

**TRƯỜNG ĐẠI HỌC KIÊN GIANG**  
**KHOA THÔNG TIN & TRUYỀN THÔNG**

---



**BÁO CÁO HỌC PHẦN**  
**LẬP TRÌNH MẠNG**  
**ĐỀ TÀI:**  
**PHÒNG HỌP TRỰC TUYẾN HALO MEET**

Sinh viên: Võ Vĩ Khang	MSSV: 21072006025
Đặng Nguyễn Tiền Hậu	MSSV: 21072006018
Quách Văn Sáng	MSSV: 21072006041
Lê Tuấn Dĩ	MSSV: 21072006072

GVHD: Ths.Nhan Thanh Nhã

*Tháng 2 năm 2024*

# MỤC LỤC

LỜI CẢM ƠN .....	2
DANH MỤC HÌNH .....	3
I. Giới thiệu chung .....	4
1. Đặt vấn đề .....	4
2. Lý do chọn đề tài .....	4
3. Mục tiêu và phạm vi của đề tài .....	4
2. Công cụ thực hiện: .....	5
III. Các mục xử lý (Demo) .....	10
1. Đăng ký .....	10
2. Đăng nhập .....	13
3. Tạo phòng họp và chia sẻ link phòng họp: .....	14
4. Tham gia vào phòng họp có sẵn.....	18
5. Share màn hình. ....	22
IV. Tổng kết.....	23
1. Kết quả đạt được.....	23
2. Ưu điểm.....	24
3. Hạn chế.....	24
4. Hướng phát triển.....	24
PHỤ LỤC .....	25
TÀI LIỆU THAM KHẢO.....	27

## LỜI CẢM ƠN

Kính gửi thầy Nhan Thanh Nhã,

Chúng em xin gửi đến thầy lời cảm ơn chân thành và sâu sắc nhất về sự hướng dẫn và hỗ trợ mà thầy đã mang lại cho em trong quá trình thực hiện đề tài Lập Trình Mạng của nhóm em. Sự hỗ trợ của thầy là phần đóng góp vô cùng to lớn vào thành công của đề tài này.

Trong suốt quá trình thực hiện đề tài, chúng em đã nhận được nhiều điều bổ ích từ kiến thức chuyên môn của thầy. Thầy đã luôn sẵn sàng, nhiệt tình chia sẻ kiến thức và đưa cho chúng em những định hướng tốt nhất trong việc tìm hiểu, nghiên cứu đề tài. Nhờ sự hướng dẫn này mà chúng em có cơ hội để mở rộng kiến thức của mình và phát triển kỹ năng nghiên cứu.

Đặc biệt, chúng em rất cảm kích sự tâm tâm của thầy trong việc nhận xét đề tài của chúng em. Những góp ý và phản hồi chi tiết của thầy đã giúp chúng em cải thiện chất lượng đề tài, làm cho nó trở nên hoàn thiện hơn.

Ngoài ra, chúng em cũng muốn gửi lời cảm ơn chân thành đến khoa Thông Tin – Truyền Thông đã tạo điều kiện cho chúng em được học tập, nghiên cứu và sáng tạo ra dự án cá nhân thông qua học phần “Lập trình mạng”. Thông qua quá trình tự học, tự nghiên cứu đã giúp chúng em có thêm kinh nghiệm trong việc xây dựng chương trình, báo cáo bảo vệ đề tài,...

Điều đó rất có ý nghĩa đối với chúng em, đặc biệt là trong hành trình nghề nghiệp sau này.

Chúng em xin trân trọng cảm ơn!

## DANH MỤC HÌNH

Ảnh 1 – Giao diện đăng ký.....	10
Ảnh 2 – Giao diện đăng nhập.....	13
Ảnh 3 – Giao diện trang chủ.....	14
Ảnh 4 – Giao diện trang phòng họp .....	15
Ảnh 5 – Giao diện trang phòng họp khi gọi .....	15
Ảnh 6 – Giao diện trang phòng họp khi join room.....	18
Ảnh 8 – Giao diện trang phòng họp khi share màn hình thành công .....	23
Ảnh 9 – Sơ đồ xử lý của Stringee Video Conference .....	25

## **I. Giới thiệu chung**

### **1. Đặt vấn đề**

Hiện nay do nhu cầu công việc và thời gian ngày càng bận rộn, để gặp nhau trao đổi về bài tập, công việc, các dự án... gặp nhiều khó khăn khi phải tốn thời gian để gặp trực tiếp, người ta chọn cuộc gặp gỡ trên nền tảng online, các cuộc gọi trực tuyến khá phổ biến, vì vậy chúng em đã nghiên cứu và lựa chọn đề tài này với mong muốn rút ngắn thời gian học tập và làm việc cho nhiều người hiện nay.

### **2. Lý do chọn đề tài**

Để giải quyết các vấn đề nan giải, giúp cho công việc và cuộc sống cải thiện hơn, mọi người có thể dễ dàng gặp nhau trao đổi và học hỏi một cách dễ dàng và thuận lợi.

### **3. Mục tiêu và phạm vi của đề tài**

- Gọi điện và nói chuyện
- Bật tắt video
- Nhắn tin
- Tạo phòng

## **II. Cơ sở lý thuyết và công cụ thực hiện**

### **1. Cơ sở lý thuyết:**

#### **Học phần: Lập trình web**

- "Lập trình Web" là một môn học thú vị và cần thiết trong thế giới công nghệ ngày nay. Môn học này không chỉ mang đến kiến thức cơ bản về lập trình web mà còn giúp sinh viên xây dựng kỹ năng thực hành và sáng tạo trong việc phát triển ứng dụng web hiện đại.
- Trong suốt quá trình học, sinh viên sẽ được làm quen với các ngôn ngữ lập trình phổ biến như HTML, CSS, và JavaScript, cũng như các framework và thư viện phổ biến như React, Angular hoặc Vue.js. Ngoài ra, khóa học còn chú trọng đến các khái niệm quan trọng như responsive design, quản lý trạng thái ứng dụng, và tối ưu hóa trải nghiệm người dùng.

- Với sự hướng dẫn của các giảng viên giàu kinh nghiệm và các dự án thực tế, sinh viên sẽ có cơ hội áp dụng kiến thức đã học vào việc tạo ra các trang web động, hiệu quả và thân thiện với người dùng. Môn học này không chỉ giúp sinh viên chuẩn bị cho sự nghiệp trong lĩnh vực phát triển web mà còn khuyến khích sự sáng tạo và tư duy logic trong giải quyết vấn đề.

## **2. Công cụ thực hiện:**

### **a. Hyper Text Markup Language (HTML)**

- HTML là viết tắt của cụm từ Hypertext Markup Language (tạm dịch là Ngôn ngữ đánh dấu siêu văn bản). HTML được sử dụng để tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, titles, blockquotes... và HTML không phải là ngôn ngữ lập trình.
- HTML: Hỗ trợ trong việc xây dựng layout, thêm nội dung dễ dàng trên website.

