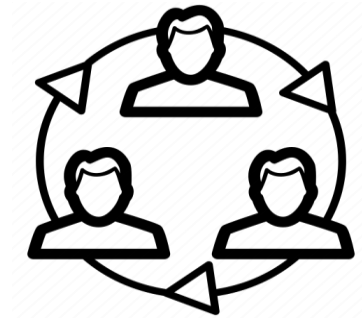


VARTALAAP [वार्तालाप]

Conversation

A simple, easy to use conversation application using video, audio and text communication

Under the mentorships of Miss Sakshi Pandey and Mr. Gopal Goyal.



Animesh Mishra

B.TECH (Computer Science) , IV Semester

Email : animeshcsejsr@gmail.com

Phone No :7091182002

Content

- Project Overview
- Agile Development Plan
- System Design
- User interface
- Learning and way forward

Overview

Objective

Is to build a Microsoft Teams clone i.e fully functional prototype with at least **one mandatory functionality** - a minimum of two participants should be able connect with each other using your product to have a video conversation

System Developed(Technology Used)

Developed a client and server communication application using WebSockets through socket-io. Peer to Peer Media Stream done thorough WebRTC(peerJS). Backend program hosted through Express Js, Frontend developed by React, bootstrap and Semantic-UI and NodeJs for developing the application.

System Feature

Following feature are successfully developed and implemented in the system.



User authentication



Meeting dashboard



Creating meeting



Multi person video conversation



Multi person text conversation



Raise hand



Take note



Polling



Meeting notification through email



Conversion of text to audio for disabled person

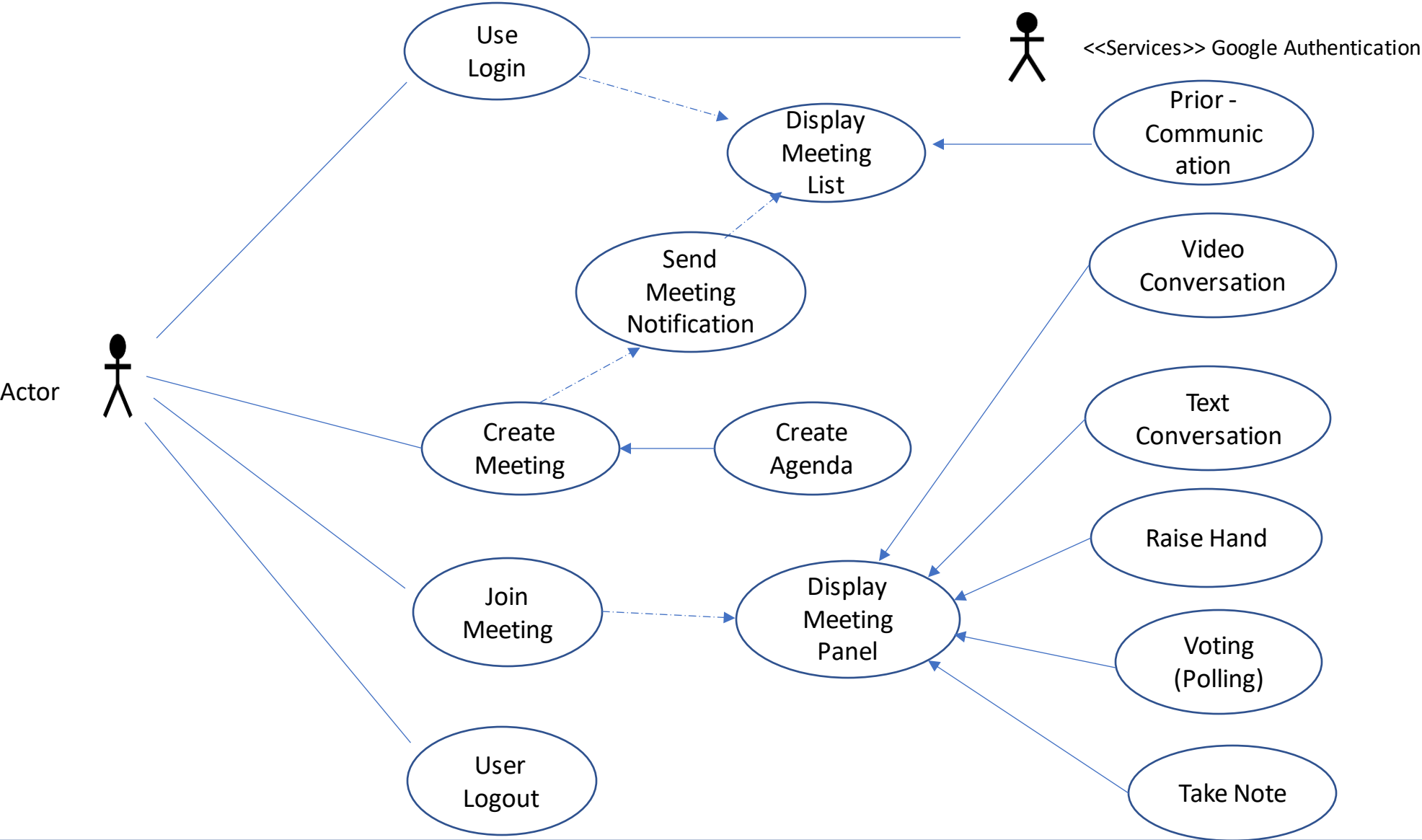
Agile Development Schedule						
Project Name : Development of conversation application using video , audio and text communication (Vartalaap)						
Start Date : 14 th July 2021 to 12 th July 2021 (Under Mentorship of Miss Sakshi Pandey and Mr. Gopal Goel)						
Sprint run		June			July	
MVP		W1 14 th - 20 th June	W2 21 st – 27 th June	W3 28 th – 4 th July	W1 5 th -10 th Jul	W2 11 th – 12 th
1. Explored technologies and designed a work flow.	P					
	S	Learnt to use socket-io, webRTC, ReactJs etc.. Developed a workflow of the application. Discussed it with mentor a who suggested some changes				
2. Developed media streaming and text sharing between two clients	P					
	S	Developed an app capable of streaming media between 2 peers.. Backlog: chat Application				
3. Development user authentication, database integration and enabling multiple clients interface.	P					
	S	Covered the backlog from previous week.. Developed the feature for allowing multiple users to join.. Added Database support. Discussed with colleagues who suggested some features				
4. UI development and additional features. (Raise Hand, Record Notes, Polling, Echo Cancellation,	P					
	S	An app was ready which supported raise hand, record notes, Polling. Brought the application to its present UI. Discussed with mentor who suggested some features.				
5. Preparation of documentation and presentation	P					
		A design document was ready along with readme files..				



