# LỜI CAM ĐOAN

Chúng em xin cam đoan đề tài “*Xây dựng website trò chuyện”* là đề tài của em và dưới đây là sản phẩm mà chúng em tự tìm hiểu, học hỏi, nghiên cứu và hoàn thiện. Vì vậy, tất cả những gì chúng em viết dưới đây đều là sự thật, là công sức của em cùng với sự giúp đỡ nhiệt tình của các thầy, cô giáo trong Viện Kỹ thuật và Công nghệ, đặc biệt là thầy giáo ThS Trần Xuân Hào đã tận tình hướng dẫn. Vì vậy, nếu phát hiện có bất kỳ gian lận nào thì chúng em xin chịu hoàn toàn trách nhiệm.

*Nghệ An, ngày 24 tháng 5 năm 2020*

Sinh viên thực hiện

Lê Văn Nhân

# MỤC LỤC

[DANH MỤC TỪ VIẾT TẮT 4](#_Toc41893611)

[DANH MỤC HÌNH VẼ, ĐỒ THỊ 5](#_Toc41893612)

[DANH MỤC BẢNG 6](#_Toc41893613)

[LỜI CẢM ƠN 7](#_Toc41893614)

[MỞ ĐẦU 8](#_Toc41893615)

[1. Lý do chọn đề tài 8](#_Toc41893616)

[2. Mục đích thực hiện 8](#_Toc41893617)

[3. Phạm vi thực hiện 9](#_Toc41893618)

[4. Nội dung thực hiện 9](#_Toc41893619)

[5. Cấu trúc đồ án 9](#_Toc41893620)

[CHƯƠNG 1. HÌNH THÀNH VÀ XÂY DỰNG Ý TƯỞNG 10](#_Toc41893621)

[1.1. Giới thiệu tổng quan, khảo sát hệ thống và đánh giá 10](#_Toc41893622)

[1.2.Yêu cầu của website cần xây dựng 10](#_Toc41893623)

[1.3. Khái quá chức năng hoạt động của website 11](#_Toc41893624)

[1.4. Các nghiên cứu liên quan đến đề tài 12](#_Toc41893625)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 14](#_Toc41893626)

[2.1. Phân tích hệ thống 14](#_Toc41893627)

[2.1.1. Quy trình đăng ký tài khoản 14](#_Toc41893628)

[2.1.2. Quy trình đăng nhập 14](#_Toc41893629)

[2.1.3. Quy trình tìm kiếm và thêm bạn bè 16](#_Toc41893630)

[2.1.4. Mô hình hoạt động call-video giữa 2 user 16](#_Toc41893631)

[2.2. Thiết kế CSDL 18](#_Toc41893632)

[CHƯƠNG 3. CÀI ĐẶT VÀ TRIỂN KHAI 22](#_Toc41893633)

[3.1. Công cụ sử dụng 22](#_Toc41893634)

[3.1.1. Ngôn ngữ và thư viện sử dụng thiết kế giao diện 22](#_Toc41893635)

[3.1.2. Ngôn ngữ lập trình và framework 23](#_Toc41893636)

[3.1.3. Môi trường và công cụ 24](#_Toc41893637)

[3.2. Giao diện khi vận hành website 24](#_Toc41893638)

[3.2.1. Trang đăng nhập, đăng ký tài khoản 24](#_Toc41893639)

[3.2.1. Thông báo kích hoạt tài khoản 26](#_Toc41893640)

[3.2.1. Giao diện trò chuyện 28](#_Toc41893641)

[KẾT LUẬN 31](#_Toc41893642)

[1. Những kết quả đạt được 31](#_Toc41893643)

[2. Những tồn tại và hạn chế 31](#_Toc41893644)

[3. Hướng phát triển đề tài 31](#_Toc41893645)

[TÀI LIỆU THAM KHẢO 32](#_Toc41893646)

DANH MỤC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Từ đầy đủ |
| HTML | Hypertext Markup Language |
| CSS | Cascading Style Sheets |
| EJS | Embedded JavaScript templating |
| API | Application Programming Interface |
| WebRTC | Web Real-Time Communication |

DANH MỤC HÌNH VẼ, ĐỒ THỊ

[Hình 1.1. Các hình thức đăng nhập 11](#_Toc41893881)

[Hình 2.1. Sơ đồ quy trình đăng kí tài khoản 14](#_Toc41893882)

[Hình 2.2.. Sơ đồ quy trình đăng nhập tài khoản tạo bằng Gmail 15](#_Toc41893883)

[Hình 2.3. Sơ đồ quy trình đăng nhập tài khoản với Google, Facebook 15](#_Toc41893884)

[Hình 2.4. Sơ đồ quy trình tìm kiếm và thêm bạn bè 16](#_Toc41893885)

[Hình 2.5. Sơ đồ quy trình thực hiên cuộc gọi giữa 2 user 17](#_Toc41893886)

[Hình 3.2. Bootstrap, EJS 22](#_Toc41893887)

[Hình 3.3. JS, NodeJS 23](#_Toc41893888)

[Hình 3.4. Express 23](#_Toc41893889)

[Hình 3.5. Vscode, Github 24](#_Toc41893890)

[Hình 3.6. Trang giao diện chính 25](#_Toc41893891)

[Hình 3.7. Giao diện đăng nhập 25](#_Toc41893892)

[Hình 3.7. Giao diện đăng ký tài khoản 26](#_Toc41893893)

[Hình 3.7. Giao diện đăng ký thành công xác nhận Gmail 26](#_Toc41893894)

[Hình 3.8. Tin nhắn xác nhận tài khoản thông qua Gmail 27](#_Toc41893895)

[Hình 3.8. Kết quả khi click kích hoạt 27](#_Toc41893896)

[Hình 3.9. Giao diện khi vào lần đầu tiên 28](#_Toc41893897)

[Hình 3.10. Giao diện tìm kiếm bạn bè 28](#_Toc41893898)

[Hình 3.11. Giao diện chat chính 29](#_Toc41893899)

[Hình 3.11. Danh bạ và một số chức năng 29](#_Toc41893900)

[Hình 3.12. Tạo nhóm trò chuyện 30](#_Toc41893901)

[Hình 3.13. Gọi video giữa 2 tài khoản 30](#_Toc41893902)

[Hình 3.14. Hình ảnh giao diện tham khảo từ facebook 33](#_Toc41893903)

[Hình 3.15. Hình thông báo tham khảo từ facebook 33](#_Toc41893904)

DANH MỤC BẢNG

[2.2.1. Bảng người dùng (users) 17](#_Toc41860490)

[2.2.2. Bảng liên hệ (contacts) 18](#_Toc41860491)

[2.2.2. Bảng tin nhắn (messages) 19](#_Toc41860492)

[2.2.2. Bảng trò chuyện nhóm (chat-groups) 19](#_Toc41860493)

[2.2.2. Bảng thông báo (notifications) 20](#_Toc41860494)

LỜI CẢM ƠN

Để hoàn thành đồ án báo cáo thực tập tốt nghiệp này em muốn gửi lời cảm ơn và biết ơn chân thành nhất của mình đến tất cả những người hỗ trợ, giúp đỡ em về kiến thức và tinh thần trong quá trình thực hiện đồ án.

Trước hết, em xin chân thành cảm ơn các quý thầy cô Viện Kỹ thuật và Công nghệ đã tận tình chỉ dạy em trong quá trình em học tập trên ghế nhà trường. Đặc biệt thầy Trần Xuân Hào người luôn chỉ dẫn tận tình, giúp em trong lúc khó khăn khi thực hiện đồ án này và em cũng đồng thời cám ơn các thầy cô khoa Công nghệ Thông tin, viện Kỹ thuật và Công nghệ đã giúp em có kiến thức và định hướng để hoàn thành đồ án này. Nhờ đó em có thêm được những bài học quý giá, những kinh nghiệm quý báu để em có thể trang bị cho bản thân vững bước trong con đường tương lai.

Do thời gian có hạn và kiến thức còn hạn chế nên đề tài còn nhiều thiếu sót. Vì vậy, em rất mong nhận được những ý kiến đóng góp từ quý thầy, cô và các bạn để đề tài được hoàn thành tốt hơn.

Em xin chân thành cảm ơn!