### **b. CSS**

- CSS là viết tắt của Cascading Style Sheets, là một ngôn ngữ thiết kế được sử dụng nhằm mục đích đơn giản hóa quá trình tạo nên một website. CSS được ra mắt vào năm 1996 bởi World Wide Web Consortium (W3C).
- CSS xử lý một phần giao diện của trang web. Sử dụng CSS, bạn có thể kiểm soát màu sắc của văn bản, kiểu phông chữ, khoảng cách giữa các đoạn văn, cách các cột được đặt kích thước và bố cục, hình ảnh hoặc màu nền nào được sử dụng, thiết kế bố cục, các biến thể hiển thị cho các thiết bị và kích thước màn hình khác nhau cũng như hàng loạt các hiệu ứng khác.
- Cách CSS hoạt động đó chính là tìm kiếm dựa trên vùng chọn chẳng hạn như thẻ HTML, ID, class, ... Sau đó, nó sẽ áp dụng những thuộc tính buộc phải thay đổi lên các vùng đã chọn.
- CSS: Hỗ trợ việc định dạng thiết kế, bố cục, style, màu sắc,...

### c. Ngôn ngữ JS

- JavaScript là ngôn ngữ lập trình website phổ biến hiện nay, nó được tích hợp và nhúng vào HTML giúp website trở nên sống động hơn. JavaScript đóng vai trò như là một phần của trang web, thực thi cho phép Client-side script từ phía người dùng cũng như phía máy chủ (Nodejs) tạo ra các trang web động.
- JavaScript là một ngôn ngữ lập trình thông dịch với khả năng hướng đến đối tượng. Là một trong 3 ngôn ngữ chính trong lập trình web và có mối liên hệ lẫn nhau để xây dựng một website sống động, chuyên nghiệp:
- JavaScript: Tạo nên những nội dung “động” trên website. Cùng tìm hiểu rõ hơn ở phần dưới đây.

### d. Framework VueJS

- Vue.js là một framework JavaScript được sử dụng để xây dựng giao diện người dùng (UI) và ứng dụng đơn trang (SPA). Nó được thiết kế để dễ dàng tích hợp và sử dụng. Dưới đây là một số đặc điểm chính của Vue.js:
- Khả năng mở rộng: Vue.js có thể được sử dụng để xây dựng từ các thành phần nhỏ cho đến ứng dụng đơn trang phức tạp. Điều này giúp nó phù hợp với nhiều loại dự án.
- Dữ liệu hai chiều binding: Vue.js cung cấp cơ chế two-way data binding, giúp đồng bộ hóa dữ liệu giữa model và view một cách tự động.
- Component-Based: Vue.js thúc đẩy việc xây dựng ứng dụng dựa trên các thành phần (components). Các thành phần này có thể được tái sử dụng và kết hợp lại với nhau để tạo nên giao diện người dùng (UI) của ứng dụng.
- Virtual DOM: Vue.js sử dụng Virtual DOM để tối ưu hóa hiệu suất của ứng dụng. Thay vì cập nhật trực tiếp DOM mỗi khi có thay đổi, Vue.js sử dụng một biểu diễn ảo của DOM để thực hiện các thay đổi, sau đó cập nhật DOM thực tế chỉ khi cần thiết.
- Cộng đồng mạnh mẽ: Vue.js có một cộng đồng lớn và đầy đủ tài liệu. Điều này giúp các nhà phát triển dễ dàng tìm kiếm hỗ trợ và tài nguyên khi làm việc với framework.

- Hỗ trợ cho TypeScript: Vue.js cung cấp hỗ trợ cho TypeScript, giúp tăng tính rõ ràng và quản lý dự án trong quá trình phát triển.
- Thích hợp cho các dự án nhỏ đến trung bình: Vue.js thường được chọn cho các dự án nhỏ đến trung bình, nhưng nó cũng có thể được sử dụng trong các dự án lớn hơn.

#### **e. Visual Studio Code**

Visual Studio Code (VSCode) là một trình biên tập mã nguồn mở và miễn phí được phát triển bởi Microsoft. Nó được thiết kế để hỗ trợ các nhà phát triển trong quá trình viết mã và phát triển ứng dụng. Dưới đây là một số đặc điểm chính của Visual Studio Code:

- Mã nguồn mở và Miễn phí: VSCode là một ứng dụng mã nguồn mở và miễn phí, có sẵn cho Windows, macOS và Linux.
- Hỗ trợ nhiều ngôn ngữ lập trình: VSCode cung cấp hỗ trợ cho nhiều ngôn ngữ lập trình thông dụng, bao gồm JavaScript, TypeScript, Python, C++, Java, HTML, CSS, và nhiều ngôn ngữ khác.
- Extensions và Marketplace: VSCode hỗ trợ các extension mở rộng giúp mở rộng tính năng của nó. Người dùng có thể cài đặt extension từ Visual Studio Code Marketplace để tùy chỉnh môi trường làm việc theo nhu cầu của họ.
- IntelliSense: Hệ thống IntelliSense của VSCode giúp nhà phát triển dễ dàng tìm kiếm và sử dụng các biến, phương thức, và các thành phần khác từ thư viện và framework.
- Debugging tích hợp: VSCode cung cấp tính năng debugging tích hợp cho nhiều ngôn ngữ, giúp nhà phát triển dễ dàng theo dõi và sửa lỗi trong mã nguồn.
- Git Integration: VSCode tích hợp sẵn với Git, cho phép nhà phát triển quản lý phiên bản mã nguồn một cách dễ dàng.
- Command Palette: Một command palette mạnh mẽ giúp người dùng thực hiện nhanh chóng các lệnh và tìm kiếm chức năng.
- Live Share: Tính năng Live Share cho phép nhiều nhà phát triển cùng làm việc trên cùng một dự án, thậm chí khi họ ở xa nhau.
- Lightweight và Nhanh chóng: VSCode được thiết kế để có hiệu suất cao và mở ra nhanh chóng, với giao diện người dùng đơn giản và trực quan.



## **f. Glitch**

Glitch là một dịch vụ hosting ứng dụng web và dự án mã nguồn mở. Nó cung cấp môi trường phát triển trực tuyến cho các nhà phát triển để xây dựng, kiểm thử và triển khai ứng dụng web một cách dễ dàng. Dưới đây là một số đặc điểm chính của Glitch:

- **Môi trường Phát triển Trực tuyến:** Glitch cung cấp một môi trường phát triển đầy đủ trực tuyến, nơi bạn có thể viết mã, kiểm thử ứng dụng và xem kết quả ngay lập tức.
- **Editor Mã Nguồn Mở:** Glitch có một trình soạn thảo mã nguồn tích hợp, hỗ trợ nhiều ngôn ngữ lập trình và có tính năng nhấn đồng thời giúp nhóm làm việc cùng nhau trực tuyến.
- **Tích hợp Git:** Glitch tích hợp với Git, giúp bạn quản lý mã nguồn và thực hiện các phiên bản khác nhau của dự án.
- **Hosting Dự Án Tự Động:** Glitch tự động triển khai ứng dụng của bạn và cung cấp một đường dẫn trực tuyến để chia sẻ với người khác.
- **Community và Remixing:** Glitch có một cộng đồng lớn, nơi người dùng có thể chia sẻ dự án của họ và người khác có thể "remix" (tạo bản sao) để sửa đổi và phát triển tiếp.
- **Hỗ Trợ Nhiều Ngôn Ngữ:** Glitch hỗ trợ nhiều ngôn ngữ lập trình và framework, bao gồm Node.js, Python, Ruby, HTML, CSS, và nhiều ngôn ngữ khác.
- **Dữ liệu và API:** Glitch cung cấp các công cụ để bạn có thể kết nối và làm việc với các dịch vụ và API bên ngoài.
- **Miễn Phí và Thanh Toán:** Glitch cung cấp một mô hình miễn phí với các giới hạn, cũng như các gói thanh toán để mở rộng tính năng và dung lượng lưu trữ.

