**Software Requirements Specification**

**for**

**Instant Wave**

**Your Seamless Communication Hub**

**Table of Contents**

**S.No Section Pg No**

1 Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 4

1.5 Overview 4

2 The Overall Description 4

2.1 Product Perspective 4

2.2 Product Functions 5

2.3 User Characteristics 7

2.4 Constraints 7

2.5 Assumptions and Dependencies 8

3 External interface Requirements 9

3.1 User Interfaces 9

3.2 Hardware Interfaces 10

3.3 Software Interfaces 10

3.4 Communications Interfaces 10

4 System Features 11

4.1 Use Case Diagram 13

5 Other Non-Functional Requirements 14

5.1 Performance Requirements 14

5.1.1 Capacity 14

5.1.2 Dynamic Requirements 14

5.1.3 Quality 16

5.2 Software System Attributes 16

5.2.1 Reliability 16

5.2.2 Availability 16

5.2.3 Security 17

5.2.4 Maintainability 17

5.3 Business Rules 17

6 Other Requirements 18

Appendix A: Glossary 18

**1. Introduction**

Instant Wave introduces a paradigm shift in online connectivity, leveraging cutting-edge WebRTC technology to create a seamless platform for real-time interactions. Users can effortlessly engage in group conversations, share audio and video, conduct screen demonstrations, and exchange text messages within an intuitive environment. The platform's user-friendly interface streamlines the process of creating and managing virtual rooms, ensuring that users can dive straight into meaningful discussions with minimal setup.

* 1. **Purpose**

Instant Wave aims to revolutionize online communication with seamless real-time interactions and prioritize security.

* Group conversations
* Multimedia sharing
* Screen demonstrations
* Text messaging
* High-quality streams with minimal latency
* Advanced features like screen sharing
* Strong focus on privacy and security

* 1. **Scope**

Instant Wave revolutionizes online communication with its WebRTC-based platform, prioritizing seamless user experiences. Key features include group conversations, multimedia sharing, and screen demonstrations, enhancing communication efficiency. Emphasizing minimal latency, the platform ensures smooth interactions, while advanced features like screen sharing facilitate collaboration across contexts.

Robust encryption protocols safeguard user data, highlighting the project's commitment to privacy and security. Instant Wave sets a new standard for digital connectivity, offering a versatile, secure, and intuitive platform.

* 1. **Definitions, Acronyms, and Abbreviations.**

|  |  |
| --- | --- |
| **SRS** | Software Requirement Specification |
| **ICE** | Interactive Connectivity Establishment |
| **SDP** | Session Description Protocol |
| **SDK** | Software Development Kit |
| **HTTPS** | Hypertext Transfer Protocol (Secure) |
| **STUN** | Session Traversal Utilities for NAT |
| **RTC** | Real Time Communication |
| **UDP** | User Datagram Protocol |

* 1. **References**

1. WebRTC Guide - <https://webrtc.org/getting-started/overview>
2. WebRTC MDN Web Docs - [https://developer.mozilla.org/docs/WebRTC\_API](https://developer.mozilla.org/enUS/docs/Web/API/WebRTC_API)
3. W3 Recommendation - <https://www.w3.org/TR/webrtc/>
4. Web.dev article - <https://web.dev/articles/webrtc-basic>

**1.5. Overview**

The document covers various aspects, including the purpose and scope of the project, external interfaces, system features, non-functional requirements such as performance and software attributes, and business rules. It provides a structured framework for understanding the intricacies of the Instant Wave, paving the way for effective collaboration, development, and ongoing improvements. The SRS document acts as a foundational document that underpins the successful realization of the Instant Wave vision in reshaping and enhancing the Real Time Communication

1. **The Overall Description**
   1. **Product Perspective**

* Instant Wave introduces a paradigm shift in online communication platforms.
* It leverages WebRTC technology for seamless, real-time interactions.
* The platform offers a user-friendly interface for effortless group conversations and multimedia sharing.
* High-quality streams with minimal latency prioritize lifelike and immersive interactions.
* Advanced features like screen sharing enhance collaboration across various contexts.
* Instant Wave caters to remote work, business meetings, and educational purposes.
* Privacy and security are paramount, with robust encryption protocols safeguarding user data.
* The platform’s versatility adapts to different user groups and scenarios.
* Instant Wave sets a new standard for digital connectivity with its intuitive interface and advanced features.
* Overall, Instant Wave prioritizes user experience, security, and adaptability in online communication.

