**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 2 CƠ SỞ TPHCM**

bÁO CÁO Thực tập tốt nghiệp



Sinh viên: Trần Nguyễn Huy Đạt

Lớp: D14CQCP-01

Mã số sinh viên: N14DCCN235

Dự án**:** Chat live posting

Giáo viên hướng dẫn:

TS.GVC. VÕ XUÂN THỂ

**LỜI CÁM ƠN**

*Lời đầu tiên, em xin chân thành cảm ơn đến Thầy Cô Học Viện Công Nghệ Bưu Chính Viễn Thông, các Thầy Cô Khoa Công Nghệ Thông Tin đã tận tình truyền dạy những kiến thức quý báu cho em trong suốt thời gian học tại trường, nguồn kiến thức đó sẽ là tài sản vô giá trong tương lai của em.*

*Đặc biệt, em xin chân thành cảm ơn sâu sắc sự hướng dẫn tận tình của Thầy Võ Xuân Thể đã cho em những ý kiến quý báu và tận tình chỉ bảo em trong suốt quá trình hoàn thành đề tài thực tập này.*

*Một lần nữa, em xin gửi đến thầy cô Học Viện Công Nghệ Bưu Chính Viễn Thông lời chúc sức khỏe và hạnh phúc.*

*Em xin chân thành cảm ơn!*

***Thành phố Hồ Chí Minh, 01/08/2018***

*Sinh viên thực hiện*

*Trần Nguyễn Huy Đạt*

**Mục lục**

[**Mục lục** 2](#_Toc521066843)

[DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH 3](#_Toc521066844)

[DANH MỤC CÁC ký hiệu và chữ viết tắt 4](#_Toc521066845)

[III. MỞ ĐẦU 5](#_Toc521066846)

[Lịch sử sửa đổi 7](#_Toc521066847)

[1. GIỚI THIỆU ĐỀ TÀI VÀ CÔNG NGHỆ 7](#_Toc521066848)

[1.1 GIỚI THIỆU ĐỀ TÀI 7](#_Toc521066849)

[1.1.1 Giới thiệu 7](#_Toc521066850)

[1.1.2 Công nghệ sử dụng 7](#_Toc521066851)

[1.2 Giới thiệu công nghệ 8](#_Toc521066852)

[1.2.1 Các tính năng mới trong HTML5 8](#_Toc521066853)

[1.2.2 ASP.NET Boilerplate 8](#_Toc521066854)

[1.2.3 Angularjs 8](#_Toc521066855)

[1.2.4 Signalr 9](#_Toc521066856)

[1.2.5 Entity Framework 9](#_Toc521066857)

[1.2.6 Version control 9](#_Toc521066858)

[2. Phân tích và thiết kế hệ thống 10](#_Toc521066859)

[2.1 Mô hình dữ liệu thực thể kết hợp (ERD - Entity Relationship Diagram) 10](#_Toc521066860)

[2.1.1 Danh sách thực thể 10](#_Toc521066861)

[2.1.2 các chức năng chính của hệ thống và xây dựng các use case chi tiết 13](#_Toc521066862)

[3. Giao diện 16](#_Toc521066863)

[3.1 1.3 Màn hình 16](#_Toc521066864)

[Mô tả chức năng 20](#_Toc521066865)

[4. Xây dựng chương trình 21](#_Toc521066866)

[4.1 Cấu trúc phân chia công việc (WBS) 21](#_Toc521066867)

[4.2 Kế hoạch dự định 21](#_Toc521066868)

[4.2.1 DANH SÁCH CÔNG VIỆC 21](#_Toc521066869)

[4.3 Thời gian thực tế 22](#_Toc521066870)

[5. kết luận 22](#_Toc521066871)

[5.1 Kết quả đạt được 22](#_Toc521066872)

[5.2 Hạn chế, vướng mắc 22](#_Toc521066873)

[5.3 Hướng phát triển, khắc phục hạn chế 23](#_Toc521066874)

**DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH**

[Hình 1. THỰC THỂ THREAD 10](#_Toc521066824)

[Hình 2. THỰC THỂ USER 11](#_Toc521066825)

[Hình 3. THỰC THỂ REPLY 12](#_Toc521066826)

[Hình 4. THỰC THỂ CHATMESSAGE 12](#_Toc521066827)

[Hình 5. Mô hình liên kết thực thể 13](#_Toc521066828)

[Hình 6.Biểu đồ usecase đăng ký đăng nhập 14](#_Toc521066829)

[Hình 7.Biểu đồ usecase tương tác thread 14](#_Toc521066830)

[Hình 8.Biểu đồ usecase tương tác reply 15](#_Toc521066831)

[Hình 9.Biểu đồ usecase quản lý user 15](#_Toc521066832)

[Hình 10. Màn hình hiển thị thread 17](#_Toc521066833)

[Hình 11. Màn hình tạo thead 17](#_Toc521066834)

[Hình 12. Màn hình sửa thead 18](#_Toc521066835)

[Hình 13. Màn hình xem reply/chi tiết thread 18](#_Toc521066836)

[Hình 14. Màn hình tạo reply 19](#_Toc521066837)

[Hình 15. Màn hình Delete thread 19](#_Toc521066838)

[Hình 16. khung chat 20](#_Toc521066839)

[Hình 17. cấu trúc thực hiện chương trình 21](#_Toc521066840)

**DANH MỤC CÁC ký hiệu và chữ viết tắt**

|  |  |  |
| --- | --- | --- |
| TT: |  | Thông tin thêm |
| HTTP | HyperText Transfer Protocol |  |
| SQL | Structure Query Language |  |
| ERD | Entity Relationship Diagram | Mô hình thực thể kết hợp |
| ABP  WBS | ASP.NET Boilerplate  Work Breakdown Structure |  |

# III. MỞ ĐẦU

Hiện nay với mạng internet ngày càng mở rộng thì nhu cầu giao tiếp của con người ngày càng phát triển. Và với sự xuất hiện của các mạng xã hội như twitter, facebook, … thì số người sử dụng các diễn đàn truyền thống ngày càng thu nhỏ lại.

Tuy nhiên không có nghĩa là các diễn đàn là không cần thiết nữa. Trái lại lại xuất hiện những loại diễn đàn khác, các diễn đàn này khá tổng hợp hợp về nội dung, thiên về chat/forum hỗn hợp và ưu tiên người dùng hơn diễn đàn truyền thống.

Cụ thể ví dụ là 2 trang web khá là nổi tiếng theo dạng này.

Thứ nhất là <https://www.reddit.com> đang đứng hạng 6 top các website có lượt truy cập cao nhất ở Mỹ [[1]](#footnote-1)

Thứ hai là <https://www.4chan.org/> cũng từng nằm trong top 100 các website có lượt truy cập cao nhất ở Mỹ, hiện tại đang ở hạng 150 [[2]](#footnote-2)

|  |  |
| --- | --- |
|  |  |

**Lịch sử sửa đổi**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ngày áp dụng** | **Phiên bản** | **Phần sửa đổi** | **Nội dung sửa** | **Người sửa** |
| 15/6 | 1.0 | Tạo mới |  |  |
| 13/7 | 1.1 | Update |  |  |
| 30/7 | 1.2 | Realtime |  |  |

# GIỚI THIỆU ĐỀ TÀI VÀ CÔNG NGHỆ

## GIỚI THIỆU ĐỀ TÀI

### Giới thiệu

Qua ví dụ ở lời mở đầu có thể thấy các dạng diễn đàn tổng hợp này được khá nhiều người sử dụng. Tuy nhiên ở Việt Nam ta theo em biết vẫn chưa có một website nào kiểu như vậy.