Nghệ An, ngày 20 tháng 05 năm 2020

**Sinh viên**

Lê Văn Nhân

MỞ ĐẦU

1. Lý do chọn đề tài

Trong thời đại hiện nay việc kết nối mọi người thông qua internet đã rất phổ biến và phát triển, nó đem đến sự thuận tiện nhất định trong việc trao đổi thông tin cũng như kết nối mọi người trong xã hội, với sự phát triển của một số mạng xã hội phổ biến như facebook, twitter, instagram…, con người dễ dàng thông qua tính năng của chúng để nhắn tin hay gọi điện dù ở khoảng cách địa lí, quốc gia khác biệt nó còn vượt lên khả năng mà điện thoại có thể đem lại trong việc liên lạc.

Mục đích của đề tài này nhằm tìm hiểu và xây dựng cách thức trao đổi như nhắn tín, call video… hay kết bạn giữa những người dùng với thời gian thực (real-time) cũng như việc thiết kế một cơ sở dữ liệu có khả năng lưu trữ và truy xuất hợp lí.

Vì vậy, với sự chỉ dẫn nhiệt tình của ThS. Trần Xuân Hào vào gợi ý thầy đưa ra em đã lựa chon đề tài “*Xây dựng website trò chuyện*” nhằm xây dưng một website mà người dùng có thể kết bạn hay trò chuyện và trao đổi thông tin.

2. Mục đích thực hiện

Xậy dưng một website cho phép người dùng có thể tạo tài khoản dễ dàng và có nhiều hình thức để vào website, người dùng cũng có thể tìm kiếm và kết bạn cũng như chấp nhận lời mời kết bạn, việc chat hay trao đổi thông tin ngay lập tức.

Tìm hiều về ngôn ngữ javascrip, cách xử lí real-time thông qua socket.io hay việc xử lí nghe gọi với Web RTC và PeerJS cũng như xậy dựng một database phù hợp với mongoDB thông qua nhữ kiến thức đã được học và các tài liệu tìm được. Xây dựng một website đáp ứng những yêu cầu trên, thêm được rất nhiều kiến thức trợ giúp cho công việc sau này.

3. Phạm vi thực hiện

Nghiên cứu trong phạm vi:

Phương pháp khảo sát : Nghiên cứu tìm hiểu phương thức hoạt động trao đổi giữa người dùng thông qua messenger của facebook.

Tìm hiểu hoạt đông quy trình gửi nhận thông tin cũng như cập nhật những công nghệ mới vào dự án.

4. Nội dung thực hiện

Xây dựng website Connect Chat mục đích kết nối mọi người cũng như tạo một không gian cho người dừng có thể trò chuyện và tậm sự với nhau.

Người dùng có thể tạo tài khoản và đăng nhập bằng nhiều hình thức khác nhau, tính năng chủ yếu của website cho phép người dùng có thể tìm kiếm, kết bạn, call-video, gửi tệp hình ảnh và icon, thông báo người dùng, tạo nhóm trò chuyện.

*Thời gian thực hiện: 5 tháng*

5. Cấu trúc đồ án

Ngoài các phần mở đầu, kết luận và tài liệu tham khảo thì báo cáo được trình bày trong ba chương, gồm những nội dung sau:

**Chương 1. Hình thành và xây dựng ý tưởng**

Chương này sẽ đưa ra những giới thiệu tổng quan về bài toán, cùng với việc trình bày các vấn đề liên quan đến hệ thống quản lý và đưa ra sơ đồ quy trình hoạt động của hệ thống.

**Chương 2. Phân tích và thiết kế hệ thống**

Chương này sẽ tiến hành phân tích và thiết kế chức năng của hệ thống, đưa ra các tác nhân ảnh hướng đến hệ thống, phác họa giao diện và thiết kế cơ sở dữ liệu.

**Chương 3. Cài đặt thử nghiệm và triển khai**

Chương này sẽ triển khai các hoạt động và trình bày về giao diện, chức năng của hệ thống đã xây dựng được.

CHƯƠNG 1. HÌNH THÀNH VÀ XÂY DỰNG Ý TƯỞNG

1.1. Giới thiệu tổng quan, khảo sát hệ thống và đánh giá

Hiện nay sự phát triển của công nghệ và internet đã ảnh hưởng đến toàn xã hội rất nhiều website được hình thành nhất là mạng xã hội như facebook, vậy tại sao nó lại phát triển mạnh mẽ như vậy? Không chỉ chia sẻ hình ảnh và tương tác như instagram mà facebook còn có tính năng kết bạn và trò chuyện riêng tư, hay call-video một tính năng rất được ưa chuông cho những người dùng ở cách xa nhau về khoảng cách địa lí, bạn có thể là sinh viên học xa nhà, nếu nhớ nhà bạn có thực hiện một cuộc gọi video là bạn có thể thoải mái trò chuyện với bố, mẹ mình mà không tốn quá nhiều chi phí như gọi điện thoại, hay việc thảo luận trao đổi riêng tư trong nhóm học tập mà bạn không cần thiết phải ngồi gặp mặt nhau, đay chỉ là phần nhỏ trong những lợi ích mà nó đem lại.

Qua những gì nói trên ta thấy tầm quan trọng của nó đã giúp facebook trở thành mạng xã hội đứng đầu cũng như lợi ích mà nó đem lại cho người dùng, trong dự án này chúng ta sẽ tìm hiểu và xây dựng một hệ thông tương tự có thể làm những việc nói trên, với một số kiến thức và công nghệ mà em học được, em đã sử dụng và nghiên cứu thông qua tính năng nổi bật đó và kết quả là sự ra đời của *website Connect Chat* đáp ứng đầy đủ yêu cầu nói trên, thuận tiện trong việc trao đổi thông tin cũng như việc liên lạc một cách nhanh chóng thuận tiện.

1.2.Yêu cầu của website cần xây dựng

Website dễ nhìn thân, giao diện thân thiện người dùng, không quá màu mè và hiệu quả, độ bảo mật thông tin tốt, tạo tài khoản dễ dàng và hộ trợ đăng nhập dưới nhiều hình thức, ví dụ: facebook, google...

Người dừng sau khi tạo tài khoản cần xác nhận Gmail để tài khoản cỏ thể kích hoạt, đối với hình thức đăng nhập bằng facebook hoặc google có thể lấy thông tin mà token cung cấp để tạo tài khoản, mật khẩu của tài khoản đăng kí thông qua gmail sẽ được bảo mật dưới dạng mã hóa.

Khi đăng nhập thành công người dùng có thể cập nhật lại thông tin cá nhận sao cho phù hợp, cũng như thay đổ mật khẩu nếu muốn.

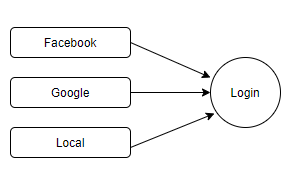
Tìm kiếm bạn bè có trong hệ thống để có thể gửi lời mời kết bạn và chờ họ xác nhận kết bạn cũng như chấp nhận lời mời kết bạn của một ai đó là hai bạn có thể chat với nhau thông qua website, khi chấp nhận kết bạn lập tức thêm ngay vòa danh sách bạn bè cũng như việc xóa bạn bè mọi thứ sẽ được xử lí real-time.

Gửi thông báo cho người gửi lời mời kết bạn nếu được chấp nhận kết bạn.

Gửi được các đoạn văn bản hay tệp hình ảnh, icon và file cũng như call-video với nhau, thông báo có tin nhắn mới.

1.3. Khái quá chức năng hoạt động của website

* Website cho phép đănng nhập bằng 3 hình thức chính:
* Đăng nhập bằng tài khoản được tạo và xác nhận thông qua Gmai.
* Đăng nhập bằng tài khoản Facebook.
* Đăng nhập bằng tài khoản Google.



Hình 1.1. Các hình thức đăng nhập

* Khi đăng nhập thành công, người dùng có thể sử dụng các tính năng như:
* Tìm kiếm bạn bè.
* Thêm bạn bè.
* Từ chối lời mời kết bạn.
* Xóa bạn bè.
* Tạo nhóm trò chuyện.
* Các tính năng trao đổi giữa người dùng
* Tin nhắn dạng văn bản.
* Tin nhắn dạng hình ảnh.
* Tin nhắn dạng tệp tin.
* Cuộc gọi Video.

