

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TPHCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 12 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

# --------------------



**ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**THIẾT KẾ VÀ XÂY DỰNG WEBSITE CHAT ONLINE**

**Giáo viên hướng dẫn: Hoàng Văn Dũng**

**Nhóm sinh viên thực hiện: Nhóm 2**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ và tên** | **Mức độ đóng góp** |
| **20142498** | **Nguyễn Đình Hiếu** | **100%** |
| **20110505** | **Trần Tấn Khoa** | **100%** |
| **20161385** | **Huỳnh Bảo Toàn** | **100%** |

**TPHCM, tháng 10 năm 2022**

**LỜI CẢM ƠN**

Để hoàn thành tốt đề tài và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, thầy Hoàng Văn Dũng, người đã trực tiếp hỗ trợ chúng em trong suốt quá trình làm đề tài. Chúng em cảm thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho chúng em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tạo Chất Lượng Cao nói chung và ngành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng với đó, chúng em xin được gửi cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của mình.

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những kiến thức còn hạn chế cùng nhiều hạn chế khác về mặt kĩ thuật và kinh nghiệm trong việc thực hiện một dự án web. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báu của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy, quý cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

**MỤC LỤC**

[MỤC LỤC HÌNH ẢNH 2](#_Toc29687)

[CHƯƠNG 1. MỞ ĐẦU 5](#_Toc30258)

[1.1. Phát biểu về bài toán: 5](#_Toc26380)

[1.2. Mục đích, yêu cầu thực hiện: 5](#_Toc5264)

[1.3. Đối tượng và phạm vi nghiên cứu: 6](#_Toc6077)

[1.3.1. Đối tượng nghiên cứu: 6](#_Toc1850)

[1.3.2. Phạm vi nghiên cứu: 6](#_Toc14629)

[1.4. Giao diện dự kiến: 6](#_Toc8169)

[1.5. Công cụ và môi trường: 6](#_Toc15615)

[CHƯƠNG 2. PHÂN CÔNG CÔNG VIỆC 7](#_Toc23756)

[CHƯƠNG 3. PHÂN TÍCH, THIẾT KẾ GIẢI PHÁP 8](#_Toc5806)

[3.1. Cơ sở lý thuyết: 8](#_Toc25051)

[3.1.1. HTML, CSS: 8](#_Toc1993)

[3.1.2. REACTJS: 9](#_Toc27088)

[3.1.3. Firebase: 1](#_Toc18870)0

[3.2. Phân tích giao diện: 1](#_Toc21442)1

[3.3. Phân tích cơ sở dữ liệu: 1](#_Toc17746)1

[3.3.1. Tổng quan các bảng trong CSDL: 1](#_Toc23222)1

[3.3.2. Mô tả các bảng: 1](#_Toc20674)2

[3.3.3. Mô tả các Field trong bảng: 1](#_Toc11941)2

[3.4. Cấu trúc: 1](#_Toc27868)4

[3.5. Thiết kế giao diện: 2](#_Toc8901)7

[CHƯƠNG 4. CÀI ĐẶT VÀ KIỂM THỬ 3](#_Toc1740)0

[4.1. Thiết kế giao diện:](#_Toc800) 30

[4.1.1. Trang chat:](#_Toc10204) 30

[4.1.2. Đăng nhập, đăng xuất:](#_Toc29616) 33

[4.1.3. Đăng ký:](#_Toc3211) 35

[4.1.4. Tìm kiếm bạn bè:](#_Toc3489) 36

[4.1.5. Thêm bạn bè:](#_Toc9563) 37

[4.1.6. Thêm, xóa group:](#_Toc21355) 38

[4.1.7. Thông tin người dùng:](#_Toc22261) 40

[4.2. Hướng dẫn chạy:](#_Toc6872) 41

[CHƯƠNG 5. KẾT LUẬN](#_Toc15516) 43

[5.1. Kết quả đạt được:](#_Toc17468) 43

[5.2. Định hướng phát triển:](#_Toc1795) 43

[5.3. Ưu điểm, hạn chế:](#_Toc4200) 43

[5.3.1. Ưu điểm:](#_Toc8959) 44

[5.3.2. Hạn chế:](#_Toc13077) 44

[TÀI LIỆU THAM KHẢO](#_Toc28340) 45

MỤC LỤC HÌNH ẢNH

[Hình 1: Database của Webchat 12](#_Toc120951552)

[Hình 2: Trang đăng nhập 27](#_Toc120951553)

[Hình 3: Trang home 27](#_Toc120951554)

[Hình 4: Trang đăng kí tài khoản 27](#_Toc120951555)

[Hình 5: Trang chat 28](#_Toc120951556)

[Hình 6: Trang danh sách bạn bè 29](#_Toc120951557)

[Hình 7: Trang danh sách nhóm 29](#_Toc120951558)

[Hình 8: Trang chat sau khi load toàn bộ danh sách tin nhắn và list tin nhắn 30](#_Toc120951559)

[Hình 9:Form mời bạn vào group 31](#_Toc120951560)

[Hình 10: Thông báo lỗi khi mời thành viên đã có trong nhóm 31](#_Toc120951561)

[Hình 11:Tìm kiếm bạn bè để mời chat 32](#_Toc120951562)

[Hình 12: Form đổi tên phòng và mô tả 33](#_Toc120951563)

[Hình 13: Login thông qua google 34](#_Toc120951564)

[Hình 14: Thể hiện hồ sơ, cài đặt, đăng xuất 34](#_Toc120951565)

[Hình 15: Form đăng nhập vào GITCHAT 35](#_Toc120951566)

[Hình 16: Đăng ký tài khoản. 36](#_Toc120951567)

[Hình 17: Danh sách bạn bè và tìm kiếm thành viên để mời vào nhóm 37](#_Toc120951568)

[Hình 18: Form tìm kiếm bạn bè theo số điện thoại 38](#_Toc120951569)

[Hình 19: Thể hiện lời mời kết bạn 38](#_Toc120951570)

[Hình 20: Thêm phòng và hiển thị danh sách phòng 40](#_Toc120951571)

[Hình 21: Form thông báo xóa phòng 40](#_Toc120951572)

[Hình 22: Form thể hiện thông tin cá nhân 41](#_Toc120951573)

MỤC LỤC BẢNG BIỂU

[Bảng 1: Bảng phân công nhiệm vụ 7](#_Toc120952731)

[Bảng 2: Bảng mô tả các trang 11](#_Toc120952732)

[Bảng 3: Mô tả các bảng sử dụng trong GitChat 12](#_Toc120952733)

[Bảng 4: Mô tả các Field trong bảng user 13](#_Toc120952734)

[Bảng 5: Mô tả các Field trong bảng addfriend 13](#_Toc120952735)

[Bảng 6: Mô tả các Field trong bảng rooms 14](#_Toc120952736)

[Bảng 7: Mô tả các Field trong bảng messages 14](#_Toc120952737)

[Bảng 8: Mô tả các Field trong bảng friend 15](#_Toc120952738)

[Bảng 9: Bảng cấu trúc của hệ thống GitChat 27](#_Toc120952739)

[Bảng 10: Bảng thiết kế giao diện 29](#_Toc120952740)

