

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**

**BÁO CÁO CUỐI KÌ**

**LẬP TRÌNH TRỰC QUAN**

**ĐỀ TÀI**

**ỨNG DỤNG LAN CHAT**

**Giảng viên Lý thuyết: Huỳnh Tuấn Anh**

**Sinh viên thực hiện:** **Nguyễn Bảo Duy 14520219**

**Hoàng Thùy Trang 16521277**

**Trần Thị Thu Tình 16521250**



**Lớp: IT008.I21 Nhóm: Lan Chat**

TP. Hồ Chí Minh, 18 tháng 06 năm 2018

**MỤC LỤC**

[**Chương 1. HIỆN TRẠNG VÀ YÊU CẦU THỰC TẾ** **1**](#_Toc517048137)

[1. Hiện trạng vấn đề 1](#_Toc517048138)

[1.1. Vấn đề 1](#_Toc517048139)

[1.2. Phương hướng giải quyết: 1](#_Toc517048140)

[2. Hiện trạng cơ sở vật chất và con người 1](#_Toc517048141)

[2.1. Tin học 1](#_Toc517048142)

[2.2. Con người 1](#_Toc517048143)

[3. Yêu cầu sơ bộ về phần mềm 1](#_Toc517048144)

[3.1. Tin học 1](#_Toc517048145)

[3.2. Con người 2](#_Toc517048146)

[**Chương 2. PHÂN TÍCH YÊU CẦU PHẦN MỀM VÀ MÔ HÌNH HÓA** **2**](#_Toc517048147)

[1. Yêu cầu phần mềm 2](#_Toc517048148)

[1.1. Yêu cầu chức năng 2](#_Toc517048149)

[1.2. Yêu cầu phi chức năng 3](#_Toc517048150)

[1.3. Bảng FURPS 3](#_Toc517048151)

[2. Mô hình hóa 4](#_Toc517048152)

[2.1. Các trường hợp sử dụng thông thường 4](#_Toc517048153)

[2.2. Mô hình thực thể - mối kết hợp 6](#_Toc517048154)

[**Chương 3. THIẾT KẾ** **7**](#_Toc517048155)

[1. Thiết kế kiến trúc phần mềm 7](#_Toc517048156)

[2. Thiết kế dữ liệu 8](#_Toc517048157)

[3. Thiết kế giao diện và thành phần xử lý giao diện 8](#_Toc517048158)

[3.1. Màn hình đăng nhập (Login Form): 8](#_Toc517048159)

[3.2. Màn hình chính của người dùng (User Form): 10](#_Toc517048160)

[3.3. Màn hình chat riêng (Private Chat Form): 10](#_Toc517048161)

[3.4. Màn hình lịch sử chat riêng (History Chat Form): 11](#_Toc517048162)

[3.5. Màn hình chat nhóm (Group Chat Form): 12](#_Toc517048163)

[3.6. Màn hình lịch sử chat nhóm (History Group Chat Form): 14](#_Toc517048164)

[**Chương 4. KIỂM THỬ** **18**](#_Toc517048165)

[**Chương 5. TỔNG KẾT** **18**](#_Toc517048166)

[5.1. Tổng Kết 18](#_Toc517048167)

[5.2. Nhận Xét & Đánh Giá 18](#_Toc517048168)

**LỜI CẢM ƠN**

Đầu tiên, nhóm em xin gửi lời cảm ơn chân thành đến tập thể quý Thầy Cô Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM đã giúp cho nhóm em có những kiến thức cơ bản làm nền tảng để thực hiện đề tài này. Đặc biệt, nhóm em xin gửi lời cảm ơn và lòng biết ơn sâu sắc nhất tới thầy Huỳnh Tuấn Anh (Giảng viên lý thuyết môn Lập trình trực quan). Thầy đã trực tiếp hướng dẫn tận tình, sửa chữa và đóng góp nhiều ý kiến giúp nhóm chúng em hoàn thành tốt báo cáo môn học của mình. Trong thời gian một học kỳ thực hiện đề tài, nhóm đã vận dụng những kiến thức nền tảng đã tích lũy đồng thời kết hợp với việc học hỏi và nghiên cứu những kiến thức mới. Từ đó, nhóm đã vận dụng tối đa những gì đã thu thập được để hoàn thành một báo cáo đồ án tốt nhất. Tuy nhiên, trong quá trình thực hiện, nhóm không tránh khỏi những thiếu sót. Chính vì vậy, nhóm chúng em rất mong nhận được những sự góp ý từ phía Thầy nhằm hoàn thiện những kiến thức mà nhóm chúng em đã học tập và là hành trang để nhóm thiết kế thực hiện tiếp các đề tài khác trong tương lai. Xin chân thành cảm ơn Thầy!

