**Experiment No-1**

**Objective: - Prepare an SRS document in line with the IEEE recommended standards for the specified case study. (Functional Requirements)**

**Functional Requirements are -**

### 1. User Login:

### Users should be able to log in by the Unique id which is generated automatically.

### 2. Video Calls:

### Users can initiate, join, and participate in real-time video calls with one or multiple participants. This includes features like mute/unmute, camera on/off, and screen sharing.

### 3. Chat Messaging:

### Users can send and receive text-based messages in real time.

### 4. Share Room Link:

### Users can share link to other participants to join the Room.

### 5. Group Chats:

### Support for creating and participating in group chat rooms, allowing multiple users to engage in a single conversation.

**Experiment No-2**

**Objective: Prepare an SRS document in line with the IEEE Recommended standards for specified case study. (Nonfunctional Requirements)**

**1. Non-Functional Requirements:**

### 1. Performance:

### Ensure the application performs smoothly, even with a large number of users. Optimize code and use efficient algorithms.

### 2. Scalability:

### Design the application to handle a growing user base. Implement strategies for load balancing, caching, and database sharding.

### 3. Privacy:

### Comply with privacy regulations and protect user data. Implement strong authentication and access controls.

### 4. Cross-Browser Compatibility:

### Ensure the application works consistently across different web browsers, including Chrome, Firefox, Safari, and Edge.

### 5. Device Compatibility:

### Design the application to be responsive and work on various devices, including desktops, laptops, tablets, and mobile phones.

**Experiment No-3**

**Objective: List out the entities and identify the relationship between them. Also, identify related attributes supposed to be recorded while considering the normalization rule.**

**1. User:**

Attributes: UserID (Primary Key), Username, Email, Password, Profile Picture, Registration Date, Last Active Timestamp.

**2.Chat:**

Attributes: ChatID (Primary Key), Participants (Foreign Key referencing User.UserID), GroupChat (Boolean indicating if it's a group chat), Creation Timestamp.

**3. Message:**

Attributes: MessageID (Primary Key), ChatID (Foreign Key referencing Chat.ChatID), SenderID (Foreign Key referencing User.UserID), Content, Timestamp.

**4. Video Call:**

Attributes: CallID (Primary Key), Participants (Foreign Key referencing User.UserID), Start Timestamp, End Timestamp.

Now, let's discuss the relationships:

**1.User-User:**

Relationship: Friendship

Description: Many-to-Many relationship indicating which users are friends with each other.

**2.User-Chat:**

Relationship: Participant

Description: Many-to-Many relationship indicating which users are part of a chat.

