

# SLACK

## Features:

- **Authentication-Authorization:** Providing the general login and signup with the leaders that can invite users.
- **Channels:** The leader may create a channel and the user may join them
- **General Chats:** The leader may create any general chats available to all
- **Private Chats:** The users may create chats to themselves
- **Direct Messaging:** One-on-one messaging between users
- **Threaded Conversations:** Reply to specific messages in a thread, keeping conversations organized.
- **Search Messages:** Search Messages
- **File Sharing:** Sharing Files
- **Voice Calls:** voice calls
- **Video Calls:** called huddling
- **Status:** Available or not
- **Dashboard:** With bookmarks

# Requirements:

## Functional Requirements:

### *User Management*

- **User Registration and Authentication**
  - Sign-up, login, and logout functionalities.
  - Support for OAuth/Social login (e.g., Google, GitHub).
  - Password reset and account recovery.
- **Profile Management**
  - Edit profile information .
  - Manage user preferences .
  - Private chats
- **User Roles**
  - Different roles (Admin, User) with varying levels of access and permissions.

### *Channels and Messaging*

- **Public and Private Channels**
  - Create, join, and manage public and private channels.
- **Direct Messaging**
  - One-on-one and group messaging.
  - Threaded conversations for organized discussions.
- **Message Formatting**
  - Rich text formatting (bold, italics, lists).
  - Emojis and reactions to messages.
  - file sharing, and image attachments.
- **Search Functionality**
  - Search across messages, channels with filters.

## ***File Sharing and Storage***

- **File Upload and Management**

Upload files in messages and channels.

Preview files directly within the platform.

- **Cloud Storage Integration**

Integration with Google Drive, Dropbox, etc., for external file management.

## ***Voice, Video calls***

- **Voice and Video Calls**

Support for one-on-one and group calls.

In-call features like muting, video on/off.

# **Non-Functional Requirements**

## ***Backend***

- **Server-side**

NodeJS, ExpressJS for apis

- **Database**

Database

- **Communication**

WebSocket for real-time messaging.

- **Authentication**

- JWT (JSON Web Token) and google auth

- **APIs**

- RESTful APIs for frontend-backend communication.

- **File Storage**

- Use cloud storage services .

## *Frontend*

- **Frontend Framework**

React.js for building interactive UIs.

- **Responsive Design**

Ensure the website is responsive and mobile-friendly.

- **WebRTC**

For voice, video calls, and screen sharing functionalities.

- **State Management**

Redux for managing application state.

# TeamManagement:

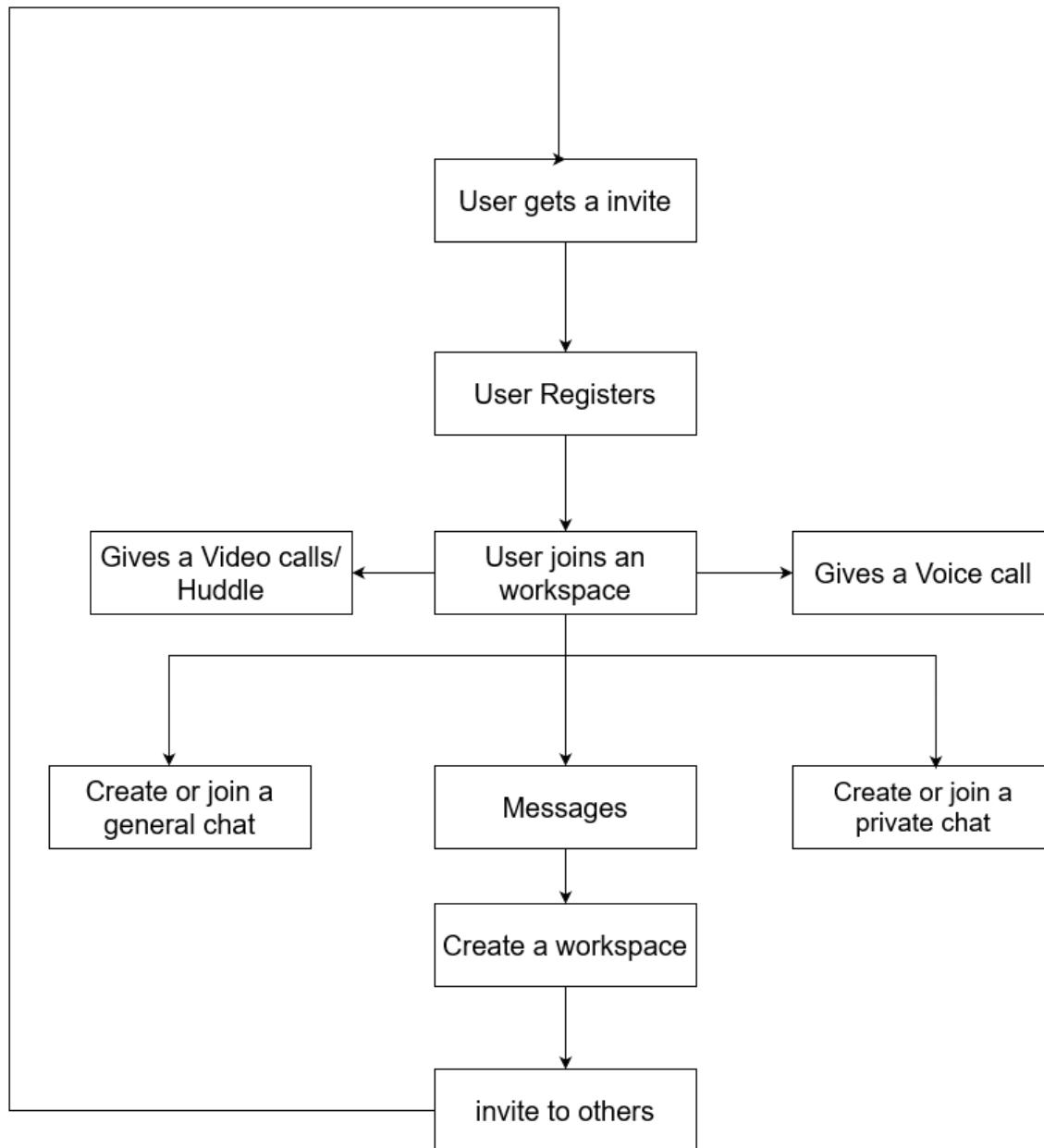
The team for this particular project will consist of 3 members and will work as alpha , beta and gamma according to their availability .

**Aplha member:** Will devote entire time to the project and develop maximum of the functionalities

**Beta member:** Will help in implementing the file uploadation to the cloud which is his expertise.

**Gamma member:** Will help in the implementation of the voice and video calls and all over if there is need.

## Modal:



- **Detemining objectives and solutions:**
- **Identifying and resolving risks:**
- **Develop and test:**

- **Review and plan:**

## **Total Time and Sprints:**

Starting date :16/08/2024

### **Sprint-1:**

Authentication-Authorization(with modals): 2 days

Channels(create,delete,update,get): 1 days

General and private Chats: 1-2 days

### **Sprint-2:**

Direct Messaging: 2 days

Threded Conversations: 2 days

Search Messages: 1 day

### **Sprint-3:**

File Sharing: 1-2 days

Status:1 day

Dashboard: 1 day

## **Sprint-4:**

Voice calls: 2-3 days

Video calls: 2-3 days

## **Sprint-5:**

Remaining Frontend: 3 days

Testing and resolutions: 2 days

## **Sprint-6:**

Version release

Review



# Technical:

**DataBase:** MongoDB

**Schemas:** In the database