**CHƯƠNG 1. MỞ ĐẦU**

## Phát biểu về bài toán:

Trong thời đại 4.0 ngày nay, chúng ta đang chứng kiến sự phát triển vượt bậc của internet trên toàn thế giới. Cùng với sự phát triển đó biến chúng thành một chiếc chìa khóa giúp chúng ta giải quyết được nhiều nhu cầu trong cuộc sống. Một trong những nhu cầu đó là trao đổi thông tin và liên lạc một cách nhanh chóng tiện lợi. Điều này đã thúc đẩy sự phát triển của rất nhiều phần mềm hỗ trợ cho việc đó. Điển hình cho các phần mềm này là các web chat, app chat chẳng hạn như facebook, instagram, zalo,... Nhưng có một vấn đề đặt ra là các ứng dụng trên phải cài đặt trên các thiết bị của chúng ta trong khi các tính năng của các ứng dụng này đều có thể làm được trên web. Web đem lại cho chúng ta sự tiện lợi, chỉ với một ứng dụng duy nhất có kết nối mạng chỉ cần nhập đường link của các trang web ta có thể truy cập và sử dụng những tính năng mà trang web đó đem lại. Điều đó mang lại sự thuận tiện cho người dùng và đồng thời là sự tiết kiệm tài nguyên máy của ta một cách đáng kể. Qua những vấn đề trên nên nhóm em quyết định thiết kế và xấy dựng một website cho phép chúng ta nhắn tin trực tiếp trên đó với đề tài “Xây dựng và thiết kế website chat online”. Với mục tiêu đề ra website sẽ có những phần tương tự như những ứng dụng đã đề ra ở trên, cùng với đó là áp dụng những công nghệ mới giúp mục đích hoàn thiện hơn các chức năng và giao diện của web chat online.

## Mục đích, yêu cầu thực hiện:

Đề tài **Thiết kế và xây dựng website chat online** đặt ra các vấn đề trọng tâm về mục tiêu mà sinh viên nghiên cứu cần thực hiện được bao gồm:

Vấn đề đầu tiên được đưa ra về yếu tố lý thuyết, sinh viên áp dựng những kiến thức, nghiên cứu hiểu được tổng quan và khả năng ứng dụng vào trong bài làm. Song song với việc áp dụng công nghệ được lựa chọn để sử dụng, nhóm sinh viên cầm tìm hiểu thêm về cách xây dựng giao diện đẹp, thân thiện với người dùng.

Vấn đề thứ hai là nâng cao khả năng làm việc nhóm, tìm hiểu và thảo luận về đề tài, các vấn đề đưa ra từ dó tìm ra câu trả lời. Xây dựng thái tích cực trong việc học, làm đồ án.

## Đối tượng và phạm vi nghiên cứu:

### Đối tượng nghiên cứu:

Xây dựng trang web chat bằng ReacJS. Kết nối với cơ sở dữ liệu bằng FireBase.

### Phạm vi nghiên cứu:

Thiết kế và xây dựng website chat online.

* 1. **Giao diện dự kiến:**

Để thích hợp với một ứng dụng chat thì nhóm dự kiến sẽ xây dựng các trang bao gồm trang chủ là nơi giới thiệu thông tin website, trang login để đăng nhập, trang đăng ký để tạo tài khoản, trang chat để thực hiện các chức năng liên quan đến chat. Trong trang chat này gồm 3 thành phần: chat thể hiện các đoạn chat, friends thể hiện danh sách bạn bè và lời mời kết bạn, groups thể thiện danh sách group tồn tại có thể thêm và xóa.

* 1. **Công cụ và môi trường:**

- IDE: Visual Studio Code.

- Ngôn ngữ: REACTJS, HTML, CSS cho font-end, firebase cho backend.

# CHƯƠNG 2. PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Danh sách thành viên** | **Công việc thực hiện được** | **Tỉ lệ hoàn thành** |
| 1 | Nguyễn Đình Hiếu | Xây dựng giao diện, viết báo cáo, xây dựng các chức năng trang chat, đăng nhập, đăng ký, kết nối firebase. | 100% |
| 2 | Trần Tấn Khoa | Xây dựng giao diện, xây dựng các chức năng trang friend, group, kết nối firebase, viết báo cáo. | 100% |
| 3 | Huỳnh Bảo Toàn | Xây dựng giao diện, xây dựng các chức năng trang friend, group, kết nối firebase, viết báo cáo. | 100% |

Bảng 1: Bảng phân công nhiệm vụ

# 

# CHƯƠNG 3. PHÂN TÍCH, THIẾT KẾ GIẢI PHÁP

* 1. **Cơ sở lý thuyết:**
     1. **HTML, CSS:**

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu tiêu chuẩn để tạo các trang Web. Mô tả cấu trúc của một trang web, bao gồm một loạt các phần tử và các phần tử HTML cho trình duyệt biết cách hiển thị nội dung. CSS (Cascading Style Sheets) mô tả cách các phần tử HTML hiển thị trên màn hình laptop, điện thoại,… HTML và CSS độc lập với nhau, CSS không nên viết bên trong HTML và ngược lại. HTML đại diện cho nội dung và CSS thể hiện sự xuất hiện của nội dung đó.

* Ưu điểm:

+ Được sử dụng rộng rãi với rất nhiều nguồn tài nguyên mở và hoàn toàn miễn phí. Được hỗ trợ cùng một cộng đồng sử dụng vô cùng lớn đằng sau nó.

+ Hoạt động được trên hầu hết mọi trình duyệt hiện nay.

+ Tiếp cân và học HTML, CSS khá đơn giản.

+ Chuẩn của web được vận hành bởi World Wide Web Consortium (W3C).

+ Dễ dàng tích hợp với các ngôn ngữ backend như Node.js, java,…

* Nhược điểm:

+ Ngôn ngữ này chỉ được áp dụng chủ yếu cho trang web còn đối với các tính năng động của trang web thì cần phải kết hợp JavaScript. Bên cạnh đó, muốn hiện những dữ liệu trong cơ sở dữ liệu lên màn hình thì cần phải kết hợp với backend.

+ Phải tạo các trang web riêng lẻ ngay cả khi các phần tử trong trang web đó giống nhau.

+ Phải có khả năng thiết kế tốt để trang web trở nên đẹp hơn.

* + 1. **REACTJS:**

REACTJS là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để để xây dựng các tương tác với các thành phần trên website. Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được trên tầng Server mà còn ở dưới Client nữa.

Ngoài việc hỗ trợ xây dựng giao diện nhanh, hạn chế lỗi trong quá trình code, cải thiện hiệu website thì chúng cũng đem lại những ưu, nhược điểm khác nhau cho người sử dụng. Sau đây ta tìm hiểu về một vài ưu nhược của chúng.

* Ưu điểm:

+ Hỗ trợ chúng ta rất nhiều trong việc chia cấu trúc file để có thể code dễ dàng hơn. Cung cấp rất nhiều loại hook thay thế cho JavaScript một cách tiện lợi nhanh chóng.

+ Phù hợp với đa dạng thể loại website: ReactJS khiến cho việc khởi tạo website dễ dàng hơn bởi vì ta không cần phải code nhiều như khi tạo trang web thuần chỉ dùng JavaScript, HTML.

+Tái sử dụng các Component: nếu xây dựng một component tốt, ta có thể sử dụng lại chúng vào nhiều trang web hoặc layout khác nhau thay cho việc làm lại từ đầu.

+ Ngoài ra con được sử dụng cho cả Mobile application.

+ Debug nhanh chóng và dễ dàng: Facebook đã phát hành 1 Chrome extension hỗ trợ việc debug khi gặp lỗi.Phần nào giúp ta tăng hiệu xuất khi xây dựng trang web.

* Nhược điểm:

+ Tích hợp Reactjs vào các framework MVC truyền thống nên yêu cầu phải cấu hình lại toàn bộ file.

