Use Case Name:	Manage Meeting Components - Accessibility Settings		
Created By:	Anuj Taparia	Last Updated By:	Ritik Kumar
Date Created:	18-02-24	Date Last Updated:	04-04-24

Actors:	- End User
Description:	This use case involves the capability to manage various meeting components, including configuring accessibility settings, to ensure a well-organized and inclusive virtual meeting for physically disabled individuals.
Preconditions:	<ol style="list-style-type: none"> 1. The user is logged into the virtual meeting platform. 2. The virtual meeting is scheduled or ongoing.
Postconditions:	Meeting components, and accessibility settings, are successfully configured, contributing to organized and accessible virtual meetings
Normal Flow:	<ol style="list-style-type: none"> 1. End User accesses the meeting management interface. 2. End User configures accessibility settings, such as closed captions and assistive technologies. 3. End User saves the configured meeting components. (A1)
Alternative Flows:	A1 - If there is an error in saving the configured meeting components, the platform displays an error message and prompts the Meeting Host to retry.

Exceptions:	
Includes:	- Real-time ASL Gesture Recognition
Priority:	High
Frequency of Use:	Frequent, especially by individuals with hearing impairments relying on ASL for communication.
Business Rules:	-----
Assumptions:	<ul style="list-style-type: none"> - The device used by the user has a functional camera and meets the minimum hardware requirements for running the ASL recognition application. - The ASL recognition model deployed in the application has been trained on a diverse dataset of ASL gestures and exhibits satisfactory performance in real-world conditions. - The ASL recognition application provides user-friendly interfaces for gesture input and output translation, ensuring ease of use for individuals with varying levels of technical proficiency.

c) Use Case Specifications 3:

Use Case ID:	IVMPD3.0		
Use Case Name:	ASL Recognition - Recognize User Gestures		
Created By:	Anuj Taparia	Last Updated By:	Ritik Kumar
Date Created:	18-02-24	Date Last Updated:	04-04-24

Actors:	- End User
Description:	This use case involves the process of recognizing American Sign Language (ASL) gestures in real time using a trained ASL recognition model. The goal is to provide accessibility and inclusivity for individuals with hearing impairments by enabling them to communicate through ASL gestures, which are translated into text or audio output.
Preconditions:	<ol style="list-style-type: none"> 1. The user has access to a device (e.g., smartphone, tablet, computer) with a camera and the ASL recognition application installed.
Postconditions:	The ASL gestures performed by the user are successfully recognized and translated into text.
Normal Flow:	<ol style="list-style-type: none"> 1. End User launches the ASL recognition application on their device. 2. The platform activates the device camera to capture ASL gestures performed by the user. 3. End User performs ASL gestures in front of the camera, aiming to express desired messages. 4. The platform processes the captured video feed and applies the trained ASL recognition model to recognize gestures. 5. The ASL recognition model identifies and interprets the ASL gestures in real time. (A1)
Alternative Flows:	A1 - If the ASL recognition model encounters difficulty in recognizing certain gestures, the platform displays no message.

Exceptions:	
Includes:	- User Role Assignment - Permission Configuration
Priority:	Medium
Frequency of Use:	Regularly before and during virtual meetings to ensure proper access control and collaboration.
Business Rules:	User roles and permissions should align with the responsibilities of individuals within the meeting.
Assumptions:	<ul style="list-style-type: none"> - Meeting Hosts have the necessary permissions to manage user roles and permissions. - The virtual meeting platform supports real-time updates to user roles and permissions.