**3.User-Message:**

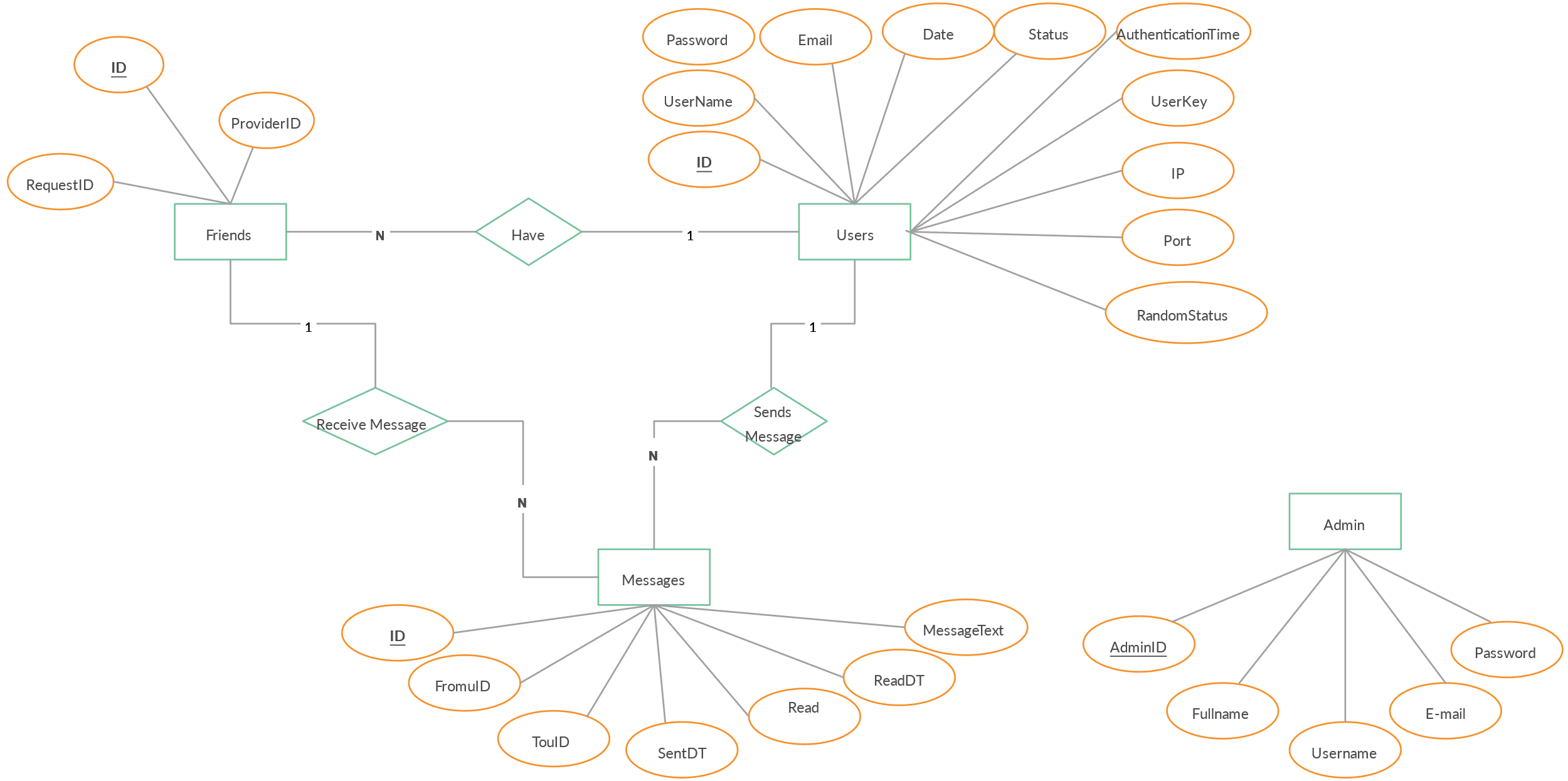
Relationship: Sender

Description: One-to-Many relationship indicating which user sent which messages.