Đa số vẫn còn là các diễn đàn truyền thống, mỗi diễn đàng chỉ tập trung vào một hoặc một số chủ đề. Vì thế kết hợp nhiều công nghệ như angularJs, EntityFramework, ASP.Net Zero, SQL Server em muốn làm nên một website như vậy

Về đối tượng và sự hữu dụng của đề tài theo em nghĩ trên thực tế do sự tổng hợp của loại site này nên đối tượng sử dụng sẽ trải rộng, ai cũng sẽ có phần. Và do tính chất không tập trung quá vào cá nhân như các loại mạng xã hội nên thiết nghĩ người dùng sẽ tập trung vào việc phát triển, thảo luận hơn là vun đắp cho trang cá nhân của mình.

Ngoài ra đề tài còn tích hợp thêm tính năng realtime. Cả hai ví dụ được nêu ra ở trang 5 đều không có tích hợp realtime nên người dùng phải tự refresh để xem reply/thead mới. Em muốn khắc phục điều này.

### Công nghệ sử dụng

* HTML 5 & CSS 3
* ASP.NET Boilerplate
* AngularJs
* Signalr
* Entity Framework

## Giới thiệu công nghệ

### Các tính năng mới trong HTML5

HTML5 cung cấp các công cụ quản lý dữ liệu, đồ họa, video, và âm thanh có hiệu quả. Nó tạo điều kiện cho sự phát triển của các ứng dụng giữa các trình duyệt với nhau cho trang web cũng như cho các thiết bị di động. HTML5 là một trong những công nghệ thúc đẩy những cải tiến trong các dịch vụ điện toán đám mây di động, vì nó tính đến tính linh hoạt rộng hơn, cho phép phát triển các trang web thú vị và có khả năng tương tác. Nó cũng đưa vào thẻ và các cải tiến mới, bao gồm cấu trúc thu nhỏ, các nút điều khiển của biểu mẫu, các API, đa phương tiện, hỗ trợ cơ sở dữ liệu, và tốc độ xử lý nhanh hơn.

### ASP.NET Boilerplate

ASP.NET Boilerplate (**ABP**) là một **application framework** [open source](https://github.com/aspnetboilerplate/aspnetboilerplate)  **và được viết tài liệu rõ ràng**. Không chỉ là framework, ABP còn **cung cấp mô hình kiến trúc** dựa trên **Domain Driven Design**. ABP hỗ trợ cả ASP.Net core & EF Core và ASP.Net MVC & EF 6.

Đề tài này sử dụng ASP.Net MVC và EF 6.

Ví dụ một số lợi ích của việc sử dụng ABP

* Repository: ABP tạo repository tự động cho mỗi entity trong đó có các hàm có ích như FirstOrDefault
* Validation: ABP tự động check xem input có null hay không hoặc input có hợp lệ dựa trên data annotation của các thuộc tính. Nếu không hợp lệ ABP sẽ trả về exception tùy theo và tất cả đều là clientside
* Audit logging: Ai, browser, địa chỉ IP, calling service, method, parameters, calling time, execution duration và một số thông tin khác đều được tự động lưu trữ với mỗi request.
* Auto Mapping: ABP cung cấp hàm MapTo để map input sang entity và ngược lại.
* Logging: Ta có thể sử dụng object logger trong log4net để lưu log.
* Exception Handling: ABP hỗ trợ rất nhiều exception tự đồng và gần như ta không phải tự bắt lỗi thủ công nữa.
* Và nhiều lợi ích khác nữa

### Angularjs

Angularjs là front-end framework dựa trên javascript và được tích hợp sẵn trong ABP. Trong đề tài này chủ yếu sử dụng các directives của angularjs. Directives là một thành phần mở rộng HTML, hay nói cách khác là các thuộc tính (properties) của các thẻ HTML mà Angular nó định nghĩa thêm.

Ngoài ra còn có scope. **$scope là một object** (đối tượng) có nhiệm vụ giao tiếp dữ liệu giữa controller và view của ứng dụng. Nó sẽ thực hiện dưới dạng biểu thức, nghĩa là ở model sẽ được khai báo đúng với quy cách thì đối tượng scope sẽ truyền hành động (function) hoặc dữ liệu tương ứng và ta có thể truyền các sự kiện thông qua đối tượng này.

### Signalr

SignalR là một thư viện phần mềm cho ASP.NET cho phép các hàm, code từ server gửi thông báo không đồng bộ xuống cho web app client-side. Thư viện này cho phép người lập trình thêm real-time vào trong application. SignalR sử dụng nhiều cách khác nhau để truyền dữ liệu và tự đồng chọn cách tốt nhất. Thường sử dụng nhất là WebSocket, một API của HTML5 hỗ trợ giao tiếp hai chiều giữa server và client.

SignalR còn cung cấp một số API cấp cao cho phép server gọi method từ client trong app ASP.Net cùng với một số móc để quản lý kết nổi như onConnect, onDisconnec, grouping, …

### Entity Framework

Entity Framework (EF) là một framework open source object-relational mapping (ORM) cho ADO.NET, một phần của .NET Framework. Nó cho phép người lập trình sử dụng dữ liệu sử dụng các đối tượng của các class tương ứng mà không phải chỉnh sửa dựa theo các bảng hay các dòng mà dữ liệu được lưu trữ. Với Entity Framework, developers có thể sử dụng dữ liệu ở cấp cao hơn và có thể tạo, bảo trì/bảo dưỡng data-oriented sử dụng ít code hơn so với applications truyền thống.

Trong đề tài này sử dụng EF-Code-First. Code First chủ yếu là hữu ích **Domain Driven Design (DDD)**. Trong cách tiếp cận Code First, devs có thể tập trung vào việc thiết kế domainvà bắt đầu tạo ra các lớp theo yêu cầu của Domain của bạn chứ không phải thiết kế cơ sở dữ liệu trước rồi sau đó tạo ra các lớp phù hợp với thiết kế cơ sở dữ liệu đó. Code First API sẽ tạo ra cơ sở dữ liệu dựa trên các lớp thực thể và lớp cấu hình đã tạo.

### Version control

Hệ thống quản lý phiên bản (còn viết là VCS từ *version control systems*) là một hệ thống lưu giữ các phiên bản của mã nguồn của sản phẩm phần mềm, giúp các lập trình viên có thể dễ dàng lấy lại phiên bản mong muốn.

Hiện tại đề tài có sử dụng version control là github để lưu trữ và push update.

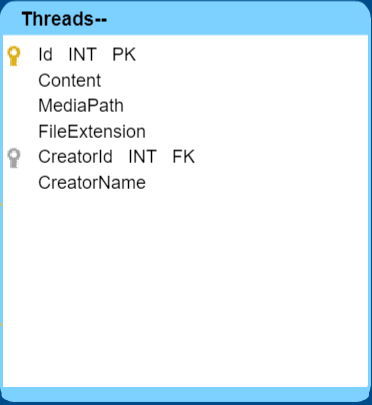
Vẫn còn rất nhiều chức năng khác của git vẫn chưa được ứng dụng như pull request, fork, branch, … Nhưng đề tài này vẫn là bước khởi đầu quý báu để cho em tập làm quen với git và verson control.