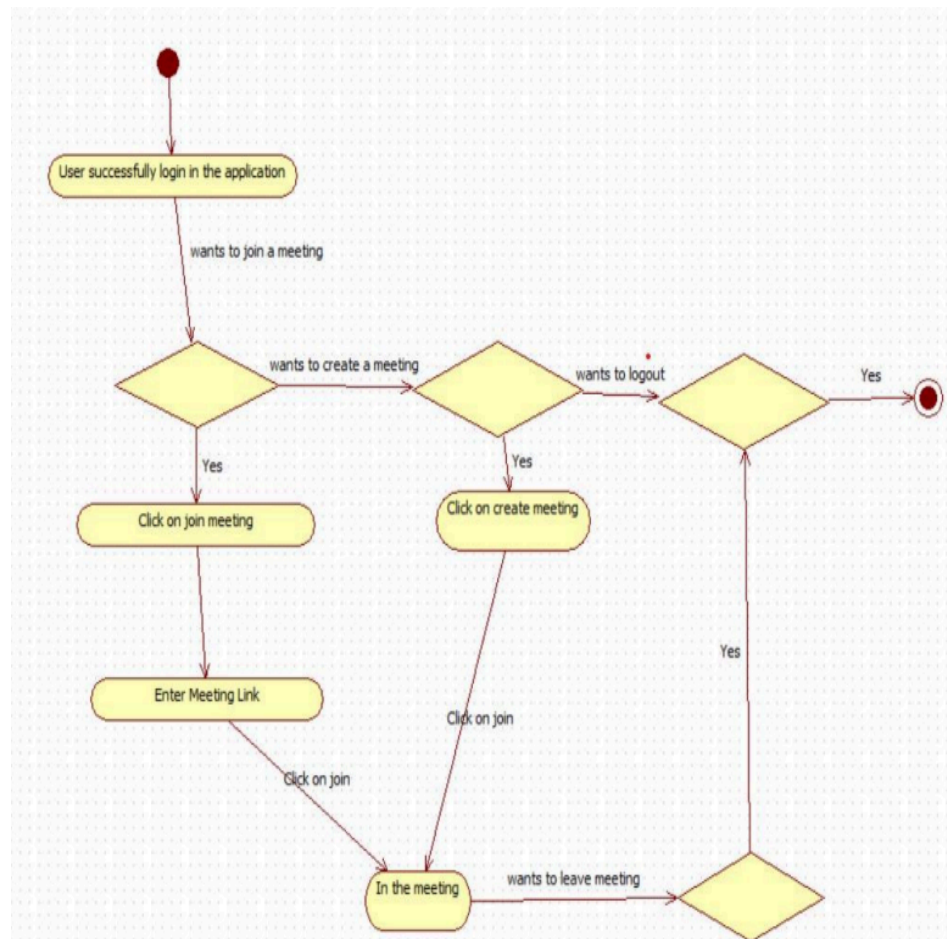
d) Use Case Diagram 4:

Use Case ID:	IVMPD4.0		
Use Case Name:	Output as Audio and Text		
Created By:	Anuj Taparia	Last Updated By:	Ritik Kumar
Date Created:	18-02-24	Date Last Updated:	04-0-24

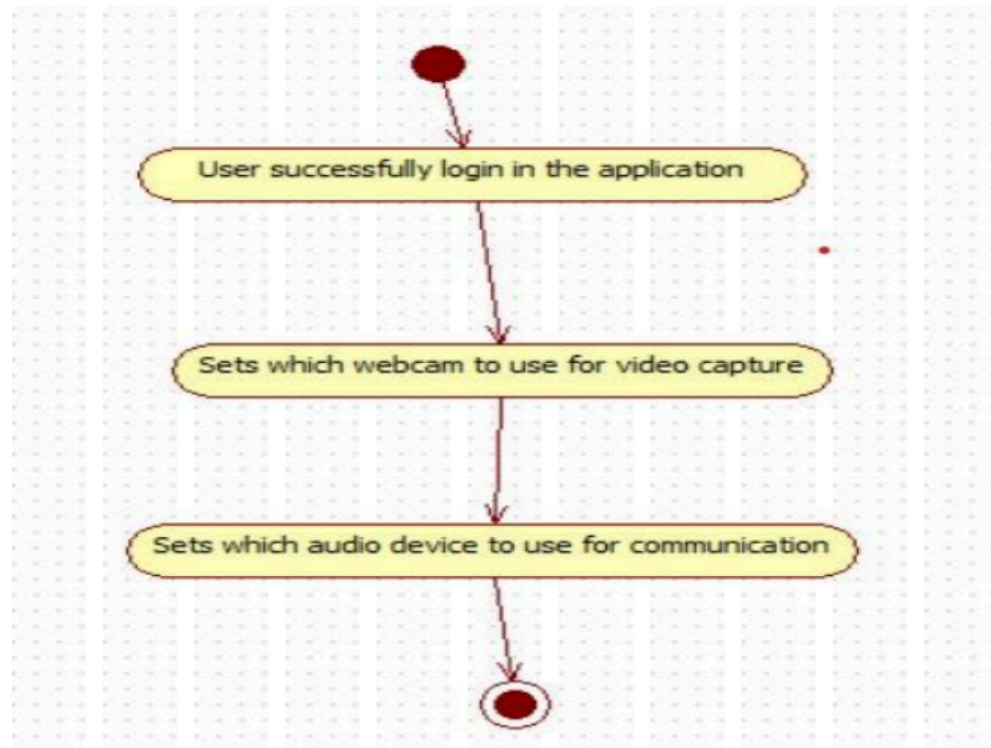
Actors:	- End User
Description:	This use case involves the generation of audio output based on the recognized American Sign Language (ASL) gestures. The purpose is to provide an additional communication channel for individuals with hearing impairments, allowing them to convey messages through both visual ASL gestures and auditory cues.
Preconditions:	<ol style="list-style-type: none"> 1. The ASL recognition application is installed and running on the user's device. 2. ASL gestures have been successfully recognized by the ASL recognition model.
Postconditions:	Audio output corresponding to the recognized ASL gestures is generated and made available to the user.
Normal Flow:	<ol style="list-style-type: none"> 1. End User performs ASL gestures in front of the device camera, and the ASL recognition model accurately recognizes the gestures. 2. The platform receives the recognized ASL gestures from the ASL recognition module. 3. The platform converts the recognized ASL gestures into corresponding audio representations using a predefined mapping or synthesis algorithm. (A1) 4. The generated audio output is played back to the user through the application interface or external speakers/headphones. 5. The End User receives the audio output and interprets the conveyed message in combination with the visual ASL gestures. (A2)
Alternative Flows:	A1 - If the ASL recognition model fails to recognize certain gestures accurately, the platform displays no message.