1.4. Các nghiên cứu liên quan đến đề tài

**Socket.IO là** một thư viện JavaScript cho các ứng dụng web thời gian thực. Nó cho phép giao tiếp hai chiều thời gian thực giữa máy khách và máy chủ web. Nó có hai phần: thư viện phía máy khách chạy trong trình duyệt và thư viện phía máy chủ cho Node.js. Cả hai thành phần có một API gần như giống hệt nhau

**WebRTC là** một web API được phát triển bởi World Wide Web Consortium, khả năng hỗ trợ trình duyệt giao tiếp với nhau thông qua VideoCall, VoiceCall hay transfer data "Peer-to-Peer" mà không cần browser phải cài thêm plugins hay phần mềm hỗ trợ nào từ bên ngoài.

**PeerJS là** một thư viện JavaScript hoạt động như một trình bao bọc cho WebRTC và cho phép bạn tạo các kết nối ngang hàng rất dễ dàng. Với **PeerJS**, bạn có thể tạo kết nối như vậy chỉ trong 3 dòng mã. **PeerJS** giao dịch với bắt tay WebRTC và cho phép kết nối bằng ID ngang hàng.

**ICE là** một giao thức được cùng để thiết lập phiên media dựa trên UDP đi qua NAT một cách nhanh nhất. ICE sẽ tìm đường tốt nhất để kết nối giữa các peer, nó thử tất cả khả năng có thể kết nối một cách song song và lựa chọn con đường hiệu quả nhất.

**Mongoose là** một thư viện mô hình hóa đối tượng (Object Data Model - ODM) cho MongoDB và Node. js. Nó quản lý mối quan hệ giữa dữ liệu, cung cấp sự xác nhận giản đồ và được sử dụng để dịch giữa các đối tượng trong mã và biểu diễn các đối tượng trong MongoDB.

**PeerJS là** một thư viện JavaScript hoạt động như một trình bao bọc cho WebRTC và cho phép bạn tạo các kết nối ngang hàng rất dễ dàng. Với PeerJS, bạn có thể tạo kết nối như vậy chỉ trong 3 dòng mã. PeerJS giao dịch với bắt tay WebRTC và cho phép kết nối bằng ID ngang hàng.

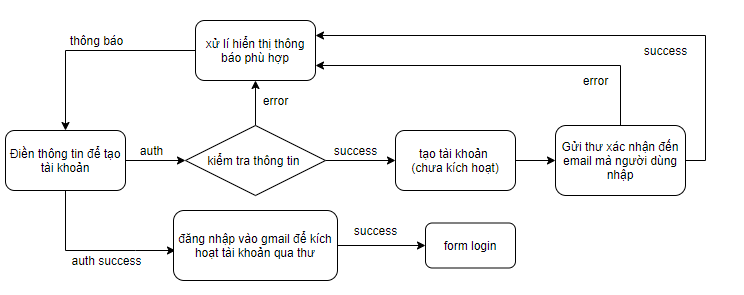
[**Passport**](https://github.com/jaredhanson/passport)**là** một thư viện javascript cung cấp cơ chế cho phép dễ dàng thiết lập hệ thống xác thực và đăng ký hỗ trợ [nhiều frameworks và các cách xác thực khác nhau](https://github.com/jaredhanson/passport" \l "strategies" \t "_blank).

CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Phân tích hệ thống

2.1.1. Quy trình đăng ký tài khoản

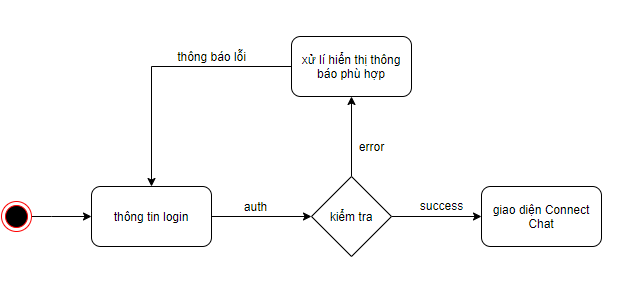
Sau khi điền đầy đủ thông tin sẽ tạo tài khoản nhưng ở trạng thái chưa kích hoạt đồng thời gửi thư qua gmail đã đăng ký nếu lỗi trả về thông báo lỗi, thành công thì thông báo người dùng vào gmail kích hoạt, khi người dừng vào gmail kích hoạt tài khoản mới sử dụng được.



Hình 2.1. Sơ đồ quy trình đăng kí tài khoản

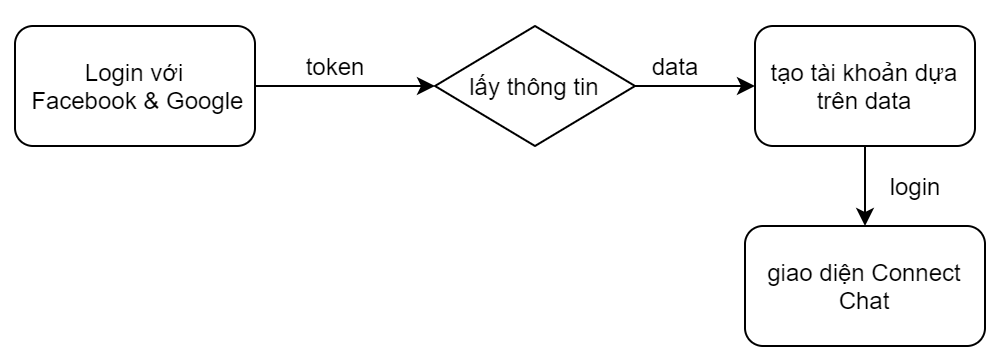
2.1.2. Quy trình đăng nhập

Sau khi đăng ký thành công và kích hoạt bạn có thể điền thông tin vào để đăng nhập, nếu thông tin nhập chưa đúng trả về thông báo lỗi, nếu đúng login vào giao diện chính của Connect Chat.



Hình 2.2.. Sơ đồ quy trình đăng nhập tài khoản tạo bằng Gmail

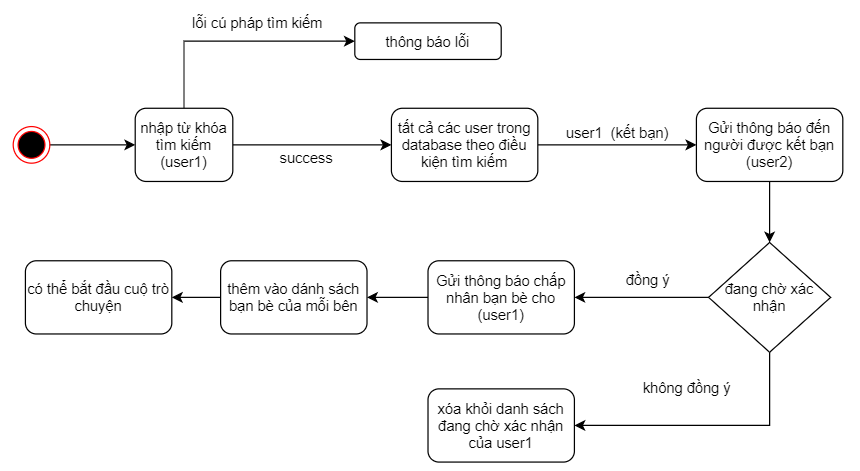
Với các hình thức còn lại như đăng nhập bằng facebook hoặc google xử lí và cấu hình để có thể lấy được thông tin từ token mà facebook và google cung cấp khởi tạo tài khoản bằng những thông tin đó và vào giao diện chính của Connect Chat.



Hình 2.3. Sơ đồ quy trình đăng nhập tài khoản với Google, Facebook

2.1.3. Quy trình tìm kiếm và thêm bạn bè

Sau khi đăng nhập thành công người dừng có thể tìm kiếm bạn bè để bắt đầu cuộc trò chuyện.



Hình 2.4. Sơ đồ quy trình tìm kiếm và thêm bạn bè

2.1.4. Mô hình hoạt động call-video giữa 2 user

Để thực hiên cuộc gọi call-video giữa 2 user ta thực hiện các bước:

Bước 1: User 1 gửi request lên sever để kiểm tra User 2 có online hay không.

Bước 2: Nếu User 2 không online kết thúc cuộc gọi.

Bước 3: Nếu User 2 online Sever sẽ lấy PeerId và data của User 2 ( PeerId là từ thư viện Peerjs mà có “ một thư viện hỗ trợ dùng để thực hiên cuộc gọi ” ).

