**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**

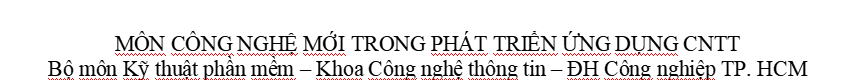
A picture containing text

Description automatically generated

**BÁO CÁO ĐỀ TÀI ỨNG DỤNG CHAT NHÓM 09**

***Nhóm 09 - Sinh viên thực hiện***

1. Hoàng Thị Ngọc Ánh –
2. Nguyễn Hải Nam –
3. Nguyễn Nhật Quang –
4. Đinh Thị Phương Thảo – 19525901
5. Trần Thị Anh Thư – 19516531



MỤC LỤC

[**1.GIỚI THIỆU CHUNG** 4](#_Toc118078258)

[Tính cấp thiết của đề tài 4](#_Toc118078259)

[Đối tượng và phạm vi nghiên cứu 4](#_Toc118078260)

[1.1 Phương pháp nghiên cứu 4](#_Toc118078261)

[1.2 Mục tiêu nghiên cứu đề tài 5](#_Toc118078262)

[1.3 Phạm vi đề tài 6](#_Toc118078263)

[1.3.1 Chức năng dành cho người dùng 6](#_Toc118078264)

[1.4 Mô tả yêu cầu chức năng 6](#_Toc118078265)

[1.4.1 Phân tích yêu cầu của hệ thống 6](#_Toc118078266)

[*1.4.2* Mô tả hoạt động của hệ thống 7](#_Toc118078267)

[**2.** **CƠ SỞ LÝ THUYẾT** 7](#_Toc118078268)

[2.1 Giới thiệu 7](#_Toc118078269)

[2.2 Giới thiệu chung về hệ thống chat 8](#_Toc118078270)

[2.3 Giới thiệu các công nghệ sử dụng cho hệ thống chat 9](#_Toc118078271)

[2.3.1 Công nghệ React Native 9](#_Toc118078272)

[2.3.2 Công nghệ NodeJS 10](#_Toc118078273)

[2.3.3 Công nghệ Socket.IO 10](#_Toc118078274)

[2.3.4 Công nghệ React .JS 11](#_Toc118078275)

[2.3.5 MongoDB 11](#_Toc118078276)

[**3.** **PHÂN TÍCH VÀ THIẾT KẾ** 12](#_Toc118078277)

[3.1 Mô hình Use case 12](#_Toc118078278)

[3.2 Danh sách các tác nhân và mô tả 13](#_Toc118078279)

[3.3 Danh sách Use case và mô tả 13](#_Toc118078280)

[**4.** **ĐẶC TẢ CÁC YÊU CẦU CHỨC NĂNG (FUNCTIONAL)** 14](#_Toc118078281)

[4.1 UC001\_Đăng ký tài khoản 14](#_Toc118078282)

[4.1.1 Mô tả use case UC001 14](#_Toc118078283)

[4.1.2 Biểu đồ 16](#_Toc118078284)

[4.2 UC002\_Đăng nhập tài khoản 18](#_Toc118078285)

[4.2.1 Mô tả use case UC002 18](#_Toc118078286)

[4.2.2 Biểu đồ 19](#_Toc118078287)

[4.3 UC003\_gửi tin nhắn Chat 21](#_Toc118078288)

[4.3.1 Mô tả use case UC003 21](#_Toc118078289)

[4.3.2 Sơ đồ 22](#_Toc118078290)

[4.4 UC004\_Tạo nhóm chat 24](#_Toc118078291)

[4.4.1 Mô tả Use case UC004 24](#_Toc118078292)

[4.4.2 Sơ đồ 24](#_Toc118078293)

[4.5 UC005\_Gửi yêu cầu kết bạn 25](#_Toc118078294)

[4.5.1 Mô tả Use case UC005 25](#_Toc118078295)

[4.5.2 Sơ đồ 26](#_Toc118078296)

[4.6 UC006\_Hủy kết bạn 28](#_Toc118078297)

[4.6.1 Mô tả Use case UC006 28](#_Toc118078298)

[4.6.2 Sơ đồ 29](#_Toc118078299)

[4.7 UC007\_Chặn người dùng 30](#_Toc118078300)

[4.7.1 Mô tả Use case UC007 30](#_Toc118078301)

[4.7.2 Sơ đồ 31](#_Toc118078302)

[4.8 UC008\_Thêm thành viên vào nhóm chat 33](#_Toc118078303)

[4.8.1 Mô tả Use case UC008 33](#_Toc118078304)

[4.8.2 Sơ đồ 33](#_Toc118078305)

[4.9 UC009\_Xóa thành viên ra khỏi nhóm chat 35](#_Toc118078306)

[4.9.1 Mô tả Use case UC009 35](#_Toc118078307)

[4.9.2 Biểu đồ 35](#_Toc118078308)

[4.10 UC010\_Tìm kiếm người dùng 37](#_Toc118078309)

[4.10.1 Mô tả Use case UC010 37](#_Toc118078310)

[4.10.2 Biểu đồ 37](#_Toc118078311)

[4.11 UC011\_Thu hồi tin nhắn 38](#_Toc118078312)

[4.11.1 Mô tả Use case UC011 38](#_Toc118078313)

[4.11.2 Sơ đồ 39](#_Toc118078314)

[**5.** **CÁC THÔNG TIN HỖ TRỢ KHÁC** 40](#_Toc118078315)

[5.1. Sơ đồ lớp (entity) 40](#_Toc118078316)

[**6.** **HIỆN THỰC** 40](#_Toc118078317)

[6.1 Cấu hình phần cứng, phần mềm 40](#_Toc118078318)

[6.2 Giao diện của hệ thống 40](#_Toc118078319)

[6.2.1 Giao diện Web 40](#_Toc118078320)

[6.2.2 Giao diện App 40](#_Toc118078321)

[6.3 Kế hoạch và kiểm thử hệ thống 40](#_Toc118078322)

[6.3.1 Những yêu cầu về tài nguyên cho kiểm thử ứng dụng 40](#_Toc118078323)

[6.3.2 Kiểm thử hệ thống 41](#_Toc118078324)

[**7. KẾT LUẬN** 41](#_Toc118078325)

[7.1 Kết quả đạt được 41](#_Toc118078326)

[7.2 Hạn chế của đồ án 42](#_Toc118078327)

[7.3 Hướng phát triển 42](#_Toc118078328)

[**TÀI LIỆU THAM KHẢO** 42](#_Toc118078329)