+ React khá nặng so với một vài framework khác (Khoảng 35kb so với 39kb của Angular).

+ Khó tiếp cận cho người mới học Web.

* + 1. **Firebase:**

Firebase là Backend-as-a-Service (Baas) nền tảng chứa các API đơn giản và mạnh mẽ giúp phát triển trang web và ứng dụng di động mà không cần Server hay Backend. Firebase hay Google firebase được phát triển bởi Google chức năng như một dịch vụ cơ sở dữ liệu có hệ thống máy chủ mạnh mẽ, chạy trên nền Cloud – Điện toán đám mây.

Cũng giống như ReactJS, HTML, CSS thì firebase cũng có những ưu và nhược điểm riêng của chúng.

* Ưu điểm:

+ Cho phép chúng ta tạo tài khoản sử dụng một cách dễ dàng.

+ Cung cấp nhiều các xác thực khác nhau khi login vào một trang web như google, facebook, số điện thoại,…

+ Có Firestore database cho phép chúng ta lưu trữ dữ liệu một cách dễ dàng nhanh chóng thay thế cho phía backend rất hữu ích.

+ Cho phép chúng ta sử dụng miễn phí.

+ Có docs hướng dẫn sử dụng một cách chi tiết dễ dàng.

* Nhược điểm:

+ Tuy cho phép chúng ta sử dụng miễn phí nhưng giới hạn sử dụng dữ liệu chẳng hạn như cho phép chúng ta lưu trữ tối đa 1 GiB, cho phép đọc 50k dữ liệu mỗi ngày, ghi 20k dữ liệu mỗi ngày, 20k delete mỗi ngày và muốn nhiều hơn thì phải trả thêm phí. Tuy nhiên chi phí không quá mắc.

+ Khó tiếp cận đối với những người mới học web.

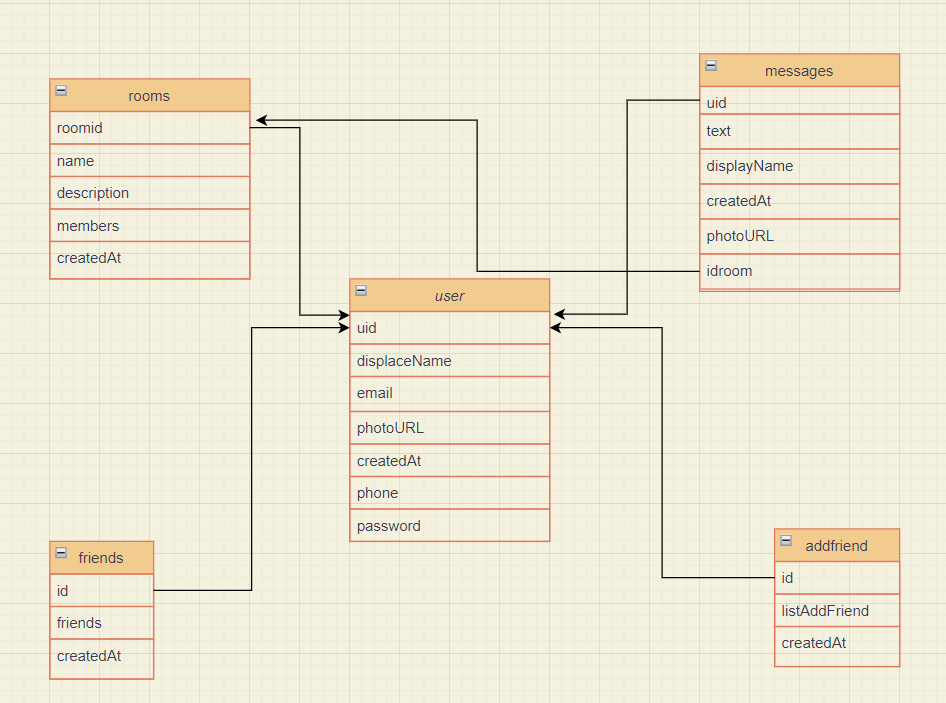
* 1. **Phân tích giao diện:**

|  |  |  |
| --- | --- | --- |
| STT | Giao diện | Chức năng |
| 1 | Trang Login | Nơi người dùng đăng nhập bằng tài khoản của mình thông qua tài khoản đã đăng ký, thông qua facebook hoặc gmail |
| 2 | Trang đăng kí tài khoản | Nơi người dùng có thể đăng kí tài khoản bằng số điện thoại hoặc có thể đăng nhập bằng gmail hoặc facebook |
| 3 | Trang chat | Nơi hiển thị danh sách tin nhắn và chi tiết của tin nhắn được chọn, cho phép kết bạn, tạo các cuộc trò chuyện với bạn bè hoặc các hội nhóm. Bên cạnh đó còn có thể phát triển một số tính năng khác. |
| 4 | Trang Home | Giới thiệu về trang web, cho phép điều hướng sang trang đăng nhập và đăng ký. |

Bảng 2: Bảng mô tả các trang

* 1. **Phân tích cơ sở dữ liệu:**
     1. **Tổng quan các bảng trong CSDL:**

Trang web có cơ sở dữ liệu đơn giản để lưu trữ thông tin các user, các room, các message và list friend của user đó. Sau đây chúng ta đi tìm hiểu kỹ hơn về từng bảng.



Hình 1: Database của Webchat

* + 1. **Mô tả các bảng:**

|  |  |  |
| --- | --- | --- |
| STT | Tên bảng | Mục đích |
| 1 | User | Thông tin về tài khoản đăng nhập |
| 2 | Rooms | Thông tin về phòng chat |
| 3 | Message | Thông tin về tin nhắn của User |
| 4 | addfriend | Chứa thông tin về danh sách bạn bè gửi lời mời kết bạn |
| 5 | friend | Chứa thông tin về danh sách bạn bè |

Bảng 3: Mô tả các bảng sử dụng trong GitChat

* + 1. **Mô tả các Field trong bảng:**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | uid | string | Mã số người dùng |
| 2 | displayName | string | Tên hiển thị của người dùng |
| 3 | email | string | Địa chỉ Email người dùng |
| 4 | photoURL | string | Link ảnh của người dùng |
| 5 | createdAt | timestamp | Ngày tạo người dùng |
| 6 | phone | string | Số điện thoại tạo tài khoản |
| 7 | password | string | Mật khẩu để đăng nhập |

Bảng 4: Mô tả các Field trong bảng user

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | id | string | Id user |
| 2 | listAddFriend | array | Id của các friend gửi lời mời kết bạn |
| 3 | createdAt | timestamp | Thời gian tạo bảng |

Bảng 5: Mô tả các Field trong bảng addfriend

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | name | string | Tên phòng |
| 2 | description | string | Mô tả phòng |
| 3 | members | array | Chứa Id của user người chat |
| 4 | createdAt | timestamp | Ngày, thời gian tạo phòng |
| 5 | roomid | string | Id của phòng chat |

Bảng 6: Mô tả các Field trong bảng rooms

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | uid | string | ID sở hữu đoạn chat |
| 2 | text | string | Nội dung chat |
| 3 | displayName | string | Tên hiển thị |
| 4 | createdAt | timestamp | Ngày, thời gian tạo |
| 5 | photoURL | string | Ảnh đại diện của user |
| 6 | idroom | string | Id phòng hiển thị đoạn chat |

Bảng 7: Mô tả các Field trong bảng messages

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | id | string | Id user |
| 2 | friends | array | Id các friend |
| 3 | createdAt | timestamp | Thời gian tạo bảng |