Bước 4: User 2 gửi data và PeerId cho server.

Bước 5: Server gửi peerId của User 2 cho User 1.

Bước 6: User 1 gọi đến server.

Bước 7: Có thể hủy yêu cầu cuộn gọi.

Bước 8: Server gọi cho User 2 sau khi xử lí thành công bước 6.

Bước 9: Sau khi thực hiên bước 7 (hủy cuộc gọi từ user 1) server thực hiện hủy cuộc gọi đến User 2.

Bước 10: User 2 từ chối cuội gọi gửi trả về Server

Bước 11: User 2 chấp nhận cuộ gọi gửi trả về server

Bước 12: Server gửi hủy cuộc gọi của User 2 cho User 1

Bước 13: Server gửi chấp nhận cuộc của User cho User 1

Bước 14: Server gửi chấp nhận cuộc gọi cho User 2

Hoàn thành kết nối 2 client thông qua Sever.

## 

Hình 2.5. Sơ đồ quy trình thực hiên cuộc gọi giữa 2 user

2.2. Thiết kế CSDL

2.2.1. Bảng người dùng (users)

|  |  |  |  |
| --- | --- | --- | --- |
| No | FieldName | Type | Description |
| 1 | \_id | String | Mã người dùng |
| 2 | username | String | Tên người dùng |
| 3 | gender | String | Giới tính |
| 4 | phone | String | Điện thoại |
| 5 | adress | String | Địa chỉ |
| 6 | avatar | String | Ảnh đại diện |
| 7 | role | String | Vai trò |
| 8 | local | Object | Tài khoản tạo bằng Gmail |
|  | email | String | Tên đăng nhập |
|  | password | String | Mật khẩu |
|  | isActive | boonlean | Trạng thái tài khoản (kích hoạt hay chưa kích hoạt) |
|  | verifyToken | String | Mã xác thực |
| 9 | facebook | Object | Tài khoản đăng nhập với Facebook |
|  | uid | String | Id nhận từ Facebook |
|  | token | String | Mã xác thực |
|  | email | String | Tài khoảng email mà Facebook cung cấp |
| 10 | google | Object | Tài khoản đăng nhập với Google |
|  | uid | String | Id nhận từ Google |
|  | token | String | Mã xác thực |
|  | email | String | Tài khoảng email mà Google cung cấp |
| 11 | createdAt | Timestamp | Ngày tạo |
| 12 | updatedAt | Timestamp | Ngày cập nhật |
| 13 | deletedAt | Timestamp | Ngày xóa |

2.2.2. Bảng liên hệ (contacts)

|  |  |  |  |
| --- | --- | --- | --- |
| No | FieldName | Type | Description |
| 1 | \_id | String | Mã liên lạc |
| 2 | userId | String | Mã người dùng |
| 3 | contactId | String | Mã người dùng là bạn bè |
| 4 | status | Boolean | Trạng thái đã kết bạn hay chưa |
| 5 | createdAt | Timestamp | Ngày tạo |
| 6 | updatedAt | Timestamp | Ngày cập nhật |
| 7 | deletedAt | Timestamp | Ngày xóa |

2.2.2. Bảng tin nhắn (messages)

|  |  |  |  |
| --- | --- | --- | --- |
| No | FieldName | Type | Description |
| 1 | \_id | String | Mã tin nhắn |
| 2 | senderId | String | Mã người gửi |
| 3 | receiverId | String | Mã người nhận |
| 4 | conversationType | String | Loại hội thoại |
| 5 | messageType | String | Loại tin nhắn |
| 6 | sender | Object | Người gửi |
|  | id | String | Mã người gửi tin nhắn |
|  | name | String | Tên người gửi tin nhắn |
|  | avatar | String | Ảnh đại diên người gửi tin nhắn |
| 7 | receiver | Object | Người nhận |
|  | id | String | Mã người nhận tin nhắn |
|  | name | String | Tên người nhận tin nhắn |
|  | avatar | String | Ảnh đại diên người nhận tin nhắn |
| 8 | text | String | Tin nhắn văn bản |
| 9 | file | Object | Tin nhắn tập tin |
|  | data | Buffer | Dữ liệu tập tin lưu dạng Base64 |
|  | contentType | String | loại định dạng |
|  | fileName | String | Tên tập tin |
| 10 | createdAt | Timestamp | Ngày tạo |
| 11 | updatedAt | Timestamp | Ngày cập nhật |
| 12 | deletedAt | Timestamp | Ngày xóa |

2.2.2. Bảng trò chuyện nhóm (chat-groups)

|  |  |  |  |
| --- | --- | --- | --- |
| No | FieldName | Type | Description |
| 1 | \_id | String | Mã nhóm |
| 2 | name | String | Tên nhóm |
| 3 | userAmount | Integer | Số thành viên của nhóm |
| 4 | messagesAmount | Integer | Số tin nhắn |
| 5 | userId | String | Mã người tạo nhóm |
| 6 | member | Object | Thành viên |
|  | userId | String | Mã thành viên |
| 7 | createdAt | Timestamp | Ngày tạo |
| 8 | updatedAt | Timestamp | Ngày cập nhật |
| 9 | deletedAt | Timestamp | Ngày xóa |

2.2.2. Bảng thông báo (notifications)

|  |  |  |  |
| --- | --- | --- | --- |
| No | FieldName | Type | Description |
| 1 | \_id | String | Mã thông báo |
| 2 | senderId | String | Mã người gửi |
| 3 | receiverId | String | Mã người nhận |
| 4 | type | String | Loại thông báo |
| 5 | isRead | Boolean | Đã đọc thông báo hay chưa |
| 6 | createdAt | Timestamp | Ngày tạo |

CHƯƠNG 3. CÀI ĐẶT VÀ TRIỂN KHAI

3.1. Công cụ sử dụng

3.1.1. Ngôn ngữ và thư viện sử dụng thiết kế giao diện

**HTML, CSS và Javascript là** bộ 3 ngôn ngữ rất quan trọng cho sự phát triển web. Cho dù bạn muốn xây dựng một website đơn giản hay phức tạp đều không thể bỏ qua việc sử dụng chúng.

*Hình 3.1. HTML, CSS và Javascript*

**Bootstrap là** một khung CSS miễn phí và mã nguồn mở hướng vào phát triển web đầu tiên đáp ứng, di động. Nó chứa các mẫu thiết kế dựa trên CSS và JavaScript cho kiểu chữ, biểu mẫu, nút, điều hướng và các thành phần giao diện khác.

**EJS là** viết tắt của "Embedded JavaScript templating", đây là một thư viện, được sử dụng để phân tích các tập tin ejs, và tạo ra HTML trả về cho client (Trình duyệt).

### Node js View Engine: Try out some of the most popular - FrontNet Blog

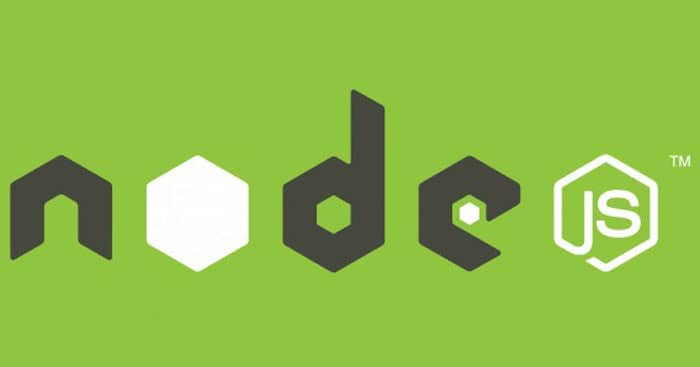
Hình 3.2. Bootstrap, EJS

3.1.2. Ngôn ngữ lập trình và framework

**JavaScript là** một ngôn ngữ lập trình thông dịch được phát triển từ các ý niệm nguyên mẫu. Ngôn ngữ này được dùng rộng rãi cho các trang web cũng như phía máy chủ.

