**Xác định cấu trúc dữ liệu âm thanh:** Trước hết, bạn cần xác định cách bạn sẽ biểu diễn dữ liệu âm thanh để truyền qua mạng. Thông thường, dữ liệu âm thanh được biểu diễn dưới dạng các gói (packets) âm thanh, mỗi gói chứa một phần của dữ liệu âm thanh.

**Chọn codec âm thanh:** Lựa chọn một codec âm thanh để mã hóa và giải mã dữ liệu âm thanh. Opus là một codec phổ biến cho ứng dụng VoIP và cuộc gọi thoại thời gian thực trên giao thức UDP.

**Xây dựng máy chủ và máy khách:** Bạn cần xây dựng ứng dụng máy chủ để lắng nghe các yêu cầu kết nối từ các máy khách và máy khách để thiết lập kết nối với máy chủ.

**Kết nối UDP:** Sử dụng UDP sockets trong Java để thiết lập kết nối UDP giữa các máy khách và máy chủ. Bạn cần quản lý cả việc gửi và nhận dữ liệu âm thanh qua các kết nối UDP này.

**Thu và mã hóa âm thanh:** Sử dụng microphone để thu âm thanh từ người dùng và sau đó mã hóa dữ liệu âm thanh bằng codec đã chọn trước khi gửi nó qua UDP.

Cụ thể, codec âm thanh sẽ mã hóa âm thanh từ dạng tín hiệu analog thành dạng số hóa (bits) để lưu trữ hoặc truyền qua mạng, sau đó giải mã nó trở lại thành âm thanh tín hiệu để người nghe có thể nghe. Tương tự, codec video mã hóa hình ảnh và video từ dạng tín hiệu tương tự thành dạng số hóa và giải mã chúng thành hình ảnh và video trên màn hình.

**Gửi và nhận dữ liệu âm thanh:** Sử dụng giao thức UDP để gửi và nhận các gói dữ liệu âm thanh giữa máy khách và máy chủ. Đảm bảo bạn xử lý độ trễ và thứ tự của các gói dữ liệu.

**Giải mã và phát âm thanh:** Tại máy chủ và máy khách đích, bạn cần giải mã dữ liệu âm thanh và phát lại nó từ loa.

**Xử lý tình trạng cuộc gọi:** Xây dựng logic để theo dõi tình trạng cuộc gọi, chẳng hạn như việc kết nối, ngắt kết nối, và thời gian cuộc gọi.

**Phía máy chủ:**

1. **Khởi tạo máy chủ:** Máy chủ được tạo để lắng nghe các kết nối đến từ các máy khách khác nhau. Máy chủ cần biết địa chỉ IP và cổng mà các máy khách sẽ kết nối đến.
2. **Lắng nghe kết nối:** Máy chủ sử dụng một socket UDP để lắng nghe các kết nối đến từ các máy khách. Nó sẽ chờ và chấp nhận các yêu cầu kết nối từ các client.
3. **Quản lý kết nối:** Máy chủ cần theo dõi và quản lý các kết nối từ các máy khách khác nhau. Mỗi kết nối sẽ có một địa chỉ IP và cổng riêng biệt.

* **Thiết lập kết nối ban đầu:** Khi máy khách muốn tham gia cuộc trò chuyện, nó phải thiết lập một kết nối với máy chủ. Máy khách sẽ gửi yêu cầu kết nối đến máy chủ và đợi phản hồi.
* **Chấp nhận kết nối:** Máy chủ cần chấp nhận yêu cầu kết nối từ máy khách. Sau khi kết nối được chấp nhận, máy chủ và máy khách có thể gửi và nhận dữ liệu âm thanh.
* **Quản lý nhiều kết nối:** Máy chủ cần hỗ trợ nhiều kết nối từ nhiều máy khách khác nhau. Điều này đặc biệt quan trọng trong các ứng dụng đa người dùng hoặc cuộc họp trực tuyến, nơi có nhiều người tham gia.
* **Theo dõi tình trạng kết nối:** Cả máy chủ và máy khách cần theo dõi tình trạng của kết nối.

1. **Nhận và giải mã âm thanh:** Khi máy chủ nhận được dữ liệu âm thanh từ một máy khách, nó sẽ giải mã dữ liệu này (nếu cần) và sau đó phát lại nó qua loa của máy chủ.

**Phía máy khách:**

1. **Khởi tạo máy khách:** Máy khách được tạo để kết nối đến máy chủ và gửi dữ liệu âm thanh.
2. **Kết nối đến máy chủ:** Máy khách sử dụng một socket UDP để kết nối đến máy chủ bằng địa chỉ IP và cổng của máy chủ.
3. **Thu và mã hóa âm thanh:** Máy khách thu âm thanh từ nguồn âm thanh, chẳng hạn như microphone, và sau đó mã hóa dữ liệu âm thanh bằng codec (ví dụ: Opus) nếu cần.
4. **Gửi dữ liệu âm thanh:** Dữ liệu âm thanh sau khi được mã hóa sẽ được gửi từ máy khách đến máy chủ thông qua giao thức UDP.
5. **Nhận và phát lại âm thanh từ máy chủ:** Máy khách cũng cần lắng nghe cho dữ liệu âm thanh từ máy chủ thông qua kết nối UDP riêng của nó và sau đó giải mã dữ liệu (nếu cần) và phát lại nó qua loa của máy khách.
6. **Bảo vệ và quản lý kết nối:** Máy khách cần xác minh tính toàn vẹn và bảo mật của dữ liệu, và quản lý kết nối với máy chủ.