**2.2 Product Functions**

The major functions that **Instant Wave** performs are described as follows: -

1. **WebRTC Integration**:

**Functionality:** WebRTC (Web Real-Time Communication), JavaScript APIs that provides web browsers and mobile applications with real-time communication via simple application programming interfaces (APIs). It enables peer-to-peer communication directly between browsers without the need for intermediate servers.

**Details:** WebRTC allows for audio, video, and data communication between browsers. It includes components such as getUserMedia (for accessing cameras and microphones), RTCPeerConnection (for establishing direct peer-to-peer connections), and RTCDataChannel (for sending arbitrary data).

**Usage in Project:** In “Instant Wave,” WebRTC is used to establish direct communication channels between users’ browsers for real-time audio, video, and data sharing without the need for additional server infrastructure.

1. **Agora SDK Integration:**

**Functionality:** Agora SDK is a third-party service that provides real-time audio and video communication capabilities. It offers APIs and SDKs for integrating features like voice and video calling, interactive broadcasting, and real-time messaging into applications.

**Details:** Agora SDK offers features such as high-quality audio and video streaming, low-latency communication, and cross-platform compatibility. It also includes advanced features like recording, transcoding, and interactive live streaming.

**Usage in Project:** In “Instant Wave,” Agora SDK is integrated to enhance the audio, video, and screen sharing functionalities within the real-time communication platform. It provides robust capabilities for handling real-time media streams and interactions among multiple users.

1. **Room Management:**

**Functionality:** Room management involves creating, managing, and maintaining chat rooms or channels where multiple users can join and interact in real-time.

**Details:** Room management functionalities typically include creating new rooms, joining existing rooms, leaving rooms, managing room settings (such as privacy and permissions), and handling room-related events (such as user join/leave notifications).

**Usage in Project:** In “Instant Wave,” room management functionality is implemented to facilitate the creation of chat rooms where users can join to participate in real-time communication sessions. This functionality ensures organized and efficient communication among participants.

1. **Bot Integration:**

**Functionality:** Bot integration involves incorporating automated bots or virtual assistants into the communication platform to perform specific tasks or provide assistance to users.

**Details:** Bots can be programmed to perform various tasks, such as greeting new users, providing information or instructions, answering frequently asked questions, moderating conversations, and facilitating interactions.

**Usage in Project**: In “Instant Wave,” a bot is integrated to greet new users when they join a chat room. This automated feature enhances user experience by providing a welcoming and engaging environment for participants.

1. **Audio Communication:**

**Functionality:** Audio communication enables users to transmit and receive audio data in real-time during communication sessions.

**Details:** Audio communication functionalities include capturing audio input from microphones, encoding audio data, transmitting audio streams over network connections, receiving and decoding audio streams, and playing back audio output through speakers or headphones.

**Usage in Project:** In “Instant Wave,” audio communication functionality is implemented using WebRTC and/or Agora SDK to enable users to communicate with each other through voice calls or audio conferences in real-time.

1. **Video Communication:**

**Functionality:** Video communication enables users to transmit and receive video data in real-time during communication sessions.

**Details:** Video communication functionalities include capturing video input from cameras, encoding video data, transmitting video streams over network connections, receiving and decoding video streams, and rendering video output on screens.

**Usage in Project:** In “Instant Wave,” video communication functionality is implemented using WebRTC and/or Agora SDK to enable users to communicate with each other through video calls or video conferences in real-time.

1. **Screen Sharing:**

**Functionality:** Screen sharing allows users to share their screens with other participants in real-time, enabling collaborative activities such as presentations, demonstrations, and remote assistance.

**Details:** Screen sharing functionality involves capturing the contents of a user’s screen or specific application window, encoding the screen capture into a video stream, transmitting the video stream over network connections, and displaying the shared screen content on the screens of remote participants.

**Usage in Project:** In “Instant Wave,” screen sharing functionality is implemented using WebRTC and/or Agora SDK to enable users to share their screens with other participants during communication sessions, facilitating collaborative interactions and content sharing.