# **1.GIỚI THIỆU CHUNG**

## Tính cấp thiết của đề tài

Ngày nay, ứng dụng công nghệ và việc tin học hóa được xem là một trong những yếu tố mang tính quyết định trong hoạt động của các chính phủ , tổ chức, cũng như các công ty, nó đóng vai trò hết sức quan trọng , có thể tạo ra những bước đột phá mạnh mẽ. Với sự phát triển ngày càng nhanh chóng này của xã hội thì việc phát triển hệ thống mạng lưới thông tin là một điều kiện tiên quyết. Và khi hệ thống mạng lưới thông tin được phủ sóng rộng rãi, chúng ta chắc chắn sẽ thực hiện được nhiều công việc, chuyển thông tin, dữ liệu với tốc độ nhanh hơn và chi phí thấp hơn nhiều so với cách thức truyền thống. Chính điều này, đã thúc đẩy sự khai sinh và phát triển ứng dụng Chat -ứng dụng truyền thông giúp cho việc truyền thông tin nay trở nên dễ dàng hơn và tiết kiệm hơn, nhanh chóng hơn, làm biến đổi đáng kể bộ mặt văn hóa, nâng cao chất lượng cuộc sống con người.

## Đối tượng và phạm vi nghiên cứu

Nghiên cứu phát triển ứng dụng chat trên hai nền tảng phổ biến nhất hiện nay là androi và web browser

## Phương pháp nghiên cứu

Sử dụng kiến thức đã học và các tài liệu liên quan trong và ngoài nước liên quan đến đề tài nghiên cứu đặc biệt là các tài liệu về phát triển ứng dụng , thiết kế hệ thống trong hệ thống thông tin di động và web.

Nghiên cứu và tìm hiểu tập trung bám sát đề cương dưới sự hướng dẫn của giảng viên

## Mục tiêu nghiên cứu đề tài

Nghiên cứu và phát triển thành công để ứng dụng có thể truyền thông tin trực tiếp theo thời gian thực và chạy ổn định trên hai nền tảng mobile và website.

Một ứng dụng hỗ trợ chat là phần thiết yếu của cuộc sống khi mà nhu cầu giao tiếp của mọi người bị cản trở bởi dịch bện, mọi người cần một nền tảng để giao tiếp và duy trì các mối quan hệ.

Với nền tảng website người dùng có thể :

* Xem và chỉnh sửa thông tin cá nhân.
* Kết bạn dựa trên số điện thoại và email.
* Xem danh sách bạn bè, danh sách nhóm.
* Gọi video call cho người khác.
* Chat cá nhân: gửi hình ảnh, video, sticker, thả reaction vào tin nhắn, thu hồi và xoá bỏ tin nhắn, mention tới người nhắn tin.
* Chat nhóm: thêm thành viên vào nhóm, các chức năng tương tự với chat cá nhân, giải tán nhóm, thêm nhóm phó, ghim tin nhắn.
* Về phía admin :có thể quản lý tài khoản người dùng,quản lý nhóm sticker và sticker.

Với nền tảng mobile người dùng có thể

* Xem và chỉnh sửa thông tin cá nhân.
* Kết bạn dựa trên số điện thoại và email.
* Xem danh sách bạn bè, danh sách nhóm.
* Gọi video call cho người khác.
* Chat cá nhân: gửi hình ảnh, video, sticker, thả reaction vào tin nhắn, thu hồi và xoá bỏ tin nhắn, mention tới người nhắn tin.
* Chat nhóm: thêm thành viên vào nhóm, các chức năng tương tự với chat cá nhân, giải tán nhóm, thêm nhóm phó, ghim tin nhắn.

## Phạm vi đề tài

### Chức năng dành cho người dùng

• Giao diện dễ sử dụng và thân thiện, dễ thao tác và sử dụng.

• Cho phép khách hàng đăng kí tài khoản và đảm bảo thông tin khách hàng.

• Cho phép khách hàng thay đổi thông tin cá nhân, ảnh đại diện, ảnh bìa…

• Xem và thay đổi các thông tin về tài khoản.

• Chat nhóm và group, video call với người khác

• Xoá, thu hồi tin nhắn, gửi các sticker, emoji, reaction các tin nhắn với các sticker đa dạng và phong phú

• Xem và tìm kiếm theo nhiều danh mục.

• lưu trữ các tệp tin đa phương tiện như hình ảnh, video...

## Mô tả yêu cầu chức năng

### Phân tích yêu cầu của hệ thống

- Người dùng có thể thay đổi thông tin cá nhân như hình ảnh, tên, giới tính…

- Người dùng thực hiện chat trên ứng dụng, hệ thống sẽ tiếp nhận tin nhắn và gửi cho đối tượng chỉ định

- Cho phép người dùng gửi tin nhắn, sticker, emoji, hình ảnh và video.

- Người dùng có thể thu hồi hoặc xoá tin nhắn, ghim tin nhắn quan trọng và lưu trữ các tệp đa phương tiện để người dùng dễ kiểm soát.

- Từ mô tả trên, có thể đưa ra yêu cầu của hệ thống với hai đối tượng chính tương tác với hệ thống như sau:

· Đối với hệ thống:

- Tìm kiếm thông tin người dùng dựa trên số điện thoại và email

· Đối với người dùng:

- Tạo tài khoản, xem và chỉnh sửa thông tin cá nhân.

- Chat đơn hoặc chat group.

- Thu hồi, xoá hoặc ghim tin nhắn quan trọng.

- Gửi sticker, hình ảnh và video…

- Kết bạn bằng số điện thoại hoặc email.

- Xem tệp tin đa phương tiện đã gửi.

### Mô tả hoạt động của hệ thống

-Đối với người dùng:

Người dùng cần phải đăng ký 1 tài khoản gồm email hoặc số điện thoại và mật khẩu trước khi sử dùng ứng dụng chat.

Người dùng có thể kết bạn dựa trên số điện thoại và email, sau khi có bạn bè trong danh sách, người dùng có thể chat với bạn bè của mình.

Thành viên trong hộp thoại chat có thể gửi tin nhắn văn bản, đi kèm với hình ảnh, sticker hoặc video và có thể thả reaction vào tin nhắn của người khác để biểu đạt cảm xúc.

Khi gửi nhầm hoặc vô tình gửi tin nhắn, người dùng có thể thu hồi hoặc xoá tin nhắn mà mình vừa gửi, ngoài ra có thể xoá chat nếu muốn.

Ngoài ra, người dùng có thể tạo group chat để tiện trong việc trao đổi thông tin theo nhóm, với cách thức hoạt động tương tự như chat đơn .