**Nodejs là** một **nền tảng** (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.





Hình 3.3. JS, NodeJS

**Express là** một khung ứng dụng web cho Node.js, được phát hành dưới dạng phần mềm miễn phí và nguồn mở theo Giấy phép MIT. Nó được thiết kế để xây dựng các ứng dụng web và API. Nó đã được gọi là khung máy chủ tiêu chuẩn thực tế cho Node.js

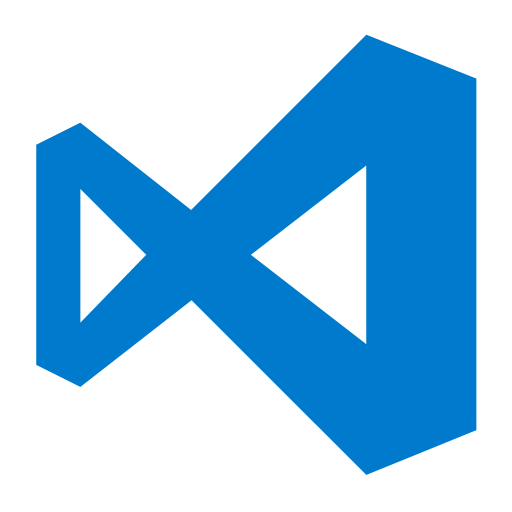


Hình 3.4. Express

3.1.3. Môi trường và công cụ

**Visual Studio Code là** một trình biên tập mã được phát triển bởi Microsoft dành cho Windows, Linux và macOS. Nó hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn.

**GitHub là** một dịch vụ cung cấp kho lưu trữ mã nguồn Git dựa trên nền web cho các dự án phát triển phần mềm. GitHub cung cấp cả phiên bản trả tiền lẫn miễn phí cho các tài khoản. Các dự án mã nguồn mở sẽ được cung cấp kho lưu trữ miễn phí.



Hình 3.5. Vscode, Github

**Socket.IO là** một thư viện JavaScript cho các ứng dụng web thời gian thực. Nó cho phép giao tiếp hai chiều thời gian thực giữa máy khách và máy chủ web. Nó có hai phần: thư viện phía máy khách chạy trong trình duyệt và thư viện phía máy chủ cho Node.js. Cả hai thành phần có một API gần như giống hệt nhau.

3.2. Giao diện khi vận hành website

3.2.1. Trang đăng nhập, đăng ký tài khoản

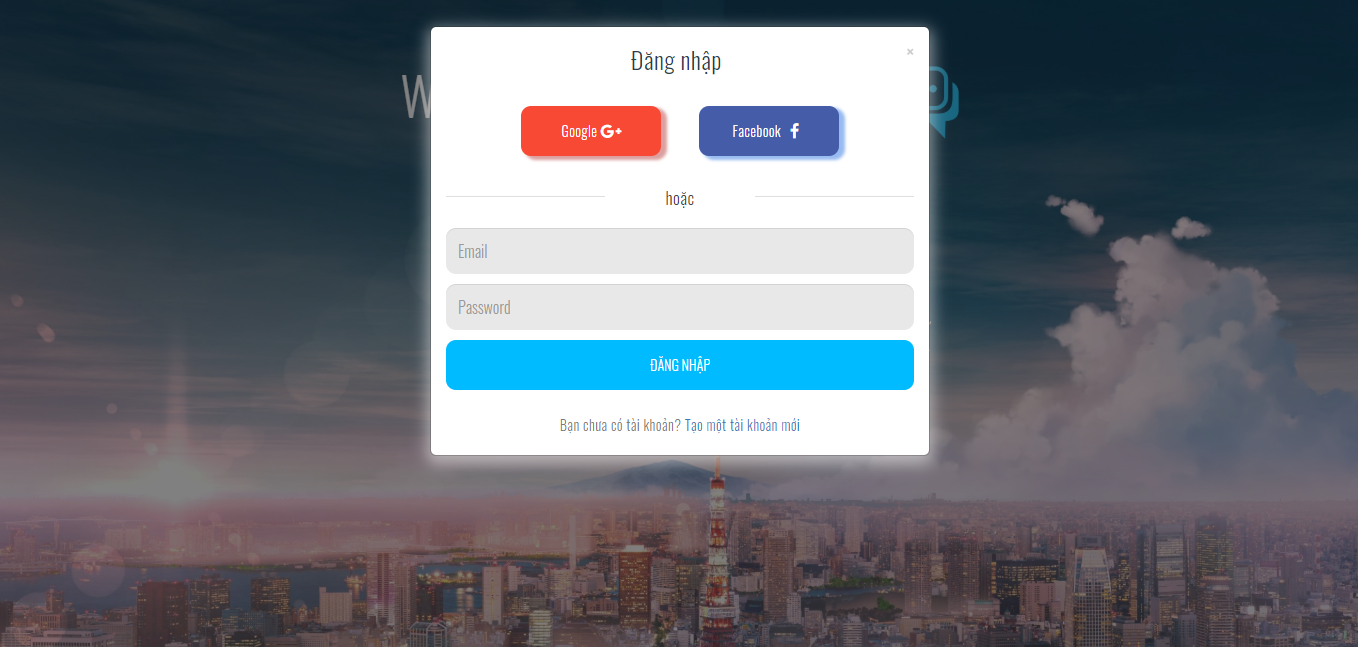
Tại đây người dùng sẽ tiến hành đăng nhập bằng 3 hình thức tài khoản local ( Tạo thông qua xác nhận Gmail ), Google, Facebook.

Sử dụng Gmail để đăng kí và xác thực tài khoản.

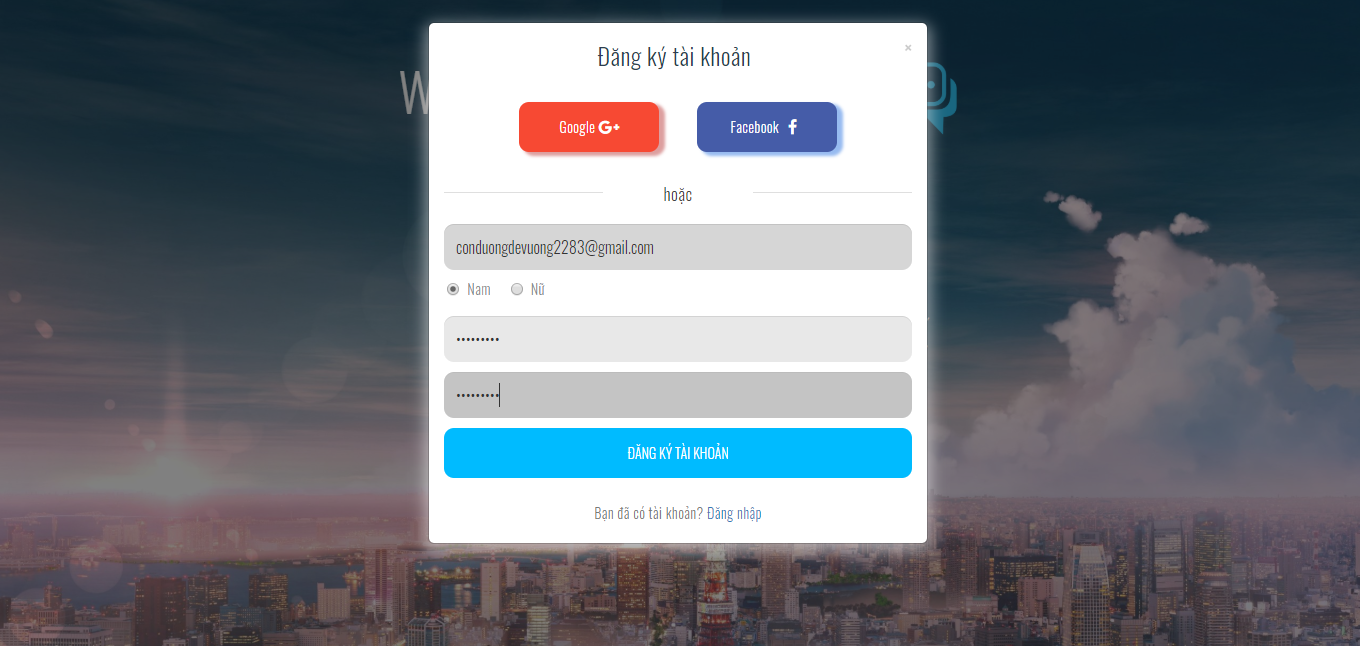
Sau đó người dùng sẽ được chuyển đến trang giao diện chính để có thể sử dụng chức năng chính của website.