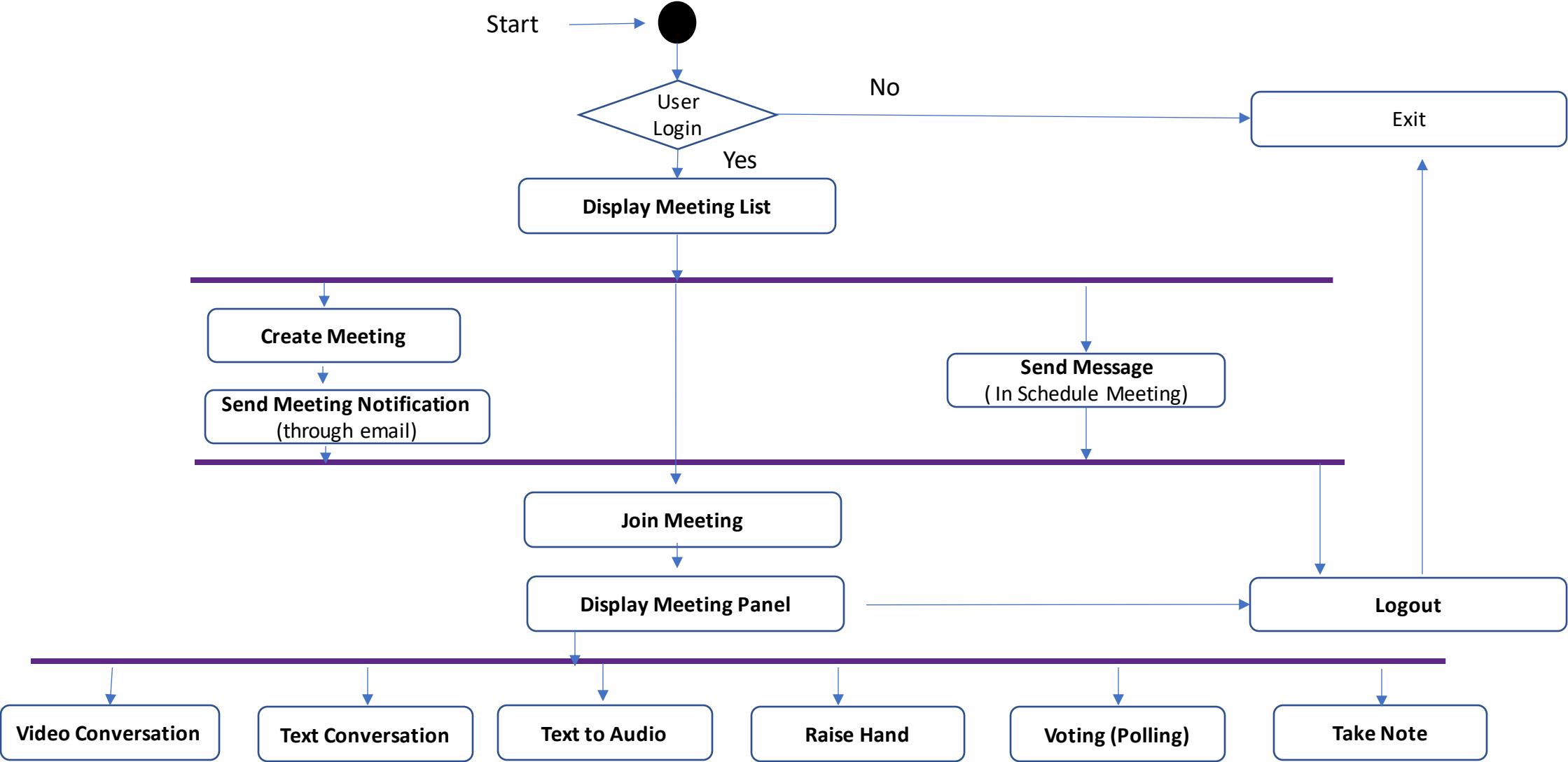
System Design

- Use Case Diagram
- Activity Diagram
- Architecture Diagram

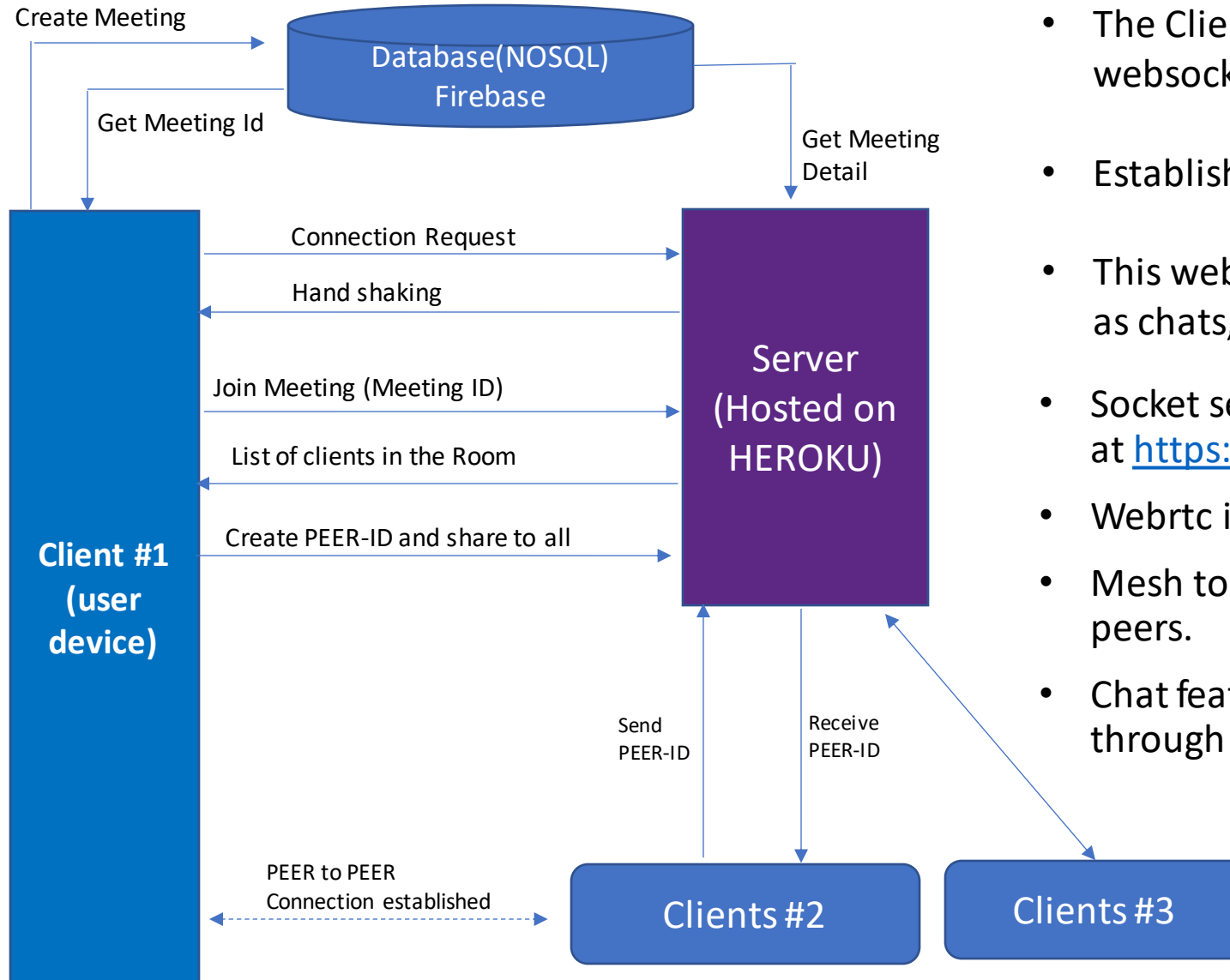
Use Case Diagram



Activity Flow Diagram



Architecture between Client and Server for video & audio streaming



- The Client and the Server are connected through websockets.
- Establishes peer connection between clients.
- This websocket architecture also facilitates features such as chats, raise hand.
- Socket server is currently hosted on a heroku server at <https://boxing-syrup-20682.herokuapp.com>
- Webrtc is used for streaming media amongst peers
- Mesh topology is used for streaming media amongst peers.
- Chat features and raise hand features are also facilitated through websockets.