# **CƠ SỞ LÝ THUYẾT**

## Giới thiệu

Dưới sự phát triển càng nhanh của xã hội thì theo đó là sự lớn mạnh một cách nhanh chóng của công nghệ thông tin. Vì lẽ đó nên chúng ta ngày càng được chứng kiến nhiều sự xuất hiện của rất nhiều công nghệ mới, nền tảng phát triển mới như hiện nay. Vậy thì, việc lựa chọn ngôn ngữ, lựa chọn nền tảng để bắt đầu một dự án là một việc hết sức phức tạp và phụ thuộc vào rất nhiều yếu tố cũng như mục đích của dự án. Và ở đây chúng ta sẽ nói về lý do để lựa chọn và đặc điểm của các công nghệ được sử dụng cho việc phát triển hệ thống chat.

## Giới thiệu chung về hệ thống chat

Để xây dựng một hệ thống Chat lấy dữ liệu thời gian thực thì ứng dụng phải gồm 2 phần chính là phần giao diện người dùng và phần máy chủ để xử lý và lưu trữ dữ liệu.

Trước tiên là phần giao diện người dùng, Ở đây mục tiêu của chúng ta là xây dựng và phát triển ứng dụng chat trên cả hai nền tảng phổ biến nhất hiện nay là android và web browser, vậy thì thông thường chúng ta sẽ cần viết riêng ứng dụng bằng ngôn ngữ java để có chạy trên nền tảng android và web.

Và những ngôn ngữ này được gọi là ngôn ngữ native, và có một sự thật là ngôn ngữ native thì đêm lại cho chúng ta rất nhiều ưu điểm như là :

Về mặt hiệu năng thì khi sử dụng ngôn ngữ để lập trình thì ứng dụng sẽ chắc chắn chạy nhanh hơn trong hầu hết trường hợp. Tuy nhiên sự chênh lệch ở các ứng dụng có thể là nhỏ, chấp nhận được và khó nhận biết bởi người dùng. Và có một sự thật là chỉ ngôn ngữ native mới có khả năng truy cập các phần cứng (như camera, GPS, thiết bị thu âm... ) và có thể truy cập đến các quyền như : sao chép, tạo, ghi, đọc tập tin (trên bộ nhớ của điện thoại), các thông tin danh bạ, thông tin cá nhân, thực hiện cuộc gọi, nhắn tin... Nói cách khác, khi sử dụng ngôn ngữ native thì ứng dụng có thể khai thác được sức mạnh phần cứng nhiều nhất của nền tảng.

Tuy nhiên thì với những dự án vừa và nhỏ thì công cuộc xây dựng ứng dụng kiểu này sẽ thực sự tốn chi phí và thời gian cũng như công sức phát triển và bảo trì ứng dụng. Và với một dự án nhỏ về hệ thống chat như thế này, thì ngôn ngữ native chắc chắn không phải là ngôn ngữ tối ưu nhất để phát triển ứng dụng. Và sự lựa chọn trong dự án này chắc chắn phải là một nền tảng giúp chỉ việc code một lần để sở hữu ứng dụng có thể vận hành trên cả hai nền tảng.

Sự thật là hiện nay có một vài framework khá tốt như react native, flux Và lựa chọn của nhóm trong ứng dụng này sẽ là React native,Chúng ta sẽ bàn về lý do chúng ta sử dụng react native ở phần sau. React native thì được viết trên ngôn ngữ javascript và đẩy cũng chính là điểm mạnh của framework này.

Vậy thì với react native và javascript chúng ta có thể dễ dàng xây dựng giao diện người dùng.

Còn lại bên phía máy chủ, có hai phần chính để xây dựng một hệ thống máy chủ hoàn chỉnh đó chính là api và cơ sở dữ liệu, và cũng tồn tại khá nhiều loại api như là graphql server, hay restful api , rest api .... Và để mỗi loại api thì có các ngôn ngữ, cách thực lập trình khác nhau. Và chọn loại api nào sẽ phụ thuộc vào mục đích sử dụng của chúng ta. Api sẽ có nhiệm vụ kết nối giữa phía người dùng và hệ thống cơ sở dữ liệu. Và có một vài cơ sở dữ liệu có thể kể đến như là hệ thống cơ sở quản lý dữ liệu quan hệ (Oracle, M, MS server, PostgreSQL), hệ cơ sở dữ liệu hướng tài liệu (MongoDB), Cơ sở dữ liệu dạng khoá ...

Ngoài ra thì chúng ta cũng cần phải có thư viện cung cấp các phương thức để phía người dùng có thể giao tiếp với api để lấy dữ liệu từ cơ sở dữ liệu. Cũng phải nói rằng mỗi loại api khác nhau thì có các cách để gửi http request khác nhau và các thư viện cung cấp các phương thức gửi http request khác nhau như là axios, apollo client ....

Tuy nhiên thì ở đây, Để tạo ra một ứng dụng chat có sử dụng dữ liệu thời gian thực, thì đòi hỏi phía máy chủ và phía máy khách phải liên tục gửi dữ liệu sang cho nhau với độ trễ thấp nhất có thể ( ms ). Vì vậy nên sự lựa chọn của tôi trong dự án này là cơ sở dữ liệu phi quan hệ ( MongoDB) Cùng với đó tôi sẽ sử dụng các phương thức cung cấp để phía máy khách có thể truy cập vào và lấy dữ liệu trực tiếp từ hệ thống cơ sở dữ liệu.

## Giới thiệu các công nghệ sử dụng cho hệ thống chat

### 2.3.1 Công nghệ React Native

React Native là công nghệ được tạo bởi Facebook, cho phép các lập trình viên sử dụng xavaScript để làm mobile apps trên cả Android và iOS.

Mặc khác, nếu chúng ta chọn lập trình trên React Native, phần lớn thời gian chúng ta sẽ chỉ cần học 1 bộ công cụ. Có rất nhiều thứ để làm quen như: javaScript, Node, React Native...

Một ưu điểm nữa là giảm chi phí tái biên dịch của Native bằng cách sử dụng Hot-Loading tức là ta không cần phải build lại ứng dụng từ đầu nên việc chỉnh sửa diễn ra rất nhanh chóng. Giúp cho lập trình viên có thể thấy được những chỉnh sửa của họ một cách nhanh chóng trực quan, không còn phải bỏ quá nhiều thời gian trong việc build và chạy ứng dụng nữa.

Ưu điểm của React Native:

- Hiệu quả về mặt thời gian khi mà bạn muốn phát triển một ứng dụng nhanh chóng.

- Hiệu năng tương đối ổn định.

- Cộng đồng phát triển mạnh.

- Tiết kiệm chi phí.

- Ứng dụng tin cậy và ổn định.

- Xây dựng cho nhiều hệ điều hành khác nhau với ít native code nhất.