1. **User Authentication and Management:**

**Functionality:** User authentication and management involve verifying the identity of users and managing their access to the communication platform and its features.

**Details:** User authentication functionalities include user registration, login, logout, password management, and session management. User management functionalities include user profiles, permissions, roles, and account settings.

**Usage in Project:** In “Instant Wave,” user authentication and management functionality may be implemented to ensure secure access to the communication platform, manage user accounts and permissions, and provide personalized experiences for users.

1. **Real-Time Event Handling:**

**Functionality:** Real-time event handling involves detecting, processing, and responding to various events that occur during communication sessions in real-time.

**Details:** Real-time events may include user join/leave notifications, message sent/received notifications, audio/video stream state changes, screen sharing requests, error messages, and other relevant events.

**Usage in Project:** In “Instant Wave,” real-time event handling functionality is implemented to manage and respond to events such as user join/leave notifications, incoming messages, audio/video stream state changes, and other relevant events, ensuring smooth and interactive communication experiences for participants.

**2.3 User Characteristics**

* **Tech-Savvy Users**: These users are comfortable with technology and can quickly adapt to new online communication tools. They may have experience using similar platforms and can navigate through features easily.
* **Collaborative Teams**: These users primarily belong to teams or groups that require frequent communication for collaborative work. They may include remote teams, project groups, educational institutions, and businesses.
* **Privacy-Conscious Individuals**: These users prioritize their privacy and seek platforms that offer secure communication channels and data encryption.
* **Multimedia Consumers**: These users are interested in sharing and consuming multimedia content like audio, video, and screen demonstrations during communication sessions.
* **Diverse User Base**: The platform should cater to users with varying technical expertise, from beginners to advanced users, and accommodate different communication needs across various industries and domains.

* 1. **Constraints**

The major constraints that the project has are as follows: -

* **Browser Compatibility**: The platform’s functionality may be limited by the compatibility of WebRTC across different web browsers, potentially affecting user experience.
* **Internet Connection**: Users require stable and high-speed internet connections for seamless real-time communication. Poor internet connectivity can lead to disruptions and affect the quality of interactions.
* **Third-Party SDK Reliability**: The platform’s performance is dependent on the reliability and stability of third-party services like Agora SDK for audio and video communication. Any issues or downtime with these services can impact the user experience.
* **Device Compatibility**: Users must have devices (computers, laptops, mobile devices) with compatible hardware components (microphones, cameras) to fully utilize the platform’s features.
* **Security Concerns**: Ensuring data privacy and security is paramount. Any breaches or vulnerabilities in encryption protocols or communication channels can compromise user data and trust in the platform.
  1. **Assumptions and Dependencies**

Assumptions and dependencies for the "Instant Wave" project can vary based on the specific requirements, technology stack, and development approach. However, here are some common assumptions and dependencies that may apply:

**1. Assumptions:**

**a. Browser Compatibility**: The project assumes that users will access the communication platform using modern web browsers that support WebRTC technology

b**. Stable Internet Connection**: Users are assumed to have a stable internet connection to facilitate real-time communication without significant disruptions.

c**. User Familiarity**: Users are assumed to be familiar with basic internet browsing and communication tools, such as web browsers and chat applications.

d. **Agora SDK Integration:** The project assumes successful integration and compatibility with Agora SDK for audio, video, and screen sharing functionalities.

e**. Serverless Architecture**: The project assumes a serverless architecture for real-time communication, relying solely on WebRTC for direct browser-to-browser connections.

f. **Bot Functionality:** The project assumes successful implementation and integration of the bot functionality to greet new users in chat rooms.

g**. Security Measures:** Basic security measures such as encryption and authentication are assumed to be implemented to ensure the privacy and integrity of communication data.

**2. Dependencies:**

**a. WebRTC API:** Successful integration and utilization of WebRTC API for establishing real-time communication between browsers.

**b. Agora SDK:** Integration and compatibility with Agora SDK for audio, video, and screen sharing functionalities.

**c. Frontend Frameworks/Libraries:** Usage of frontend frameworks or libraries (e.g., React, Vue.js) for building the user interface and managing application state.

**d. Network Infrastructure:** Reliance on a stable network infrastructure to ensure seamless data transmission and communication between users.

**e. Browser Support:** Availability of modern web browsers supporting WebRTC technology for users to access the communication platform.