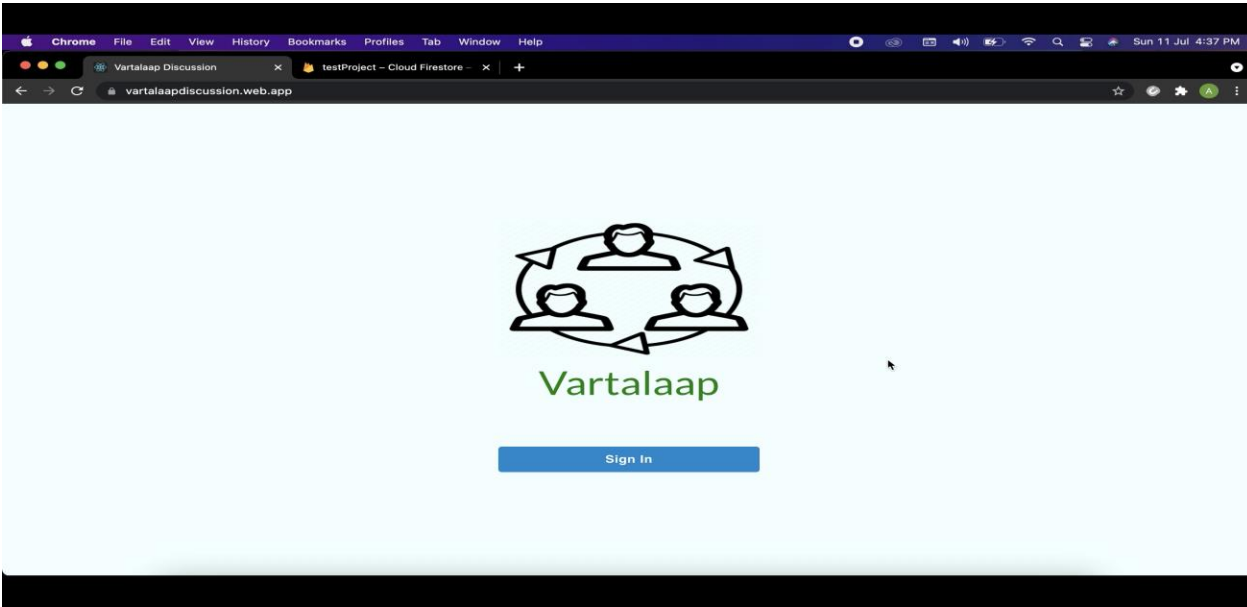


VARTALAAP [वार्तालाप]
Conversation

User Interface

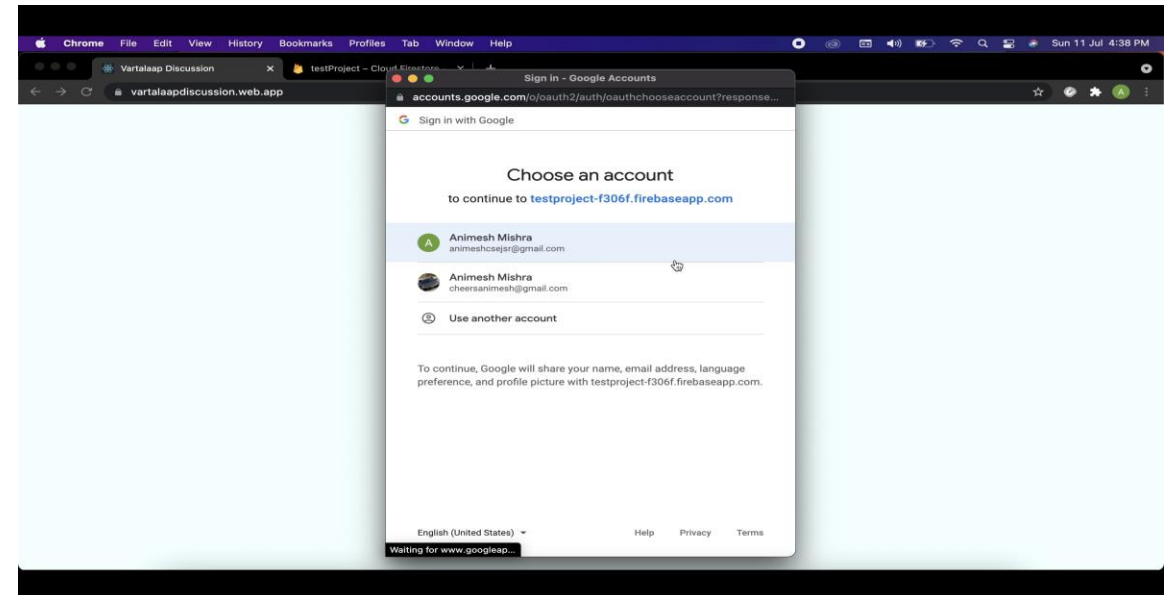
- User login page
- User Authentication Page
- Meeting Dashboard.
- Create Meeting
- Online Chart
- Meeting panel
 - Taking Note
 - Raise Hand
 - Text chart
 - Polling

User Interface

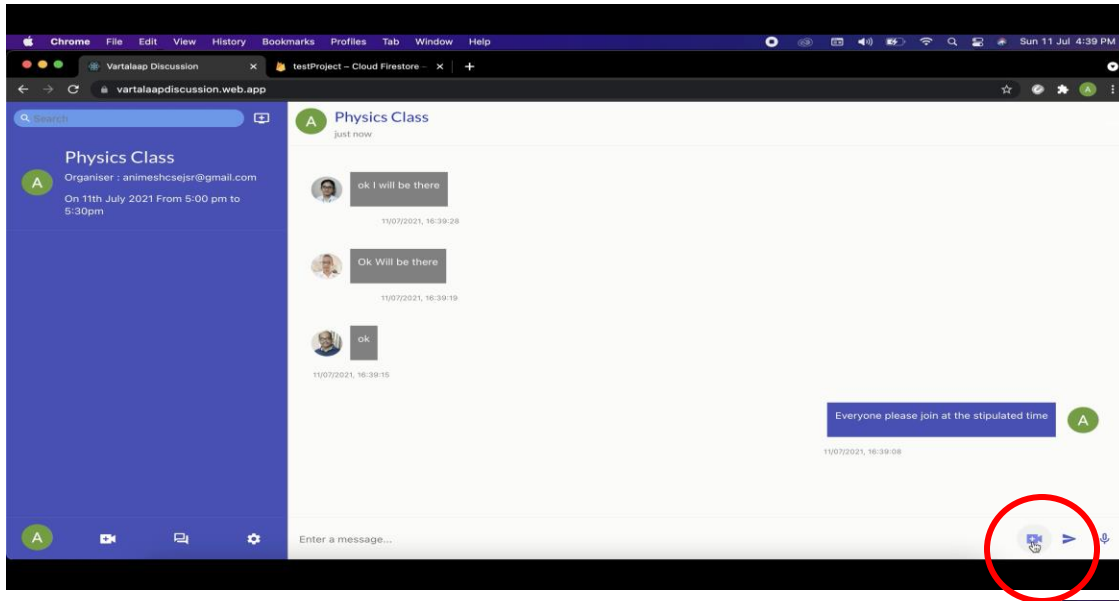


- Google Authentication is required to get into the Application.
- So on clicking sign, In one has to use his/her google account to log in.
- Feature is enabled through Firebase Authentication

- This is first page to get into the application . User has to type URL : <https://vartalaapdiscussion.web.app/>
- The website is hosted on Firebase.
- Recommended to be used on Google Chrome