## **g. Github**

- **Kiểm Soát Phiên Bản:** GitHub sử dụng Git, một hệ thống kiểm soát phiên bản phổ biến, để theo dõi sự thay đổi trong mã nguồn. Điều này giúp đội ngũ phát triển làm việc cùng nhau trên dự án mà không gặp xung đột và quản lý lịch sử thay đổi của mã nguồn.

- **Lưu Trữ Từ Xa (Remote Repositories):** GitHub cung cấp các kho lưu trữ từ xa (remote repositories) trên máy chủ của họ. Điều này cho phép nhóm làm việc từ xa và chia sẻ mã nguồn của họ trên toàn cầu.
- **Quản Lý Dự Án:** GitHub cung cấp các tính năng quản lý dự án, bao gồm các công cụ để theo dõi công việc, quản lý vấn đề, và tổ chức mã nguồn một cách hiệu quả.
- **Colaboration và Social Networking:** GitHub không chỉ là một nơi để lưu trữ mã nguồn, mà còn là một cộng đồng nơi người dùng có thể tương tác, đóng góp vào dự án của người khác và theo dõi các dự án mà họ quan tâm.
- **Cộng Tác Nhóm:** GitHub hỗ trợ cộng tác nhóm thông qua các tính năng như nhánh (branching), hợp nhất (merging), và các công cụ quản lý xung đột.
- **Trang Web Giao Diện Người Dùng (UI):** GitHub cung cấp một giao diện người dùng web thân thiện, giúp người dùng xem và quản lý mã nguồn của họ một cách dễ dàng.
- **Lưu Trữ Công Khai và Riêng Tư:** Người dùng có thể chọn làm cho dự án của họ công khai hoặc riêng tư, tùy thuộc vào yêu cầu của dự án.
- **Actions và CI/CD:** GitHub Actions cung cấp công cụ để tự động hóa các công việc trong quy trình phát triển, bao gồm kiểm thử liên tục (continuous integration) và triển khai liên tục (continuous deployment).

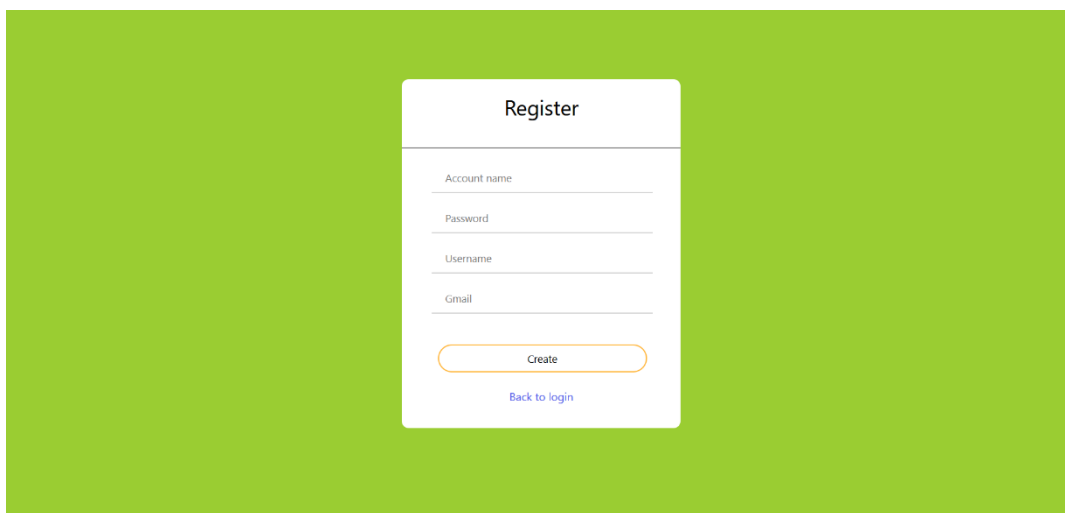
#### **h. Stringee API**

- Stringee là một dịch vụ cung cấp API cho việc tích hợp các tính năng liên quan đến cuộc gọi thoại và tin nhắn trong ứng dụng di động hoặc web. Đặc biệt, Stringee API thường được sử dụng để xây dựng các ứng dụng như ứng dụng gọi video, ứng dụng gọi điện thoại, hệ thống trò chuyện, và các ứng dụng liên quan đến việc truyền dữ liệu âm thanh và video trong thời gian thực.
- Các tính năng chính của Stringee API bao gồm:
  - **Cuộc Gọi Thoại và Video:** Cho phép triệu gọi thoại và video trong ứng dụng di động hoặc web.
  - **Tin Nhắn Trực Tiếp và Nhóm:** Hỗ trợ gửi tin nhắn trực tiếp giữa các người dùng cũng như trong các nhóm.

- **Quản lý Người Dùng:** Cung cấp các API để quản lý người dùng, đăng ký, đăng nhập, và quản lý thông tin tài khoản.
- **Sự Kiện và Thông Báo:** Hỗ trợ các sự kiện trong thời gian thực như cuộc gọi đến, tin nhắn mới và các sự kiện khác.
- **Tích hợp Dễ Dàng:** Cung cấp các SDK và API để tích hợp vào ứng dụng di động (iOS, Android) và web.
- **Bảo mật:** Cung cấp các giải pháp bảo mật để đảm bảo an toàn cho dữ liệu gửi và nhận trong quá trình truyền thông.
- **Quản lý Thiết Bị:** Hỗ trợ quản lý các thiết bị được đăng ký và sử dụng trong ứng dụng.
- **Stringee** thường được sử dụng trong các ứng dụng có yêu cầu về giao tiếp âm thanh và video trong thời gian thực, như ứng dụng gọi video, ứng dụng cuộc gọi điện thoại VoIP, và các ứng dụng trò chuyện trực tuyến.