### 2.3.2 Công nghệ NodeJS

NodeJS là một môi trường chạỵ javascript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phát triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện không đồng bộ.

Node.js là một Framework mạnh mẽ dựa trên nền tảng Google Chrome Javascript V8 Engine. Node.js được dùng để phát triển các ứng dụng hướng nhiều đến I/O trên nền tảng web như Video Streaming Site, các ứng dụng Single-page App hay các ứng dụng web khác.

Ưu điểm của Nodejs:

* Các ứng dụng Nodejs được viết bằng javascript, ngôn ngữ này là một ngôn ngữ khá thông dụng.
* Nodejs chạy đa nền tảng phía Server, sử dụng kiến trúc hướng sự kiện Event-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả.
* Có thể chạy ứng dụng Nodejs ở bất kỳ đâu trên máy Mac – Window –
* Linux, hơn nữa cộng đồng Nodejs rất lớn và hoàn toàn miễn phí.

### 2.3.3 Công nghệ Socket.IO

Socketio sẽ giúp các bên ở những địa điểm khác nhau kết nối với nhau, truyền dữ liệu ngay lập tức thông qua server trung gian. Có thể được sử dụng trong nhiều ứng dụng như chat, game online, cập nhật kết quả của một trận đấu đang xảy ra

Ưu điểm:

* Khả năng bảo mật
* Kết nối lại tự động + phát hiện tình trạng ngắt kết nối
* Hỗ trợ nhị phân
* Hỗ trợ tạo phòng và kênh

### 2.3.4 Công nghệ React .JS

React.JS là một thư viện Javascript dùng để xây dựng giao diện người dùng, React hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được.

Ưu điểm của Reactjs:

* Sử dụng DOM ảo là một đối tượng JavaScript. Điều này sẽ cải thiện hiệu suất ứng dụng, vì DOM ảo JavaScript nhanh hơn DOM thông thường.
* Có thể được sử dụng trên máy khách và máy chủ cũng như với các khuôn khổ khác.
* Các mẫu thành phần và dữ liệu cải thiện khả năng đọc, giúp duy trì các ứng dụng lớn hơn.

### 2.3.5 MongoDB

MongoDB là một chương trình cơ sở dữ liệu mã nguồn mở được thiết kế theo kiểu hướng đối tượng trong đó các bảng được cấu trúc một cách linh hoạt cho phép các dữ liệu lưu trên bảng không cần phải tuân theo một dạng cấu trúc nhất định nào.

Ưu điểm của MongoDB:

* Hiệu suất cao
* Tính sẵn sàng cao – Nhân rộng
* Khả năng mở rộng cao – Sharding
* Năng động – Không có lược đồ cứng nhắc.
* Linh hoạt – thêm / xóa trường có ít hoặc không ảnh hưởng đến ứng dụng
* Dữ liệu không đồng nhất
* Không joins
* Phân phối được
* Biểu diễn dữ liệu trong JSON hoặc BSON

# **PHÂN TÍCH VÀ THIẾT KẾ**

## Mô hình Use case

*Diagram

Description automatically generated*

## Danh sách các tác nhân và mô tả

|  |  |  |
| --- | --- | --- |
| Tác nhân | Mô tả tác nhân | Ghi chú |
| User | Người dùng (User) khi chưa có tài khoản có thể thực hiện chức năng đăng ký tài khoản mới (register account).Nếu người dùng đã có tài khoản thực hiện chức năng: đăng nhập, đăng xuất, chat, tạo nhóm, thêm bạn, …. |  |
|  |  |  |

## Danh sách Use case và mô tả

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tên Use case | Mô tả ngắn gọn Use case | Chức năng | Ghi chú |
| UC001 | Đăng ký tài khoản | Use case mô tả các bước đăng ký một tài khoản |  |  |
| UC002 | Đăng nhập tài khoản | Use case mô tả cách đăng nhập trên ứng dụng |  |  |
| UC003 | Gửi tin nhắn Chat | Use case mô tả các bước để gửi tin nhắn trên ứng dụng |  |  |
| UC004 | Tạo nhóm chat | Use case mô tả các bước để tạo một nhóm chat trên ứng dụng |  |  |
| UC005 | Gửi yêu cầu kết bạn | Use case mô tả các bước kết bạn trên ứng dụng |  |  |
| UC006 | Hủy kết bạn | Use case mô tả các bước hủy kết bạn trên ứng dụng |  |  |
| UC007 | Chặn người dùng | Use case mô tả các bước chặn một người dùng trên ứng dụng |  |  |
| UC008 | Thêm thành viên vào nhóm | Use case mô tả các bước thêm một thành viên vào nhóm chat trên ứng dụng |  |  |
| UC009 | Xóa thành viên ra khỏi nhóm chat | Use case mô tả các bước xóa một thành viên ra khỏi nhóm chat trên ứng dụng |  |  |
| UC010 | Tìm kiếm người dùng | Use case mô tả các bước tìm kiếm |  |  |
| UC011 | Thu hồi tin nhắn |  |  |  |

# **ĐẶC TẢ CÁC YÊU CẦU CHỨC NĂNG (FUNCTIONAL)**

Xác định yêu cầu chức năng:

* Người dùng tạo một tài khoản mới
* Người dùng đăng nhập vào hệ thống
* Người dùng đăng xuất tài khoản ra khỏi hệ thống
* Người dùng thêm bạn bè mới
* Người dùng hủy kết bạn
* Người dùng gửi tin nhắn chat
* Người dùng tạo nhóm chat mới
* Người dùng chặn người dùng
* Người dùng thêm thành viên vào nhóm chat
* Người dùng xóa thành viên ra khỏi nhóm chat
* Người dùng tìm kiếm người dùng
* Người dùng thu hồi tin nhắn