**Nhóm thực hiện đồ án**

# HIỆN TRẠNG VÀ YÊU CẦU THỰC TẾ

1. Hiện trạng vấn đề
   1. Vấn đề

Hiện nay các phòng máy tại trường Đại học công nghệ thông tin, nhóm thấy được sự bất tiện khi cần phải nộp bài cho thầy qua kênh học tập moodle, gữi file giữa các sinh viên và trao đổi trong khi học vì mạng trường không ổn định khi đi ra internet và khi trao đổi thường gây ra sự khó chịu cho những người xung quanh.

* 1. Phương hướng giải quyết:

Để giải quyết sự khó khăn và những bất cập trên, nhóm quyết định chọn thực hiện ứng dụng Lan Chat vì những ưu điểm sau: tiết kiệm băng thông ra internet, tốc độ cao hơn so với các ứng dụng chat online, khả năng bảo mật cao vì dữ liệu không được di chuyển ra ngoài internet phù hợp với các mô hình tổ chức cục bộ như các phòng máy trong các trường học, các phòng ban trong cơ quan xí nghiệp, công ty.

1. Hiện trạng cơ sở vật chất và con người
   1. Tin học

Hiện nay, việc sử dụng máy tính cá nhân rất phổ biến. Hầu như bất cứ lúc nào sinh viên cũng mang theo bên mình 1 chiếc laptop.

* 1. Con người

Hiện nay, Số lượng sinh viên trường truy cập rất nhiều vào internet thông qua mạng wifi trường để chat với nhau, chia sẻ nhau tài liệu học tập.

1. Yêu cầu sơ bộ về phần mềm
   1. Tin học

Phần mềm chạy được trên những máy tính có cấu hình từ trung bình trở lên, và không yêu cầu quá cao vào phần cứng.

Một mô hình mẫu:

* Processor: Intel Core i7-5500U, 2.40GHz (4CPUs)
* Memory: 4G
* Operating System: Windows 10 Education
  1. Con người
* Độ phức tạp: Thấp, dễ học.
* Ngôn ngữ chính: Tiếng việt
* Giao diện đẹp, quen thuộc dễ sử dụng

# PHÂN TÍCH YÊU CẦU PHẦN MỀM VÀ MÔ HÌNH HÓA

1. Yêu cầu phần mềm
   1. Yêu cầu chức năng

Bảng tổng hợp và định dạng các yêu cầu

|  |  |  |  |
| --- | --- | --- | --- |
| ***Định danh*** | | ***Độ Ưu tiên*** | ***Mô tả Yêu Cầu*** |
| YC\_1 | SignIn | 1 | Đăng nhập và hệ thống |
| YC\_2 | SignUp | 2 | Đăng kí tài khoản |
| YC\_3 | PrivateChat | 3 | Chat riêng giữa các người dùng |
| YC\_4 | GroupChat | 4 | Chat nhóm |
| YC\_5 | SendFile | 5 | Gửi file |
| YC\_6 | CreateGroup | 6 | Tạo nhóm chat |
| YC\_7 | HistoryChat | 7 | Xem lịch sử chat |
| YC\_8 | HistoryGroupChat | 8 | Xem lịch sử chat nhóm |
| YC\_9 | ShowAvatar | 9 | Hiển thị avatar của người dùng |
| YC\_10 | UpdateAvatar | 10 | Cập nhật avatar của người dùng |

* 1. Yêu cầu phi chức năng

|  |  |  |  |
| --- | --- | --- | --- |
| ***Định danh*** | | ***Độ Ưu tiên*** | ***Mô tả Yêu Cầu*** |
| YC\_A | Lưu trữ thời gian của các đoạn đối thoại | 1 | Các đoạn chat được lữu trữ cùng với thời gian chat để tăng lượng thông tin và độ tin cậy cho ứng dụng |
| YC\_B | Có thể theo dõi các kết nối và message từ giao diện Server | 1 | Server có các phần hiển thị các thông điệp giữa nó và các clients để tiện theo dõi |
| YC\_C | Ứng dụng nhẹ, tương thích và thân thiện với thiết bị | 1 | Ứng dụng cần đáp ứng được trên nhiều máy tính có giới hạn về phần cứng |
| YC\_D | Giao diện thân thiện, dễ sử dụng | 1 | Ứng dụng cần đáp ứng được về tính đơn giản và thân thiện với người dùng |
| YC\_E | Quản lý cơ chế đăng nhập, đăng xuất hợp lý | 1 | Khi người dùng đăng nhập, phải kiểm tra xem tài khoản đó có đang kết nối hay chưa, khi đăng xuất cũng có các tác vụ hợp lý. |