# Phân tích và thiết kế hệ thống

## Mô hình dữ liệu thực thể kết hợp (ERD - Entity Relationship Diagram)

### Danh sách thực thể

* Thread



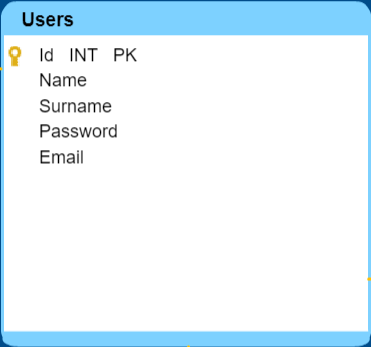
Hình 1. THỰC THỂ THREAD

Trong đó Id là Id của thread.

Các thuộc tính Content, MediaPath và FileExtension thuộc kiểu string và chứa nội dung thread, đường dẫn file media và đuôi file.

CreatorId và CreatorName là Id và tên của người tạo thead. Thuộc tính CreatorId khóa ngoại đến table User.

* User



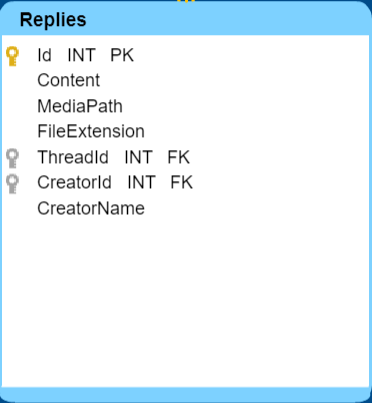
Hình 2. THỰC THỂ uSER

Ta có các trường Id là Id của User. Name và Surname là tên của user.

Password kiểu string chứa hash của account.

Email chứa email của user.

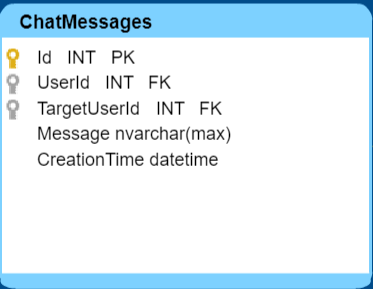
* Reply



Hình 3. THỰC THỂ rEPLY

Table Reply cũng chứa các thuộc tính tương tự như table Thread. Ngoài ra có thêm thuộc tính ThreadId khóa ngoại đến Id của table Thread.

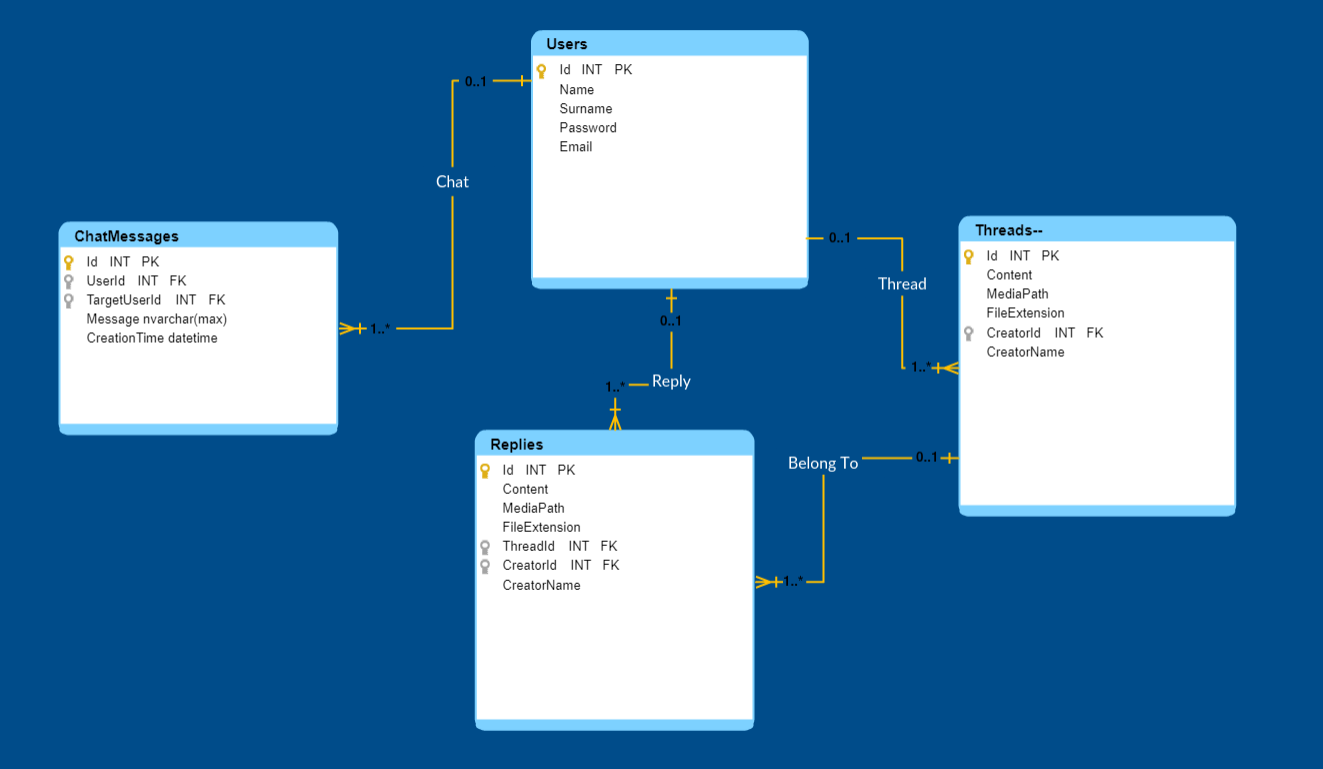
* ChatMessage



Hình 4. THỰC THỂ cHATmESSAGE

Table ChatMessage gồm Id của Message tương ứng. UserId là Id của người gửi, TargetUserId là Id của người nhận. Message chứa nội dung của tin nhắn.

