**NOTE: FOCUS ON FEATURE, NOT ON UI AND DESIGN.**

Features of the App:

* Group Video call
* Chat
* Send voice chat
* MAX\_PARTICIPANTS = 3
* Share files. When we send file it will have a preview in the chat.
* Screen Share
* Record meet
* Handle large image sharing, limit 2mb.
* Handling image compression.
* Emoji support (render them properly on frontend side)
* **Implement Url Metadata properly 1st show chat instantlt , the call the rest api to extract url metadata then updatate in the front end side.**
* Implement llm on the chat.
* Command support(with a drop down to show all existing commands)
* Optional Features: white board without sync feature(redirect to a new page with white board component), background blur, ai integration in the chat.
* Implement later: can share any type of file , like pdf , docs and can download it directly form chat.

Staring Page:

* Name of the app
* Text Field To enter username and Topic of meeting, room id.
* Two buttons: Create new room, Join Existing room (in the center)

Functions:

Create New Room : Takes Name, Topic of meeting and create a new Room with a room id. Then redirect to the video calling page.

Join Existing Room: Takes Name, Topic of meeting , room id to join an existing room. Then redirect to the video calling page.

Video Calling page:

* Will show a participants list, this list will be stored in backend server. And will be fetched through api, when a new user joined.
* Chat will have group chat feature only. We can send emoji, send file, will have link preview(react-link-preview).
* On join new user a sound must be used to notify to everyone. And chat must flashed with user left.
* On leaving the same must happen. And chat must be flashed with user left.
* On new messge new sound must be generated.

Index.js Backend:

* Const Users= [Username:{username:test,personal\_room\_id:84274}]

19.07.24:

* ContextAPi: implement socket.io client , write all functions for chats all states for the chats
* Design Controlss
* JoinRoom.jsx, complete the input fileds, make a generate roomId button.

20.07.24:

* Video part ui design, must be of grid nature and must be responsive, according to users.
* Implement Audio on/off functions, video on/off functions.
* Video Showing on camera on.
* Full Screen Button
* Screen Record functions.

21.07.24:

* Chat section , chat design. Scroll-Bar design in chat section.
* Implement axios to get participant list
* Copy roomId to clipboard.
* Create a White Board button , onClick it redirects to WhiteBoard.jsx page
* Create WhiteBoard.jsx and implement white board.

22.08.24:

* Implement the chatContextApi.jsx , and implement different chat functions and logics
* When /video-call/:roomID route is hitted call a api to check if roomId exist or not , if doesn’t exist redirect to “Room doesn’t exist”.
* When, join room button is clicked , make the user joined the meet.
* Understand the myPeer.on(“open”) event, videogrid.jsx file.
* the grid implementation of videos according to number of participants
* full screen function on video.
* mute myself and video off, mute others.

FIX :

* Chat position left and right properly.
* *after sending an image chat stop reciving further messages. : HANDLE LARGE IMAGE SENDING*