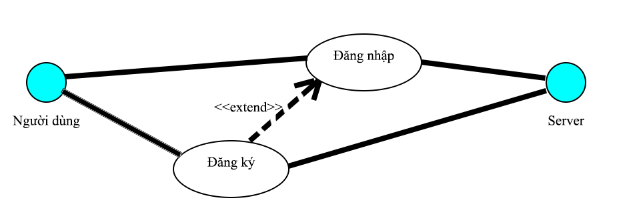
* 1. Bảng FURPS

|  |  |
| --- | --- |
| ***Tiêu chí chất lượng*** | ***Mô tả*** |
| Functionality | * Hệ thống hướng tới phục vụ người dùng đơn lẻ. |
| Usability | * Cung cấp khả năng liên lạc và trao đổi thông tin mà không cần có kết nối internet. * Giao diện thân thiện, đơn giản, dễ sử dụng. * Chức năng lên lịch du lịch hỗ trợ WYSIWIG editor và hệ thống template đẹp, bắt mắt. |
| Reliability | * Đã được kiểm tra và sửa lỗi trước khi sử dụng. * Các thông tin được lữu trữ và có lượng thông tin đáp ứng được các yêu cầu của ứng dụng. * Có khả năng kiểm soát các hoạt động thông qua Server. |
| Performance | * Ứng dụng nhẹ, thân thiện, không yêu cầu cấu hình cao. * Khả năng tra cứu tiện lợi. |
| Supportability | * Hỗ trợ Server trong quá trình kiểm soát các liên kết và thông tin. * Áp dụng các phương pháp thiết kế tiêu chuẩn để hỗ trợ người dùng trong quá trình sử dụng. |

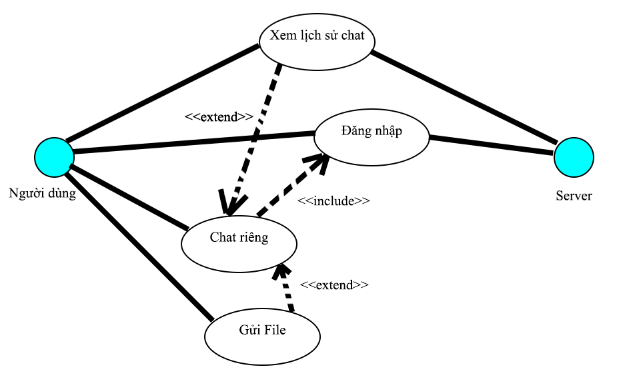
1. Mô hình hóa
   1. Các trường hợp sử dụng thông thường

|  |  |  |  |
| --- | --- | --- | --- |
| ***Use Case*** | ***Tên*** | ***Mô tả*** | ***Yêu cầu liên quan*** |
| UC-1 | Đăng nhập | Đăng nhập vào hệ thống | SignIn, SignUp |
| UC-2 | Chat riêng | Gửi tin nhắn qua lại cho nhau, gửi file, xem lịch sử chat | PrivateChat, SendFile, HistoryChat, SignIn |
| UC-3 | Chat nhóm | Tạo nhóm, chat trong nhóm, xem lịch sử chat nhóm | CreateGroup, GroupChat , HistoryGroupChat, SignIn |
| UC-4 | Hiển thị Avatar | Hiển thị Avatar trên form User và khung chat riêng | ShowAvatar, UpdateAvatar, SignIn |

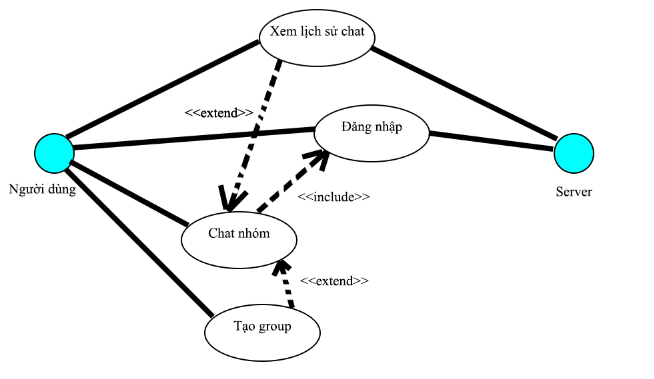
* + 1. Use Case 1 (Đăng nhập):



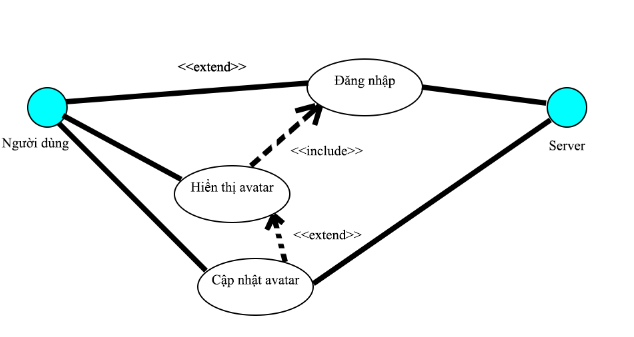
* + 1. Use Case 2 (Chat riêng):