**f. Bot Integration:** Successful integration and functionality of the bot component to automate welcome messages for new users.

**g. Third-Party Services:** Reliance on third-party services or APIs for additional functionalities such as analytics, error logging, or user management (if applicable).

These assumptions and dependencies provide a foundational understanding of the project's requirements and the external factors that may impact its development and functionality.

1. **External Interface Requirements**

**3.1 User Interface Requirements**

The interface provided to the user should be a very user-friendly one and it should provide optional interactive help for each of the services listed. The interface provided is a menu driven one and the following screens will be provided: -

* **Intuitive Interface**: The user interface should be intuitive and easy to navigate, with clear labeling and visual cues for different features and functions.
* **Clear Communication**: Visual indicators like active speaker highlights, participant status icons, and chat bubbles facilitate clear communication and interaction.
* **Multimedia Controls**: User-friendly controls for managing audio and video settings, such as volume adjustment, mute/unmute options, camera toggles, and screen sharing controls.
* **Room Management**: Interface elements for creating, joining, and managing chat rooms, including options for setting room permissions, inviting participants, and managing user roles.
* **Bot Interaction**: Visual cues such as automated welcome messages and bot avatars enhance user engagement and provide a seamless experience during bot interactions.

**3.2 Hardware Interface Requirements**

There are various hardware components with which the machine is required to interact. Various hardware interface requirements that need to be fulfilled for successful functioning of the software are as follows: -

* **Audio Input/Output**: Support for microphone input and speaker output for audio communication, including compatibility with built-in or external microphones and speakers/headphones.
* **Video Input/Output**: Compatibility with cameras for video communication, including support for built-in webcams or external cameras connected to the user's device.
* **Screen Sharing Compatibility**: Devices capable of capturing and sharing screen content, with support for screen capture software or built-in screen recording functionality.
* **Device Connectivity**: Compatibility with various devices including desktop computers, laptops, tablets, and mobile devices running on different operating systems (Windows, macOS, iOS, Android).

**3.3 Software Interface Requirements**

In order to perform various different functions, this software needs to interact with various other software’s. So, there are certain software interface requirements that need to be fulfilled which are listed as follows: -

* **WebRTC Integration**: User interface elements for utilizing WebRTC APIs, including controls for accessing camera and microphone devices, establishing peer-to-peer connections, and managing real-time communication sessions.
* **Agora SDK Integration**: Interface components for integrating Agora SDK features, such as APIs for audio and video streaming, interactive broadcasting, and real-time messaging into the platform.
* **Room Management Interface**: Interface elements for managing chat rooms, including options for creating new rooms, joining existing rooms, managing room settings, and handling room-related events.
* **Bot Integration Interface**: User interface elements for incorporating automated bot interactions, including visual cues for bot messages, commands, and responses.
* **User Authentication Interface**: Interface components for user authentication and management, including user registration forms, login screens, password management options, and account settings pages.

**3.4 Communication Interface Requirements**

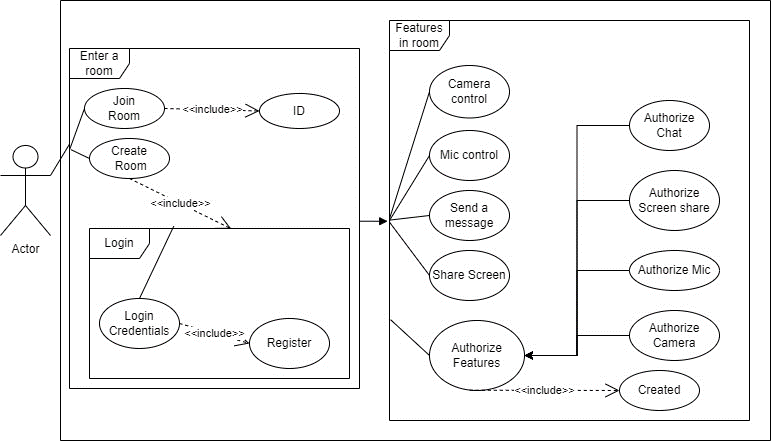
The machine needs to communicate with the main branch for each session for various functions such as login verification, account access etc. so the following are the various communication interface requirements that are needed to be fulfilled in order to run the software successfully: -