### III. Các mục xử lý (Demo)

#### 1. Đăng ký



Ảnh 1 – Giao diện đăng ký

Phần code:

```
<body>
  <div class="container">
```

```

<div class="register-form">
    <div class="title">
        Register
    </div>
    <hr>
    <form action="create.php" method="POST">
        <table>
            <tr>
                <th>
                    <input type="text" name="ten_tk" placeholder="
Account name">
                </th>
            </tr>
            <tr>
                <td>
                    <?php if(empty($_GET['error'])){?>
                    <?php }else {?>
                </td>
                <td>
                    <th>
                        <div class="error">
                            <i><?php echo $_GET['error'];?></i>
                        </div>
                    </th>
                </td>
            </tr>
            <tr>
                <td>
                    <?php }?>
                </td>
                <td>
                    <th>
                        <input type="password" name="mat_khau"
placeholder=" Password">
                    </th>
                </td>
            </tr>
            <tr>
                <td>
                    <th>
                        <input type="text" name="ten_nd" placeholder="
Username">
                    </th>
                </td>
                <td>
                    <th>

```

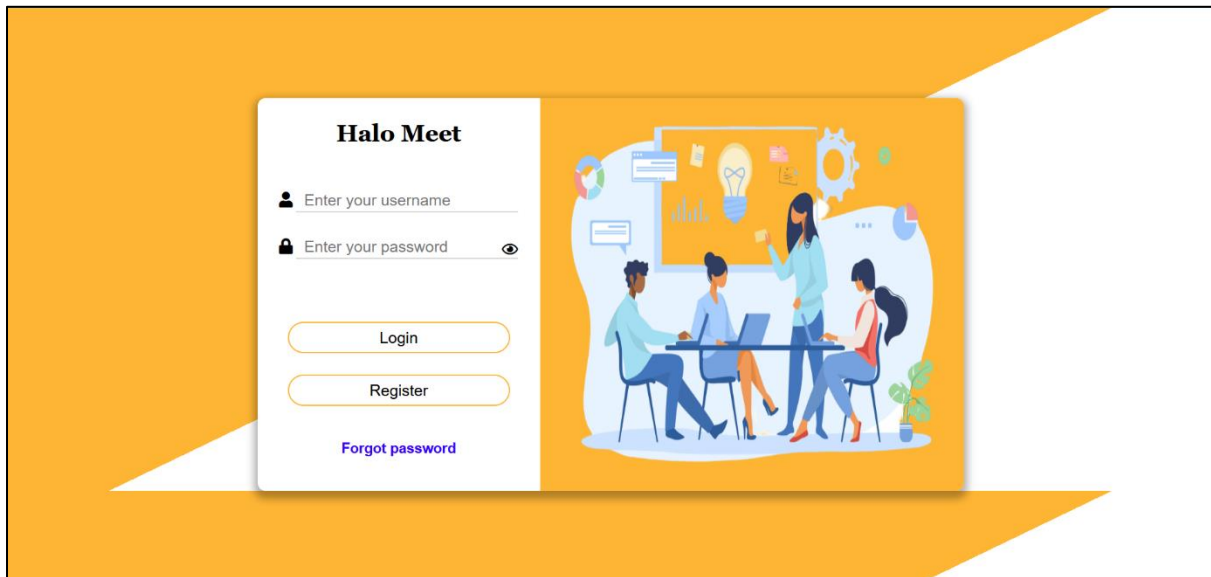
```

        <input type="text" name="gmail" placeholder="
Gmail">
        </th>
    </tr>
    <tr>
        <th>
            <a href="">
                <button class="bt-create">
                    Create
                </button>
            </a>
        </th>
    </tr>
</table>
</form>
<div class="back-to-login">
    <a href="index.php">
        Back to login
    </a>
</div>
</div>
</div>
</body>

```

## 2. Đăng nhập

Phần giao diện:



Ảnh 2 – Giao diện đăng nhập

Phần code:

```
<?php
session_start();
require 'connect.php';
$username = $_POST['Username'];
$password = $_POST['Password'];

$sql = "select user.* from user where ten_tk = '$username' and mat_khau = '$password'";
$result = mysqli_query($connect, $sql);
$num_rows = mysqli_num_rows($result);

if ($num_rows == 1) {
    $first = mysqli_fetch_array($result);
    $_SESSION['username'] = $first['ten_tk'];
    $_SESSION['name'] = $first['ten_nd'];
    header("location:../GiaoDienChinh/");
    exit();
} else {
    header("location:index.php?error=Đăng nhập không thành công!");
}
```

```
mysqli_close($connect);  
}
```

*Giải thích:*

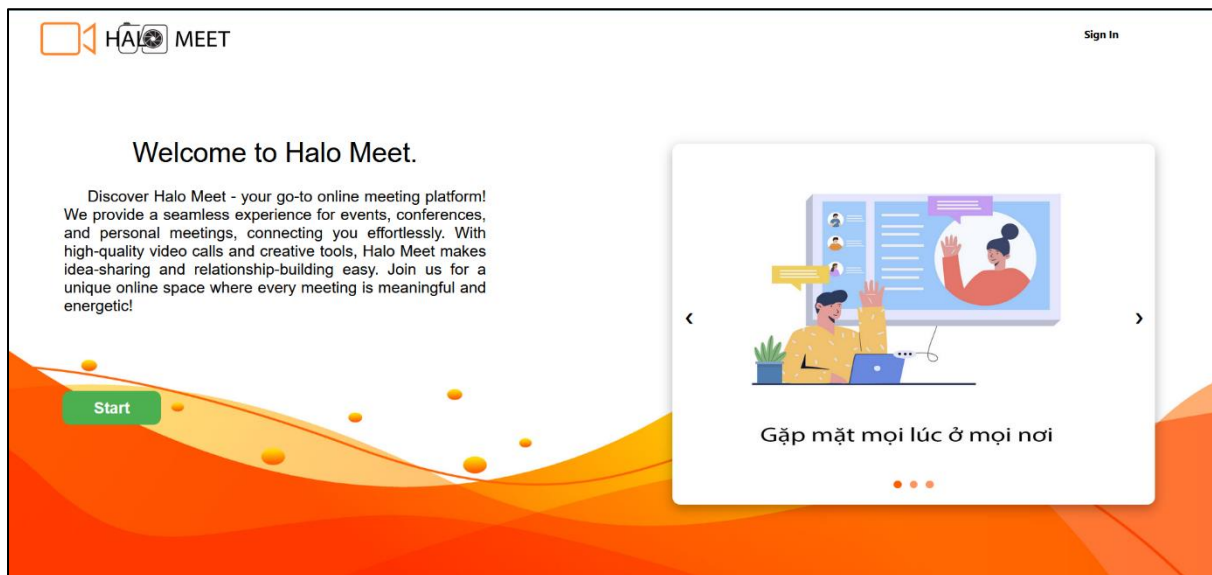
Gọi `session_start()`; để tạo session lưu thông tin người dùng khi đăng nhập vào.

Lấy username và password từ phương thức POST và đưa vào cơ sở dữ liệu để xác thực tài khoản có tồn tại không.