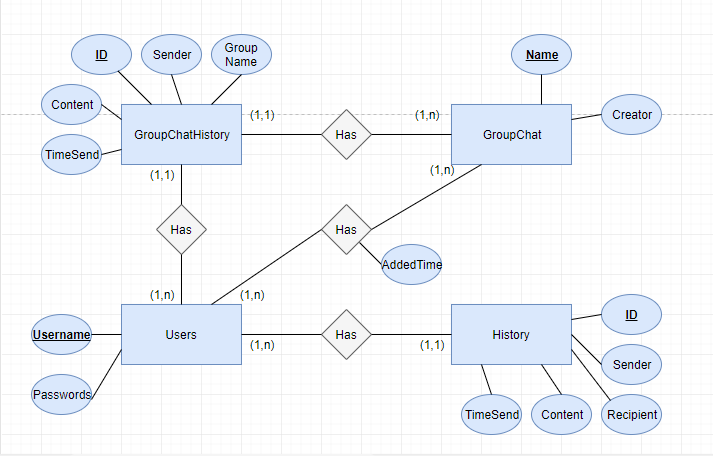
* + 1. Use Case 3 (Chat nhóm):



* + 1. Use Case 4 (Hiển thị Avatar):



* 1. Mô hình thực thể - mối kết hợp



User (Username, Passwords)

GroupChat (Name, Creator)

GroupChatDetail (GroupName, UserName, AddedTime)

History (ID, Sender, Receipient, Content, TimeSend)

GroupChatHistory (ID, Sender, GroupName, Content, TimeSend)

# THIẾT KẾ

1. Thiết kế kiến trúc phần mềm

Phần mềm sẽ được thiết kế theo mô hình MVC:

MVC là viết tắt của Model – View – Controller. Là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nói cho dễ hiểu, nó là mô hình phân bố source code thành 3 phần, mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

* Controller

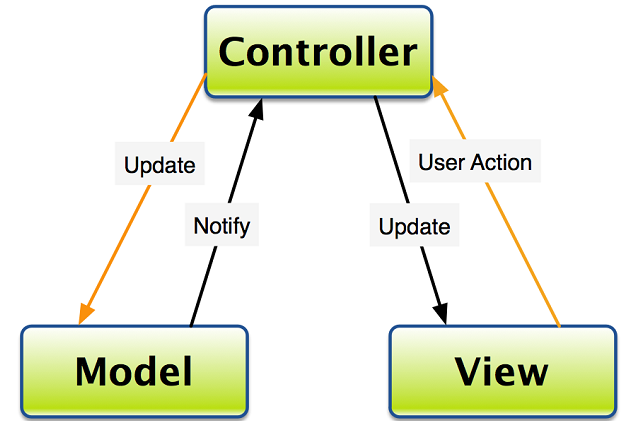
Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng… Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.

* Model

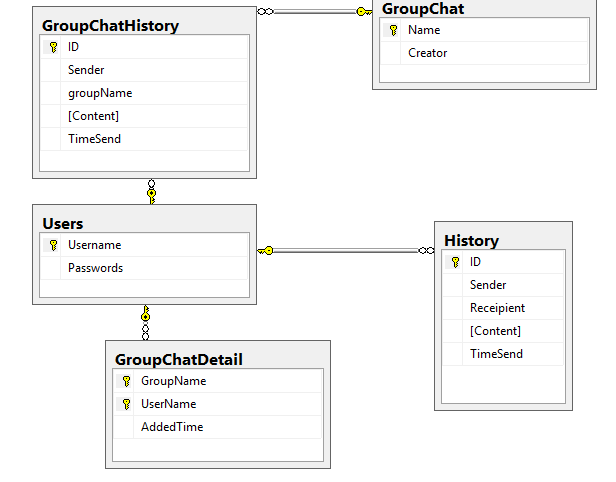
Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…

* View

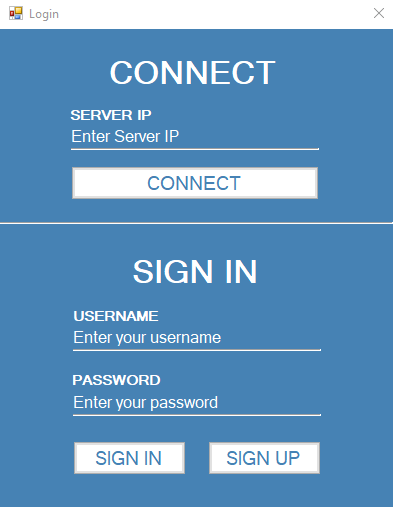
Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images… Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.