Bảng 8: Mô tả các Field trong bảng friend

* 1. **Cấu trúc:**

|  |  |
| --- | --- |
| **PAGES** | |
| Chat | Chat()  const [tab,setTab] = useState(0)  const chat = useRef()  const friend = useRef()  const group = useRef()  const setting = useRef()  const componentSetting = useRef()  Export default Chat  tonggle()  => render ra giao diện cho trang chat |
| Home | Home()  => Xây dựng trang home  export default Home |
| Login | const handleLogin = async (provider) => {  const { additionalUserInfo, user } = await auth.signInWithPopup(provider);  if (additionalUserInfo?.isNewUser) {  addDocument('users', {  displayName: user.displayName,  email: user.email,  photoURL: user.photoURL,  uid: user.uid,  providerId: additionalUserInfo.providerId,  });    addDocument('friend',{  id: user.uid,  friends: [],  });  addDocument('addfriend',{  id: user.uid,  listaddfriend: [],  });  }  };  => login thông qua facebook và google, nếu login thành công và là người mới thì thêm vào trong cơ sở dữ liệu.  const handleBtnLogin = () => {  if(phoneInput === undefined || passwordInput === undefined)  {  alert("Hãy nhập đầy đủ số điện thoại và mật khẩu!!!")  }  else{  let loginUser = db.collection('users');  loginUser.where('phone', '==', phoneInput);  loginUser = loginUser.where('password', '==', passwordInput);  getDocs(loginUser)  .then((snapshot) => {  const {displayName,photoURL,uid, phone}=snapshot.docs[0].data()  const userLogin=[displayName,photoURL,uid, phone]  const jsonUser=JSON.stringify(userLogin)  localStorage.setItem("user", jsonUser);  setUser(snapshot.docs[0]?.data())  })  .catch((error) => {  alert("Số điện thoại hoặc mật khẩu không chính xác!!!")  console.log(error)  });  }  };  => Login thông qua số điện thoại: kiểm tra nếu trùng số điện thoại và mật khẩu thì cho vào còn các trường hợp khác in ra thông báo. |
| Signup | const onClose = () => {  setPreview(null);  };  const onCrop = (view) => {  setPreview(view);  };  const createRandom = () => {  var randomstring = '';  var characters = 'QWERTYUIOPASDFGHJKLZXCVBNM123456789qwertyuiopasdfghjklzxcvbnm';  for (var i, i = 0; i < 28; i++) {  randomstring += characters.charAt(Math.floor(Math.random() \* 28));  }  return randomstring;  };  => Tạo ra chuỗi random  const handlesubmit = () => {  if((document.getElementsByClassName(`${cx('name')}`)[0].value==='' || document.getElementsByClassName(`${cx('phone')}`)[0].value==='' || document.getElementsByClassName(`${cx('pass')}`)[0].value==='' || document.getElementsByClassName(`${cx('repass')}`)[0].value===''))  {  alert("Hãy điền đầy đủ các trường")  }  else  {  if (document.getElementsByClassName(`${cx('pass')}`)[0].value === document.getElementsByClassName(`${cx('repass')}`)[0].value ) {  const uid=createRandom();  addDocument('users', {  displayName: document.getElementsByClassName(`${cx('name')}`)[0].value,  phone: document.getElementsByClassName(`${cx('phone')}`)[0].value,  photoURL: preview,  uid: uid,  providerId: 'phone',  password: document.getElementsByClassName(`${cx('pass')}`)[0].value,  });  addDocument('friend',{  id: uid,  friends: [],  });  addDocument('addfriend',{  id: uid,  listaddfriend: [],  });  alert('Đăng ký thành công');  navigate("/login");  } else {  alert('Mật khẩu không trùng khớp');  }  }  };  => Kiểm tra nếu không điền đủ các ô và không trùng pass thì in ra thông báo còn nếu đăng ký thành thông thì chuyển sang trang login đồng thời thêm thông tin vào trong cơ sở dữ liệu.  //  const handleLogin = async (provider) => {  const { additionalUserInfo, user } = await auth.signInWithPopup(provider);  if (additionalUserInfo?.isNewUser) {  addDocument('users', {  displayName: user.displayName,  email: user.email,  photoURL: user.photoURL,  uid: user.uid,  providerId: additionalUserInfo.providerId,  });  addDocument('friend',{  id: user.uid,  friends: [],  });  addDocument('addfriend',{  id: user.uid,  listaddfriend: [],  });  }  };  => xử lý cho việc login thông qua google và facebook khi đứng tại trang signup. |
| **LAYOUT COMPONENTS** | |
| Button | Button()  Export default Button |
| Service | * Xử lý việc add thành viên nhóm   export const addGroup = (user,idgroup) =>  {  let collectionUser = db.collection('rooms');  getDocs(collectionUser)  .then(res => {  const movss = res.docs.find(doc => doc.id === idgroup);    const dataa = movss.data().members;    if(dataa.includes(user)) {               alert("User đã có trong nhóm")     }     else  {  const dataaa = [...dataa,user];  db.collection('rooms').doc(movss.id).update({     createdAt: movss.data().createdAt,     description: movss.data().description,     members: dataaa,     name: movss.data().name,    }   )   alert('Thêm thành viên thành công')  }  }).catch(err => err)  } |
| Friends | Friends()  Export default Friends   * Hàm chấp nhận kết bạn   const acpfriend = (useraddfr) => {  updateFriend(uid,useraddfr.uid)  }   * Hàm xóa bạn bè   const deleteaddfriend = (useraddfr) => {  updateListAddFriend(uid, useraddfr.uid);  }  - Khi nhấn vào tin nhắn với bạn bè thì hiển thị những chức năng giống với messchat. Tuy nhiên sẽ xóa đi một số tính năng chẳng hạn như thêm bạn bè vào cuộc trò chuyện,… |
| Group | Group()  Export default Group   * Hàm hiển thị Form thêm Group, tắt hiển thị các nhóm   showModal()   * Xử lý thêm nhóm mới vào Database   handleOk()   * Hàm hủy việc thêm nhóm   handleCancel()   * Hàm hiển thị Form xóa Group   DelRoom()   * Hàm hủy hiển thị việc xóa Group   handlePerformCancel()   * Hàm xử lý việc xóa Group ra khỏi Database   handlePerformOK()   * Form xóa phòng    <Form form={form} layout="vertical" className="perform-del" style={{ display: 'none' }}></Form>   * Form thêm phòng   <Form form={form} layout="vertical" className="test" style={{ display: 'none' }}></Form> |
| Header | Header ()  Export default Header |
| HeaderChat | const choose = useRef();      const handleInfor = () => {          if (choose.current.style.display === 'inline-block') {              choose.current.style.display = 'none';          } else {              choose.current.style.display = 'inline-block';          }      };      const {          user: { displayName, photoURL },      } = useContext(AuthContext);      const handlelogout=()=>{          auth.signOut();      }  HeaderChat ()  Export default HeaderChat |
| messChat | * Lấy ra thông tin của Room   Const selectedRoom = React.useMemo(() => {          return rooms.find((room) => {              if (room.id === roomid) {                  return room;              }          });      }, [rooms, roomid]);   * Hàm xử lý gửi tin nhắn và lưu thông tin tin nhắn lên Database   handlesubmit()     * Hàm xử lý để hiện ra Form mời thành viên   showInvite()   * Hàm hủy thao tác Rename    handleCancelRename()   * Hàm Update tên Group     handleCancelOK()   * Hàm xử lý việc Update tên mô tả   handleOkDes ()   * Hàm hủy thao tác Rename mô tả    handleCancelDes()   * Hiển thị Avatar các thành viên đang có trong nhóm   <Avatar.Group size="small" maxCount={2}>     {  members.map((member, index) => (   < StatusAvatar  status="green"      size={30}      name={member.displayName}       githubHandle={member.photoURL}   />  ))}  </Avatar.Group>  const handlesubmit = () => {  if(chat!=='')  {  addDocument('messages', {  text: chat,  uid,  photoURL,  displayName,  idroom: roomid,  });  document.getElementsByClassName(`${cx('input-chat')}`)[0].value = '';  setChat('')  }  };  const handleKeyDown = (event) => {  if (event.key === 'Enter') {  if(chat!=='')  {  addDocument('messages', {  text: chat,  uid,  photoURL,  displayName,  idroom: roomid,  });  document.getElementsByClassName(`${cx('input-chat')}`)[0].value = '';  setChat('')  }  }  }  => Kiểm tra nếu như nhập rỗng thì không cho gửi, ngược lại gửi thành công.  let listchat = db.collection('messages');  //listchat = listchat.orderBy('createdAt', 'asc');  listchat = listchat.where('idroom', '==', roomid);  //  react.useEffect(() => {  onSnapshot(listchat, (snapshot) => {  const chat = snapshot.docs.map((doc) => ({  ...doc.data(),  id: doc.id,  }));  setDocuments(chat);  });  }, [roomid]);  documents.sort((a,b)=>a.createdAt-b.createdAt)  => Lấy toàn bộ tin nhắn và hiển thị ra màn hình. |
| ModalInviterMembers | * Xử lý thao tác mời thành viên   handleInviteMember()   * Tắt hiện thị Form mời thành viên     handleOpenInviteModal() |
| settingChat | * Hiểm thị thông tin phòng đang đươc chọn      const selectedRoom = react.useMemo(() => {          return rooms.find((room) => {              if (room.id === roomid) {                  return room;              }          });      }, [rooms, roomid]);  settingChat ()  Export default settingChat |
| ShowChat | const {          user: { photoURL, uid },      } = react.useContext(AuthContext);      const roomsCondition = react.useMemo(() => {          return {              fieldName: 'members',              operator: 'array-contains',    compareValue: uid,          };      }, [uid]);  ShowChat ()  Export default ShowChat |