* **Real-Time Audio Communication**: Interface components for transmitting and receiving audio data in real-time, including controls for starting/stopping audio streams, adjusting volume levels, and managing audio quality settings.
* **Real-Time Video Communication**: User interface elements for transmitting and receiving video data in real-time, including controls for initiating video calls, toggling camera views, and managing video quality settings.
* **Real-Time Text Communication**: Interface components for sending and receiving text messages in real-time, including chat input fields, message display areas, and controls for sending/receiving messages.
* **Screen Sharing Interface**: User interface elements for initiating and managing screen sharing sessions, including options for selecting screen/application windows to share, controlling screen sharing settings, and managing screen sharing permissions.
* **Event Handling Interface**: Interface components for managing real-time events such as user join/leave notifications, message sent/received notifications, audio/video stream state changes, screen sharing requests, error messages, and other relevant events.

1. **System Features**

* **Real-Time Communication**:
* **Description**: The system enables real-time communication between users without the need for intermediate servers, utilizing WebRTC technology. This allows users to engage in audio, video, and text communication instantly.
* **Functionality**: Users can initiate and participate in communication sessions with minimal latency, creating a seamless conversational experience.
* **Group Conversations**:
* **Description**: Instant Wave supports group conversations, allowing multiple users to communicate simultaneously within the same virtual room.
* **Functionality**: Users can join existing chat rooms or create new ones, enabling collaborative discussions and interactions among participants.
* **Multimedia Sharing**:
* **Description**: The system enables users to share multimedia content, including audio, video, and screen demonstrations, with other participants during communication sessions.
* **Functionality**: Users can upload and share various types of multimedia files or stream live audio and video content to enhance communication and collaboration.
* **Screen Sharing**:
* **Description**: Instant Wave facilitates screen sharing, allowing users to broadcast their screen content in real-time to other participants.
* **Functionality**: Users can choose specific applications or their entire screen to share, enabling presentations, demonstrations, remote assistance, and collaborative work on documents or projects.
* **Bot Integration**:
* **Description**: The system includes a bot that automatically greets new users when they join a chat room, enhancing user experience and engagement.
* **Functionality**: The bot interacts with users by delivering predefined messages or performing specific tasks, such as providing information, instructions, or moderating conversations.
* **Room Management**:
* **Description**: Instant Wave offers room management functionality, allowing users to create, join, and manage chat rooms for organizing communication sessions.
* **Functionality**: Users can set room preferences, manage participant permissions, invite others to join, and configure settings related to room privacy and access control.
* **Security**:
* **Description**: The system prioritizes security by implementing robust encryption protocols to safeguard user data and ensure secure communication channels.
* **Functionality**: Encryption mechanisms are employed to protect user information, messages, and media shared within the platform, mitigating the risk of unauthorized access or data breaches.
* **User Authentication and Management**:
* **Description**: Instant Wave features user authentication and management capabilities to control access to the platform and personalize user experiences.
* **Functionality**: Users can register, log in, and manage their accounts securely, while administrators have tools to manage user profiles, permissions, roles, and account settings.
* **Real-Time Event Handling**:
* **Description**: The system manages and responds to real-time events, such as user join/leave notifications, incoming messages, and audio/video stream state changes.
* **Functionality**: Event handling mechanisms ensure smooth and interactive communication experiences by notifying users of relevant events and updating the interface accordingly in real-time.
* **High-Quality Streams with Minimal Latency**:
* **Description**: Instant Wave prioritizes high-quality audio and video streams with minimal latency to provide lifelike and immersive communication experiences.
* **Functionality**: The system optimizes audio and video transmission to minimize delays, ensuring clear and uninterrupted communication even in real-time scenarios.
* **Cross-Platform Compatibility**:
* **Description**: The system is compatible with various devices and platforms, including desktop computers, laptops, tablets, and mobile devices running on different operating systems (Windows, macOS, iOS, Android).
* **Functionality**: Users can access Instant Wave from their preferred devices and operating systems, ensuring flexibility and accessibility across different platforms.

**4.1 Use Case Diagram**



1. **Other Nonfunctional Requirements**

**5.1 Performance Requirements**

The following list provides a brief summary of the performance requirements for the software:

* + 1. **Capacity**

Capacity requirements specify the system's ability to handle a certain amount of data, users, or transactions within a given timeframe. These requirements are crucial for ensuring that the system can accommodate expected usage levels and scale as needed.

