# BỘ THÔNG TIN VÀ TRUYỀN THÔNG

# HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

# 



**BÁO CÁO THỰC TẬP**

**TỐT NGHIỆP ĐẠI HỌC**

***Đề tài*:**

**“Xây dựng website tìm kiếm đối tượng phù hợp để hẹn hò, kết bạn”**

**Người hướng dẫn : TS. HỒ MẠNH TÀI**

**Sinh viên thực hiện : TRẦN QUANG HUY**

**Mã số sinh viên : N14DCCN102**

**Lớp : D14CQCP01-N**

**Khoá** **: 2014-2019**

**Hệ** **: ĐẠI HỌC CHÍNH QUY**

**TP.HCM, tháng 08/2018**

# LỜI CẢM ƠN

Lời đầu tiên, em xin gửi đến quý thầy cô của Trường Học Viện Công Nghệ Bưu Chính Viễn Thông lời cảm ơn chân thành nhất.Vì trong thời gian học vừa qua các thầy cô đã tạo điều kiện tốt nhất để em được học hỏi và truyền dạy cho em những kiến thức quý báu, những kinh nghiệm bổ ích.Và những kiến thức, những kinh nghiệm ấy sẽ là một hành trang để em bước vào đời.Với tất cả tấm lòng, em xin bày tỏ sự biết ơn chân thành đến quý thầy cô lời cảm ơn sâu sắc nhất.

Em cũng xin cảm ơn quý thầy cô trong khoa công nghệ thông tin , đặc biệt là thầy Hồ Mạnh Tài đã tận tâm, hết lòng giúp đỡ và hướng dẫn em để em có thể hoàn thành tốt đề tài đồ án thực tập tốt nghiệp này.

Sinh viên thực hiện

Trần Quang Huy

# CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

**Độc lập- Tự do- Hạnh phúc**

TP. Hồ Chí Minh, ngày tháng năm 20……

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

**THỰC TẬP TỐT NGHIỆP ĐẠI HỌC**

1. **Tên đề tài:** Xây dựng website tìm kiếm đối tượng phù hợp để hẹn hò, kết bạn**.**
2. **Sinh viên:** Trần Quang Huy **Lớp:** D14CQCP01-N
3. **Giáo viên hướng dẫn:** TS. Hồ Mạnh Tài
4. **Nơi công tác:** Khoa Công nghệ thông tin 2

**NỘI DUNG NHẬN XÉT**

1. ***Đánh giá chung:***

……………………………………………………………………………………………………………………………………………………………………………………………………

1. ***Đánh giá chi tiết:***

……………………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………

1. ***Nhận xét về tinh thần, thái độ làm việc:***

…………………………………………………………………………………………………………………………………………………………………………………………………..

1. ***Kết luận:***

……………………………………………………………………………………………………………………………………………………………………………………

1. ***Điểm hướng dẫn ():***

**GIẢNG VIÊN HƯỚNG DẪN**

(Ký, ghi rõ họ tên)

# MỤC LỤC

[MỞ ĐẦU 1](#_Toc521332780)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 2](#_Toc521332781)

[1.1. Tổng quan 2](#_Toc521332782)

[1.2. Tìm hiểu các công nghệ hỗ trợ thiết kế hệ thống website 2](#_Toc521332783)

[1.3. Công nghệ ứng dụng 2](#_Toc521332784)

[1.4. Công cụ sử dụng 3](#_Toc521332785)

[1.5. Mục tiêu đề tài 3](#_Toc521332786)

[CHƯƠNG 2: GIỚI THIỆU CÁC CÔNG NGHỆ VÀ CÔNG CỤ SỬ DỤNG 4](#_Toc521332787)

[2.1. Giới thiệu về các công nghệ 4](#_Toc521332788)

[**2.1.1.** Spring Framework 4](#_Toc521332789)

[**2.1.2.** Angular 4](#_Toc521332790)

[**2.1.3.** MySQL 5](#_Toc521332791)

[**2.1.4.** WebSocket 5](#_Toc521332792)

[**2.1.5.** RESTful API 5](#_Toc521332793)

[2.2. Giới thiệu về các công cụ 7](#_Toc521332794)

[2.2.1. IntelliJ IDEA 7](#_Toc521332795)

[2.2.2. Visual Studio Code 7](#_Toc521332796)

[2.2.3. MySQL Workbench 7](#_Toc521332797)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 9](#_Toc521332798)

[3.1. Tổng quan ứng dụng 9](#_Toc521332799)

[3.1.1. Đối tượng sử dụng ứng dụng 9](#_Toc521332800)

[3.1.2. Các chưc năng chính của hệ thống 9](#_Toc521332801)

[3.2. Xây dựng lượt đồ use case 11](#_Toc521332802)

[3.3. Xây dựng Database và lượt đồ thực thể kết hợp (ERD) 14](#_Toc521332803)

[3.3.1 Xác định các lớp thực thể 14](#_Toc521332804)

[3.3.2 Vẽ lượt đồ thực thể kết hợp 15](#_Toc521332805)

[3.4. Xây dựng ứng dụng web 20](#_Toc521332806)

[3.4.1. Tổng quan ứng dụng 20](#_Toc521332807)

[3.4.2. Quá trình thực hiện 20](#_Toc521332808)

[3.5. Kết quả 21](#_Toc521332809)

[3.5.1. Trang chủ 21](#_Toc521332810)

[3.5.2. Trang đăng nhập 24](#_Toc521332811)

[3.5.3. Trang đăng ký 25](#_Toc521332812)

[3.5.4. Trang Thông tin tài khoản 26](#_Toc521332813)

[3.5.5. Trang cài đặt tài khoản 29](#_Toc521332814)

[3.5.6. Trang bài viết mới 30](#_Toc521332815)

[3.5.7. Phòng chat 31](#_Toc521332816)

[3.5.8. Bài viết 32](#_Toc521332817)

[3.5.9. Tìm kiếm 33](#_Toc521332818)

[KẾT LUẬN 34](#_Toc521332819)

[DANH MỤC TÀI LIỆU THAM KHẢO 35](#_Toc521332823)

# DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT

**HTTP**: Hypertext Transfer Protocol.

**HTML**: Hypertext Markup language.

**SPA**: Single Page Application.

**API**: Application Programming Interface.

# DANH MỤC CÁC HÌNH VẼ

[Hình 3.2. Sơ đồ Use case 11](#_Toc521333588)

[Hình 3.3.2. Mô hình thực thể kết hợp 15](#_Toc521333589)

[Hình 3.5.1.a. Trang chủ (Your Feed) 22](#_Toc521333590)

[Hình 3.5.1.b. Trang chủ (Global Feed) 23](#_Toc521333591)

[Hình 3.5.2. Trang đăng nhập 24](#_Toc521333592)

[Hình 3.5.3. Trang đăng ký 25](#_Toc521333593)

[Hình 3.5.4.a. Trang Thông tin tài khoản (Xem bài viết) 26](#_Toc521333594)

[Hình 3.5.4.b. Trang thông tin tài khoản (Thư viện hình) 27](#_Toc521333595)

[Hình 3.5.4.c. Trang thông tin tài khoản (Bài viết yêu thích) 28](#_Toc521333596)

[Hình 3.5.5. Trang cài đặt tài khoản 29](#_Toc521333597)

[Hình 3.5.6. Trang bài viết mới 30](#_Toc521333598)

[Hình 3.5.7. Trang phòng chat 31](#_Toc521333599)

[Hình 3.5.8. Trang bài viết 32](#_Toc521333600)

[Hình 3.5.9. Trang tìm kiếm 33](#_Toc521333601)

# 

# MỞ ĐẦU

Trong nhiều năm gần đầy, nhu cầu kết nối internet của mọi người rất cao. Việc sử dụng internet gần như trong suốt các hoạt động của mọi người như làm việc, học tập và giải trí. Nắm bắt được xu thế đó, các nhà phát triển web đã cho ra đời rất nhiều trang mạng xã hội để mọi người có thể kết nối với nhau, điển hình là Facebook với hơn 2 tỉ người dùng trên toàn thế giới.

Trong quá trình học tại trường, em có học và tìm hiểu cách thức để thiết kế và vận hành một hệ thống web. Từ những kiến thức đó, em quyết định chọn đề tài đồ án thực tập tốt nghiệp là “Xây dụng website tìm kiếm đối tượng phù hợp để hẹn hò, kết bạn”.

Đây là một website giúp mọi người có thể kết nối với nhau như các mạng xã hội thông thường, đặc biệt đây là trang web để cho những người độc thân tìm được một nửa của mình thông qua các chức năng tìm kiếm mà hệ thống cung cấp.

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

## Tổng quan

Đề tài “Xây dụng website tìm kiếm đối tượng phù hợp để hẹn hò, kết bạn” là một ứng dụng trên nền web. Người dùng truy cập vào đăng ký tài khoản để sử dụng. Giống như một mạng xã hội, người dùng có thể theo dỗi người khác (following) và họ cũng sẽ có những người theo dõi lại (followers). Mọi người được đăng bài viết, hình ảnh để người khác có thể thích (like), bình luận (comment) hoặc chia sẻ (share).

Chức năng chính của ứng dụng là tìm kiếm, kết bạn và hẹn hò với người khác. Người dùng tìm kiếm đối tượng phù hợp dựa vào công cụ của trang web cung cấp như tìm kiếm theo tên, tuổi, sở thích,…Khi hai người đồng ý hẹn hò với nhau, họ sẽ được gọi là một cặp (couple), và những cặp này sẽ có trang dành riêng cho họ để quản lý mối quan hệ như hẹn lịch đi ăn, thêm hình ảnh chung, chia sẻ bài viết, đếm ngày yêu nhau…

## Tìm hiểu các công nghệ hỗ trợ thiết kế hệ thống website

* + - Lý thuyết :
      * Tìm hiểu Spring framework.
      * Tìm hiểu Angular.
      * Tìm hiểu Websocket.
      * Tìm hiểu RESTful API.
    - Thực hành:
      * Phân tích, thiết kế cơ sở dữ liệu cho hệ thống.
      * Xây dựng website đơn giản có các chức năng:
        + Tạo và quản lý tài khoản.
        + Tìm kiếm (lọc) đối tượng phù hợp.
        + Đăng bài (ảnh, video), bình luận, thích, chia sẻ.
        + Chat và nhắn tin giữa các users.

## Công nghệ ứng dụng

* + - Spring Framework.
    - Angular 6.
    - MySQL.
    - WebSocket.
    - RESTful API.

## Công cụ sử dụng

* + - IntelliJ IDEA
    - Visual Studio Code
    - MySQL Workbench

## Mục tiêu đề tài

* + - Hệ thống phải dễ sử dụng, có tính khả thi, đầy đủ thông tin, tránh dư thừa thông tin.
    - Đăng ký tài khoản.
    - Cung cấp chức năng đăng bài viết, hình ảnh cho người dùng.
    - Người dùng có quyền thích, bình luận, chia sẻ bài viết, hình ảnh.
    - Lọc được đối tượng phù hợp.
    - Chat được giữa các users.

# CHƯƠNG 2: GIỚI THIỆU CÁC CÔNG NGHỆ VÀ CÔNG CỤ SỬ DỤNG

## Giới thiệu về các công nghệ

### Spring Framework

Spring là một Framework phát triển các ứng dụng Java được sử dụng bởi hàng triệu lập trình viên. Nó giúp tạo các ứng dụng có hiệu năng cao, dễ kiểm thử, sử dụng lại code…

Spring nhẹ và trong suốt (nhẹ: kích thước nhỏ, version cơ bản chỉ khoảng 2MB; trong suốt: hoạt động một cách trong suốt với lập trình viên).

Spring là một mã nguồn mở, được phát triển, chia sẻ và có cộng đồng người dùng rất lơn.

Spring Framework được xây dựng dựa trên 2 nguyên tắc design chính là: Dependency Injection và Aspect Oriented Programming.

