* **Yêu cầu cơ bản**: Ứng dụng cần hỗ trợ nhiều người dùng tham gia hội thảo, truyền tải âm thanh, video và dữ liệu văn bản.
* **Tính năng**:
  + Truyền tải dữ liệu (âm thanh, video, tin nhắn).
  + Giao diện cho người dùng (thêm người tham gia, chọn chế độ hội thảo).
  + Quản lý người dùng (tham gia/thoát).
  + Hỗ trợ phát hiện mất kết nối, xử lý độ trễ hoặc nhiễu tín hiệu..

### 3. ****Thiết kế hệ thống****

* **Mô hình Client-Server**:
  + **Server**: Lắng nghe kết nối từ nhiều Client, quản lý phiên hội thảo (conference), gửi và nhận dữ liệu từ các Client.
  + **Client**: Gửi dữ liệu (âm thanh, video, tin nhắn) đến Server và nhận dữ liệu từ Server.
* **Kiến trúc**:
  + **Server** xử lý gói tin UDP, truyền dữ liệu đến tất cả các Client đã tham gia.
  + **Client** gửi gói tin UDP chứa âm thanh/video đến Server, và nhận gói tin UDP từ Server.

### 4. ****Cài đặt truyền thông UDP****

* **Server UDP**:
  + Tạo DatagramSocket để nhận và gửi dữ liệu.
  + Lắng nghe tại một cổng cố định.
  + Xử lý gói tin nhận được và gửi tới các Client khác (broadcast).
* **Client UDP**:
  + Tạo DatagramSocket để gửi và nhận dữ liệu từ Server.
  + Đóng gói dữ liệu (audio, video) vào DatagramPacket và gửi đến Server.

### ****ền tải âm thanh và video****

* **Âm thanh**: Ghi âm từ microphone và nén dữ liệu (có thể sử dụng thư viện như Java Sound API), sau đó đóng gói và gửi qua UDP.
* **Video**: Sử dụng webcam, nén khung hình (sử dụng OpenCV hoặc thư viện tương tự), đóng gói và gửi qua UDP.
* **Gửi dữ liệu**: Phân chia dữ liệu thành các gói tin nhỏ để truyền qua UDP.

### 6. ****Xây dựng giao diện người dùng (GUI)****

* **Client GUI**:
  + Hiển thị luồng video/âm thanh nhận từ Server.
  + Tạo giao diện đăng nhập và tham gia hội thảo.
* **Server GUI**:
  + Hiển thị danh sách người tham gia.
  + Quản lý kết nối và điều hành hội thảo.