Hình 3.6. Trang giao diện chính



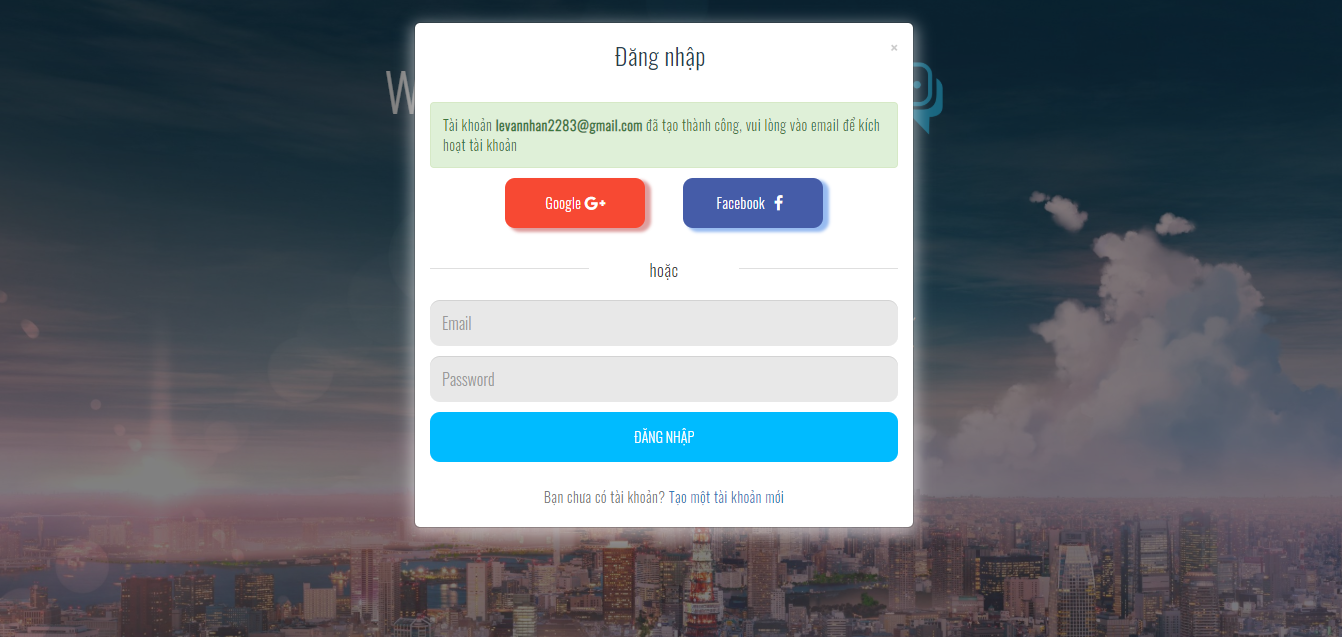
Hình 3.7. Giao diện đăng nhập



Hình 3.7. Giao diện đăng ký tài khoản

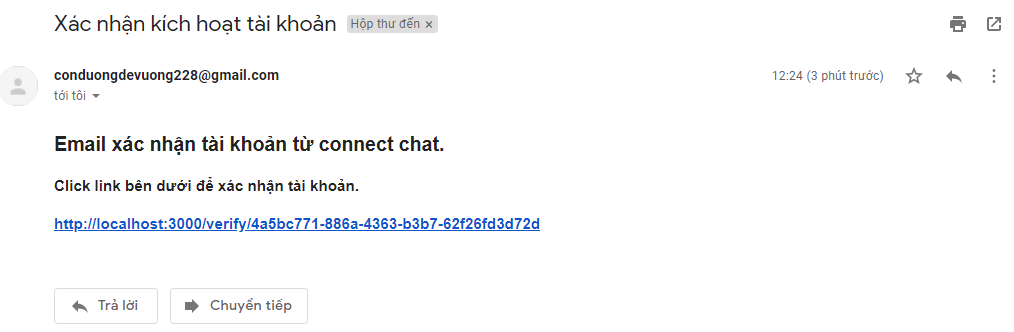
3.2.1. Thông báo kích hoạt tài khoản

Sau khi thực hiện đăng ký thành công không xuất hiện lỗi thì bạn sẽ có một thông báo về việc kích hoạt tài khoản thông qua Gmail mà bạn đã đăng ký.

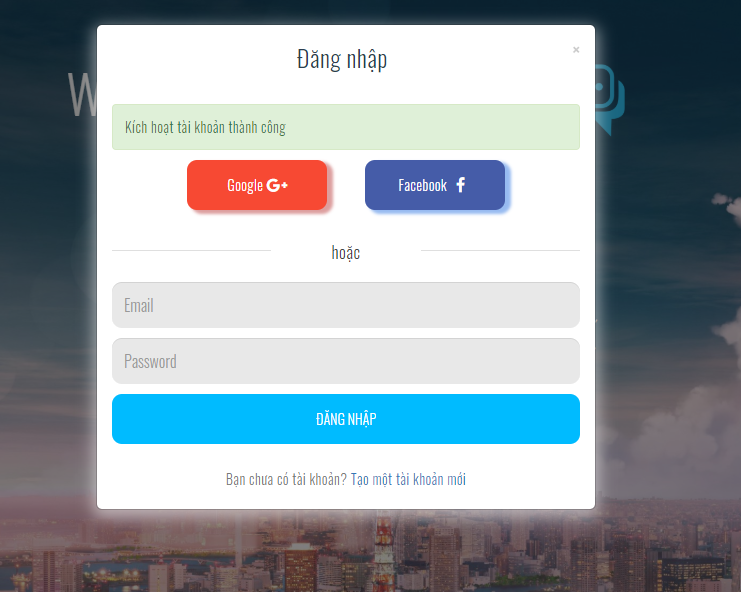


Hình 3.7. Giao diện đăng ký thành công xác nhận Gmail

Vào kiểm tra Gmail bạn sẽ thấy một tin nhắn thông báo cho việc kích hoạt tài khoản, click vào đường link để kích hoạt.



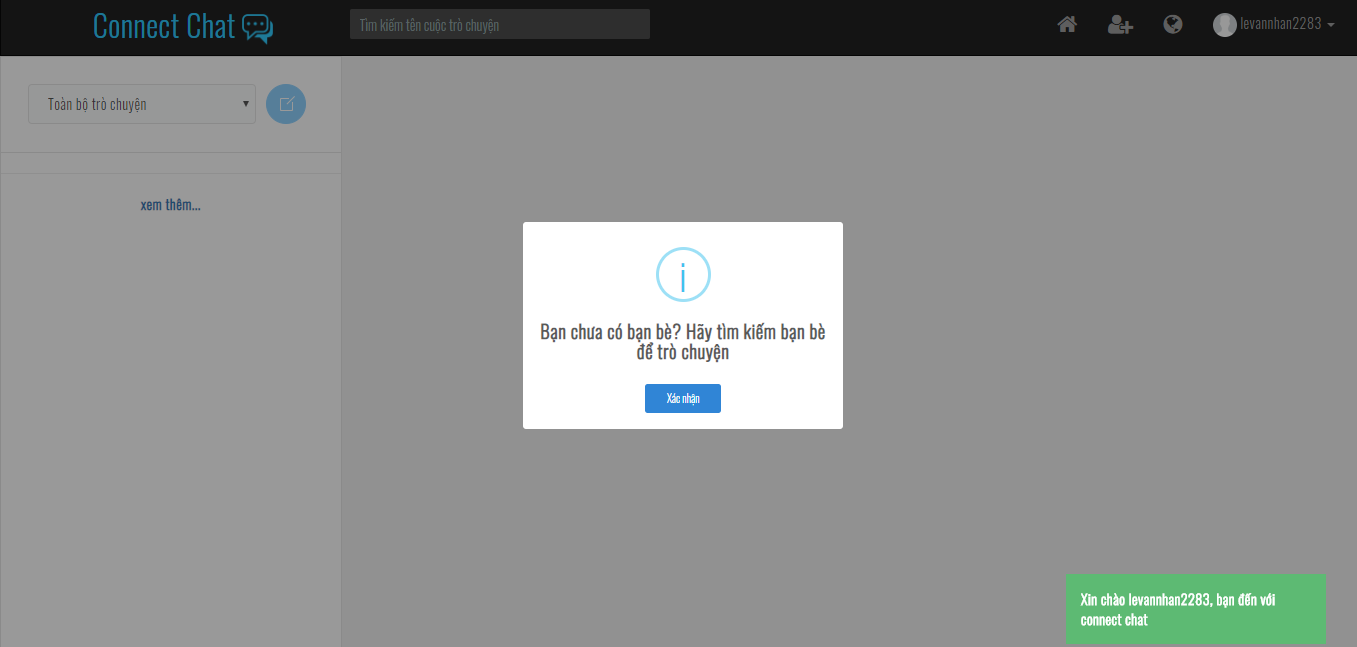
Hình 3.8. Tin nhắn xác nhận tài khoản thông qua Gmail