Những tính năng core (cốt lõi) của Spring có thể được sử dụng để phát triển Java Desktop, ứng dụng mobile, Java Web. Mục tiêu chính của Spring là giúp phát triển các ứng dụng J2EE một cách dễ dàng hơn dựa trên mô hình sử dụng POJO (Plain Old Java Object)

### Angular

Angular là một nền tảng web phía người dùng, mã nguồn mở dựa trên Kiểu dữ liệuScript (Kiểu dữ liệuScript-based open-source front-end web application platform) dẫn đầu bởi nhóm Angular của Google và cộng đồng các cá nhân cũng như tổ chức đã đóng góp cho quá trình phát triển.

Angular giúp xây dựng các ứng dụng phía client bằng HTML và Javascript hoặc Kiểu dữ liệuScript (một ngôn ngữ lập trình mã nguồn mở phát triển bởi Microsoft, là một bộ các cú pháp chặt chẽ for Javascript nhưng có thể tùy chọn thêm vào kiểu cho biến, Kiểu dữ liệu cuối cùng sẽ được dịch ra Javascript). Angular gồm các thành phần chính: các module, component, service …

Angular có các đặc trưng mang lại lợi ích: đa nền tảng, tốc độ và hiệu năng …

Phiên bản hiện tại của Angular là 6 phát hành vào ngày 03/05/2018.

### MySQL

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,...

### WebSocket

**WebSocket** là một giao thức truyền thông (communication protocol), nó giúp thành lập một kênh liên lạc 2 chiều (two-way communication channel) giữa **client** và **server**. So sánh giữ HTTP và WebSocket:

**HTTP** (Hypertext Transfer Protocol): Là một giao thức request-response (Yêu cầu - Đáp ứng). Client (Trình duyệt) muốn một cái gì đó, nó gửi yêu cầu tới Server, và Server đáp ứng yêu cầu đó. HTTP là một giao thức truyền thông một chiều, mục đích ở đây là để giải quyết *"Làm thế nào để tạo một yêu cầu tại client, và làm thế nào để đáp ứng yêu cầu của client"*, và đây là lý do để HTTP tỏa sáng.

**WebSocket**: Không phải là giao thức request-response (Yêu cầu - Đáp ứng), nơi mà chỉ Client có mới thể gửi yêu cầu tới Server. Một khi kết nối với giao thức WebSocket được thiết lập, client & server có thể gửi dữ liệu tới cho nhau, cho tới khi kết nối ở tầng dưới là TCP được đóng lại. WebSocket về cơ bản rất giống với khái niệm TCP Socket, sự khác biệt là WebSocket được tạo ra để sử dụng cho các ứng dụng Web.

### RESTful API

**RESTful Web Service** là các **Web Service** được viết dựa trên kiến trúc **REST**. **REST**đã được sử dụng rộng rãi thay thế cho các **Web Service** dựa trên **SOAP**và **WSDL**. **RESTful Web Service** nhẹ (lightweigh), dễ dàng mở rộng và bảo trì.

Những khái niệm đầu tiên về **REST** ( **R**epresentational **S**tate **T**ransfer) được đưa ra vào năm 2000 trong luận văn tiến sĩ của ***Roy Thomas Fielding*** (đồng sáng lập giao thức HTTP). Trong luận văn ông giới thiệu khá chi tiết về các ràng buộc, quy ước cũng như cách thức thực hiện với hệ thống để có được một hệ thống **REST**.

**REST** định nghĩa các quy tắc kiến trúc để bạn thiết kế **Web services**, chú trọng vào tài nguyên hệ thống, bao gồm các trạng thái tài nguyên được định dạng như thế nào và được truyền tải qua **HTTP**, và được viết bởi nhiều ngôn ngữ khác nhau. Nếu tính theo số dịch vụ mạng sử dụng, **REST** đã nổi lên trong vài năm qua như là một mô hình thiết kế dịch vụ chiếm ưu thế. Trong thực tế, **REST**đã có những ảnh hưởng lớn và gần như thay thế **SOAP** và **WSDL** vì nó đơn giản và dễ sử dụng hơn rất nhiều.

**REST** là một bộ quy tắc để tạo ra một ứng dụng **Web Service**, mà nó tuân thủ 4 nguyên tắc thiết kế cơ bản sau:

* Sử dụng các phương thức HTTP một cách rõ ràng
* Phi trạng thái
* Hiển thị cấu trúc thư mục như các URls
* Truyền tải **JavaScript Object Notation (JSON)**, **XML**hoặc cả hai.

## Giới thiệu về các công cụ

## IntelliJ IDEA

IntelliJ IDEA là phần mềm lập trình và phát triển các ngôn ngữ lập trình các ứng dụng cho điện thoại, java, hoặc trên máy tính với việc hỗ trợ nhiều ngôn ngữ lập trình khác nhau phần mềm là một trong những giải pháp, chìa khóa trao tay dành cho những kỹ sư lập trình nhằm tạo ra những chương trình, ứng dụng cho điện thoại hoặc bất kỳ thứ gì khác.

IntelliJ IDEA là công cụ nhằm tạo ra những dự án lập trình cho điện thoại hoặc cho với khả năng mã hóa sâu sắc và điều hướng nhanh phần mềm còn cung cấp cho người dùng một danh sách các biểu tượng và ký hiệu phục vụ trong công việc lập trình của người dùng.

IntelliJ IDEA thực hiện phân tích dòng dữ liệu đầu vào của người dùng, phần mềm phân tích lưu lượng dữ liệu để đoán biểu tượng thể loại thời gian phần mềm thực hiện phân tích kịch bản được đưa vào của người dùng với các để xuất khác nhau tự động thêm các lớp phôi để phù hợp với chương trình mà bạn đang làm việc.

Những tính năng chính của IntelliJ IDEA:

* + - Lập trình các dự án cho điện thoại, java…
    - So sánh, tìm kiếm các đoạn mã trùng nhau.
    - Phân tích dữ liệu.
    - Tìm kiếm nhanh chóng.
    - Ứng dụng rộng rãi.

## Visual Studio Code

Visual Studio Code là công cụ soạn thảo mã nguồn (code) được phát triển bởi Microsoft và có thể chạy trên Windows, Linux và MacOS. Nó hỗ trợ debug, nhúng Git control, highlight cú pháp, đoán code thông minh, snippets và code refactor.