1. Thiết kế dữ liệu

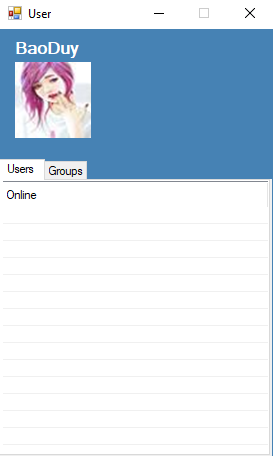
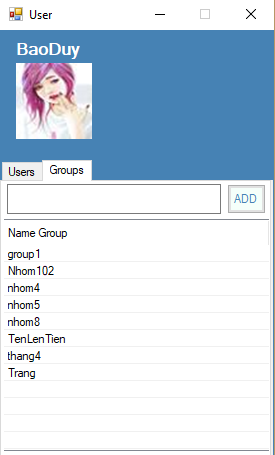


1. Thiết kế giao diện và thành phần xử lý giao diện
   1. Màn hình đăng nhập (Login Form):



Mô tả và cách thức hoạt động của giao diện:

* Trước tiên là khung để nhập vô IP server và nhấn nút (Connect) kết nối để có thể kết nối tới Server mà mình muốn kết nối vào. Nếu server đang mở thì sẽ xuất ra thông báo kết nối thành công ngay bước này, còn không sẽ xuất ra thông báo kết nối không thành công.
* Sau khi đã kết nối tới server giờ đây nếu chúng ta đã có tài khoản thì có thể đăng nhập bằng tài khoản (điền thông tin username, password) của mình và nhấn nút (Sign In) thì luồng dữ liệu thông tin đăng nhập của chúng ta sẽ được gữi đến server để kiểm tra, nếu đúng thì sẽ được đăng nhập vào màn hình User sai thì sẽ báo lỗi với người dùng.
* Nếu chúng ta chưa có tài khoản thì cũng sẽ điền thông tin vào 2 trường username, password nhưng sẽ nhấn nút Sign up để đăng kí và thông tin này cũng sẽ được gữi đến server để kiểm tra có tồn tại hay chưa, thông báo đăng nhập thành công nếu chưa tồn tại, thông báo trùng nếu đã tồn tại.
  1. Màn hình chính của người dùng (User Form):

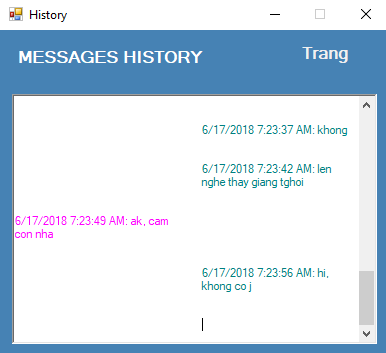
Mô tả và cách thức hoạt động của giao diện:

* Sau khi đăng nhập thành công, màn hình User hiện lên. Bên Users gồm 1 ListView lớn hiển thị danh sách những người đang online. Nếu bạn kích đúp chuột vào tên của người online thì 1 form chat sẽ hiện ra và bạn đã thực hiện tác vụ chat với người đó. Bên Groups gồm 1 ListView hiển thị danh sách các nhóm. Đồng thời còn có 1 ô textbox chức năng để tạo ra 1 nhóm mới. Nếu sau khi bạn nhập tên nhóm vào ô textbox đó và ấn nút button ADD thì thì bạn đã tạo ra được 1 nhóm. Mặc khác nếu bạn kích đúp chuột vào tên nhóm thì 1 form chat nhóm sẽ hiện ra để bạn thực hiện các tác vụ chat nhóm.
  1. Màn hình chat riêng (Private Chat Form):



Mô tả và cách thức hoạt động của giao diện:

* Sau khi chọn người chat với mình, màn hình sẽ hiện ra một cửa sổ như hình. Giao diện gồm có một vùng soạn thảo tin nhắn. Sau khi soạn thảo tin nhắn nhấn button SEND để gửi tin nhắn đi (hoặc có thể nhấn enter). Có một vùng richtexbox để hiển thị nội dung chat giữa hai bên. Để gửi file, ta nhấn vào nút Chosse File sau đó chọn file cần gửi và nhấn nút Send File để gửi. Nhấn vào nút History để mở màn hình lịch sử chat riêng để xem lịch sử chat với đối tượng.
  1. Màn hình lịch sử chat riêng (History Chat Form):



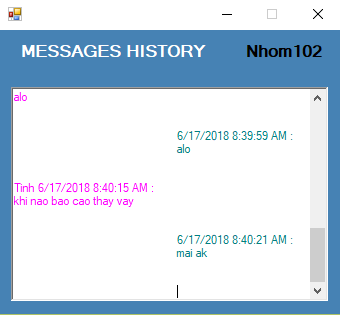
Mô tả và cách thức hoạt động của giao diện:

* Sau khi kích vào button History ở màn hình chat riêng thì một form lịch sử chat riêng sẽ xuất hiện. Form có một richtexbox chứa nội dung tin nhắn. đồng thời có một lable hiển thị tên của đối tượng.
  1. Màn hình chat nhóm (Group Chat Form):



Mô tả và cách thức hoạt động của giao diện:

* Khung chat chính của group chat gồm 2 danh sách (danh sách có chữ “BaoDuy” là danh sách các người dùng đang online nhưng chưa được add vào group còn phía dưới là các người dùng đã được add vào group), một khung để soan tin nhắn, nút SEND để gữi tin nhắn, nút QUIT để thoát ra khỏi group chat, HISTORY để xem lịch sử chat và nút ADD bên form Chat nhóm để thêm một người đang online nhưng chưa được vào group từ danh sách và khung để hiển thị tin nhắn giữa các thành viên trong group chat. Để thêm thành viên mới vào group chat cần phải thông qua các thành viên đã có trong group.
* Form còn có vùng soạn thảo tin nhắn. Nút SEND để gửi tin nhắn (hoặc nhấn enter), một richtexTextBox để hiện thị nội dung tin nhắn của nhóm.
  1. Màn hình lịch sử chat nhóm (History Group Chat Form):

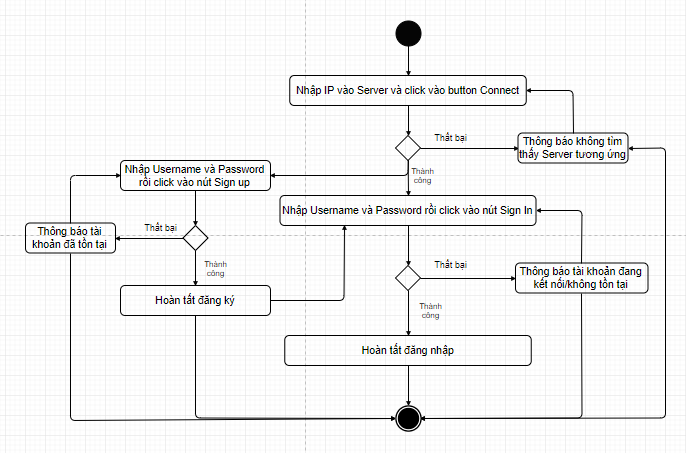


Mô tả và cách thức hoạt động của giao diện:

* Giao diện gồm có một RichTextBox để hiện lịch sử chat của nhóm, đồng thời có một lable để hiện thị tên của nhóm chat.

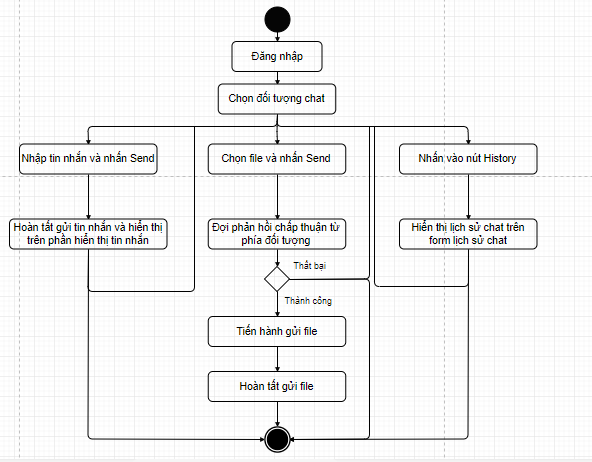
1. Thiết kế xử lý
   1. Đăng nhập:

Trong chức năng đăng nhập, người dùng sẽ tiến hành nhập thông tin vào trường Server IP rồi nhấn Connect hoặc ENTER để kết nối đến Server. Nếu kết nối thành công thì vùng nhập Server IP và button Connect sẽ bị disable. Sau khi kết nối thành công, người dùng sẽ nhận được một thông báo cho biết đã kết nối thành công và có thể nhập thông tin vào các trường Username và Password. Nếu người dùng nhấn Sign In hoặc ENTER, Server sẽ kiểm tra xem tài khoản có đang đăng nhập vào hệ thống và tài khoản có tồn tại hay không, nếu tài khoản chưa đăng nhập và có tồn tại thì người dùng sẽ đăng nhập thành công và màn hình Login sẽ đóng lại, đồng thời màn hình User sẽ mở ra; nếu người dùng nhấn Sign Up thì Server sẽ kiểm tra xem tài khoản có tồn tại hay chưa, nếu tài khoản chưa tồn tại thì Server sẽ tạo tài khoản mới ứng với thông tin ở các trường Username và Password đồng thời thông báo lại cho người dùng biết đã đăng ký thành công.