* Mô hình liên kết thực thể



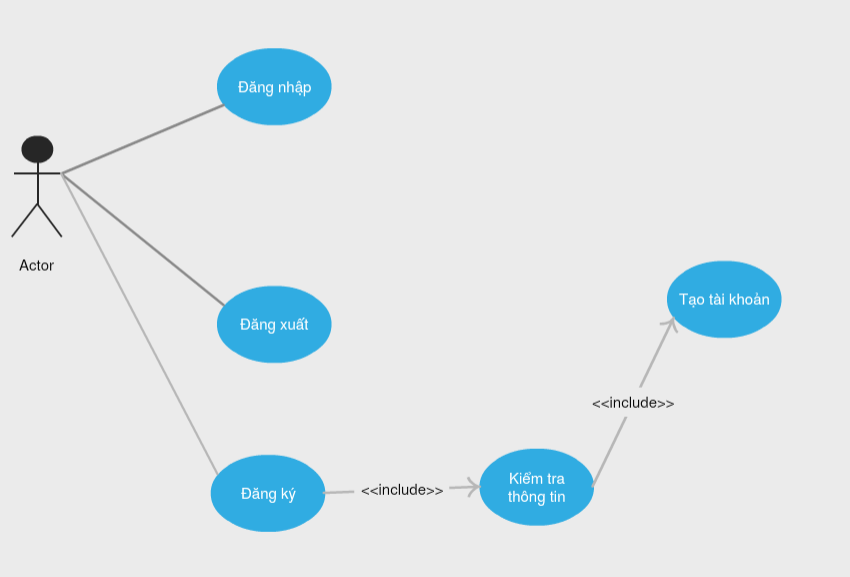
Hình 5. Mô hình liên kết thực thể

### các chức năng chính của hệ thống và xây dựng các use case chi tiết

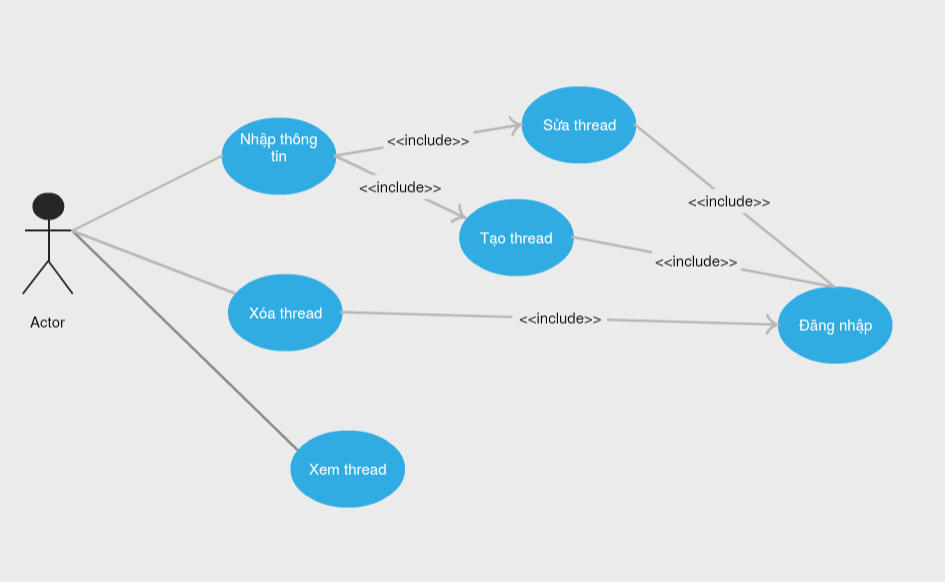
* Đối với người dùng

Hệ thống có các chức năng hỗ trợ và giải quyết nhu cầu của người dùng:

* Xem tất cả thread đang hiện có: Tin tức về thread, người đăng, nội dụng, có kèm theo media hay ko.
* Đăng ký thành viên của hệ thống để có thể đăng bài, tham gia bình luận- trao đổi ý kiến.

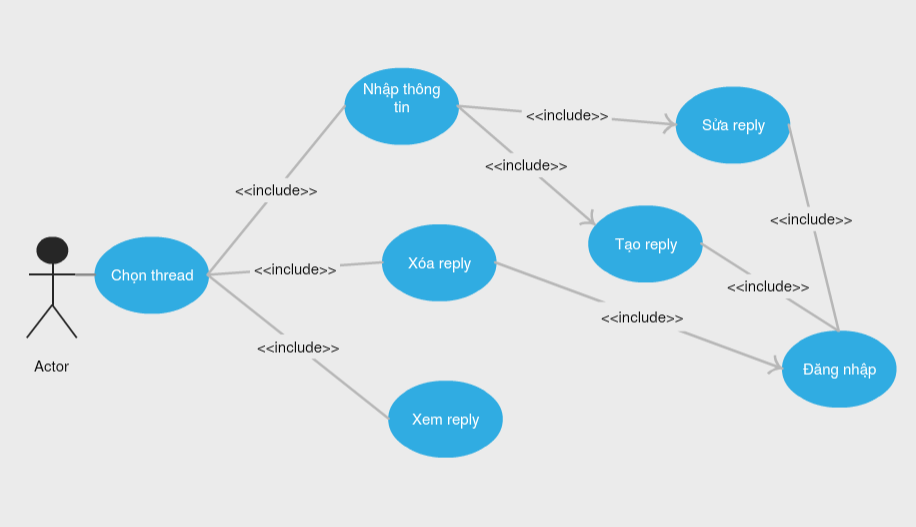


Hình 6.Biểu đồ usecase đăng ký đăng nhập



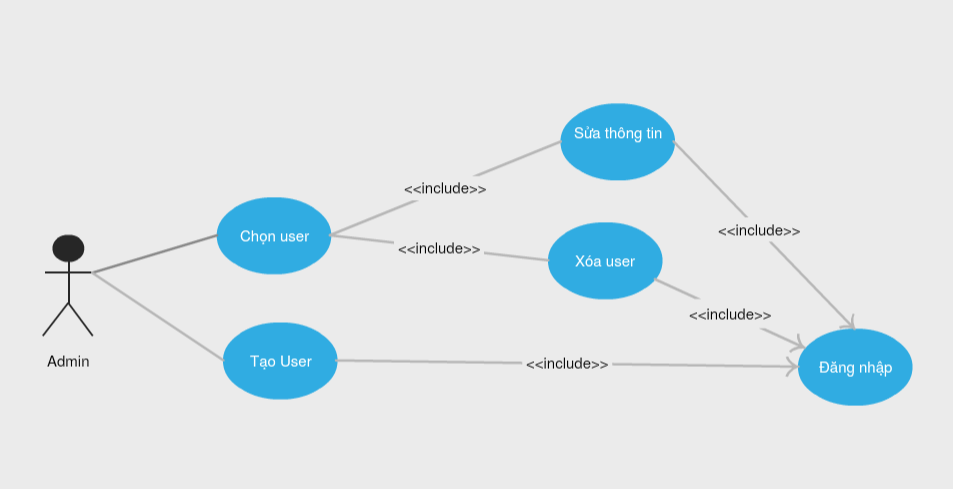
Hình 7.Biểu đồ usecase tương tác thread

Tương tự với thread user cũng tương tác được với reply trong thread như vậy



Hình 8.Biểu đồ usecase tương tác reply

Ngoài ra người quản trị còn có thể quản lý các user khác như



Hình 9.Biểu đồ usecase quản lý user

Và các user có thể chat với nhau nữa.