## UC001\_Đăng ký tài khoản

### Mô tả use case UC001

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case: Đăng ký tài khoản** | | | |
| Mục đích: | Giúp người dùng đăng ký tài khoản để vào hệ thống sử dụng dịch vụ của ứng dụng | | |
| Mô tả: | Use case mô tả các bước đăng ký tài khoản | | |
| Tác nhân: | User | | |
| Điều kiện trước: | Người dùng phải có số điện thoại và chưa có tài khoản ứng dụng | | |
| Điều kiện sau: | Nếu tạo tài khoản thành công người dùng có thể đăng nhập vào hệ thống | | |
| Luồng sự kiện chính (Basic flows): | | | |
| Actor | | System | |
| 1. Người dùng chọn chức năng đăng ký tài khoản | |  | |
|  | | 1. Hệ thống hiển thị màn hình đăng ký tài khoản | |
| 1. Người dùng nhập thông tin đăng ký | |  | |
| 1. Người dùng bấm chọn đăng ký | |  | |
|  | | 1. Hệ thống kiểm tra thông tin đăng ký | |
|  | | 1. Hệ thống kiểm tra tài khoản đã tồn tại chưa | |
|  | | 1. Hệ thống gửi mã xác nhận | |
| 1. Người dùng nhập mã | |  | |
| 1. Người dùng bấm gửi | |  | |
|  | | 1. Hệ thống kiểm tra mã OTP | |
|  | | 1. Hệ thống thông báo đăng ký thành công và chuyển hướng tới giao diện chat | |
| Luồng sự kiện phụ (Alternative Flows): | | | |
|  | | | 5.1 Hệ thống hiển thị màn hình đăng ký kèm thông báo lỗi |
| 5.2 Quay lại bước 3 | | |  |
|  | | | 6.1 Hệ thống thông báo tài khoản đã tồn tại |
| 6.2 Quay lại bước 3 | | |  |
|  | | | 10.1 Hệ thống thông báo sai mã OTP |
| 10.2 Quay lại bước 8 | | |  |

### Biểu đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Senquence

**Diagram

Description automatically generated**

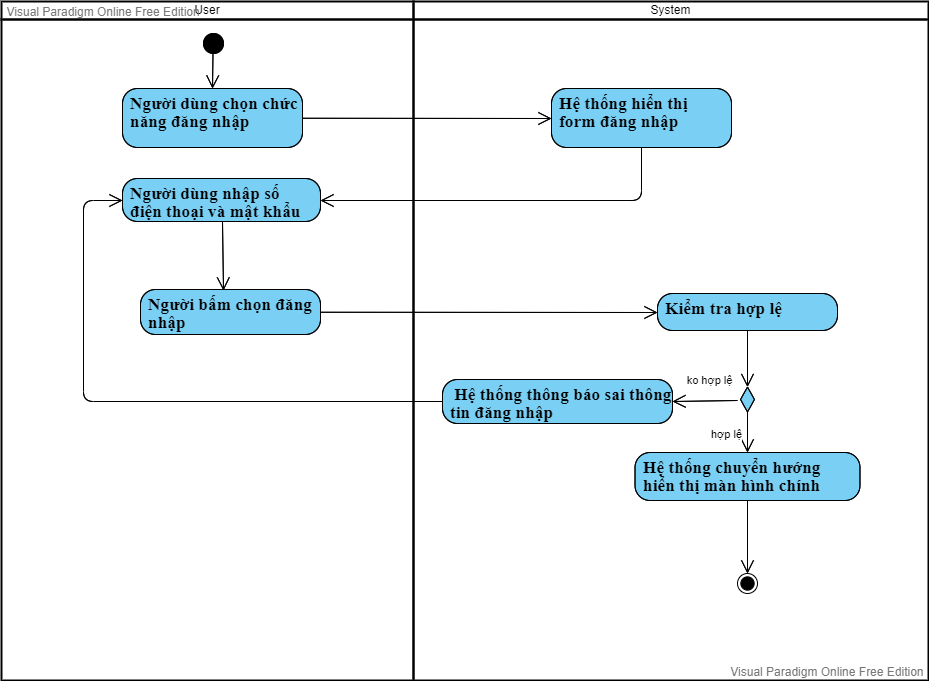
## UC002\_Đăng nhập tài khoản

### Mô tả use case UC002

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case: Đăng nhập tài khoản** | | | |
| Mục đích: | Giúp người dùng đăng nhập vào hệ thống | | |
| Mô tả: | Use case mô tả các bước đăng nhập vào hệ thống | | |
| Tác nhân: | User | | |
| Điều kiện trước: | Phải có tài khoản của hệ thống trước đó | | |
| Điều kiện sau: | Sau khi đăng nhập thành công thì hệ thống sẽ phân quyền sử dụng chức năng ứng dụng . | | |
| Luồng sự kiện chính (Basic flows): | | | |
| Actor | | System | |
| 1. Người dùng chọn chức năng đăng nhập | |  | |
|  | | 1. Hệ thống hiển thị form đăng nhập | |
| 1. Người dùng nhập số điện thoại và mật khẩu | |  | |
| 1. Người bấm chọn đăng nhập | |  | |
|  | | 1. Kiểm tra hợp lệ | |
|  | | 1. Hệ thống chuyển hướng hiển thị màn hình chính | |
| Luồng sự kiện phụ (Alternative Flows): | | | |
|  | | | 4.1 Hệ thống thông báo sai thông tin đăng nhập |
| 5.2 Quay lại bước 3 | | |  |

### Biểu đồ

1. Sơ đồ activity



1. Sơ đồ Sequence

Chart, diagram

Description automatically generated with medium confidence

## UC003\_gửi tin nhắn Chat

### Mô tả use case UC003

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case: Gửi tin nhắn chat** | | | |
| Mục đích: | Giúp người dùng gửi tin nhắn văn bản, file,… tới nhóm chat hay user khác | | |
| Mô tả: | Use case mô tả các bước gửi tin nhắn chat | | |
| Tác nhân: | User | | |
| Điều kiện trước: | User đã đăng nhập vào hệ thống và đã tạo cuộc trò chuyện với user khác hoặc nhóm muồn nhắn tin | | |
| Điều kiện sau: | Người dùng có thể nhắn tin với bạn bè hoặc nhóm trong cửa sổ chat. | | |
| Luồng sự kiện chính (Basic flows): | | | |
| Actor | | System | |
| 1. Người dùng chọn người dùng hoặc nhóm chat cần kết nối | |  | |
|  | | 1. Hệ thống hiển thị cửa sổ trò chuyện | |
| 1. Người dùng nhập tin nhắn/ file/… | |  | |
| 1. Người dùng chọn gửi | |  | |
|  | | 1. Hệ thống kiểm tra tin nhắn và xử lý tin nhắn | |
|  | | 1. Hệ thống hiển thị tin nhắn vừa gửi và cập nhật danh sách tin nhắn được gửi | |
| Luồng sự kiện phụ (Alternative Flows): | | | |
|  | | | 5.1 Hệ thống kiểm tra nội dung tin nhắn không hợp lệ (rỗng, chỉ chứa kí tự khoảng trắng) |
|  | | | 5.2 Hệ thống xóa nội dung tin nhắn trong ô nhập tin nhắn |

### Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence

Diagram

Description automatically generated

## 4.4 UC004\_Tạo nhóm chat

### 4.4.1 Mô tả Use case UC004

|  |  |  |  |
| --- | --- | --- | --- |
| **Use case: Tạo nhóm chat** | | |  |
| Mục đích: | Giúp cho người dùng tạo nhóm trò chuyện nhiều thành viên | | |
| Mô tả: | Use case mô tả các bước tạo một nhóm chat mới | | |
| Tác nhân: | Người dùng | | |
| Điều kiện tiên quyết: | User đã đăng nhập vào hệ thống | | |
| Điều kiện sau: | Sau khi tạo nhóm chat thành công các thành viên trong nhóm có thể gửi tin nhắn, file cho nhau. | | |
| Luồng sự kiện chính: | |  | |
| Actor | | | System |
| 1. Người dùng chọn vào icon tạo nhóm phía trên màn hình | | |  |
|  | | | 2. Hệ thống hiển thi màn hình tạo mới một nhóm chat |
| 3. Người dùng đặt tên nhóm và thêm thành viên vào nhóm | | |  |
|  | | | 4. Hệ thống kiểm tra tính hợp lệ (tên nhóm không để trống, số thành viên tối thiểu của nhóm là 3) |
| 5. Người dùng chọn nút tạo nhóm | | |  |
|  | | | 6. Hệ thống tạo nhóm mới và thêm nhóm vừa tạo vào danh sách nhóm. |
| Luồng sự kiện thay thế: | | |  |
|  | | | 4.1 Hệ thống thông báo lỗi |
|  | | | 4.2 Quay lại bước 3 |

### 4.4.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ sequence

## 4.5 UC005\_Gửi yêu cầu kết bạn

### 4.5.1 Mô tả Use case UC005

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use case: Thêm bạn bè** | | |  | |
| Mục đích: | Giúp người dùng thêm liên kết với người dùng khác | | | |
| Mô tả: | Use case mô tả các bước thêm bạn bè | | | |
| Tác nhân: | User | | | |
| Điều kiện tiên quyết: | Người dùng đã đăng nhập vào hệ thống và đang ở màn hình chính | | | |
| Điều kiện sau: | Người dùng có thể gửi yêu cầu kết bạn với người dùng khác | | | |
| Luồng sự kiện chính: | |  | | |
| Actor | | | | System |
| 1. Người dùng bấm chọn icon thêm bạn bè phía trên cửa sổ chat | | | |  |
|  | | | | 2. Hệ thống hiển thị màn hình tìm kiếm |
| 3. Người dùng nhập số điện thoại của người cần kết bạn vào ô tìm kiếm | | | |  |
|  | | | | 4. Hệ thống xử lý tìm kiếm |
|  | | | | 5. Hệ thống hiển thị kết quả tìm kiếm |
| 6. Người dùng bấm chọn xem thông tin tài khoản của người cần kết bạn | | | |  |
|  | | | | 7. Hệ thống hiển thị thông tin của người cần kết bạn |
| 8. Người dùng chọn gửi lời mời kết bạn | | | |  |
|  | | | | 9. Hệ thống xử lý yêu cầu kết bạn |
|  | | | | 10. Hệ thống hiển thị thông báo gửi yêu cầu kết bạn thành công. |
| Luồng sự kiện thay thế: | | | |  |
|  | | | | 5.1 Hệ thống thông báo không tìm thấy thông tin tài khoản |
|  | | | |  |

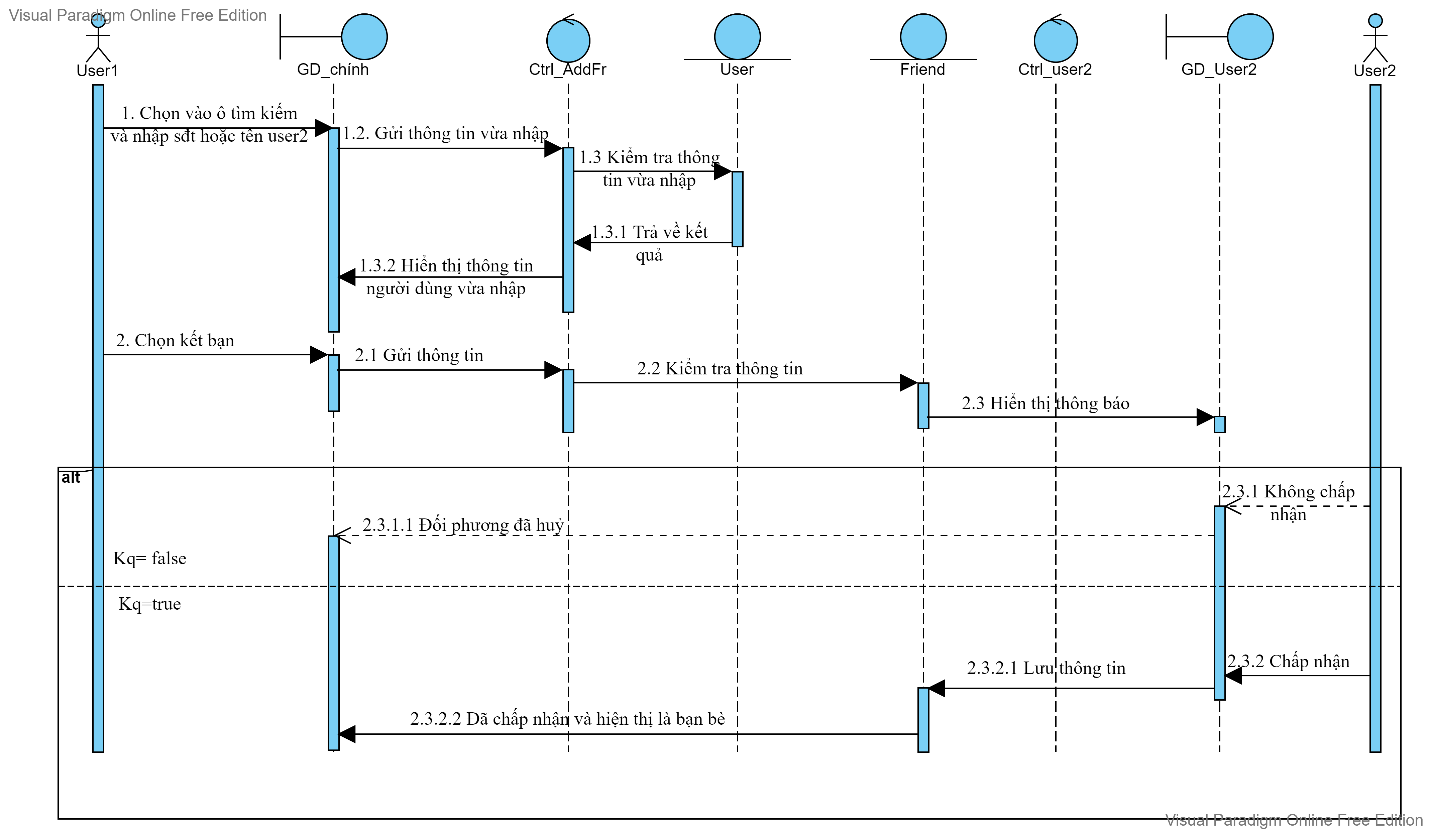
### 4.5.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence



## 4.6 UC006\_Hủy kết bạn

### 4.6.1 Mô tả Use case UC006

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use case: Hủy kết bạn** | | |  | |
| Mục đích: | Giúp cho người dùng có thể xóa một user khỏi danh sách bạn bè | | | |
| Mô tả: | Use case mô tả các bước hủy kết bạn | | | |
| Tác nhân: | User | | | |
| Điều kiện tiên quyết: | Người dùng phải đăng nhập vào hệ thống và đang ở màn hình danh bạ | | | |
| Điều kiện sau: | Sau khi người dùng xóa một người dùng khác ra khỏi danh sách bạn bè thì hệ thống sẽ cập nhật lại danh sách bạn bè | | | |
| Luồng sự kiện chính: | |  | | |
| Actor | | | | System |
| 1. Người dùng chọn một bạn bè cần hủy kết bạn | | | |  |
| 2. Người dùng chọn chức năng “Xóa bạn” tại giao diện danh sách bạn bè | | | |  |
|  | | | | 3. Hệ thống yêu cầu xác nhận xóa |
| 4. Người dùng xác nhận xóa | | | |  |
|  | | | | 5. Hệ thống xử lý yêu cầu xóa bạn |
|  | | | | 6. Hệ thống cập nhật lại danh sách bạn bè và thông báo “xóa thành công” |
| Luồng sự kiện thay thế: | | | |  |
| 4.1 Người dùng chọn hủy | | | |  |
|  | | | |  |

### 4.6.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence

## 4.7 UC007\_Chặn người dùng

### 4.7.1 Mô tả Use case UC007

|  |  |  |
| --- | --- | --- |
| **Use case: Chặn người dùng** | |  |
| Mục đích: | Giúp cho người dùng chặn một người dùng | |
| Mô tả: | Use case mô tả các bước chặn một người dùng | |
| Tác nhân: | Người dùng | |
| Điều kiện tiên quyết: | Người dùng phải đăng nhập vào hệ thống | |
| Điều kiện sau: | Sau khi người dùng chặn một user khác thì user bị chặn không thể trả lời tin nhắn của user chặn được nữa | |
| Luồng sự kiện chính | |  |
| Actor | | System |
| 1.Người dùng chọn vào mục danh bạ trên màn hình chính | |  |
|  | | 2. Hệ thống hiển thị danh sách bạn bè và nhóm chat |
| 3. Người dùng chọn một user cần chặn | |  |
|  | | 4. Hệ thống hiển thị cửa sổ trò chuyện |
| 5. Người dùng click chọn icon tác vụ phía trên cửa sổ chat | |  |
|  | | 6. Hệ thống hiển thị cửa sổ tác vụ |
| 7. Người dùng chọn chặn người dùng | |  |
|  | | 8. Hệ thống hiển thị cửa sổ chat |
| Luồng sự kiện thay thế | |  |
| Actor | | System |
| 7.1. Xác nhận không chặn | |  |
|  | | 6.2. Quay lại bước 6 |

### 4.7.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence

## 4.8 UC008\_Thêm thành viên vào nhóm chat

### 4.8.1 Mô tả Use case UC008

|  |  |
| --- | --- |
| **Tên Use case: Thêm thành viên vào nhóm** | |
| Miêu tả Use Case: | Giúp người dùng có thể thêm thành viên vào nhóm chat |
| Actor: | **User** |
| Điều kiện tiên quyết: | Người dùng phải đăng nhập vào hệ thống và đang ở cửa sổ chat của nhóm |
| Điều kiện hậu kỳ: | Sau khi thêm thành viên vào nhóm chat thành công thì số lượng thành viên sẽ được cập nhật |
| **Luồng sự kiện chính:** | |
| Actor | Hệ thống |
| 1/ Người dùng bấm chọn icon thêm thành viên phía trên cửa sổ chat |  |
|  | 2/ Hệ thống hiển thị màn hình thêm thành viên vào nhóm chat |
| 3/ Người dùng chọn các thành viên cần thêm vào nhóm |  |
| 4/ Người dùng bấm thêm |  |
|  | 5/ Hệ thống gửi yêu cầu đến server |
|  | 6/ Hệ thống thêm các thành viên được lựa chọn vào nhóm, cập nhật lại số lượng thành viên nhóm |
|  | 7/ Hệ thống hiển thị thông báo tin nhắn “Thành viên X đã tham gia nhóm” đến nhóm chat |
| **Luồng sự kiện thay thế:** | |
|  |  |
| **Luồng sự kiện ngoại lệ:** | |

### 4.8.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence

## 4.9 UC009\_Xóa thành viên ra khỏi nhóm chat

### 4.9.1 Mô tả Use case UC009

|  |  |  |
| --- | --- | --- |
| **Tên Use Case: Xóa thành viên ra khỏi nhóm chat** | | |
| Miêu tả Use Case: | Giúp người dùng có thể xóa một thành viên ra khỏi nhóm | |
| Actor: | User | |
| Điều kiện tiên quyết: | Người dùng phải đăng nhập vào hệ thống và đang ở cửa sổ chat của nhóm | |
| Điều kiện hậu kỳ: | Sau khi xóathành viên ra khỏi nhóm chat thành công thì số lượng thành viên nhóm sẽ được cập nhật | |
|  | | |
| 1/ Người dùng chọn xem thông tin nhóm | |  |
|  | | 2/ Hệ thống hiển thị màn hình thông tin nhóm |
| 3/ Người dùng chọn vào mục thành viên | |  |
|  | | 4/ Hệ thống hiển thị danh sách thành viên nhóm |
| 5/ Người dùng chọn thành viên cần xóa và bấm chọn xóa | |  |
|  | | 6/ Hệ thống yêu cầu xác nhận xóa |
| 7/ Người dùng chọn xác nhận xóa | |  |
|  | | 8/ Hệ thống gửi yêu cầu đến server |
|  | | 9/ Hệ thống xóa thành viên được chọn ra khỏi nhóm |
|  | | 10/ Hệ thống cập nhật lại số lượng thành viên |
|  | | 11/ Hệ thống gửi tin nhắn thông báo “thành viên X đã bị xóa khỏi nhóm” đến nhóm chat |
| **Luồng sự kiện thay thế:** | | |
| 7.1/ Người dùng bấm chọn hủy | |  |
|  | | 7.2/ quay lại bước 4 |