Here are the capacity requirements for the Instant Wave project:

* **User Capacity**:
* The system should be able to support a minimum of 1,000 concurrent users participating in communication sessions simultaneously.
* **Room Capacity**:
* Each chat room created within the platform should have the capacity to accommodate up to 50 participants at any given time.
* **Message Capacity**:
* The system should support the storage and retrieval of a minimum of 10,000 messages per chat room, ensuring that historical messages are accessible to users.
* **Media Capacity**:
* The platform should allow users to upload and share multimedia content (e.g., audio, video, images) with a maximum file size of 10 MB per file.
* **Session Capacity**:
* Each communication session (e.g., audio call, video call, screen sharing session) should have the capacity to support up to 20 participants without significant degradation in performance.
* **Bandwidth Capacity**:
* The system should have sufficient bandwidth capacity to handle peak data transfer rates during communication sessions, ensuring that audio and video streams are transmitted smoothly without buffering or delays.
* **Storage Capacity**:
* The platform should provide sufficient storage capacity to store user data, messages, multimedia content, and system logs for at least one year without exceeding storage limits.
* **Processing Capacity**:
* The system's processing capacity should be able to handle simultaneous user interactions, data processing tasks, and real-time event handling without experiencing bottlenecks or performance issues.
  + 1. **Dynamic requirements**

Dynamic requirements refer to those aspects of a system that can change or evolve over time in response to various factors such as user interactions, system load, andenvironmental conditions. Here are some dynamic requirements of the project:

* **Real-Time Updates**:
* The system should provide real-time updates for user actions, such as new message notifications, participant join/leave notifications, and changes in communication session status.
* **Adaptive Bandwidth Usage**:
* The system should dynamically adjust bandwidth usage based on network conditions and available resources to ensure optimal performance during communication sessions.
* **Load Balancing**:
* The system should employ load balancing techniques to distribute incoming user requests evenly across servers and resources, ensuring efficient resource utilization and preventing overloads.

* **Auto-Scaling**:
* The system should automatically scale resources (e.g., servers, bandwidth) up or down based on demand to maintain performance levels during peak usage periods and minimize costs during periods of low activity.
* **Session Management**:
* The system should dynamically manage communication sessions, including session initiation, termination, and reconnection handling, to ensure seamless user experiences and minimize disruptions.
* **Dynamic Resource Allocation**:
* The system should dynamically allocate resources (e.g., CPU, memory, bandwidth) based on user demand and system load, optimizing resource usage and maintaining performance levels.
* **Adaptive Security Measures**:
* The system should adaptively adjust security measures (e.g., encryption strength, access controls) based on threat levels and security risks to protect user data and ensure system integrity.
* **Adaptive User Interfaces**:
* The system's user interface should adapt dynamically based on user preferences, device characteristics, and accessibility needs to provide personalized and optimized user experiences.
* **Dynamic Content Delivery**:
* The system should dynamically adjust content delivery (e.g., media streaming quality, file compression) based on network conditions and device capabilities to optimize performance and user experience.
* **Dynamic User Authorization**:
* The system should dynamically adjust user permissions and access levels based on user roles, organizational changes, and security policies to ensure secure and appropriate access to system resources.

**5.1.3. Quality**

Quality requirements, also known as quality attributes or non-functional requirements, describe the overall quality characteristics that the system must possess to meet user expectations and organizational standards.

  Here are some quality requirements for the Instant Wave project:

* **Performance**:
* The system should perform efficiently, with fast response times and minimal latency, to provide smooth and responsive communication experiences for users.
* **Usability**:
* The system should be intuitive and easy to use, with a user-friendly interface that minimizes the learning curve for new users and enhances overall user satisfaction.
* **Scalability**:
* The system should be scalable to accommodate increasing numbers of users and communication sessions, ensuring that performance levels are maintained as the user base grows.
* **Compatibility**:
* The system should be compatible with a wide range of devices, browsers, and operating systems, ensuring accessibility and seamless user experiences across different platforms.
* **Reliability**:
* The system should operate reliably without unexpected failures, ensuring continuous availability and minimal downtime during communication sessions.
* **Interoperability**:
* The system should seamlessly integrate with other third-party services or applications if needed, allowing for interoperability and ease of use in heterogeneous environments.
* **Accessibility**:
* The system should be accessible to users with disabilities, following accessibility guidelines and standards to ensure equal access and usability for all users.
* **Compliance**:
* The system should comply with relevant laws, regulations, and industry standards related to data privacy, security, accessibility, and interoperability to maintain legal and regulatory compliance.