User Interface

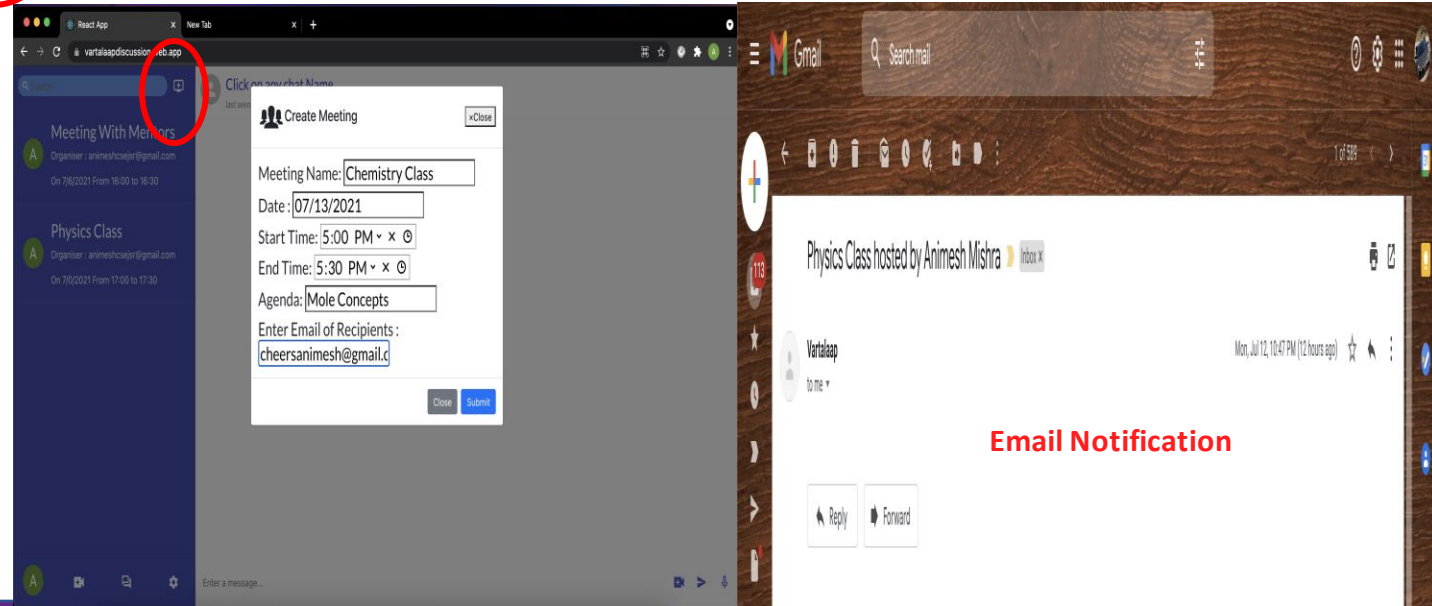


Meeting List (Dashboard)

- Displays all the scheduled meetings
- User can select the respective meeting and chat with meeting members or could just drop a message.
- User can just clicking button and can join a meeting
- User can also create a meeting by pressing button number

Create Meeting

- User has to enter the required details such as meeting name, meeting date, start time, end time and agenda to create a meeting.
- A email notification will send to the invitees.



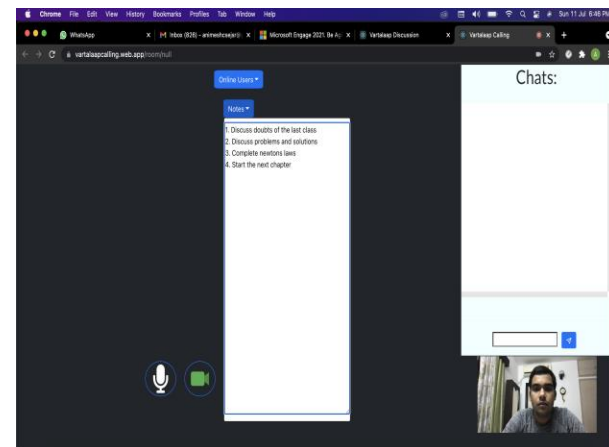
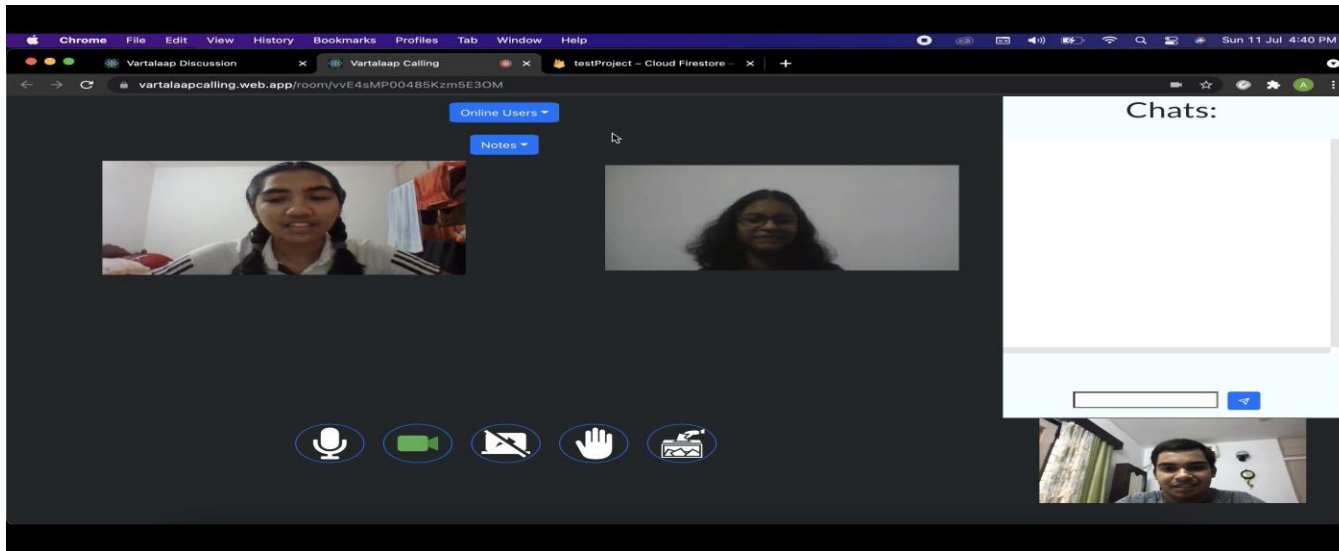
User Interface