Bảng 9: Bảng cấu trúc của hệ thống GitChat

* 1. **Thiết kế giao diện:**

|  |  |  |
| --- | --- | --- |
| STT | Trang | Giải thích thiết kế |
| 1 | Hình 2: Trang đăng nhập | - 2 textbox để nhập tên đăng nhập và mật khẩu  - Check box để yêu cầu ghi nhớ mật khẩu.  - Button Đăng nhập để gọi Controller Login.  - 2 button dùng để đăng nhập bằng gmail và facebook. |
| 2 | Hình 3: Trang home | Trang home thực hiện giới thiệu về trang web và điều hướng đến trang đăng nhập củng như đăng kí tài khoản. |
| 3 | Hình 4: Trang đăng kí tài khoản | - 4 textbox để nhập tên người dùng, số điện thoại, mật khẩu, xác nhận mật khẩu  - Button Đănh ký để gọi Controller đăng ký.  - 2 button dùng để đăng ký bằng gmail và facebook.  - Thẻ đăng nhập để quay lại trang đăng nhập |
| 4 | Hình 5: Trang chat | - Thể hiện thông tin tài khoản đang đăng nhập  - Ta có thể tìm kiếm chức năng thông qua icon kính lúp  - Chứa các icon như danh sách bạn bè, hội nhóm.  - Thể hiện nội dung đoạn chat, các user trong room.  - Các chức năng như tìm kiếm tin nhắn, được phép sửa đổi tên phòng và description của phòng, cho phép mời thêm frienh vào room. |
| 5 | Hình 6: Trang danh sách bạn bè | - Thể hiện danh sách lời mời kết bạn chứa các button xóa, đồng ý.  - Thể hiện danh sách bạn bè.  - Cho phép tìm kiếm bạn bè có trong danh sách và hiển thị tin nhắn đối với bạn bè đó. |
| 6 | Hình 7: Trang danh sách nhóm | - Trang danh sách hội nhóm đã tham gia  - Hiển thị danh sách bạn vè và tìm kiếm bạn bè.  - Thêm xóa các phòng. |

Bảng 10: Bảng thiết kế giao diện

# CHƯƠNG 4. CÀI ĐẶT VÀ KIỂM THỬ

* 1. **Thiết kế giao diện:**

Ngoài việc thiết kế hoàn chỉnh giao diện thì nhóm em đã hoàn thành một số tính năng sau. Ứng với mỗi tính năng sẽ có những trường hợp kiểm tra thích hợp và những trường hợp thường là những trường hợp bị lỗi nhất mà nhóm đã phát hiện và sửa lỗi để có đầu vào phù hợp.