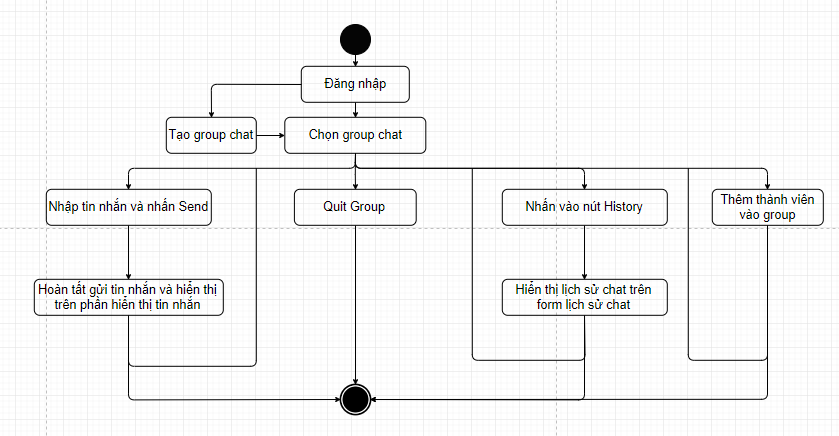
* 1. Chat riêng:

Sau khi người dùng đăng nhập thành công, Ở form User người dùng có thể chuyển sang tab Users và nhấn double click vào tên của các đối tượng đang online để mở ra màn hình chat riêng. Trên màn hình chat riêng, người dùng có thể nhập tin nhắn vào phần textbox để nhập tin nhắn và nhấn Send hoặc ENTER để gửi tin nhắn, lúc này trên màn hình sẽ hiển thị nội dụng tin nhắn vừa gửi, nếu client kia gửi tin nhắn lại thì cũng được hiển thị trên phần hiển thị tin nhắn; người dùng có thể click vào nút Choose File, lúc này một dialog chọn File sẽ được mở ra và người dùng có thể chọn 1 file, sau khi chọn xong nhấn vào nút Send File để gửi file, lúc này người dùng chờ đối tượng xác nhận nhận file, nếu xác nhận thành công thì file sẽ được gửi thông qua Server, nếu đối tượng từ chối thì sẽ không có thêm hành vi nào xảy ra; người dùng có thể click vào nút History để màn hình hiển thị ra phần lịch sử chat với đối tượng trên form History mới.



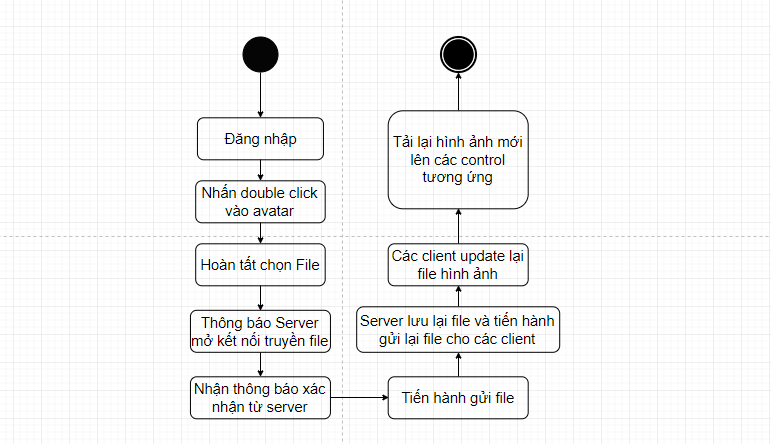
* 1. Chat nhóm:

Sau khi người dùng đăng nhập thành công, Ở form User người dùng có thể chuyển sang tab Groups và nhập thông tin vào phần textbox tạo group mới rồi nhấn nút ADD để tạo group mới, hoặc có thể nhấn double click vào tên group có trong danh sách. Lúc này, màn hình chat nhóm hiển thị, người dùng có thể nhập thông tin vào phần textbox nhập tin nhắn và nhấn nút Send hoặc ENTER để gửi tin nhắn, tin nhắn sau khi được gửi sẽ hiển thị trên phần hiển thị nội dung tin nhắn; người dùng có thể nhấn vào nút History để xem lịch sử chat của nhóm trên form History mới tạo ra; người dùng có thể nhấn vào nút Quit để thoát group.



* 1. Hiển thị avatar:

Sau khi người dùng đăng nhập, có thể nhấn double click vào phần hình ảnh avatar, lúc này một dialog cho phép chọn các file dạng hình ảnh có các kiểu đuôi (.jpg, .bmp, .png, .jpeg) sẽ được mở ra, sau khi người dùng chọn xong thì sẽ gửi một thông báo đến Server yêu cầu mở ra một luồng gửi nhận file. Sau khi nhận được thông báo xác nhận từ Server thì File sẽ được gửi cho Server, Server lưu lại file này rồi thông báo đến cho các client khác và mở kết nối gửi nhận file và đợi nhận được thông báo xác nhận từ các client thì Server tiến hành gửi file. Các client nhận được file, lưu lại trong phần hình ảnh avatar của các client khác và cập nhật lại hình ảnh trên các control có liên quan.