## MySQL Workbench

MySQL Workbench là một công cụ thiết kế cơ sở dữ liệu đa nền tảng được phát triển bởi MySQL. Nó là một ứng dụng thành công phát triển từ dự án DBDesigner4.

MySQL Workbench cung cấp một công cụ giao diện đồ hoạ để làm việc với MySQL Server và CSDL. MySQL Workbench hỗ trợ toàn vẹn cho MySQL Server phiên bản 5.1 và mới hơn, nó cũng tương thích với MySQL Server 5.0 (nhưng không phải là với tất cả chức năng).

MySQL Workbench có thể sử dụng như là các công cụ giao diện người dùng riêng trên các hệ điều hành như Windows, Linux và OS X trong các sản phẩm và phiên bản khác nhau.

MySQL Workbench cung cấp ba vùng chức năng chính:

* + - **SQL Development**: cho phép bạn tạo và quản lý kết nối đến các CSDL trên server. Cho phép bạn thiết lập các tham số kết nối, MySQL Wordbench cung cấp khả năng thực thi các truy vấn SQL trên những kết nối của CSDL bằng cách sử dụng SQL Editor tích hợp. Chức năng này thay thế ứng dụng Query Browser độc lập trong phiên bản trước.
    - **Data Modeling**: cho phép tạo và chỉnh sửa các mô hình đồ hoạ của các lược đồ CSDL của bạn bằng cách sửa dụng bộ Table Editor. Table Editor cung cấp các chức năng dễ sử dụng để sửa các Table, Column, Index, Trigger, Partitioning, Options, Inserts và Privileges, Routines and Views.
    - **Server Administration**: Cho phép bạn tạo và quản trị các thể hiện của server. Chức năng này thay thế ứng dụng MySQL Administrator độc lập của các phiên bản trước.

# CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Tổng quan ứng dụng

### 3.1.1. Đối tượng sử dụng ứng dụng

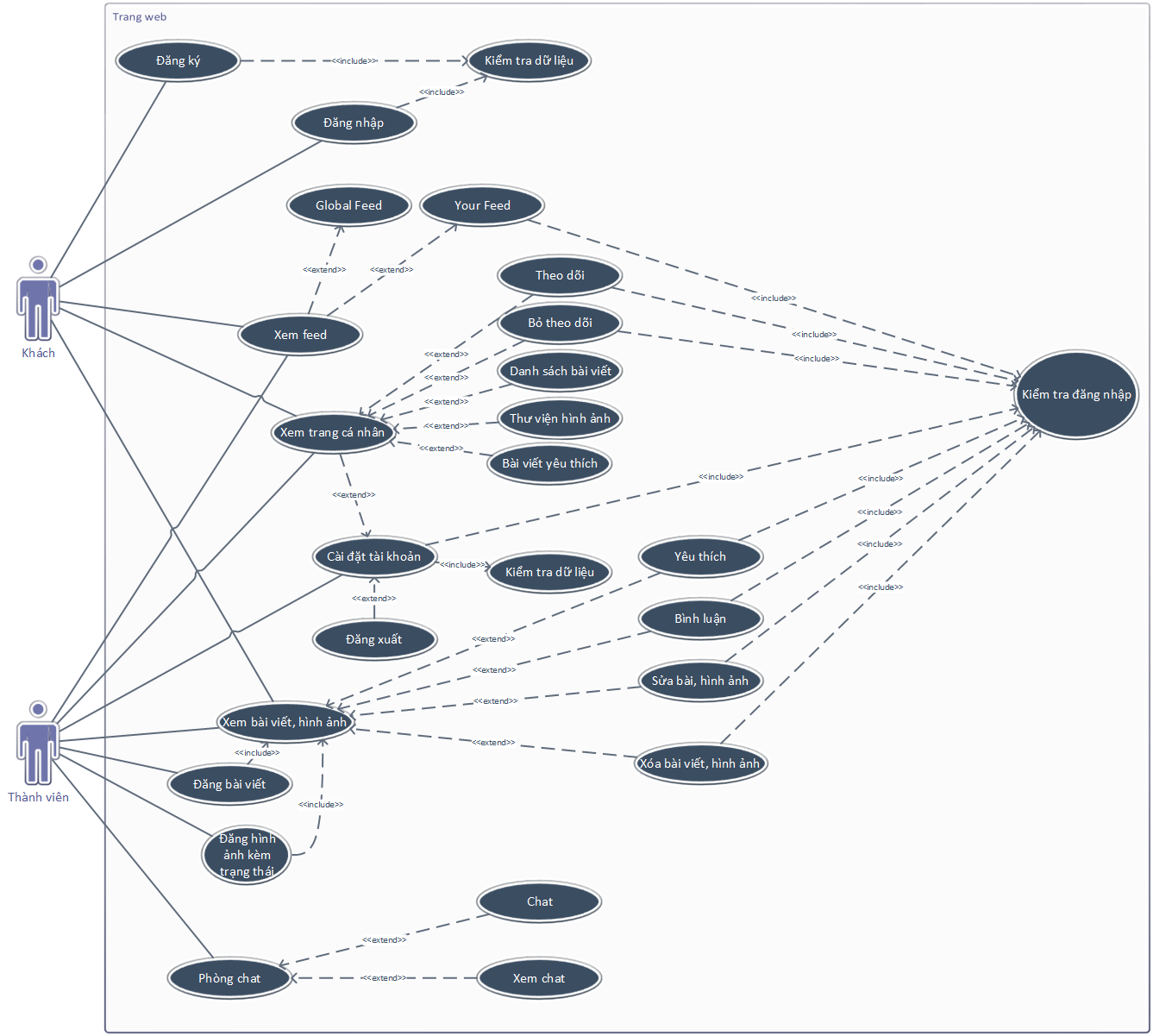
* + - Khách: Người truy cập vào trang web chưa đăng ký tài khoản.
    - Thành viên: Người đã đăng nhập vào trang web.