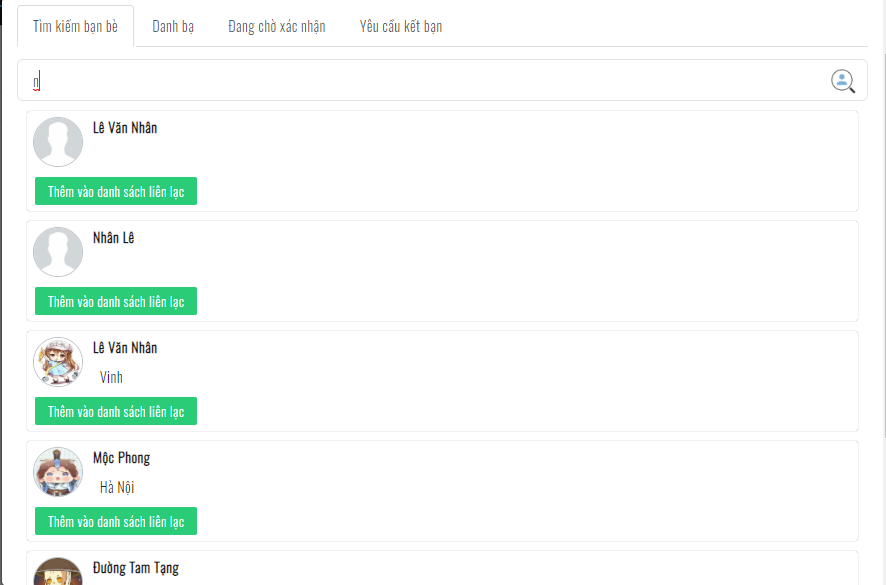
Hình 3.8. Kết quả khi click kích hoạt

3.2.1. Giao diện trò chuyện

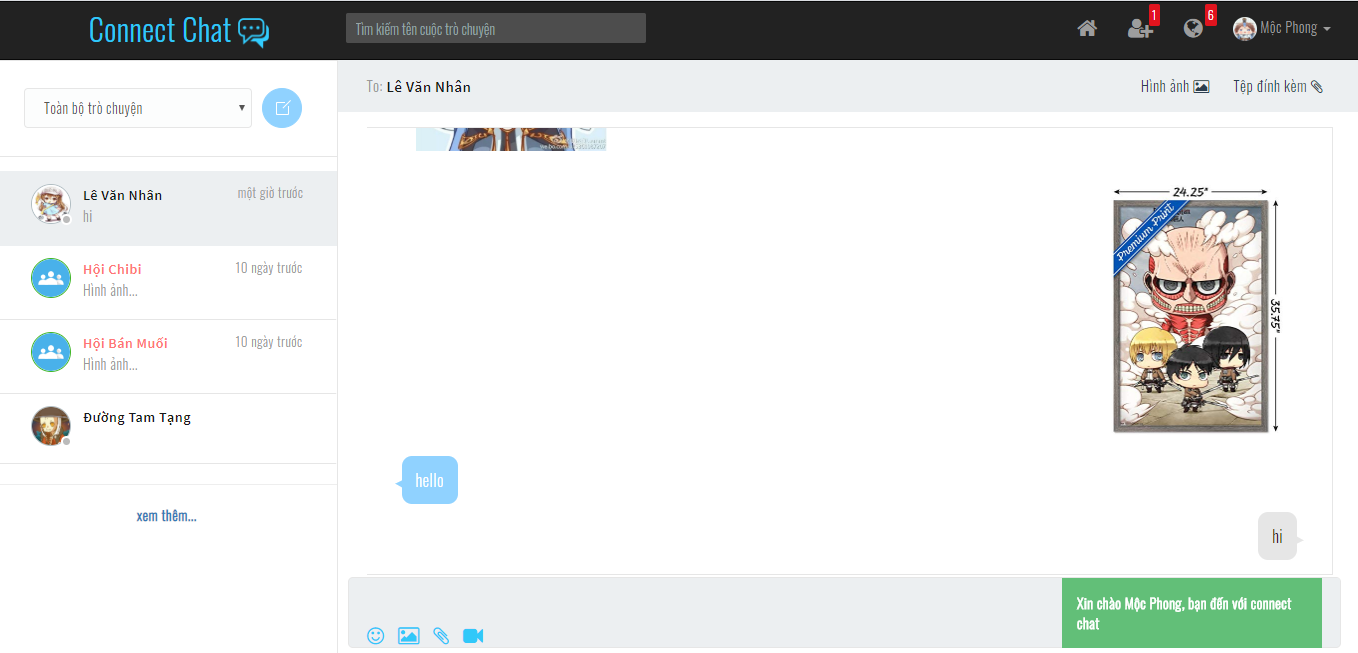
Đây là giao diện lần đầu khi bạn đăng nhập vào hệ thống, website sẽ thông báo cho bạn biết rằng bạn chưa có bạn bè nào và gợi ý kết bạn.



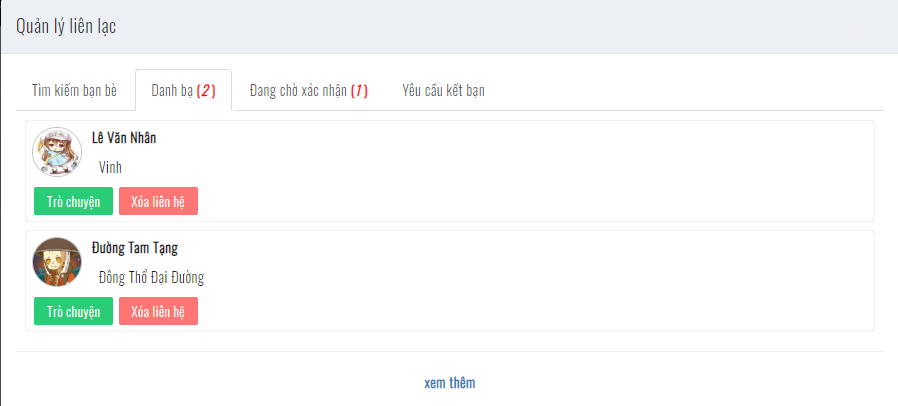
Hình 3.9. Giao diện khi vào lần đầu tiên