**4.User-Video Call:**

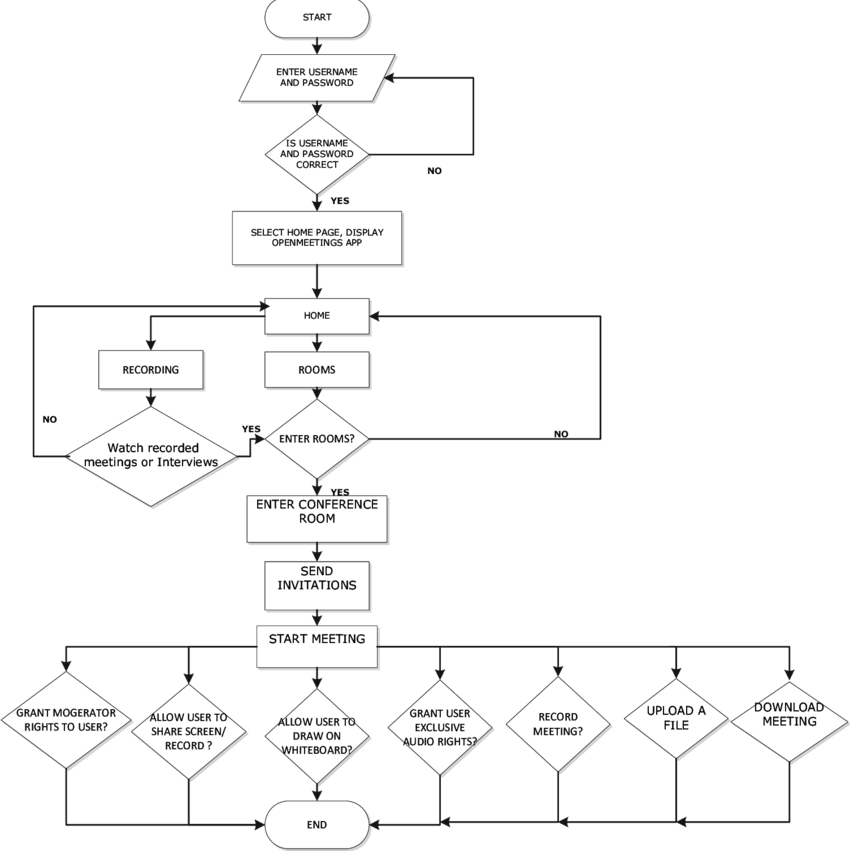
Relationship: Participant

Description: Many-to-Many relationship indicating which users participated in a video call.