# Giao diện

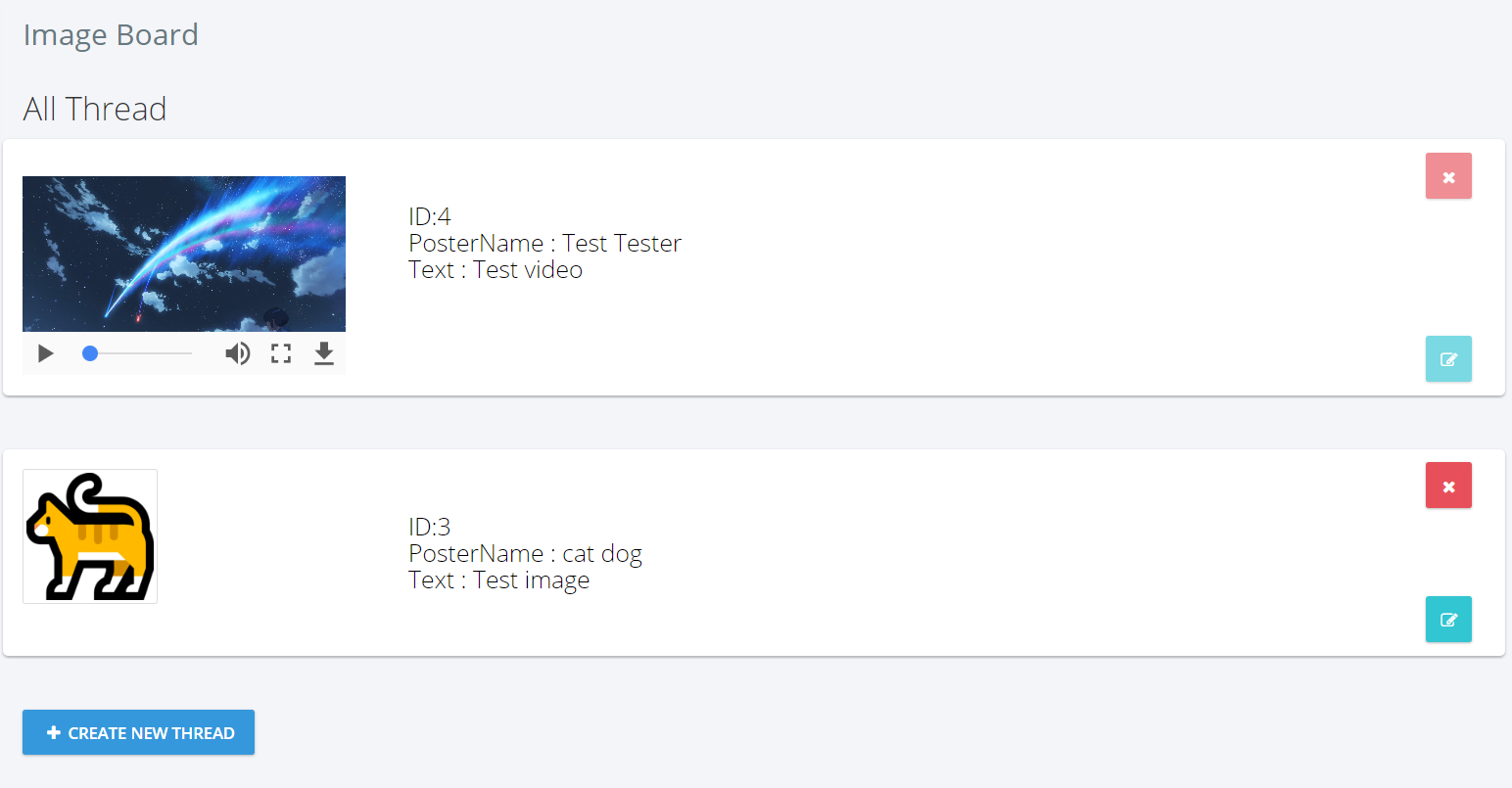
## 1.3 Màn hình

Gồm có các màn hình:

* Hiện thị toàn bộ thread, cho phép thêm xóa sửa upload
* Hiện thị toàn bộ reply khi nhấn vào thread tương ứng, cho phép thêm xóa sửa upload
* Slidebar menu cho việc chat

a) Màn hình chính liệt kê các thread.

Màn hình này sẽ liệt kê toàn bộ các thread hiện có.



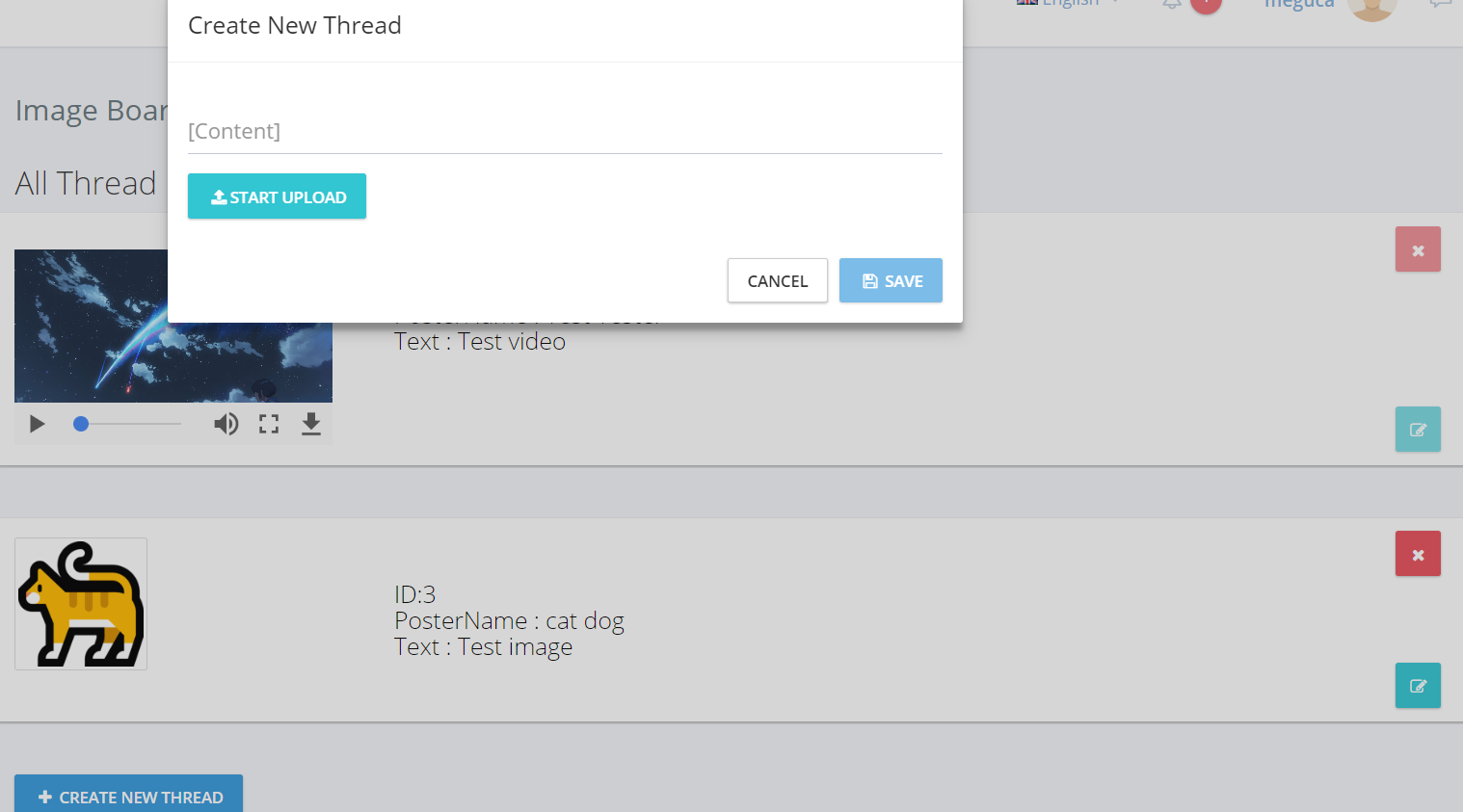
Hình 10. Màn hình hiển thị thread

Sau khi đăng nhập thì người dùng sẽ có quyền xem các thread hiện có, tạo thread mới. Cùng với đó là các nút cho phép edit và delete.