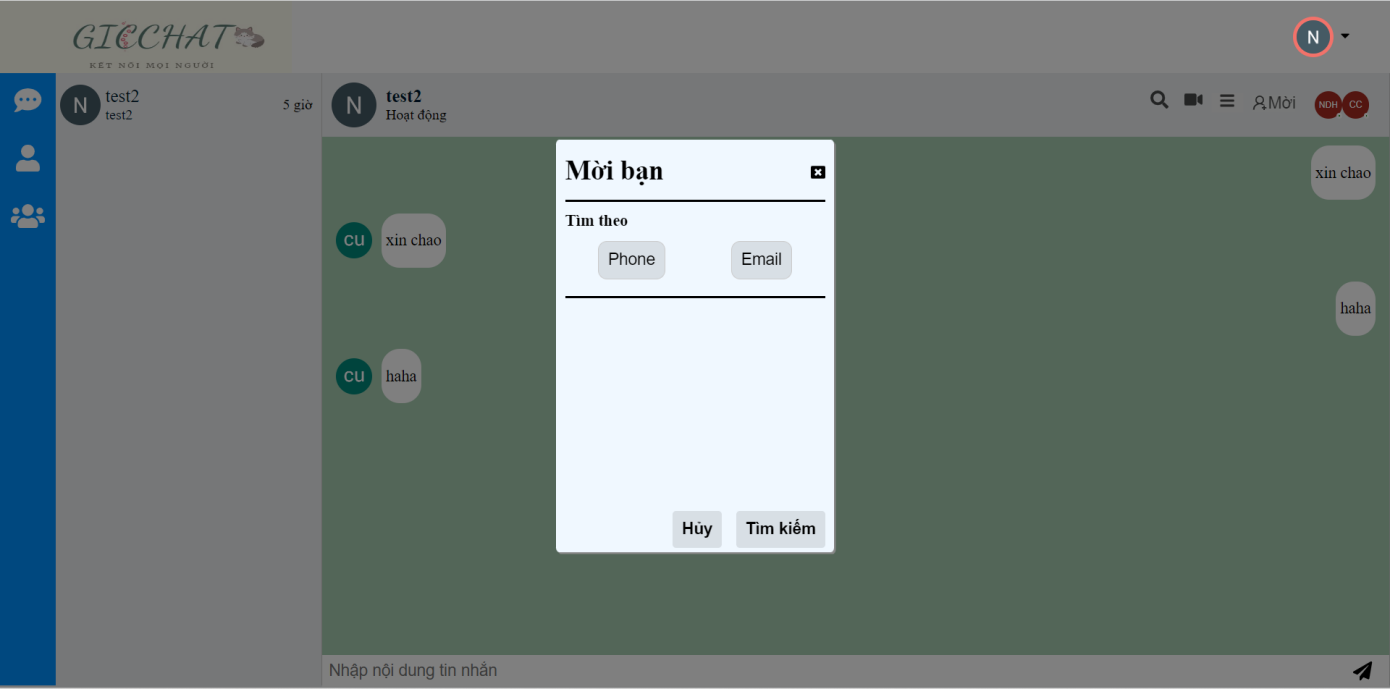
* + 1. **Trang chat:**



Hình 8: Trang chat sau khi load toàn bộ danh sách tin nhắn và list tin nhắn

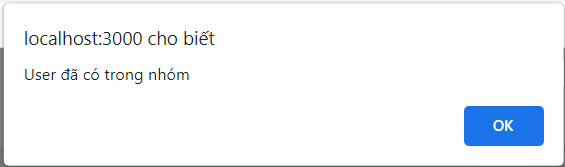
Khi nhập nội dung tin nhắn và nhấn nút gửi hoặc enter thì tin nhắn sẽ hiển thị lên giao diện người dùng theo phía người nhận người gửi. Trong trường hợp tin nhắn được gửi là rỗng thì sẽ không gửi được tin nhắn.

Cho phép thêm người vào cuộc hội thoại bằng cách nhấn vào nút mời. Khi nhấn vào nút mời sẽ hiển thị lên một form cho phép ta nhập email hoặc số điện thoại đại diện cho user đó. Khi nhấn vào tìm kiếm thì sẽ hiện thị lên thông tin của user kèm theo đó là nút mời user vào trong đoạn chat.

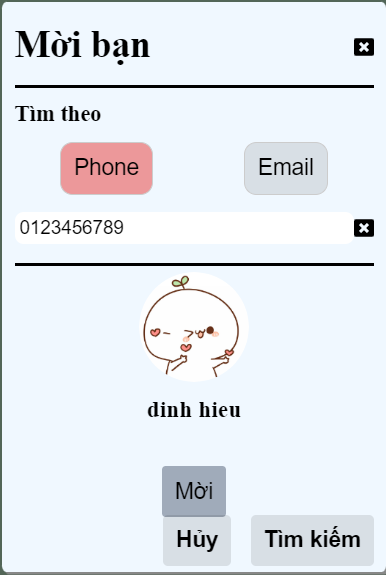


Hình 9:Form mời bạn vào group

Ví dụ như hình bên khi nhập số điện thoại 0123456789 thì hiện lên thông tin user có tên là “Dinh Hieu” và nút mời. Nếu nhập sai hoặc để trống thì sẽ không hiển thị lên gì cả. Nếu user đã tồn tại trong room thì khi nhấn mời sẽ hiển thị lên thông báo “user đã có trong nhóm”.



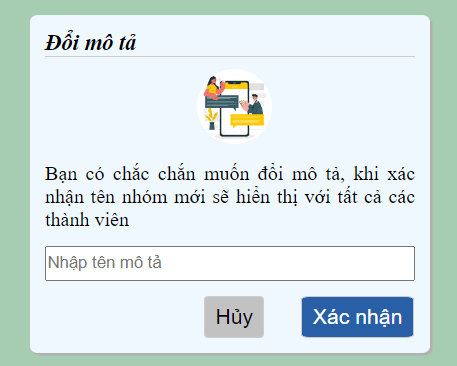
Hình 10: Thông báo lỗi khi mời thành viên đã có trong nhóm



Hình 11:Tìm kiếm bạn bè để mời chat

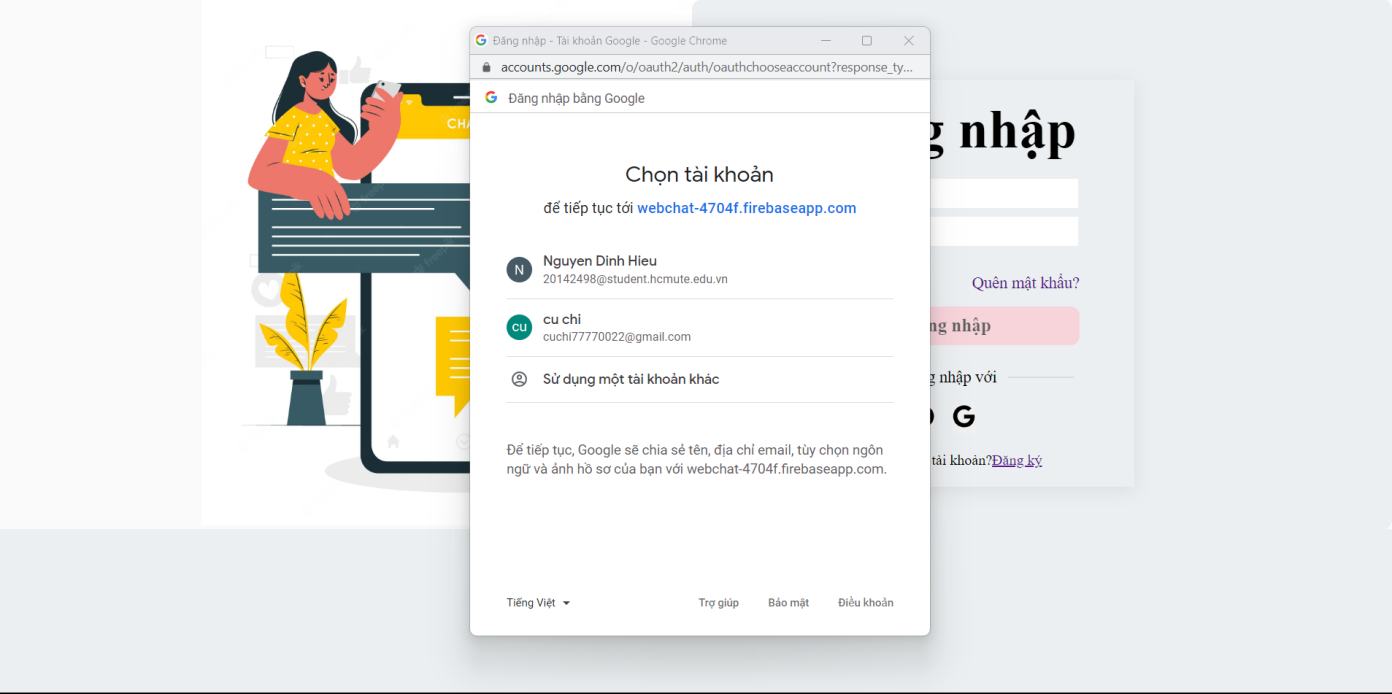
Bên phải của màn hình chính trong hình có biểu tượng cho phép chỉnh sửa thông tin phòng. Khi nhấn vào tên và mô tả sẽ hiển thị lên form như các hình bên dưới cho phép ta nhập vào tên phòng mới và mô tả mới cho phòng. Trong trường hợp tên phòng hoặc mô tả trống nhưng vẩn nhấn vào nhút xác nhận thì sẽ hiển thị lên một thông báo “Hãy nhập tên phòng!!!” hoặc “Hãy nhập mô tả!!!” và không cập nhập lại tên phòng.





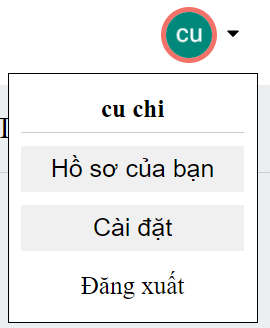
Hình 12: Form đổi tên phòng và mô tả

* + 1. **Đăng nhập, đăng xuất:**



Hình 13: Login thông qua google

Login qua google và sau khi login thành công thì sẽ hiện lên thông tin của user trong phần header chat như hình bên.