	A2 - If the audio generation process encounters technical issues or limitations, such as insufficient resources for real-time synthesis, the platform notifies the user and prompts for retry or suggests troubleshooting steps.
Exceptions:	
Includes:	<ul style="list-style-type: none"> - ASL Gesture Recognition Integration - Audio Generation from Recognized Gestures
Priority:	Medium
Frequency of Use:	Frequent, especially in scenarios where individuals with hearing impairments rely on auditory cues for communication alongside visual ASL gestures.
Business Rules:	
Assumptions:	<ul style="list-style-type: none"> - The ASL recognition application provides seamless integration between gesture recognition and audio generation functionalities, ensuring a cohesive user experience. - The audio generation module employs robust algorithms or synthesis techniques capable of accurately representing ASL gestures in auditory form. - Users have access to compatible audio output devices (e.g., speakers, headphones) to receive the generated audio output effectively.

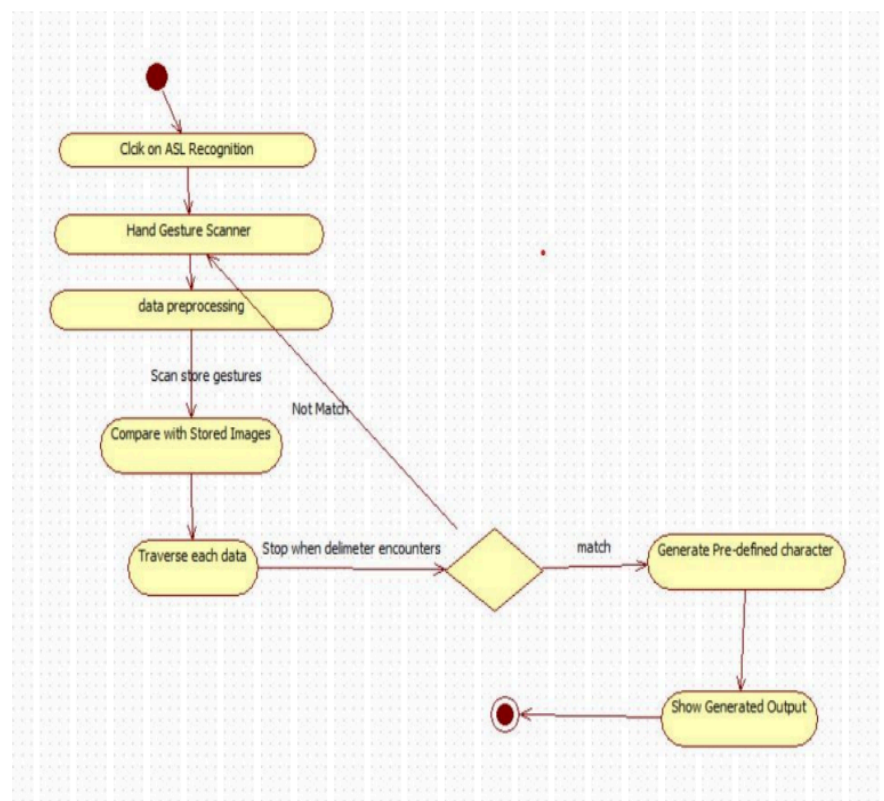
- 3.2.3 Activity Diagram:
 - a) For Meeting Components - Join, leave and Create Meeting:



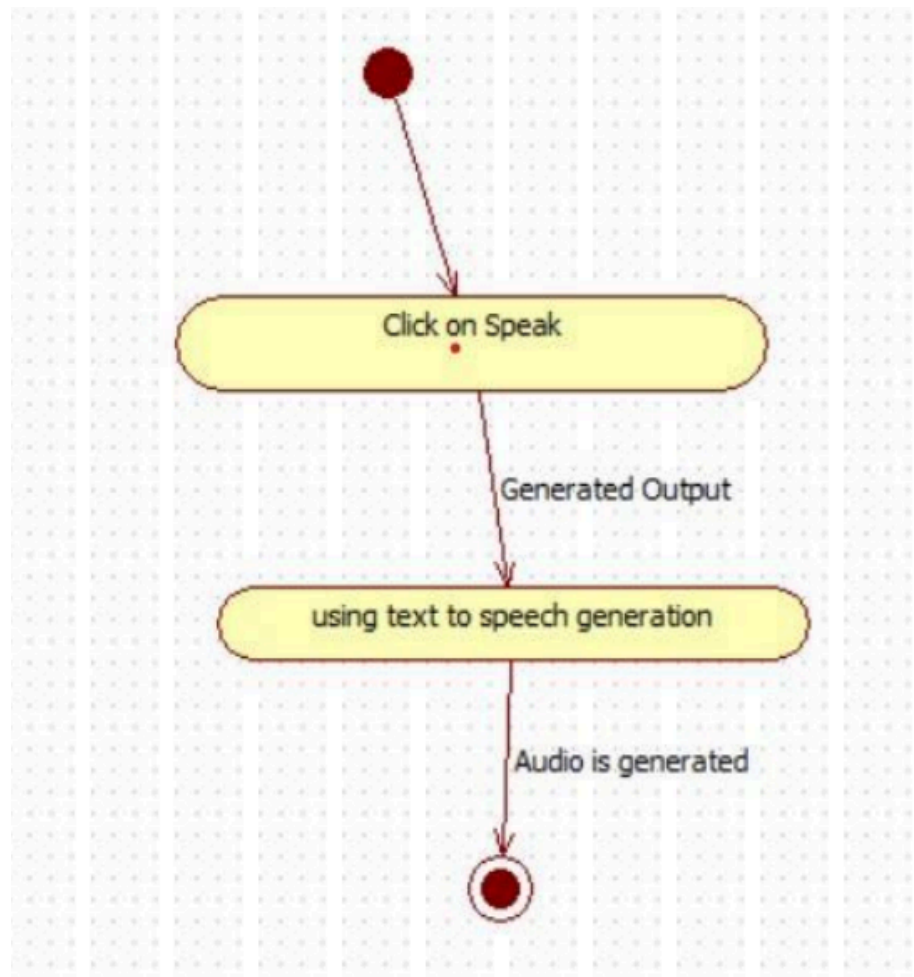
- b) For Manage Meeting Components:



c) For Start ASL Recognition:



d) For Output as Audio and Text:



- 3.2.4 Analysis Classes:
 - Process Description:
 - a) **Create Meeting:**

Pre-condition: User has appropriate permissions to create a meeting.

Input: Meeting details (e.g., title, agenda, participant list).

Output: Confirmation message indicating successful creation of the meeting.