b) Màn hình tạo thread

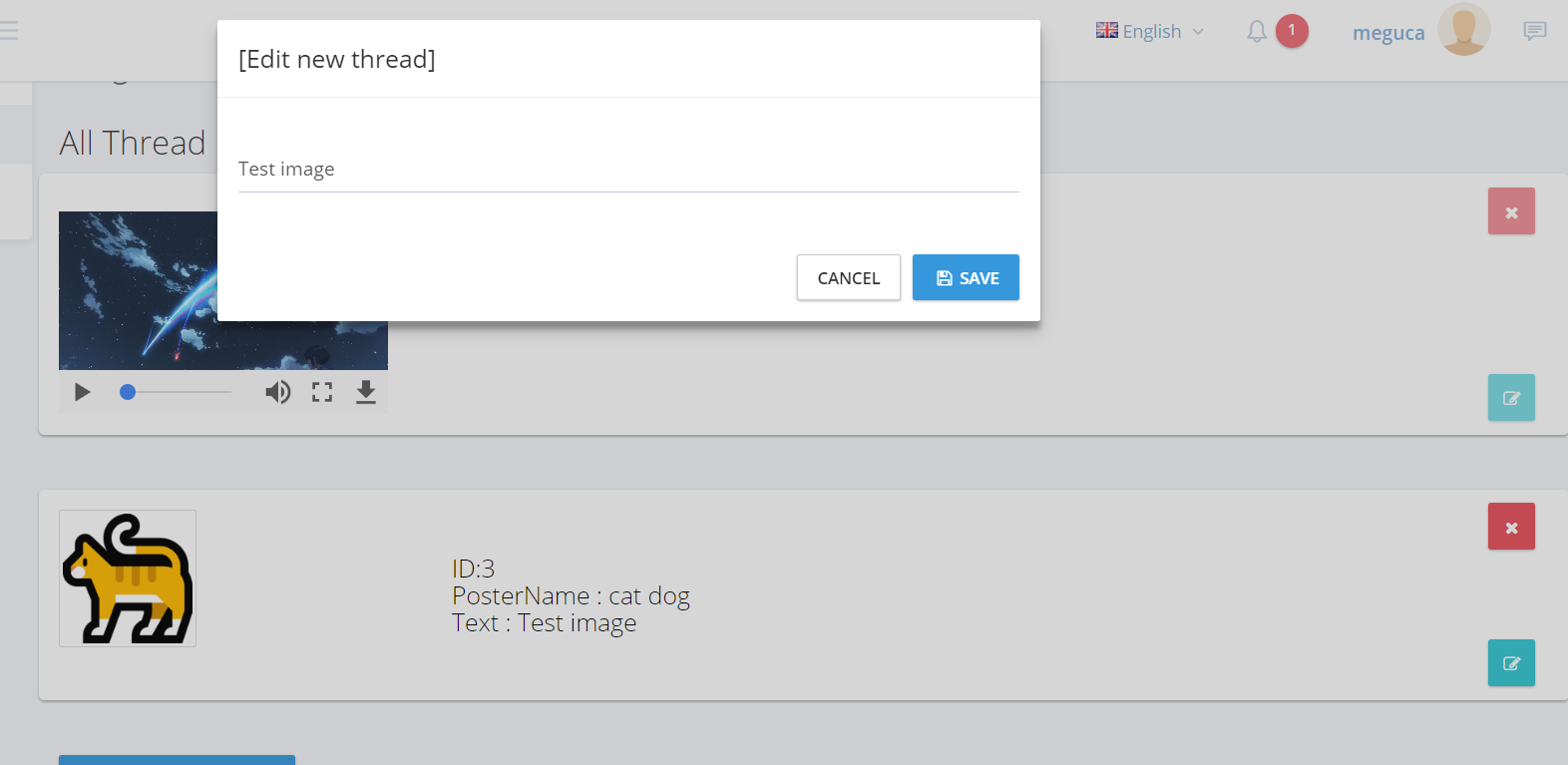
Nhấn vào Nút Create New Thread thì 1 modal sẽ hiện ra cho phép người dùng nhập nội dung thread. Người dùng cũng có thể upload hình hoặc video (Không bắt buộc).

Nội dung là 1 string có giới hạn 168 ký tự. Tương ứng với thuộc tính Content trong table Thread.



Hình 11. Màn hình tạo thead

Khi người dùng tạo thread click vào nút sửa thì sẽ hiện ra modal sửa



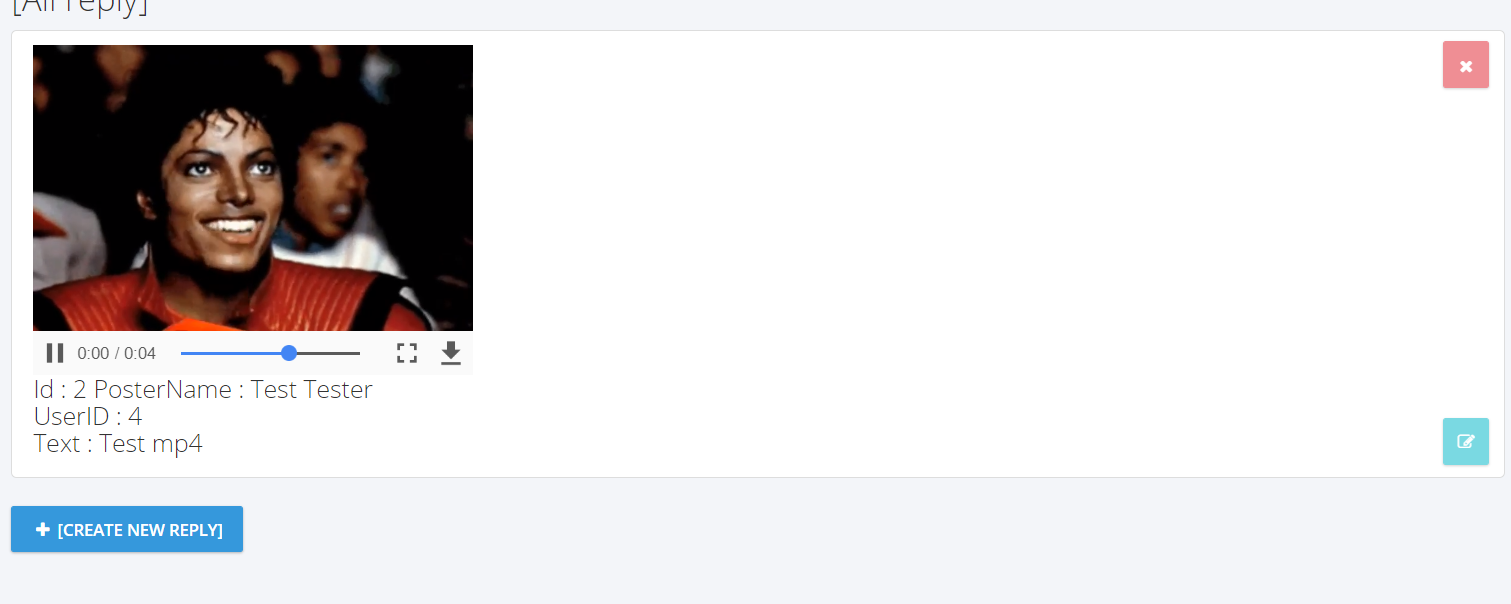
Hình 12. Màn hình sửa thead

Người dùng chỉ được sửa nội dung. Không được sửa hình.

c) Màn hình chi tiết trong thread

Khi nhấn vào 1 thread bất kì thì sẽ đi vào trong thread đấy.