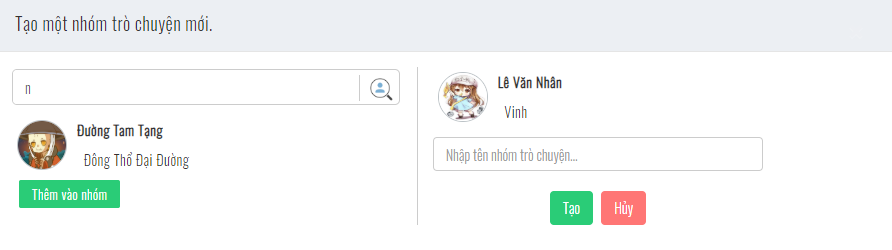
Hình 3.10. Giao diện tìm kiếm bạn bè



Hình 3.11. Giao diện chat chính

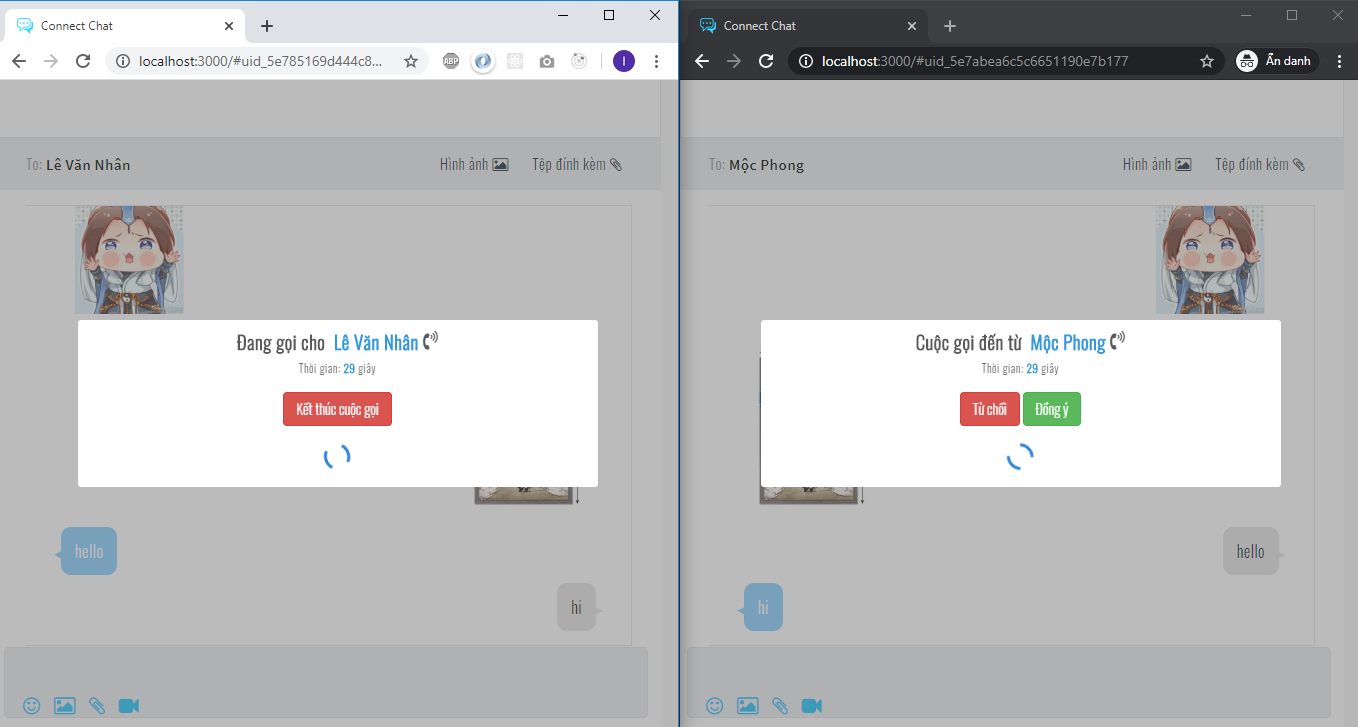


Hình 3.11. Danh bạ và một số chức năng

Hình 3.12. Tạo nhóm trò chuyện

Để thực hiện cuộc gọi nếu với chỉ thông qua Socket Server thì bạn chỉ có thể gọi thông mạng LAN thông thường, nếu ở ngoài mạng LAN thì sẽ không thực hiên được cho nên ta cần có thêm turn server để thực hiện điều đó.

Đối với hầu hết các ứng dụng WebRTC để hoạt động, cần có một máy chủ để chuyển tiếp lưu lượng giữa các thiết bị ngang hàng, vì một ổ cắm trực tiếp thường không thể thực hiện được giữa các máy khách (trừ khi chúng nằm trên cùng một mạng cục bộ). Cách phổ biến để giải quyết vấn đề này là sử dụng máy chủ TURN. Thuật ngữ này là viết tắt của Traversal bằng Relay NAT và nó là một giao thức để chuyển tiếp lưu lượng mạng.



Hình 3.13. Gọi video giữa 2 tài khoản

KẾT LUẬN

1. Những kết quả đạt được

Sau khi hoàn thành đề tài này em đã học được rất nhiều thứ, đọc và tìm hiểu thêm rất nhiều kiến thức từ đề tài này.

Nắm vững hơn kiến thức về Javascript và thư viện mạnh mẽ như Jquery. Xử lí bảo mật và xác thực người dùng với thư viện passport. Kết nối và cấu hình đăng nhập với nhiều hình thức.

Năm bắt giao thức truyền tải dữ liệu, xử lí real – time và kết nối thông qua Socket.IO cũng như việc lưu cuộc gọi với Turn Server.

Tìm hiểu thêm được một số cách thức render như ( server-side rendering or client-side rendering ).

Xây dựng thành công website chat xử lí real – time, có đầy đủ tính năng cần thiết để trao đổi giữa người dùng.

2. Những tồn tại và hạn chế

Do thời gian có hạn nên việc đồ án hoàn thành có phần hơi gấp gáp, rất nhiều tính năng chưa được phát triển thêm, code xử lý nhiều chỗ còn phức tạp chưa được tối ưu.

Chưa thể deploy lên vì gặp một số trục trặc chưa xử lý được nên phạm vi đồ án chỉ chạy trên localhost.

3. Hướng phát triển đề tài

Bổ sung thêm các chức năng thêm hoặc loại bỏ thành viên khi nhóm chat đã được tạo. Phát triển tính năng gọi nhóm cải thiện hiệu suất của website.

TÀI LIỆU THAM KHẢO

[1] <https://peerjs.com/>

[2] <https://stackoverflow.com/>

[3] <https://codepen.io/>

[4] <https://techtalk.vn/>

[5] <https://developers.facebook.com/>

[6] <https://developer.mozilla.org/>

[7] <https://www.youtube.com/>

[8] <https://viblo.asia/>

[9] <https://www.w3schools.com/>

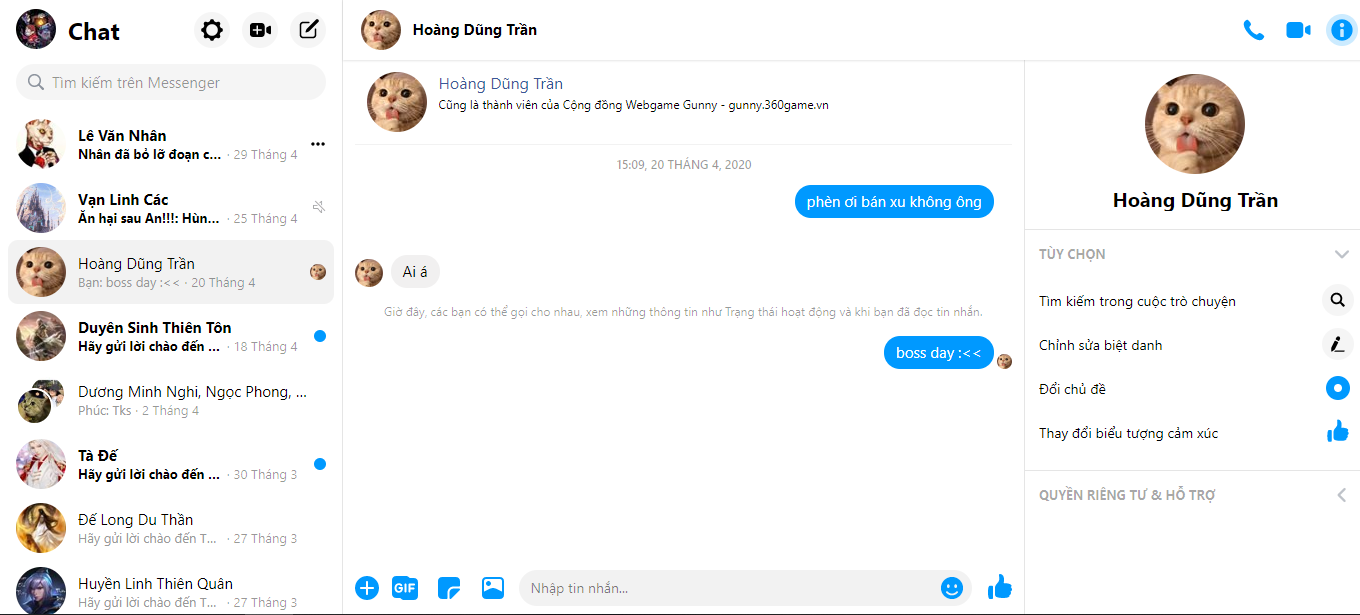
[10] <https://webrtc.org/>

[11] <https://xirsys.com/>

[12] <https://mongoosejs.com/>

[13] <https://expressjs.com/>

[14] <https://www.mongodb.com/>



Hình 3.14. Hình ảnh giao diện tham khảo từ facebook



Hình 3.15. Hình thông báo tham khảo từ facebook