# KIỂM THỬ

Quá trình kiểm thử được diễn ra xuyên suốt trong quá trình xây dụng và phát triển. Áp dụng các phương pháp kiểm thử Unit test để kiểm tra sự chính xác trong từng module và Integration test để kiểm tra sự chính xác về logic trên giao diện và tính năng của chương trình.

# TỔNG KẾT

## 5.1. Tổng Kết

Phần mềm bước đầu đã được phát triển đi đúng theo dự tính và kế hoạch. Tuy gặp khá nhiều khó khăn trong vấn đề tiếp cận với công nghệ lập trình mới, nhóm vẫn đảm bảo được các tiến độ làm việc do đã đặt ra. Thường xuyên liên lạc và trao đổi trong quá trình thực hiện đồ án qua công cụ chia sẻ source code Github và mạng xã hội Facebook.

## 5.2. Nhận Xét & Đánh Giá

Trong thời gian có hạn, việc hoàn thành dự án đã vượt qua được nhiều thử thách nhất định nhưng phần mềm vẫn còn nhiều điểm có thể cải tiến nhằm nâng cao chất lượng phục vụ và trải nghiệm của người dùng:

* Cải thiện giao diện người dùng.
* Cài đặt và hoàn thiện nhiều tính năng nâng cao.
* Mở rộng dữ liệu trong database hoặc hướng đến sử dụng dịch vụ cloud database.
* Kiểm soát toàn bộ các bug, lỗi có thể xảy ra.
* Mở rộng thêm nhiều tính năng mới.

5.1. Phân chia công việc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Công việc** | **Duration (days)** | **Assign To** | **% Finished** |
| 1 | Tìm hiểu sơ bộ & đăng ký đồ án |  | Duy | 100% |
| 2 | Tìm hiểu công nghệ liên quan |  | Duy, Trang, Tình | 100% |
| **I. VIẾT BÁO CÁO** | | | | |
| 3 | **Chương 1 – Hiện trạng** |  |  |  |
| 4 | 1.1. Hiện trạng vấn đề |  | Duy, Trang, Tình | 100% |
| 5 | 1.2. Hiện cơ sở vật chất và con người |  | Trang, Tình | 100% |
| 6 | 1.3. Yêu cầu về phần mềm |  | Duy, Tình | 100% |
| 7 | **Chương 2: Phân tích** |  |  |  |
| 8 | 2.1. Yêu cầu phần mềm |  | Duy, Trang, Tình | 100% |
| 9 | 2.2. Mô hình hóa |  | Duy, Trang, Tình | 100% |
| 10 | **Chương 3: Thiết kế** |  |  |  |
| 11 | 3.1. Thiết kế kiến trúc phần mềm |  | Duy, Tình | 100% |
| 12 | 3.2. Thiết kế dữ liệu |  | Duy, Trang, Tình | 100% |
| 13 | 3.2.1. Thiết kế dữ liệu bộ nhớ chính |  | Duy, Trang | 100% |
| 14 | 3.3. Thiết kế giao diện và thành phần xử lí của giao diện |  | Duy, Trang, Tình | 100% |
| 19 | **Chương 4: Kiểm thử** |  |  |  |
| 20 | 5.1. Unit Test |  | Duy, Trang | 75% |
| 21 | 5.2. Integration Test |  | Duy, Tình | 75% |
| 23 | **Chương 5: Kết luận** |  |  |  |
| 24 | **Tài liệu tham khảo** |  | Duy, Tình | 100% |
| **II. LẬP TRÌNH** | | | | |
|  | 1. Database | 10 | Duy, Trang, Tình | 90% |
|  | 2. Giao diện | 20 | Duy, Tình | 90% |
|  | 3. Tính năng | 45 | Duy, Trang, Tình | 90% |
| **III. KIỂM THỬ** | | | | |
|  | 1. Unit Test |  | Duy, Trang |  |
|  | 2. Integration Test |  | Duy, Tình |  |
| **IV. NỘP BÁO CÁO TIẾN ĐỘ & SẢN PHẨM** | | | **Tiến độ sản phẩm** | |
|  | **Ngày 18 tháng 06 năm 2018** | | Báo cáo cuối kì. | |

**TÀI LIỆU THAM KHẢO**

<https://stackoverflow.com/>

<https://www.codeproject.com/>

<https://github.com/>

<https://socketprogramming.blogspot.com/>

https://msdn.microsoft.com/en-us/dn308572.aspx