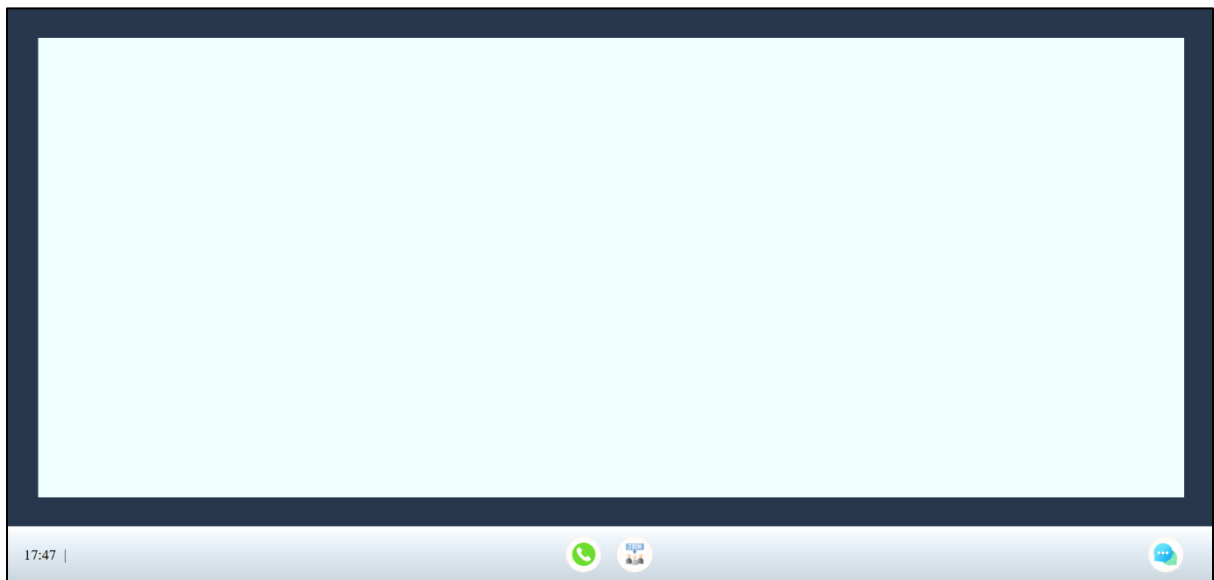
Nếu có tồn tại thì đưa thông tin vào session và chuyển hướng đến trang chủ. Nếu không tồn tại thì hiện thông báo lỗi.

### 3. Tạo phòng họp và chia sẻ link phòng họp:

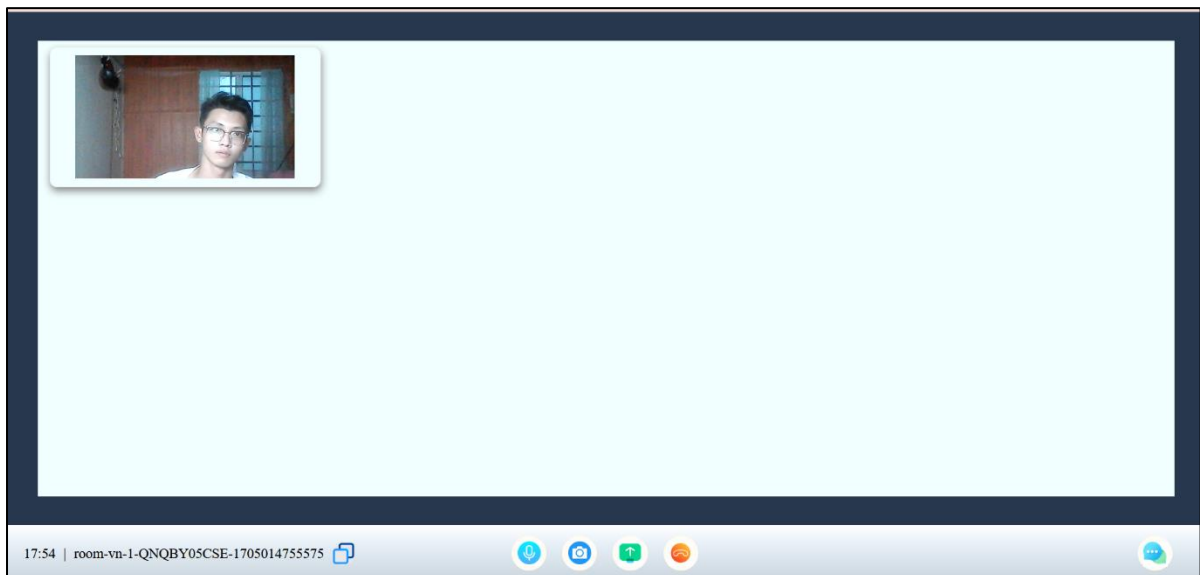
Phần giao diện:



Ảnh 3 – Giao diện trang chủ



Ảnh 4 – Giao diện trang phòng họp



Ảnh 5 – Giao diện trang phòng họp khi gọi



Phần code:

Tại trang chủ, ta tạo một nút <a> để điều hướng vào trang phòng họp:

```
<div class="contact" id="app">
  <table class="table_contact">
    <tr>
      <th class="table_contact_create-room_left">
        <a href="../GiaoDienPhongHop/" class="create-room">Start</a>
      </th>
    </tr>
  </table>
</div>
```

Trong phòng họp, ta ấn vào nút có hình ảnh chiếc điện thoại để bắt đầu cuộc gọi:

```
<!-- Button Call -->
<div class="btn-wrapper" v-if="!room">
  <button class="toolbar-btn" id="btn-call" @click="createRoom">
    
  </button>
  <script>
    const btnCall = document.getElementById("btn-call");
    btnCall.click();
  </script>
</div>
```

*Giải thích:*

v-if="!room": Nếu chưa có phòng thì mới được tạo phòng.

@click="createRoom": Sự kiện click vào nút thì gọi đến hàm createRoom trong API của Stringee.

API thực hiện gọi điện:

```
async createRoom() {
  const roomName = Math.random().toFixed(4);
  const response = await axios.post(
    `${BASE_URL}/create`,
```

```

    {
      name: roomName,
      uniqueName: roomName,
    },
    {
      headers: this._authHeader(),
    }
  );

```

Sau khi gọi xong, ta ấn nút gọi đi và hiển thị nút copy room ID và ID room:

```

// Hiện nút copy ID room khi ấn vào nút gọi:
document.getElementById("btn-call").addEventListener("click", function () {
  document.getElementById("toolbar-center").style.paddingLeft = "8%";
  document.getElementById("btn-copy").style.display = "block";
});

```

```

<div class="toolbar-left time-linkroom info">
  <p id="time-display"></p>
  <span> | </span>
  <p class="linkroom" v-if="roomId">{{ roomId }}</p>
  <input type="hidden" value="" id="roomId_saved" />
  <button id="btn-copy" onclick="copyRoomId()">
    
  </button>
</div>

```

Hàm copyRoomID:

```

// Copy room id to clip board
function copyRoomId() {
  var copyText = document.getElementById("roomId_saved");