Logic: The process validates the input meeting details, creates a new meeting instance, and stores the meeting information in the database. It then notifies the user about the successful creation of the meeting.

b) Join Meeting:

Pre-condition: User has been invited to or has access to the meeting.

Input: Meeting ID or invitation link.

Output: Confirmation message indicating successful joining of the meeting.

Logic: The process verifies the provided meeting ID or invitation link, checks the user's credentials, and grants access to the meeting. It then updates the meeting attendance records and notifies the user about successful joining.

c) Leave Meeting:

Pre-condition: User is currently part of the meeting.

Input: Leave request.

Output: Confirmation message indicating successful leaving of the meeting.

Logic: The process receives a leave request from the user, removes the user from the list of meeting participants, and updates the meeting attendance records accordingly. It then notifies the user about successful leaving of the meeting.

d) Gesture Recognition:

Pre-condition: Sign language recognition feature is activated.

Input: Live video stream of user's sign language gestures.

Output: Translated text/audio of recognized gestures.

Logic: The process continuously analyzes the incoming video stream, detects sign language gestures, and translates them into text or audio using gesture recognition algorithms. It then sends the translated output to the appropriate components for further processing or display

e) Text Generation:

Pre-condition: Text generation feature is activated.

Input: Text input from gesture recognition process.

Output: Text-to-speech output or displayed text.

Logic: The process receives text input from gesture recognition, processes it as needed (e.g., formatting, language processing), and generates corresponding text-to-speech output or displays the text within the meeting interface. It may also handle any additional functionalities related to text generation, such as language translation or speech synthesis.

- Data Dictionary:

a) Name:

Description: The name of the meeting participant.

Type: String

Constraints: Maximum character limit may apply.

b) Email:

Description: The email address of the meeting participant.

Type: String (Email format)

Constraints: Valid email format required.

c) Mobile Number:

Description: The mobile number of the meeting participant.

Type: String

Constraints: Valid phone number format may apply.

d) Password:

Description: The password of the meeting participant.

Type: String

Constraints: Password strength requirements may apply.

e) Meeting Link (ID):

Description: The unique identifier or link associated with the meeting.

Type: String

Constraints: Unique identifier or link format may vary.

f) Join/Create:

Description: Indicates whether the participant joined the meeting or created it.

Type: String (Enumerated)

Values: "Join", "Create"

3.3 Performance Requirements

- The system should respond promptly to user interactions, with minimal latency.
- Real-time translation and recognition features should operate swiftly and accurately.

3.4 Logical Database Requirements

- The system requires a database to store user profiles, agenda details, and accessibility settings.

3.5 Design Constraints

- The system's design should prioritize simplicity and ease of use for a diverse user base.
- Compatibility with various web browsers and devices should be ensured.

3.6 Software System Attributes

- 3.6.1 Reliability: The system should operate reliably, with minimal downtime and errors. The virtual meeting system ensures reliability by implementing robust error handling mechanisms to minimize system downtime.
- 3.6.2 Availability: The system should be available 24/7, with robust infrastructure and redundancy measures. It utilizes scalable infrastructure and load balancing techniques to handle varying levels of user traffic and maintain consistent availability. It implements multi-factor authentication and access controls to prevent unauthorized access to meetings and sensitive information.
- 3.6.3 Security: The system should implement robust security measures to protect user data and ensure privacy.
- 3.6.4 Maintainability: The system should be easy to maintain and update, with modular components and documentation.
- 3.6.5 Portability: The system should be compatible with a wide range of devices and platforms.

4. Supporting Information

- Glossary: Definitions of technical terms, acronyms, and abbreviations used throughout the document.
- Appendices: Additional documents, diagrams, or reference materials that provide further context or detail for understanding the software requirements.
- Contact Information: Details for stakeholders, project managers, and development team members, including names, roles, email addresses, and phone numbers, to facilitate communication and collaboration.

- Legal Information: Any legal disclaimers, copyrights, or licenses relevant to the software project, ensuring compliance with legal regulations and protecting intellectual property rights.
- Acknowledgments: Recognition of individuals or organizations that contributed to the development of the SRS document or the software project, expressing gratitude for their support and assistance.

Overall, this SRS document provides a detailed overview of the requirements for the development of the virtual meeting collaboration system. It serves as a reference for the design, development, and testing phases of the project.