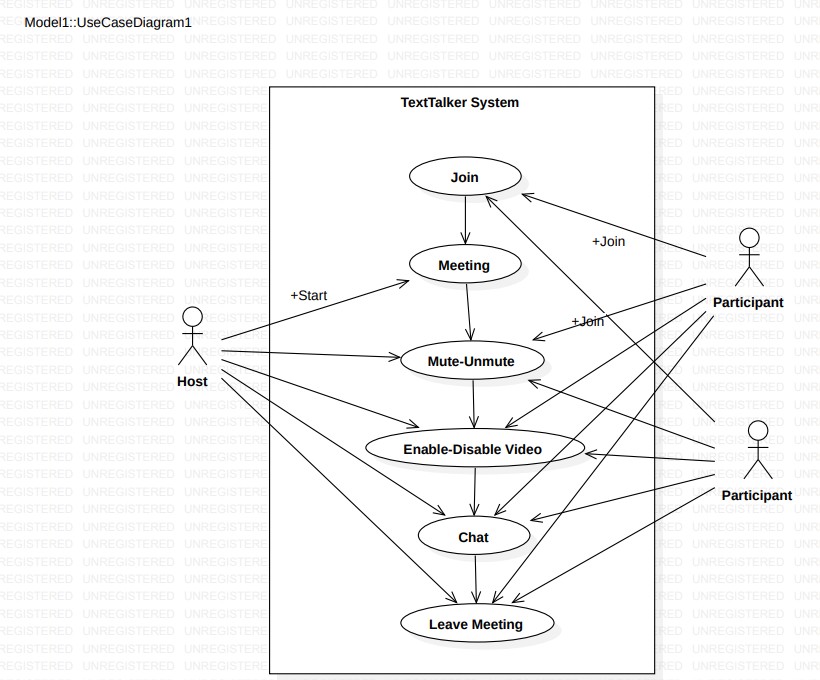
**Experiment No-4**

**Objective:** List out the required processes to be coded along with the required flow charts.



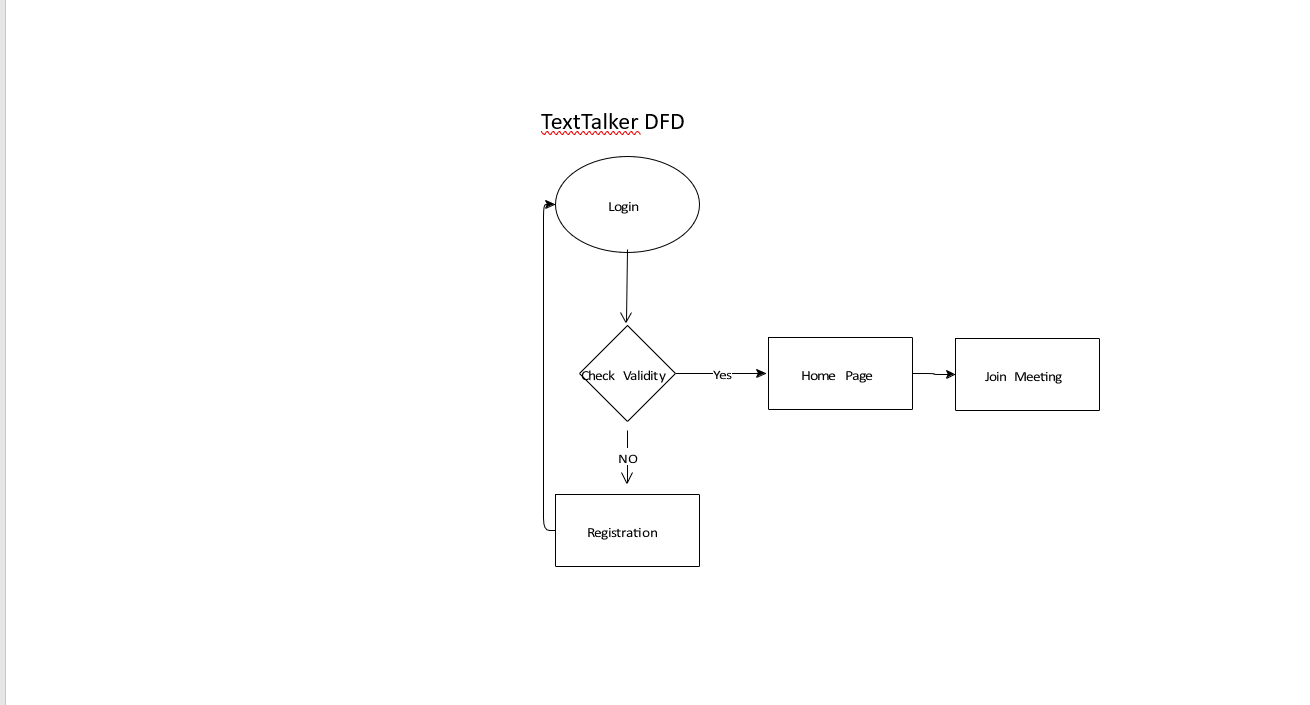
**Experiment No-5**

**Objective: Draw the activity diagram for the specified Case Study.**

****

**Experiment No- 9**

**Objective: Draw the DFD to be considered while codding the individual processes or functions.**

****

**Experiment No-7**

**Objective: List out the categories of reports and the required data to be represented with them. Also, determine the layout of the reports that may be used by individual authorities of the intended organization.**

1. User Registration Report:

* Table with columns for User ID
* Username
* Email
* Registration Date.

1. Chat History Report:

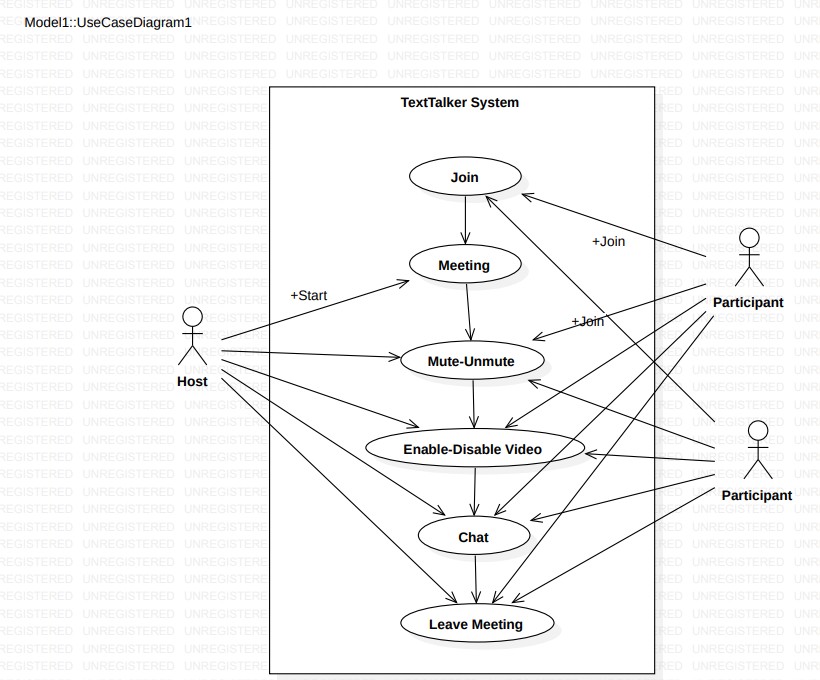
* Table with columns for Chat ID
* Participants
* Message Content.