```

```

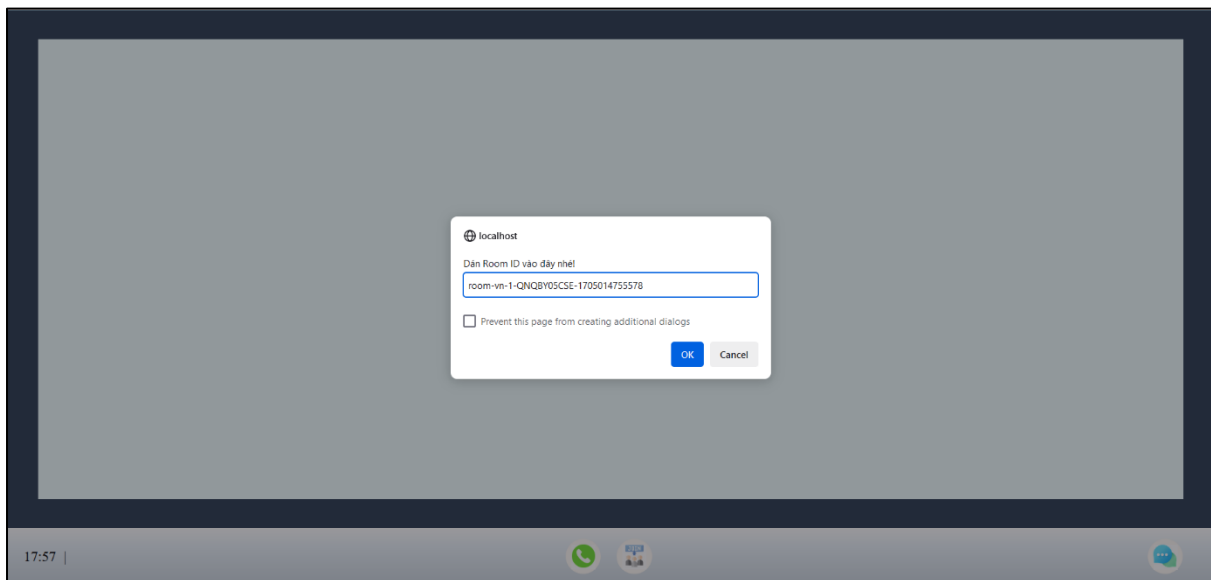
copyText.value = document.querySelector(".linkroom").innerText;

copyText.select();
navigator.clipboard.writeText(copyText.value);
alert(
    "Đã copy link phòng. Bạn có thể gửi nó cho những người khác cùng tham
gia."
);
}

```

#### 4. Tham gia vào phòng họp có sẵn.

Để tham gia vào một phòng họp, ta ấn vào nút có biểu tượng Join và nhập room ID vào bảng:



Ảnh 6 – Giao diện trang phòng họp khi join room

Đoạn code cho phép tham gia vào phòng họp từ room ID của API:

```

join: async function () {
    const roomToken = await api.getRoomToken(this.roomId);
    this.roomToken = roomToken;
    // Lấy lại roomToken vì khi được share thì người join vào chỉ có room
ID.

```

```

    await this.authen();
    await this.publish();
  },
  joinWithId: async function () {
    const roomId = prompt("Dán Room ID vào đây nhé!"); // Hiển thị hộp
    thoại để dán room Id vào.
    if (roomId) {
      // Nếu room ID đúng thì thực hiện hàm join.
      this.roomId = roomId;
      await this.join();
    }
  },
},

```

Hàm publish, hiển thị video người dùng:

```

publish: async function (screenSharing = false) {
  const localTrack = await StringeeVideo.createLocalVideoTrack(
    this.callClient,
    {
      audio: true,
      video: true,
      screen: screenSharing,
      videoDimensions: { width: 345, height: 180 },
    } // Cấu hình video
  );

  const videoElement = localTrack.attach();
  this.addVideo(videoElement);
  // Gắn local track vào video element bằng cách sử dụng phương thức
  attach.
  // Để hiển thị video trong giao diện người dùng.

  const roomData = await StringeeVideo.joinRoom(
    this.callClient,
    this.roomToken
  );
  // Lấy thông tin về phòng và danh sách các track có sẵn trong phòng
  sau khi quá trình join room hoàn tất.

```

```

    const room = roomData.room; // Lấy phòng từ roomData để quản lý các sự
    kiện và thêm các track vào phòng.
    console.log({ roomData, room });

    if (!this.room) {
        this.room = room;
        room.clearAllOnMethos(); // Xóa các track (video, user) khi họ rời
        khỏi room.
        room.on("addtrack", (e) => {
            const track = e.info.track;

            console.log("addtrack", track);
            if (track.serverId === localTrack.serverId) {
                console.log("local");
                return;
            }
            this.subscribe(track);
        });
        // Đặt một sự kiện lắng nghe cho sự kiện "addtrack" của phòng.
        // Khi một track mới được thêm vào phòng, hàm callback được gọi.
        // Trong trường hợp này, nó kiểm tra xem track đó có phải là local
        track không.
        // Nếu không phải, nó gọi phương thức subscribe để đăng ký track
        mới.

        room.on("removetrack", (e) => {
            const track = e.track;
            if (!track) {
                return;
            }

            const mediaElements = track.detach();
            mediaElements.forEach((element) => element.remove());
        });
        // Đặt một sự kiện lắng nghe cho sự kiện "removetrack" của phòng.
        Khi một track bị xóa khỏi phòng, hàm callback được gọi.
        // Xóa đối tượng video element liên quan đến track đã bị xóa.

```

```

    // Join existing tracks
    roomData.listTracksInfo.forEach((info) => this.subscribe(info));
  }

  await room.publish(localTrack);
  // Gọi phương thức publish trên đối tượng room để xuất bản (publish)
  localTrack lên phòng.
  // Giúp người dùng hiển thị video và audio của mình trong phòng.
  console.log("Room publish successful");
}

```

Hàm authen để xác thực người dùng:

```

authen: function () {
  return new Promise(async (resolve) => {
    // Phương thức authen trả về một promise.
    const userId = `${(Math.random() * 100000).toFixed(6)}`; // Tạo một
    userId ngẫu nhiên bằng cách nhân một số ngẫu nhiên với 100000 và giới hạn số
    lượng chữ số thập phân thành 6.
    const userToken = await api.getUserToken(userId); // Gọi hàm
    getUserToken từ đối tượng api bằng cách sử dụng await để đợi cho hàm này
    thực hiện xong.
    this.userToken = userToken;

    if (!this.callClient) {
      // Kiểm tra xem thuộc tính callClient có tồn tại không. Nếu chưa
      tồn tại, nghĩa là chưa có đối tượng StringeeClient được khởi tạo.
      const client = new StringeeClient();

      client.on("authen", function (res) {
        console.log("on authen: ", res);
        resolve(res);
        // Đặt một sự kiện lắng nghe ("authen") trên đối tượng client.
        // Khi sự kiện này xảy ra, hàm callback được gọi và in ra thông
        tin res của sự kiện.
      });
    }
  });
}