### 3.1.2. Các chưc năng chính của hệ thống

* + - Đối với Khách:
      * Đăng ký tài khoản.
      * Đăng nhập vào hệ thống.
      * Xem Global Feed (Danh sách bài viết mới nhất của tất cả Thành viên).
      * Xem trang cá nhân của thành viên bao gồm: Danh sách các bài đăng, Thư viện hình ảnh và Danh sách bài viết yêu thích.
      * Xem một bài viết/ hình ảnh bao gồm: Nội dung bài viết/ hình ảnh, Tác giả, Số lượt yêu thích, Danh sách bình luận của bài viết/ hình ảnh.
    - Đối với Thành viên:
      * Xem Global Feed và Your Feed (Chỉ hiện danh sách bài viết của Thành viên được theo dõi).
      * Xem trang cá nhân của thành viên bao gồm: Danh sách các bài đăng, Thư viện hình ảnh và Danh sách bài viết yêu thích, theo dõi/ bỏ theo dõi thành viên.
      * Xem một bài viết/ hình ảnh bao gồm: Nội dung bài viết/ hình ảnh, Tác giả, Số lượt yêu thích, Danh sách bình luận của bài viết/ hình ảnh, bình luận bài viết/ hình ảnh, yêu thích bài viết/ hình ảnh.
      * Cài đặt tài khoản: Chỉnh sửa thông tin tài khoản.
      * Đăng xuất.
      * Đăng bài viết mới.
      * Đăng hình ảnh kèm trạng thái.
      * Vào phòng Chat
      * Xem Chat
      * Chat với thành viên khác.

## Xây dựng lượt đồ use case

Từ tổng quan và yêu cầu của ứng dụng. Vẽ lượt đồ use case thể hiện các chức năng của hệ thống.



Hình 3.2. Sơ đồ Use case

*Mô tả Actor*

Ứng dụng có 2 Actor:

* + - *Khách:* Người truy cập vào trang web chưa đăng ký tài khoản.
    - *Thành viên:* Người đã đăng nhập vào trang web.

*Mô tả Use case*

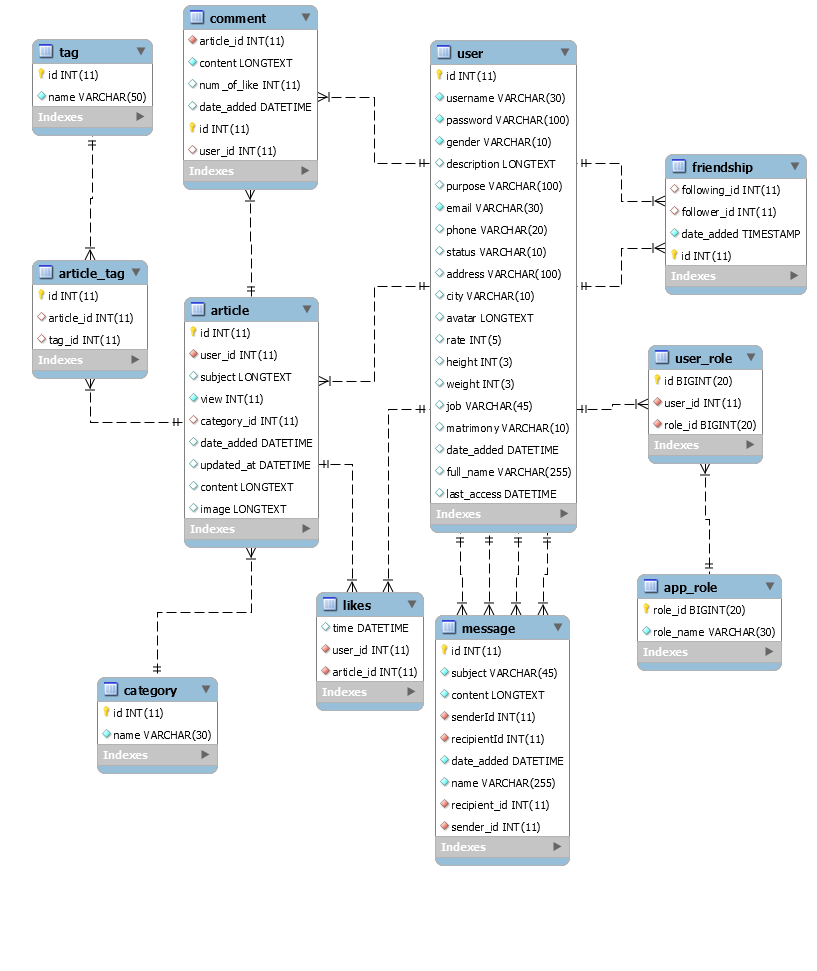
|  |  |  |
| --- | --- | --- |
| **Tên use case** | **Actor tham gia** | **Mô tả** |
| Đăng ký | Khách | Khách cung cấp thông tin để đăng ký tài khoản mới |
| Đăng nhập | Khách | Khách đăng nhập vào trang web với email và mật khẩu |
| Kiểm tra dữ liệu | Khách | Kiểm tra dữ liệu khi khách thực hiện các chức năng đăng ký và đăng nhập |
| Xem Feed | Khách | Kiểm tra dữ liệu khi khách thực hiện các chức năng đăng ký và đăng nhập |
| Global Feed | Khách, Thành viên | Hiển thị danh sách bài viết, nội dung ngắn gọn và số lượt yêu thích của từng bài viết của tất cả Thành viên trong hệ thống. Thành viên được quyền like bài viết ở danh sách này |
| Your Feed | Thành viên | Hiển thị danh sách bài viết của những người mà Thành viên đang theo dõi |
| Xem trang cá nhân | Khách, Thành viên | Hiển thị trang cá nhân của Thành viên bao gồm tên tài khoản, mô tả, ảnh đại diện  Thành viên khác có thể theo dõi Thành viên tại trang này  Hiển thị danh sách bài viết, thư viện ảnh, các bài viết yêu thích |
| Theo dõi | Thành viên | Theo dõi Thành viên |
| Bỏ theo dõi | Thành viên | Bỏ theo dõi thành viên |
| Danh sách bài viết | Khách, Thành viên | Hiển thị danh sách bài viết của một thành viên. |
| Thư viện hình ảnh | Khách, Thành viên | Hiển thị thư viện hình ảnh của một thành viên. |
| Bài viết yêu thích | Khách, Thành viên | Hiển thị danh sách bài viết mà Thành viên đã thích |
| Cài đặt tài khoản | Thành viên | Chỉnh sửa thông tin cá nhân |
| Đăng xuất | Thành viên | Đăng xuất khỏi hệ thống |
| Xem bài viết, hình ảnh | Khách, Thành viên | Hiển thị bài viết chi tiết, bao gồm Tiêu đề, nội dung, lượt thích, lượt bình luận |
| Yêu thích | Thành viên | Thích một bài viết |
| Bình luận | Thành viên | Đăng bình luận cho bài viết |
| Sửa bài, hình ảnh | Thành viên | Sửa một bài viết, hình ảnh. |
| Xóa bài, hình ảnh | Thành viên | Xóa một bài viết, hình ảnh. |
| Đăng bài viết | Thành viên | Đăng một bài viết mới. |
| Đăng hình ảnh kèm trạng thái | Thành viên | Đăng trạng thái mới cùng với hình ảnh |
| Phòng chat | Thành viên | Tham gia phòng chat |
| Chat | Thành viên | Gửi nội dung chat đến phòng chat. |
| Xem chat | Thành viên | Xem nội dung cuộc trò chuyện trong phòng chat. |