Meeting Room (Panel)

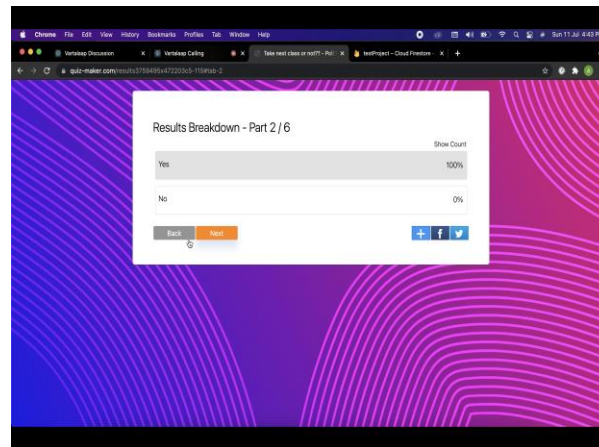
Conversation Panel which includes video audio and text conversation.

The lower right section consists of media stream from the client's machine

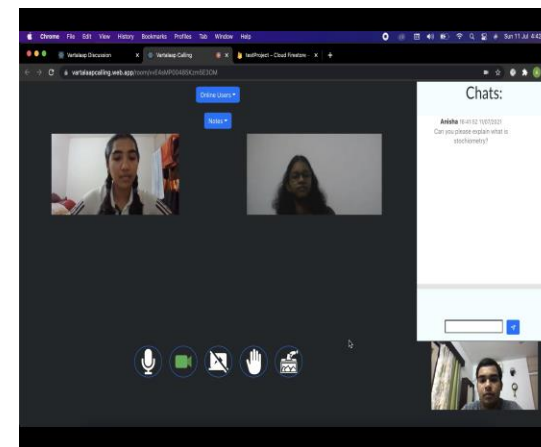
The other video streams in the upper left portion contains media stream from other clients.