```

```

    this.callClient = client;
    // Gán giá trị của client cho thuộc tính callClient của đối tượng
Vue.

    // Đảm bảo rằng callClient chỉ được khởi tạo một lần và có thể
được sử dụng cho các kết nối tiếp theo.
}

    this.callClient.connect(userToken);
    // Gọi phương thức connect trên đối tượng callClient với tham số
userToken.

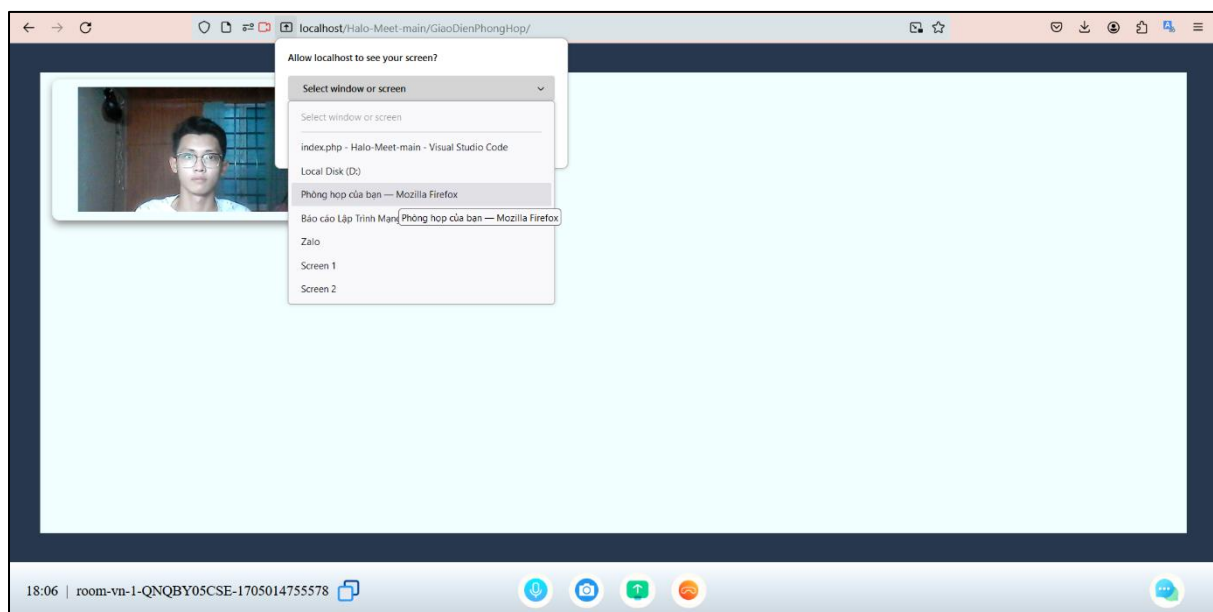
    // Khởi tạo quá trình kết nối của callClient với server, sử dụng
userToken để xác thực.
});
}

```

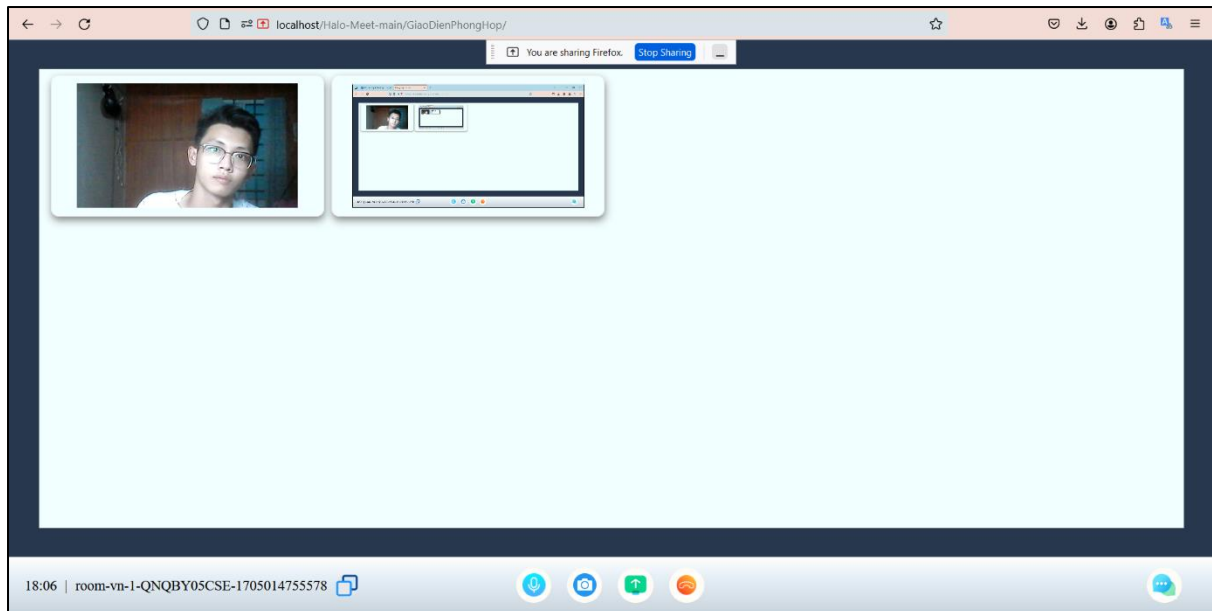
## 5. Share màn hình.

Trong sự kiện click của nút share màn hình, ta gọi đến hàm publish trên để hiển thị video quay màn hình:

Giao diện:



Ảnh 7 – Giao diện trang phòng họp chọn màn hình để share



Ảnh 8 – Giao diện trang phòng họp khi share màn hình thành công

Code:

```
<!-- Button Share Screen -->
<div class="btn-wrapper" v-if="room">
  <button href="" class="toolbar-btn" @click="publish(true)">
    
  </button>
</div>
```

## IV. Tổng kết

### 1. Kết quả đạt được

- Khi sử dụng người dùng cần phải tạo tài khoản để đăng nhập.
- Người dùng có thể truy cập vào Halo Meet mọi lúc và ở mọi nơi.
- Có thể sử dụng chức năng call video, chia sẻ màn hình.
- Giao diện đơn giản, thân thiện với người dùng và dễ sử dụng.



- Có thể tạo một phòng họp nhanh hoặc dùng mã được chia sẻ vào phòng một cách nhanh chóng.

## **2. Ưu điểm**

- Có thể tạo một cuộc họp để kết nối mọi người lại với nhau ở mọi nơi.
- Giao diện đơn giản dễ sử dụng.

## **3. Hạn chế**

- CSS phòng họp chính chưa được hoàn chỉnh.
- Chưa làm được chức năng nhắn tin trong phòng họp, quay video màn hình cuộc họp.