## Xây dựng Database và lượt đồ thực thể kết hợp (ERD)

### 3.3.1 Xác định các lớp thực thể

* + Các lớp thực thể có trong cơ sở dữ liệu:
    - User: Thông tin thành viên.
    - Friendship: Mối quan hệ theo dõi giữa thành viên và thành viên.
    - App\_role: Danh sách các quyền của thành viên.
    - User\_role: Thành viên được gán quyền hạn.
    - Message: Lưu các tin nhắn của thành viên.
    - Article: Thông tin bài viết/ hình ảnh.
    - Comment: Thành viên bình luận vào bài viết.
    - Likes: Thành viên Thích bài viết.
    - Tag: Thông tin thẻ.
    - Article\_tag: Lưu thông tin tất cả các thẻ của bài viết.
    - Category: Thông tin thể loại bài viết.

### 3.3.2 Vẽ lượt đồ thực thể kết hợp



Hình 3.3.2. Mô hình thực thể kết hợp

1. **Table “user”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh một Thành viên. |
| username | varchar(50) | Tên tài khoản |
| password | varchar(50) | Mật khẩu |
| gender | datetime | Giới tính |
| description | longtext | Mô tả |
| purpose | varchar(50) | Mục đích |
| email | varchar(50) | Email |
| phone | varchar(50) | Số điện thoại |
| status | varchar(50) | Trạng thái (ACT: Hoạt động | DEA: Khoong hoạt động) |
| address | varchar(50) | Địa chỉ |
| city | varchar(50) | Thành phố |
| avatar | longtext | Ảnh đại diện |
| rate | int(5) | Đánh giá |
| height | int(3) | Chiều cao |
| weight | int(3) | Cân nặng |
| job | varchar(50) | Công việc |
| matrimony | varchar(50) | Tình trạng hôn nhân |
| date\_added | datetime | Ngày thêm vào |
| full\_name | vatchar(255) | Tên đầy đủ |

1. **Table “message”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | int(11) | Khóa chính, mã số định danh một tin nhắn |
| Subject | varchar(40) | Tiêu đề |
| Content | varchar(40) | Nội dung |
| sender\_id | int(11) | Người gửi |
| recipient\_id | int(11) | Người nhận |
| date\_added | date | Ngày thêm vào |
|  |  |  |

1. **Table “friendship”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | int(11) | Khóa chính, mã số định danh Mối quan hệ (Follow) |
| following\_id | int(11) | Người gửi yêu cầu theo dõi |
| follower\_id | int(11) | Người được theo dõi |
| date\_added | datetime | Ngày thêm vào |

1. **Table “comment”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh một bình luận |
| content | longtext | Nội dung |
| date\_added | datetime | Ngày thêm vào |
| num\_of\_like | int(11) | Số lượng thích bình luận |
| user\_id | int(11) | Người đăng |
| article\_id | int(11) | Mã số của bài viết |

1. **Table “article”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh một bài viết |
| subject | longtext | Tiêu đề |
| content | longtext | Nội dung |
| view | int(11) | Số lượt xem |
| category\_id | int(11) | Mã số của thể loại |
| date\_added | datetime | Ngày thêm vào |
| updated\_at | datetime | Ngày cập nhật |
| user\_id | int(11) | Mã số Thành viên |
| image | longtext | Hình anh bài viết |

1. **Table “likes”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| time | datetime | Thời gian yêu thích |
| user\_id | int(11) | Khóa chỉnh, mã số Thành viên |
| article\_id | int(11) | Khóa chính, mã số bài viết |

1. **Table “article\_tag”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh bảng |
| article\_id | int(11) | Mã số bài viết |
| tag\_id | int(11) | Mã số của thẻ |

1. **Table “tag”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh thẻ |
| name | varchar(40) | Tên thẻ |

1. **Table “category”**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| id | int(11) | Khóa chính, mã số định danh thể loại |
| name | varchar(40) | Tên thể loại |

## Xây dựng ứng dụng web

Dựa vào các thiết kế bên trên, bắt đầu quá trình lập trình để hiện thực các yêu cầu hệ thống.

### Tổng quan ứng dụng

Trang web được viết theo kiến trúc REST, nên sẽ chia ứng dụng thành 2 phần đề lập trình: Server và Client.

* + - Phần Server: Dùng Spring để lập trình các chức năng, cung cấp API cho phía Client.
    - Phần Client: Dùng Angular để lập trình các chứng năng, gọi API từ Server cung cấp để lấy dữ liệu.

### Quá trình thực hiện

*Bước 1*: Phân tích, thiết kế, mô tả các chức năng cần thiết của website.

*Bước 2*: Tạo Project và lập trình ở phần Backend.

*Bước 3*: Viết API cho trang web.

*Bước 4*: Thêm dữ liệu vào cơ sở dữ liệu và kiểm tra tính đúng đắn của dữ liệu thông qua việc gọi API.