### 4.9.2 Biểu đồ

1. Sơ đồ activity

Diagram

Description automatically generated

1. Sơ đồ Sequence

## 4.10 UC010\_Tìm kiếm người dùng

### 4.10.1 Mô tả Use case UC010

|  |  |  |
| --- | --- | --- |
| **Tên Use Case: Tìm kiếm người dùng** | | |
| Miêu tả Use Case: | Giúp người dùng tìm kiếm một người dùng khác đang sử dụng ứng dụng | |
| Actor: | User | |
| Điều kiện tiên quyết: | Người dùng đã đăng nhập vào hệ thống và đang ở trang chủ | |
| Điều kiện hậu kỳ: | Người dùng tìm kiếm được tài khoản người dùng khác đang sử dụng ứng dụng | |
|  | | |
| 1/ Người dùng bấm chọn thanh tìm kiếm phía trên màn hình trang chủ | |  |
|  | | 2/ Hệ thống hiển thị màn hình tìm kiếm |
| 3/ Người dùng nhập số điệnt thoại của người dùng muốn tìm kiếm | |  |
|  | | 4/ Hệ thống gửi yêu cầu đến server |
|  | | 5/ Hệ thống nhận kết quả trả về và hiển thị thông tin người dùng tìm được lên màn hình |
| **Luồng sự kiện thay thế:** | | |
|  | | 5.1/ Hệ thống thông báo không tìm thấy thông tin tài khoản |
|  | | 5.2/ Quay lại bước 3 |

### 4.10.2 Biểu đồ

1. Sơ đồ activity

Diagram

Description automatically generated

## 4.11 UC011\_Thu hồi tin nhắn

### 4.11.1 Mô tả Use case UC011

|  |  |  |
| --- | --- | --- |
| **Tên Use Case: Thu hồi tin nhắn** | | |
| Miêu tả Use Case: | Giúp người dùng thu hồi tin nhắn đã gửi với bạn bè hoặc trong nhóm chat | |
| Actor: | User | |
| Điều kiện tiên quyết: | Người dùng đã đăng nhập vào hệ thống và đang ở cửa sổ chat | |
| Điều kiện hậu kỳ: | Danh sách tin nhắn được cập nhật, bạn bè hay thành viên nhóm không thể thấy tin nhắn đã được thu hồi được nữa | |
|  | | |
| 1/ Người dùng chọn chức năng thu hồi với tin nhắn được chọn | |  |
|  | | 2/ Hệ thống xử lý thu hồi tin nhắn |
|  | | 3/ Hệ thống cập nhật lại danh sách tin nhắn và hiển thị nội dung tin nhắn lên giao diện nhắn tin |
| **Luồng sự kiện thay thế:** | | |

### 4.11.2 Sơ đồ

1. Sơ đồ activity

Diagram

Description automatically generated

# **CÁC THÔNG TIN HỖ TRỢ KHÁC**

## 5.1. Sơ đồ lớp (entity)

1. **HIỆN THỰC**

## Cấu hình phần cứng, phần mềm

* Cấu hình Server: Visual code studio, MongoDB Compass
* Cấu hình Client: Visual code studio, MongoDB Compass

## Giao diện của hệ thống

### 6.2.1 Giao diện Web

### 6.2.2 Giao diện App

## 6.3 Kế hoạch và kiểm thử hệ thống

### 6.3.1 Những yêu cầu về tài nguyên cho kiểm thử ứng dụng

6.3.1.1 Phần cứng

Máy tính cá nhân phải có kết nối mạng LAN

|  |  |  |  |
| --- | --- | --- | --- |
| **CPU** | **RAM** | **HDD** | **Architecture** |
| Intel Core i5, 1.3 GHz trở lên | 4 GB | 120 GB | 64bit |

6.3.1.2 Phần mềm

|  |  |  |
| --- | --- | --- |
| **Tên phần mềm** | **Phiên bản** | **Loại** |
| Visual studio code | 1.62.0 | IDE cho nodejs & reactjs |
| MongoDB | 5.0.2 2008R2 Plus | Hệ quản trị cơ sở dữ liệu |
| Microsoft Windows 10 | 10 | Hệ điều hành |

### 6.3.2 Kiểm thử hệ thống

**UNIT TEST CASE LIST**

|  |  |
| --- | --- |
| Project Name | Test Black-box |
| Project Code | KiemThu\_Nhom09 |
| Normal number of test cases |  |
| Test Environment setup description |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Requipment Name | Class Name | Funtion Name | Sheet Name | Description |
| 1 | Module Đăng nhập |  |  |  |  |
| 2 | Module đăng ký |  |  |  |  |
| 3 | Module chat |  |  |  |  |
| 4 | Module tạo nhóm |  |  |  |  |
| 5 | Module kết bạn |  |  |  |  |
| 6 | Module thu hồi tin nhắn |  |  |  |  |
| 7 | Module cập nhật thông tin |  |  |  |  |

# **7. KẾT LUẬN**

## 7.1 Kết quả đạt được

Nhóm đã hoàn thành các chức năng đã đặt ra như:

· Đối với người dùng:

- Có thể thực hiện đăng ký, đăng nhập tài khoản, sửa đổi thông tin cá nhân

- Thực hiện chat đơn, chat theo nhóm, có thể gửi kèm file, hình ảnh, video và sticker… xoá, thu hồi tin nhắn.

- Thực hiện cuộc gọi đơn giữa người dùng với người dùng khác.

- Có thể kết bạn, tạo và tham gia các group chat.

## 7.2 Hạn chế của đồ án

- Giao diện chưa thực sự dễ sử dụng,chưa được hài hòa

## 7.3 Hướng phát triển

Tiếp tục hoàn thành project, fix các lỗi và thêm vào 1 số chức năng như đăng tải bài viết cá nhân, tương tác bài viết giữa người dùng với nhau…

# **TÀI LIỆU THAM KHẢO**

[1]. [Authenticate phone number với Firebase trong ReactJS (viblo.asia)](https://viblo.asia/p/authenticate-phone-number-voi-firebase-trong-reactjs-aWj53o9o56m)

[2]. [Documentation | Node.js (nodejs.org)](https://nodejs.org/en/docs/)

[3]. [Introduction | Socket.IO](https://socket.io/docs/v4/)

[4]. [Reactjs, Socket.io with React and Node.js (w3guides.com)](https://w3guides.com/tutorial/socket-io-with-react-and-node-js)