## **4. Hướng phát triển**

- Phát triển thêm phần nhắn tin để người dùng có thể nói chuyện với nhau qua tin nhắn.
- Phát triển thêm chức năng ghi lại video.
- Phát triển thêm ứng dụng điện thoại để có thể tương tác điện thoại với người dùng sử dụng trên web.

## PHỤ LỤC

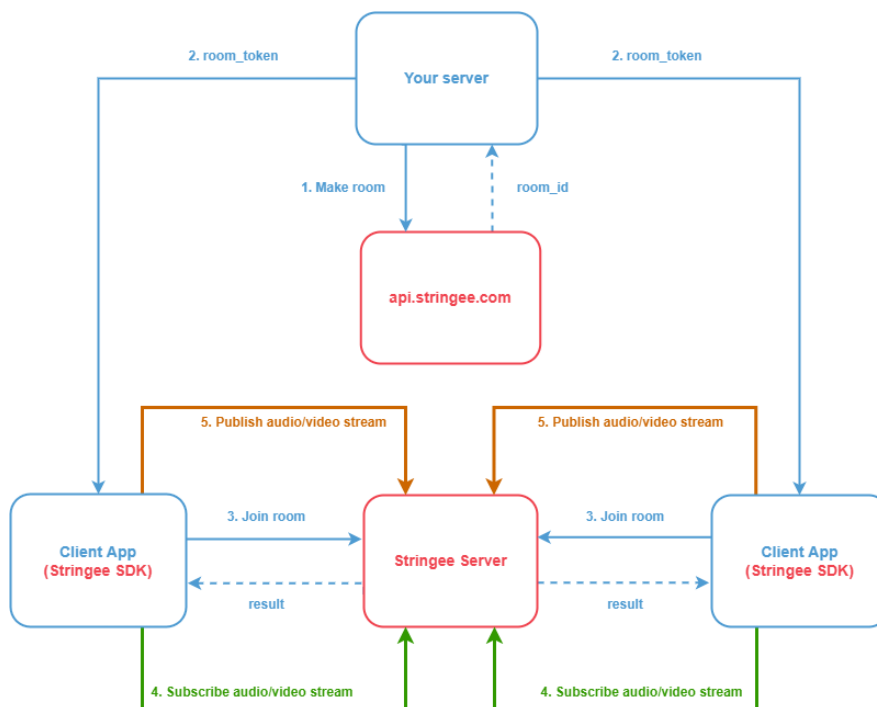
### Tổng quan về API Hội nghị Video Stringee.

- Ứng dụng khách hàng của bạn sẽ sử dụng SDK Hội nghị Video Stringee cho Android, iOS hoặc JavaScript để kết nối với một Phòng (trên Máy chủ Stringee) và xuất bản các luồng âm thanh và video.
- Các kết nối đa phương tiện sẽ được định tuyến thông qua cơ sở hạ tầng máy chủ truyền thông toàn cầu của Stringee trong tất cả các trường hợp. Bạn kiểm soát trình bày video và hành vi trong mã phía máy khách của mình.

### Xác thực khách hàng

- Để sử dụng API Hội nghị Video Stringee, ứng dụng của bạn phải được xác thực.
- Quá trình xác thực được triển khai giữa Ứng dụng của bạn, Máy chủ của bạn và Máy chủ Stringee. Nó đảm bảo an ninh hệ thống (khóa bí mật không được lưu trữ tại máy khách) và cho phép Máy chủ của bạn chặn hoặc mở chặn bất kỳ khách hàng nào mọi lúc bạn muốn.
- Quá trình xác thực được mô tả tại đây: <https://developer.stringee.com/docs/client-authentication>

### Làm thế nào Stringee Video Conference hoạt động



Ảnh 9 – Sơ đồ xử lý của Stringee Video Conference

- 1.** Máy chủ của bạn gọi REST API <https://api.stringee.com> để tạo phòng
- 2.** Máy chủ của bạn sử dụng ID phòng để tạo room\_token
- 3.** Ứng dụng Client sử dụng room\_token gọi phương thức joinRoom(stringeeClient, roomToken) để tham gia phòng.
- 4.** Ứng dụng Client gọi publish(StringeeStream) để xuất bản một luồng đến Máy chủ Stringee.
- 5.** Ứng dụng Client gọi subscribe(Stringeestream) để đăng ký luồng của client khác.

## TÀI LIỆU THAM KHẢO

1. JavaScript là gì? Kiến thức cơ bản về JavaScript cho người mới bắt đầu 2022 (2022), *JavaScript là gì? Kiến thức cơ bản về JavaScript cho người mới bắt đầu 2022*, <https://tnten.vn/tin-tuc/javascript-la-gi/> (6/12/2023)
2. Lê Hoàng (2023), *JavaScript là gì? Kiến thức cơ bản cho người mới bắt đầu*, <https://stringee.com/vi/blog/post/javascript-la-gi-cach-viet-co-ban#:~:text=2,-,L%E1%BB%8Bch%20s%E1%BB%AD%20ph%C3%A1t%20tri%E1%BB%83n%20c%E1%BB%A7a%20JavaScript,t%C3%A1c%20C4%91%E1%BB%99ng%20cho%20ng%C6%B0%E1%BB%9Di%20d%C3%B9ng.> (6/12/2023)
3. *JavaScript là gì? - Giải thích về JavaScript (JS)* (2023), *JavaScript là gì?*, <https://aws.amazon.com/vi/what-is/javascript/> (6/12/2023)
- Tò Tee Te (2018), *JavaScript là gì? Có vai trò gì? Cách bật JavaScript trên điện thoại*, <https://www.thegioididong.com/hoi-dap/javascript-la-gi-co-vai-tro-gi-cach-bat-javascript-tren-1321801> (6/12/2023)
4. Tò Tee Te (2018), *JavaScript là gì? Có vai trò gì? Cách bật JavaScript trên điện thoại*, <https://www.thegioididong.com/hoi-dap/javascript-la-gi-co-vai-tro-gi-cach-bat-javascript-tren-1321801> (6/12/2023)
5. HTML là gì? (2020), *HTML là gì?*, <https://topdev.vn/blog/html-la-gi/> (6/12/2023)
6. HTML là gì? Ưu điểm, nhược điểm và cách hoạt động của HTML (2020), *HTML là gì? Ưu điểm, nhược điểm và cách hoạt động của HTML*, <https://itnavi.com.vn/blog/html-la-gi/> (6/12/2023)
7. Glints (2022), *CSS Là Gì? Ngôn Ngữ CSS Dùng Để Làm Gì?*, <https://glints.com/vn/blog/css-la-gi/> (6/12/2023)
8. CSS là gì? Những thông tin cần biết về ngôn ngữ CSS (2023), *CSS là gì? Những thông tin cần biết về ngôn ngữ CSS*, <https://mona.media/css-la-gi/> (6/12/2023)

9. Hung Nguyễn (2022), *Visual Studio Code là gì? Những ưu điểm vượt trội khi sử dụng Visual Studio Code*, <https://vietnix.vn/visual-studio-code-la-gi/> (6/12/2023)

*Hết*