1. Security Audit Log:

* Table with columns for User ID
* Action/Operation
* Timestamp,
* Outcome.

**Experiment No-8**

**Objective: While considering the scope of the individual functionalities draw the use case diagram for each of it.**

****

**Experiment No-9**

**Objective: On the basis of required validations map the different test cases to handle all possible critical cases which may arise during the life cycle of the software.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | **Test Scenario** | **Test Steps** | **Expected Result** | **Status** |
| 1. | User Account Creation | **1.Navigate to the sign-up page.**   1. **Enter valid user details.** 2. **Click on 'Sign Up'.** | User account is created successfully. |  |
| 2. | User Login | 1. Navigate to the login page. 2. Enter valid   credentials. 3. Click on 'Login'. | User is successfully  logged into the  application |  |
| 3. | Invalid Login Attempt | 1. Navigate to the login page. 2. Enter invalid   credentials. 3. Click on 'Login'. | User receives an  appropriate error  message,  unable to log in. |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4. | UI Navigation and  Responsiveness | 1. Open the application   on different devices.   1. Navigate through the UI. | UI is intuitive, and the  application is responsive on various devices. |  |
| 5. | Camera and  Microphone Access | 1. Start a video call. 2.  Verify access to  the camera and  microphone. | Camera and microphone  are  accessible,  and video call starts  successfully. |  |
| 6. | Audio and Video Quality | 1. Initiate a video call. 2. Check the quality   of audio and video. | Audio and video quality  are clear and synchronized. |  |
| 7. | Network Conditions | 1. Test under 3G,   4G, and Wi-  Fi  conditions.   1. Simulate low   bandwidth. | Application performs  well under different network conditions. |  |

**Experiment No-10**

**Objective: Identify the classes. Classify them as weak and strong classes and draw the class diagram for the specified Case Study.**

Classes:

**1.User Class (Strong Class):**

Attributes: UserID, Username, Password, Email, etc.

Methods: SignUp(), LogIn(), LogOut(), UpdateProfile()

**2. ChatRoom Class (Strong Class):**

Attributes: RoomID, RoomName, Participants, Messages, etc.

Methods: AddParticipant(), RemoveParticipant(), SendMessage(), GetMessages()

**3.VideoRoom Class (Strong Class):**

Attributes: RoomID, RoomName, Participants, VideoStream, etc.

Methods: AddParticipant(), RemoveParticipant(), StartVideoStream(),

StopVideoStream()

**4.Message Class (Weak Class):**

Attributes: MessageID, SenderID, ReceiverID, Content, Timestamp, etc.

**5.Notification Class (Weak Class):**

Attributes: NotificationID, UserID, Content, Timestamp, etc.

**Relationships:**

**1.User-ChatRoom Relationship:**

One user can participate in multiple chat rooms.

One chat room can have multiple participants.

**2.User-VideoRoom Relationship:**

One user can participate in multiple video rooms.

One video room can have multiple participants.

1. **ChatRoom-Message Relationship:**

One chat room can have multiple messages.

One message belongs to one chat room.

1. **VideoRoom-Message Relationship:**

One video room can have multiple messages.

One message belongs to one video room.

1. **User-Message Relationship:**

One user can send/receive multiple messages.

One message has one sender and one or more receivers.

**6.User-Notification Relationship:**

One user can receive multiple notifications.

One notification belongs to one user.