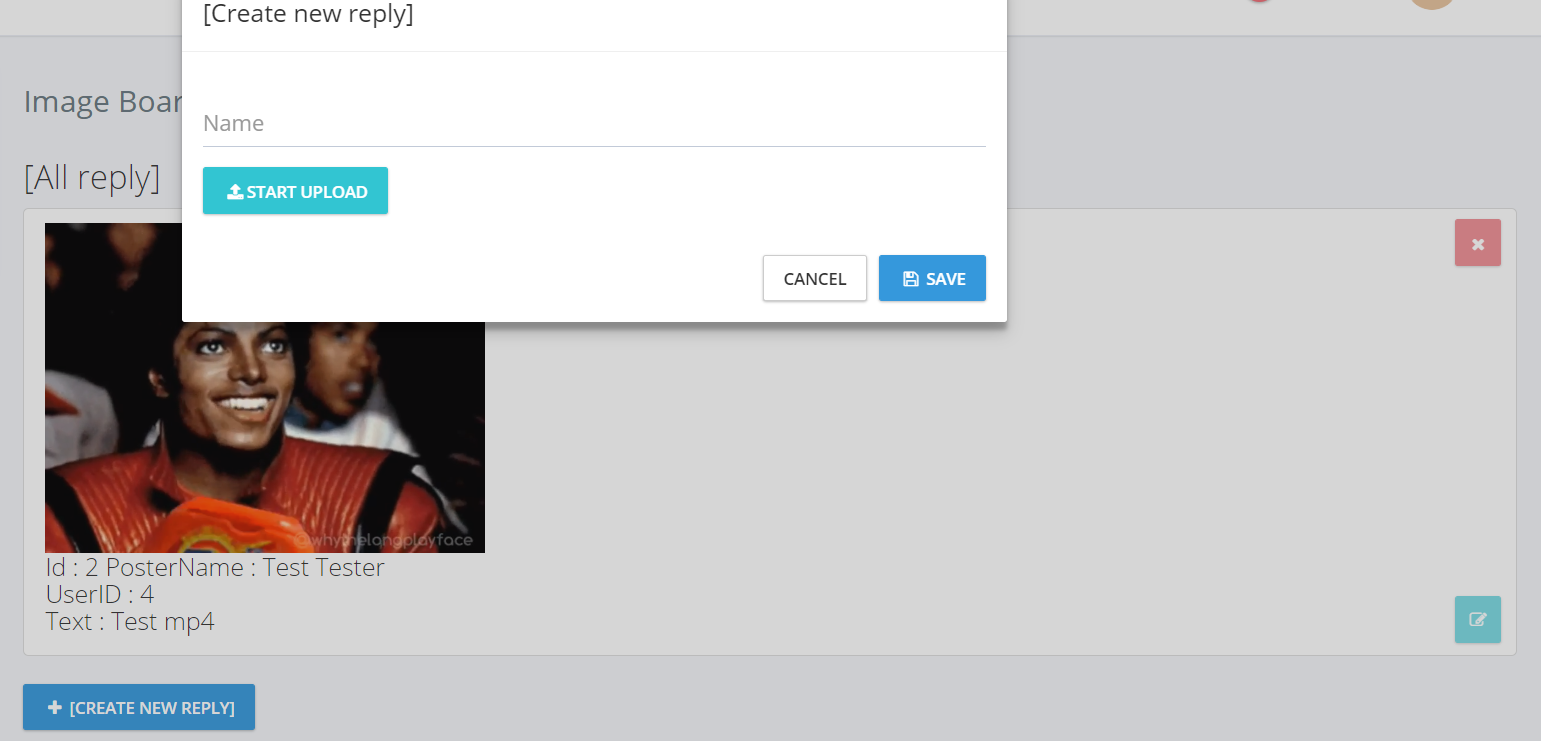
Trong thread sẽ liệt kê toàn bộ reply của thread tương ứng.



Hình 13. Màn hình xem reply/chi tiết thread

d) Màn hình tạo/edit reply mới trong thread

Tương tự như tạo mới thread. Người dùng cũng có thể tạo/edit reply như thread.

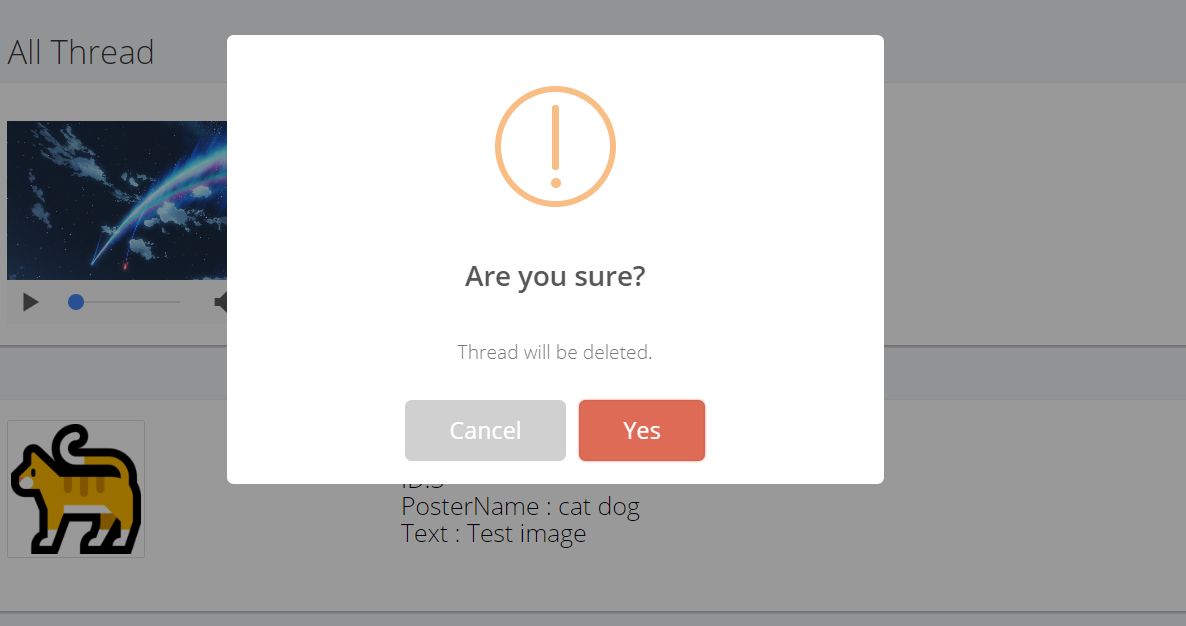


Hình 14. Màn hình tạo reply

e) Delete

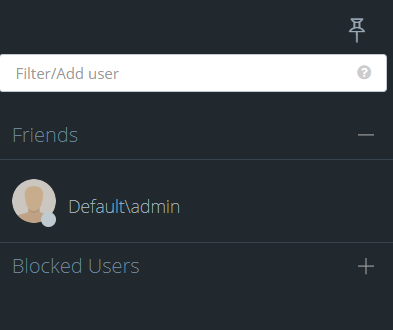
Khi nhấn nút delete 1 màn hình xác nhận sẽ hiện ra hỏi người dùng có chắc chắn xóa hay ko.