Hình 14: Thể hiện hồ sơ, cài đặt, đăng xuất

Bên cạnh đó cũng cho phép người dùng đăng xuất sau khi login. Đối với login thông qua google và facebook firebase sẽ duy trì login cho đến khi ta đăng xuất. Tuy nhiên một bất lợi đó là khi login qua facebook hiện tại chỉ cho phép login thông qua facebook đã được lưu vào trong firebase mà không dùng được cho facebook nào khác.

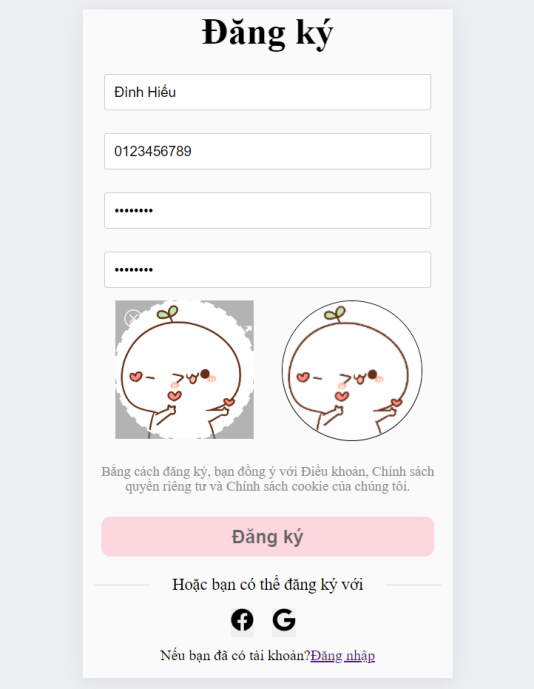
Bên cạnh việc login thông qua facebook và google, ta còn có thể login thông qua số điện thoại mà đã đăng ký tài khoản. Đối với login thông qua số điện thoại, cho phép lưu thông tin đăng nhập để khi lần sau vào không cần đăng nhập lại.

Khi đăng nhập thông qua số điện thoại mật khẩu, khi người dùng nhập đúng thì sẽ dẫn người dùng đến trang chat. Nhưng nếu không nhập gì cả hoặc nhập sai thông tin thì sẽ hiển thị lên thông báo đăng nhập không thành công.



Hình 15: Form đăng nhập vào GITCHAT

* + 1. **Đăng ký:**

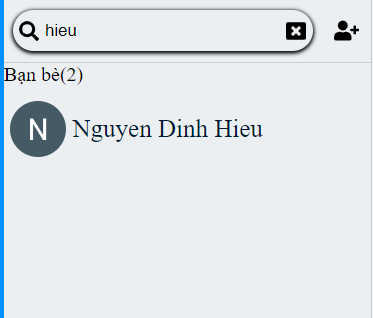
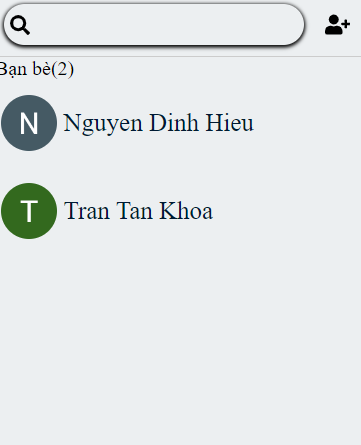


Hình 16:Đăng ký tài khoản.

Khi nhấn đăng ký thì lập tức user sẽ được thêm vào trong cơ sở dữ liệu, và sẽ chuyển trang sang trang login cho phép chúng ta login tài khoản đã tạo. Nếu nhập thiếu trường thì sẽ in ra thông báo đăng ký thất bại, nếu nhập không trùng mật khẩu cũng sẽ thông báo không đăng ký thất bại.

* + 1. **Tìm kiếm bạn bè:**

Khi vào trong trang bạn bè sẽ hiển thị toàn bộ danh sách bạn bè. Bên trên danh sách bạn bè cho phép chúng ta tìm kiếm bạn bè theo tên như hình bên. Khi nhập vào “hieu” thì sẽ cập nhập lại danh sách bạn bè như hình bên. Bên cạnh đó, mỗi lần nhập sẽ có icon hiện bên ô nhập khi nhấn vào sẽ xóa các ký tự trong ô search. Khi nhấn vào từng bạn bè thì sẽ hiển thị lên toàn bộ cuộc trò chuyện với người bạn đó. Đồng thời tại cuộc trò chuyện này vẩn cho phép nhắn tin lẫn nhau, tuy nhiên không thể thêm bạn bè khác vào cuộc trò chuyện được nữa.



Hình 17: Danh sách bạn bè và tìm kiếm thành viên để mời vào nhóm

Nếu lựa chọn vào bạn bè mà chưa từng có cuộc hội thoại thì sẽ hiển thị lên thông báo . Còn nếu có thì hiển thị lên như hình bên.

**

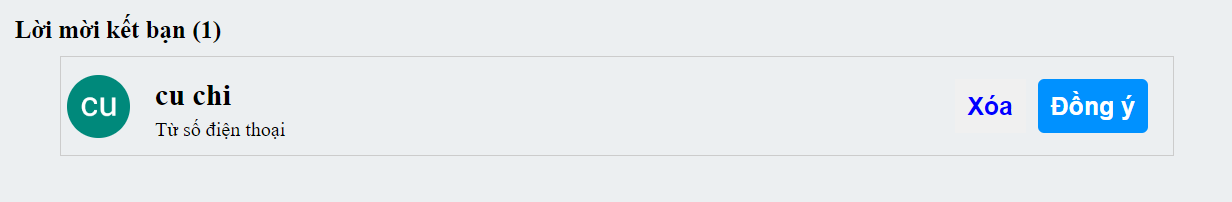
*Hình 18: Khi nhấn vào tin nhắn với bạn có tên là “hieu”*

* + 1. **Thêm bạn bè:**

Bên phải ô tìm kiếm còn có một icon khi nhấn vào icon đó sẽ hiển thị lên form cho phép tìm kiếm bạn bè theo số điện thoại và email như hình bên. Khi nhập vào số điện thoại hay email sẽ hiển thông tin người dùng tương ứng với thông tin nhập vào. Khi nhấn vào nút thêm bạn bè thì bên người dùng tương ứng sẽ hiển thị lên màn hình lời mời kết bạn. Đồng thời khi thêm bạn bè mới sẽ tạo ra phòng chat giữa hai người. Trường hợp nếu đã từng là tồn tại phòng chat giữa hai người thì khi đồng ý kết bạn sẽ không tạo ra phòng mới nữa. Khi đã đồng ý kết bạn ta có thể hủy kết bạn bằng cách tìm kiếm bạn bè theo thông tin liên hệ của bạn qua số điện thoại hoặc email sau đó nhấn vào bạn bè thì lập tức sẽ hủy kết bạn.



Hình 18: Form tìm kiếm bạn bè theo số điện thoại

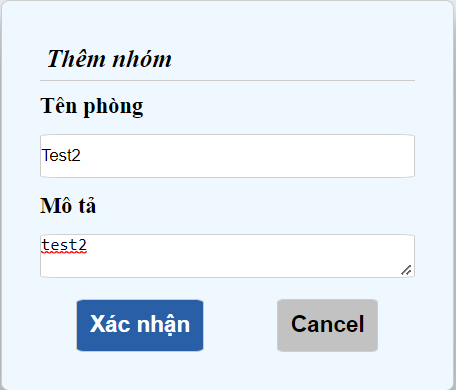


Hình 19: Thể hiện lời mời kết bạn

Như hình trên khi login user tương ứng thì hiển thị lời mời kết bạn cho phép ta đồng ý hoặc xóa lời mời đi. Nếu đồng ý thì lập tức user này sẽ được thêm vào danh sách bạn bè còn nếu xóa đi thì sẽ không tồn lại lời mời nữa.