* 1. **Software System Attributes**

* + 1. **Reliability**
* The system should operate reliably without unexpected failures, ensuring continuous availability and minimal downtime during communication sessions.
  + 1. **Availability**
* The system should be consistently available for use during specified periods to support users' communication needs, ensuring uninterrupted access to communication features.

* + 1. **Security**
* The system should prioritize security and privacy, with robust encryption protocols, access controls, and data protection measures to safeguard user data and communication channels.

* + 1. **Maintainability**
* The system should be easy to maintain and update, with well-documented code and modular architecture that allows for efficient troubleshooting, debugging, and enhancements.

**5.3 Business Rules**

       Business rules are specific guidelines or constraints that dictate how business processes and operations should be conducted within an organization. They are typically derived from business policies, regulations, or best practices. Here are some example business rules for the Instant Wave project:

* **User Authentication**:
* Users must authenticate themselves with valid credentials (e.g., username and password) before accessing communication features within the platform.
* **Room Creation**:
* Users can create new chat rooms for communication sessions, but each room must have a unique name and adhere to specific naming conventions (e.g., alphanumeric characters only, no special characters).
* **Participant Limit**:
* Each chat room has a maximum limit of participants, and users cannot exceed this limit when inviting others to join the room.
* **Bot Greeting**:
* The automated bot within the platform must greet new users with a predefined welcome message when they join a chat room for the first time.
* **Message Formatting**:
* Users must adhere to message formatting guidelines (e.g., text length limits, supported file formats) when sending messages or sharing multimedia content within communication sessions.
* **Moderation Policies**:
* Chat room moderators have the authority to enforce community guidelines and remove participants who violate the rules or engage in inappropriate behavior during communication sessions.
* **Data Retention**:
* The platform retains user data, messages, and multimedia content for a specified period (e.g., one year) before automatically deleting or archiving them to comply with data retention policies.
* **Privacy Settings**:
* Users have control over their privacy settings and can adjust who can join their chat rooms, view their profile information, or access their communication history within the platform.
* **Session Timeouts**:
* Inactive users are automatically logged out of communication sessions after a predefined period of inactivity to ensure security and optimize resource usage.
* **Feature Restrictions**:
* Certain features within the platform (e.g., screen sharing, recording) may be restricted or limited based on user roles, subscription plans, or organizational policies.

1. **Other Requirements**

**Appendix A: Glossary**

**ICE (Interactive Connectivity Establishment):** ICE is a protocol used in real-time communication systems to establish a connection between two devices, even if they are behind NAT (Network Address Translation) or a firewall. It helps in finding the best communication path between the devices.

**SDP (Session Description Protocol):** SDP is a protocol that describes multimedia sessions for the purpose of session initiation, session announcement, and parameter negotiation. It is commonly used for describing the media content of a session.

**SDK (Software Development Kit):** An SDK is a set of tools, libraries, and documentation that developers use to create applications for specific platforms or software frameworks. It provides a comprehensive set of resources to streamline the development process.

**HTTPS (Hypertext Transfer Protocol Secure):** HTTPS is a protocol for secure communication over a computer network. It is widely used on the Internet to provide secure and encrypted communication between a web browser and a web server.

**STUN (Session Traversal Utilities for NAT):** STUN is a protocol used to assist devices behind a NAT firewall or router in determining their public IP address and the type of NAT being used. It helps in establishing direct communication between devices.

**RTC (Real Time Communication):** RTC refers to any type of live communication that occurs in real time, such as voice calls, video calls, and instant messaging. It often involves the use of various protocols and technologies to enable seamless communication.

**UDP (User Datagram Protocol):** UDP is a connectionless protocol that allows for the transmission of data packets over a network. It is commonly used for applications where speed and efficiency are more important than guaranteed delivery, such as real-time streaming and online gaming.