Khi đồng ý thì ABP sẽ xóa reply khỏi danh sách hiển thị nhưng soft delete không xóa khỏi database.



Hình 15. Màn hình Delete thread

f) Menu chat



Hình 16. khung chat

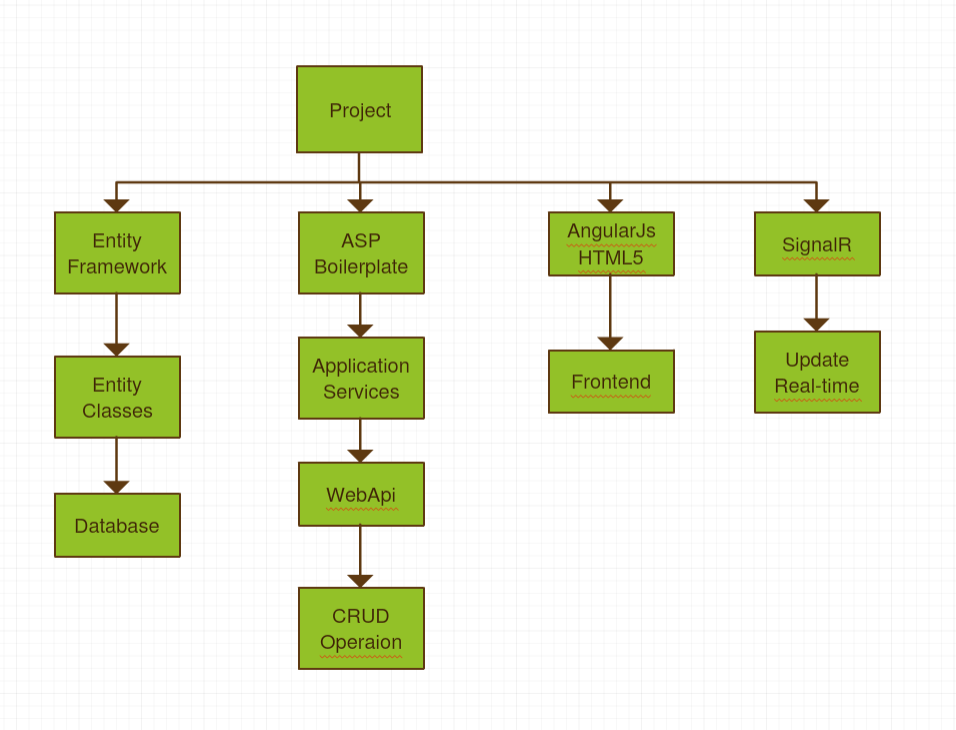
Cho phép người dùng tìm và chọn user khác rồi cho phép chat khi chọn người dùng đó.

### Mô tả chức năng

* Cho phép người dùng tạo thread, trong mỗi thread cho phép tạo reply.
* Đầy đủ các chức năng thêm xóa và sửa.
* Ai cũng có thể xem thread và reply nhưng chỉ user tạo thread hay reply mới được quyền xóa, sửa.
* Cho phép upload hình, gif, video mp4, webm kèm theo text. Có kiểm tra lọc file.
* Liệt kê và hiển thị toàn bộ danh sách thead và reply hiện có.
* Chat giữa các user
* Khi có thay đổi thread hoặc reply thì server sẽ update xuống client trong realtime

# Xây dựng chương trình

## Cấu trúc phân chia công việc (WBS)



Hình 17. cấu trúc thực hiện chương trình

## Kế hoạch dự định

### DANH SÁCH CÔNG VIỆC

| **TT** | **Công việc** | **Thời gian** | **Tiến độ** | **Ghi chú** |
| --- | --- | --- | --- | --- |
| 1. | Đọc tài liệu về các công nghệ | 25/6/2018-  7/7/2018 |  |  |
| 2. | Thực hiện backend, database, … | 1/7/2018-  20/7/2018 | 100% |  |
| 3. | Frontend | 7/7/2018-  24/7/2018 | 100% |  |
| 4. | Hỗ trợ realtime | 24/7/2018-  30/7/2018 | 100% |  |

## Thời gian thực tế

| **TT** | **Công việc** | **Thời gian** | **Tiến độ** | **Ghi chú** |
| --- | --- | --- | --- | --- |
| 1. | Đọc tài liệu về các công nghệ | 25/6/2018-  15/7/2018 |  |  |
| 2. | Thực hiện backend, database, … | 1/7/2018-  25/7/2018 | 100% |  |
| 3. | Frontend | 7/7/2018-  30/7/2018 | 100% |  |
| 4. | Hỗ trợ realtime | 24/7/2018-  31/7/2018 | 100% |  |

# kết luận

## Kết quả đạt được

* Những chức năng chính của website và trong phiếu đăng ký đã đạt được.
* Có hỗ trợ realtime.
* Trong quá trình thực tập cũng như trong thời gian thực hiện đồ án, bản thân em đã học hỏi và tìm hiểu được rất nhiều các công nghệ mới.

## Hạn chế, vướng mắc

* Lần đầu làm web bằng c# kết hợp với các framework , nên chưa thành thạo và còn làm chậm.
* Do hạn chế thời gian nên các chức năng vẫn còn sơ sài, một vài chức năng vẫn còn chưa hoàn thiện

## Hướng phát triển, khắc phục hạn chế

* Đề tài lần này là một cơ hội và thử thách lớn đối với em. Vì vậy sản phẩm có thể chưa được hoàn thiện. Em hi vọng nhận được nhiều ý kiến, trợ giúp để làm cho sản phẩm tốt hơn.
* Vẫn còn nhiều chức năng vẫn có thể thêm vào, xây dựng dự án hoàn thiện hơn. Ví dụ như mở rộng scale của đề tài, quản lý reply, tạo ra các đề mục riêng để phân loại, tìm kiếm, …

**DANH MỤC CÁC tài liệu tham khảo**

1. <http://www.w3schools.com/w3css/default.asp>
2. <http://getbootstrap.com>
3. <https://docs.angularjs.org>
4. <https://stackoverflow.com/>
5. <http://www.entityframeworktutorial.net/>
6. <https://aspnetboilerplate.com/Pages/Documents/>
7. <https://www.asp.net/signalr>
8. <https://www.aspnetzero.com/Documents>

1. https://www.alexa.com/siteinfo/reddit.com [↑](#footnote-ref-1)
2. https://www.alexa.com/siteinfo/4chan.org [↑](#footnote-ref-2)