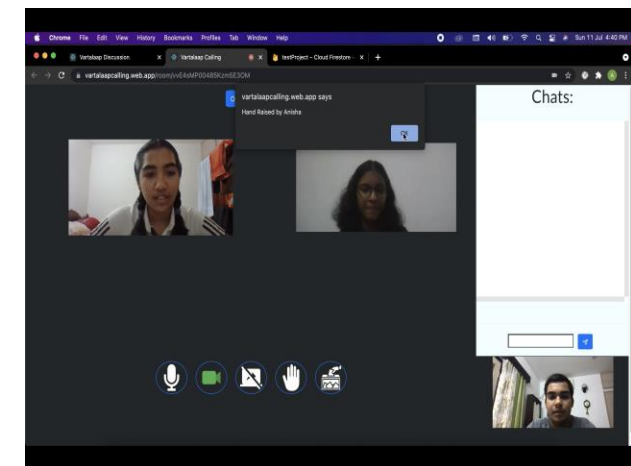
Take Notes During Meeting



To Vote During the Meeting

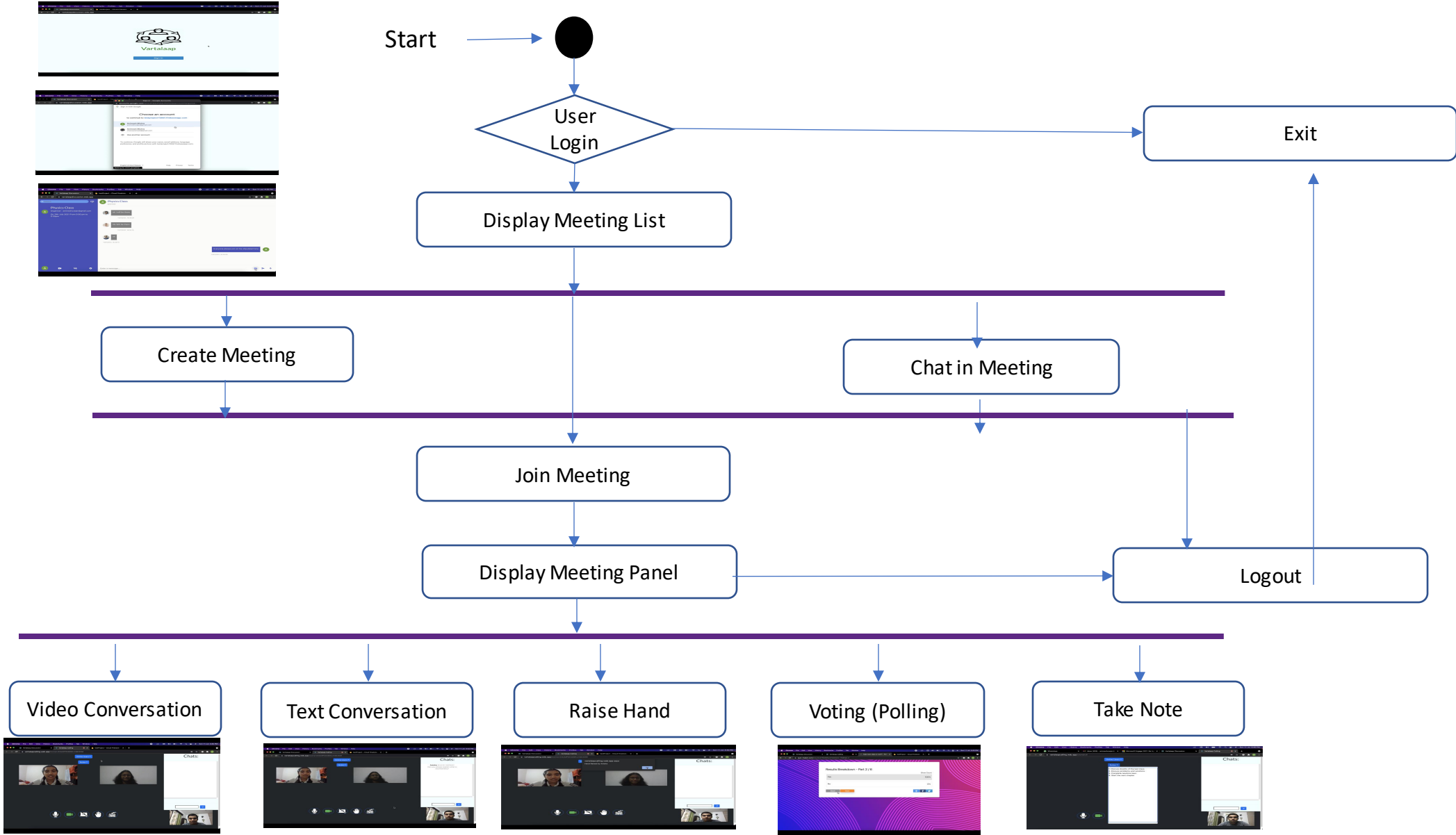


Message in between the meetings



Raise your hand to grab attention

Activity Flow Diagram with HMI



Learning and Way forward

- System is successfully developed and I have tested and validated by using it with my colleagues .
- Learnt to develop projects using agile methodology.. Was exposed to various frameworks in Javascript.. Learnt to stream media through WebRTC and trigger events through web sockets.. Learnt to develop through React.. Interacted with mentors and gained useful insights..
- Could have improved UI and add various features of the app had there been a little more time..
- I have planned to integrate the following features to the application to make it fully functional.



Background noise filtration



Background picture filtration



Screen /Presentation/control sharing



Integration of meeting schedule with Calendar



Recording of conversation.



Generate minutes of meeting(MOM)



Blackboard feature using stylus and writing pen