* + 1. **Thêm, xóa group:**

Khi nhấn vào “Thêm phòng” trên màn hình thì sẽ hiển thị ra form cho phép ta nhập tên phòng và mô tả của phòng. Sau khi nhấn xác nhận sẽ hiển thị toàn bộ các phòng ra màn hình. Trong trường hợp một trong thông tin để trống thì sẽ không thêm được phòng. Đồng thời để có thể thêm được phòng thì người dùng khi tạo phòng bắt buộc phải thêm hai người khác vào cuộc trò chuyện.

.



Hình 20: Thêm phòng và hiển thị danh sách phòng

Khi nhấn vào phòng sau đó nhấn vào nút xóa phòng thì sẽ hiển thị lên một thông báo xác nhận ta sẽ xóa phòng đó. Khi nhấn có thì phòng lập tức sẽ bị xóa và hiển thị lại danh sách các phòng đang có đồng thời thông báo xóa phòng thành công. Và nếu user không phải là chủ phòng thì sẽ không xóa được phòng.



Hình 21: Form thông báo xóa phòng

* + 1. **Thông tin người dùng:**

Khi nhấn vào nút mũi tên chỉ xuống bên phải ảnh đại diện ở góc trên bên phải màn hình sẽ hiển thị lên một form. Trong form đó có thông tin hồ sơ của ta và đăng xuất. Khi nhấn vào thông tin hồ sơ sẽ xuất hiện một form hiển thị các thông tin cơ bản như ảnh đại diện, tên sử dụng, liên hệ là email hoặc số điện thoại đăng ký cho tài khoản đó.



Hình 22: Form thể hiện thông tin cá nhân

* 1. **Hướng dẫn chạy:**

Để chạy được đồ án này đầu tiên cần phải cài ứng dụng visual studio code trên máy. Sau khi cài xong trên thanh công cụ chọn terminal -> new terminal, tại màn hình terminal gõ lệnh “git clone https://github.com/nguyendinhhieu12345/Webchat.git” nếu đã cài git trên máy để cài đặt đồ án về máy. Nếu không ta có thể vào link github được cung cấp nhấn vào  như hình và download ZIP về máy.

Sau khi đã có đồ án trên máy thì mở ứng dụng visual studio code và mở terminal gõ lệnh “npm install” hoặc “npm i” để cài đặt các thư viện hỗ trợ. Sau đó, gõ lệnh “npm start” chờ đến lúc trang web hiển thị thì có thể sử dụng. Tuy nhiên, vì sử dụng firebase nên cần có mạng để có thể lấy được dữ liệu lưu trữ.

Tên miền “http://localhost:3000” sẽ hiển thị lên trang chủ, “http://localhost:3000/chat” sẽ hiển thị lên trang chat, “http://localhost:3000/login” là dường dẫn vào trang đăng nhập, “http://localhost:3000/signup” là đường dẫn vào trang đăng ký.

**CHƯƠNG 5. KẾT LUẬN**

## Kết quả đạt được:

Về cơ bản nhóm tự nhận xét chương trình tạo ra đã giải quyết được những yêu cầu đề ra. Nhóm đã sử dụng reactjs và firebase để xây dựng nên giao diện website và thực hiện được các tính năng đề ra như đăng nhập được thông qua số điện thoại, facebook, google, đăng ký được tài khoản thông qua số điện thoại,… Hiểu rõ hơn về ngôn ngữ đang sử dụng. Bên cạnh đó, nhóm còn nâng cao kỹ năng làm việc nhóm, hiểu rõ hơn về các thành viên. Từ đó, nâng cao được nhiều kỹ năng cho bản thân mình.

## Định hướng phát triển:

Trong quá trình thực hiện cũng gặp phải nhiều khó khăn trong việc code chẳng hạn như firebase thay đổi liên tục nhóm phải tìm hiểu về những sự thay đổi đó để thay đổi code cho phù hợp. Gặp nhiều lỗi phát sinh có những lỗi sửa được nhưng cũng có lỗi chưa sửa được nhóm đã cố gắng tìm cách khắc phục. Sử dụng quá số lượng data mà firebase cho phép và khắc phục bằng cách phải tạo ra cái mới để sài cùng với nhiều những khó khắn khác. Tuy nhiên nếu có nhiều thời gian hơn nhóm sẽ tiếp tục phát triển theo một số định hướng như sau:

- Tối ưu hóa code để thời gian chạy tốt hơn, dễ hiểu hơn thuận tiện cho việc phát triển sau này.

- Thiết kế lại giao diện đẹp hơn, các tính năng theo tác dễ hơn.

- Cấu trúc file dễ hiểu, dễ nhìn hơn.

- Tìm ra những trường hợp bị lỗi và sửa lại chúng. Sau đó phát triển thêm những tính năng mới, áp dụng thêm những công nghệ mới về chẳng hạn như boot chat tự động trả lời,…

* 1. **Ưu điểm, hạn chế:**

Bên cạnh việc xây dựng trang web thì nhóm em nhận thấy được những ưu nhược điểm như sau.

* + 1. **Ưu điểm:**

Xây dựng hoàn thiện trang giao diện web với đầy đủ tính năng đưa ra một cách hợp lý, phù hợp với người sử dụng dễ thao tác. Đăng nhập với nhiều phương thức giúp dễ dàng kết nối với mọi người hơn.

* + 1. **Hạn chế:**

Vẩn còn lỗi ở một vài trường hợp có thể nhóm vẩn chưa tìm ra. Chỉ phù hợp sử dụng được cho website và chưa thiết kế để phù hợp với điện thoại. Firebase thường xuyên cập nhập những cái mới làm cho ta cũng phải cập nhập lại những câu lệnh một cách hợp lý. Trong một số trường hợp bị lỗi làm cho lượng dữ liệu bị giới hạn so với firebase cho phép không sử dụng thêm được nữa. Login qua facebook chỉ cho phép một tài khoản facebook được đăng nhập vào.

# TÀI LIỆU THAM KHẢO

1. <https://firebase.google.com/docs> , hướng dẫn sử dụng firebase
2. https://tenten.vn/tin-tuc/firebase-la-gi-nen-tang-tu-google/#:~:text=Firebase%20l%C3%A0%20g%C3%AC%3F%20Firebase%20l%C3%A0%20Backend-as-a-Service%20%28Baas%29%20n%E1%BB%81n,tr%C3%AAn%20n%E1%BB%81n%20Cloud%20%E2%80%93%20%C4%90i%E1%BB%87n%20to%C3%A1n%20%C4%91%C3%A1m%20m%C3%A2y, firebase là gì?
3. <https://viblo.asia/p/reactjs-uu-diem-va-nhuoc-diem-V3m5WzexlO7> ,Bắt đầu với Reactjs.
4. <https://reactjs.org/docs/add-react-to-a-website.html> , hướng dẫn cụ thể về reactjs.
5. <https://bizflycloud.vn/tin-tuc/html-la-gi-20211119162558126.htm> , HTML là gì? Ưu nhược điểm và vai trò của HTML trong lập trình web?