Project Design Phase-II

# Technology Stack (Architecture & Stack)

Date: 26-05-2025

Project Name: Video Conferencing App

Maximum Marks: 4 Marks

## Technical Architecture:

The Video Conferencing App is built with a scalable 3-tier architecture including the presentation layer (frontend), business logic (backend), and data storage layer. It integrates WebRTC for real-time video/audio communication and offers support for live chat, screen sharing, and room-based connections.

### Table-1: Components & Technologies

|  |  |  |
| --- | --- | --- |
| Component | Description | Technology |
| User Interface | Web-based interface for video conferencing | React.js, Tailwind CSS |
| Backend Logic | Handles room creation, signaling, authentication | Node.js, Express.js |
| WebRTC | Real-time media communication | WebRTC APIs, PeerJS |
| Database | Stores user data and meeting metadata | MongoDB |
| Chat Service | Real-time chat during meetings | Socket.IO |
| Recording | Capture and download sessions | MediaRecorder API, Node backend storage |

### Table-2: Application Characteristics

|  |  |
| --- | --- |
| Characteristics | Description |
| Open-Source Frameworks | React.js, Node.js, Bootstrap, Tailwind CSS |
| Real-Time Communication | Powered by WebRTC and PeerJS for live video and audio. |
| Scalable Architecture | 3-tier RESTful design using WebSockets and service APIs. |
| Modular Design | Separation of frontend, backend, signaling, and storage. |
| Security | Encrypted streams, JWT-based auth, and role management. |

## References:

• React.js Documentation

• Node.js Best Practices

• WebRTC.org

• https://developer.mozilla.org/en-US/docs/Web/API/MediaRecorder