*Bước 5*: Thiết kế giao diện cho trang web.

*Bước 6*: Sử dụng Angular, Bootstrap, HTML, CSS,…để tiến hành lập trình theo thiết kế.

*Bước 7*: Lập trình xử lý dữ liệu từ Backend gửi về thông qua API, hiển thị thông tin cho người dùng.

*Bước 8*: Chạy thử Website và sửa lỗi.

## Kết quả

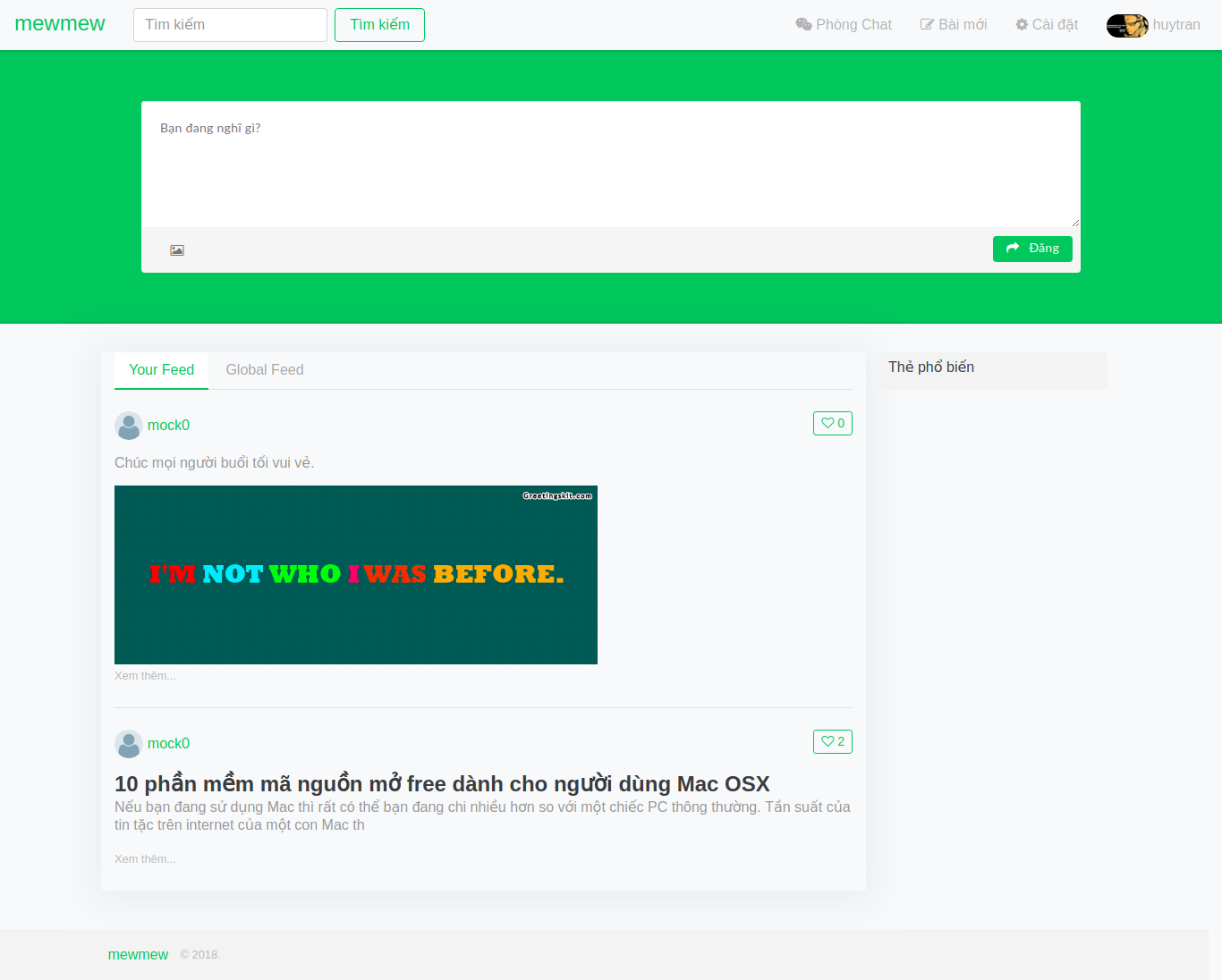
Sau khi thực hiện tuần tự các bước đã đặt ra, website đã có những tính năng cở bản theo yêu cầu ban đầu, bênh cạnh đó cũng có một số tính năng cải tiến mới và đảm bảo thuận lợi nhất cho người sử dụng.

### 3.5.1. Trang chủ

Mục đích: Cho người dùng sử dụng các tính năng dưới đây:

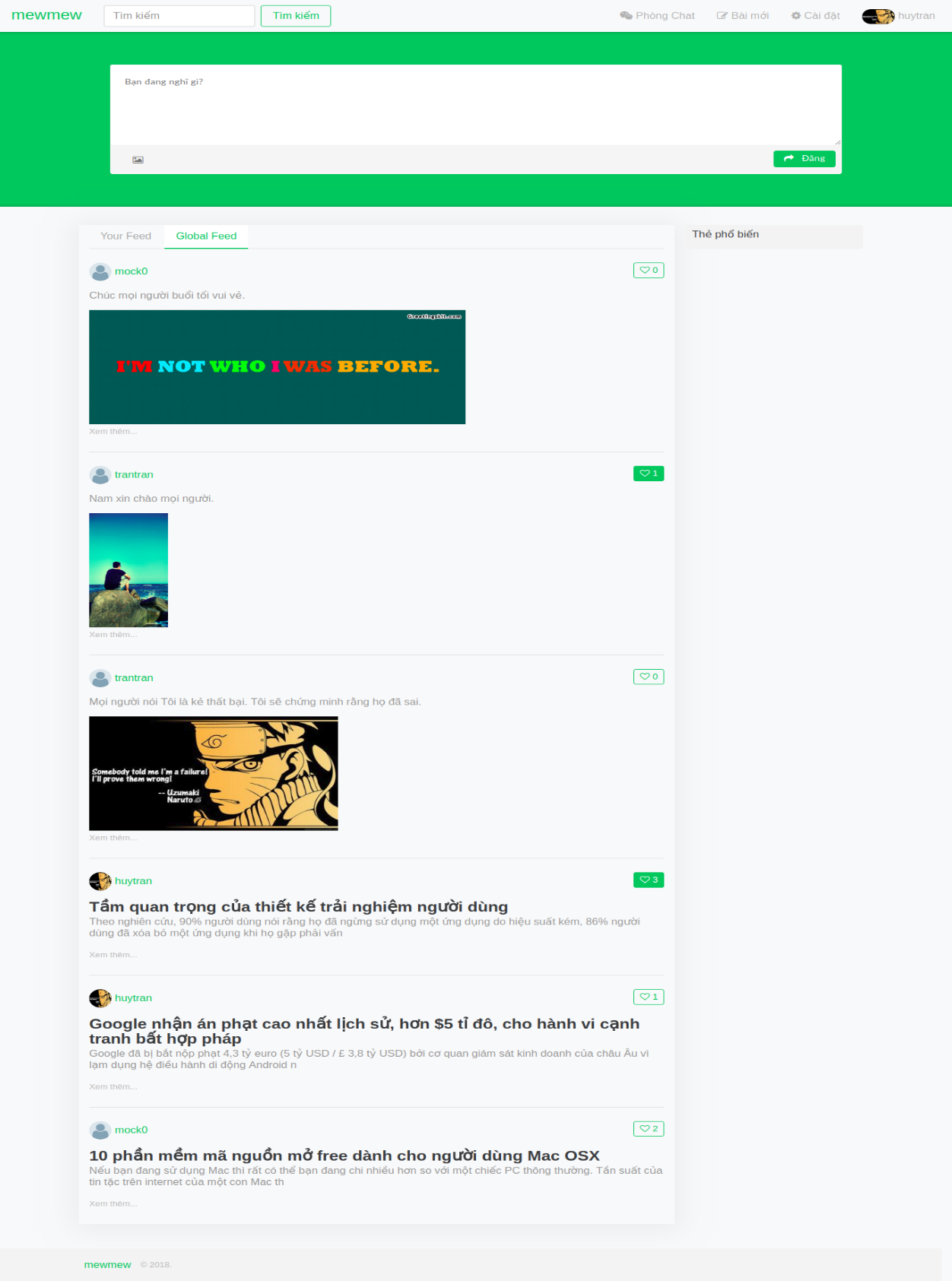
* Xem Global Feed.
* Xem Your feed (Yêu cầu đăng nhập).
* Đăng ảnh kèm trạng thái (Yêu cầu đăng nhập)
* Chọn bài viết để xem.

* Your Feed



Hình 3.5.1.a. Trang chủ (Your Feed)

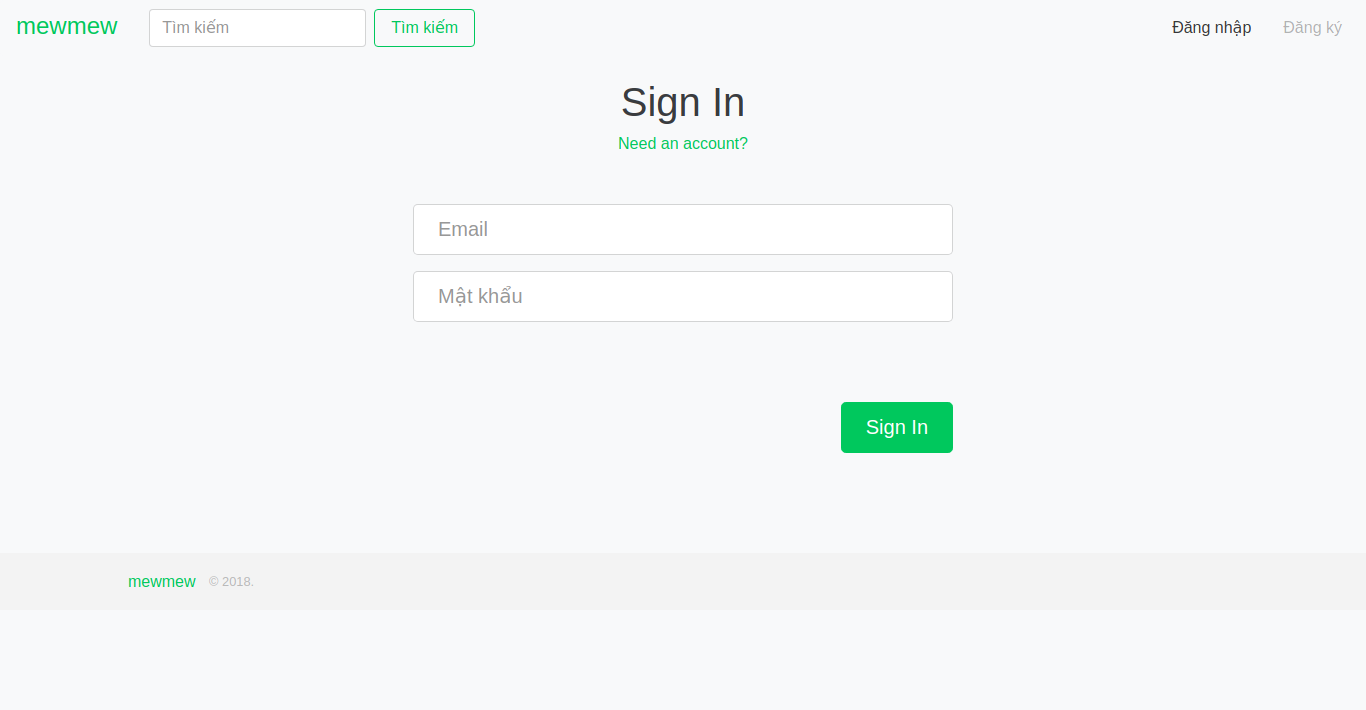
* Global Feed



Hình 3.5.1.b. Trang chủ (Global Feed)

### 3.5.2. Trang đăng nhập

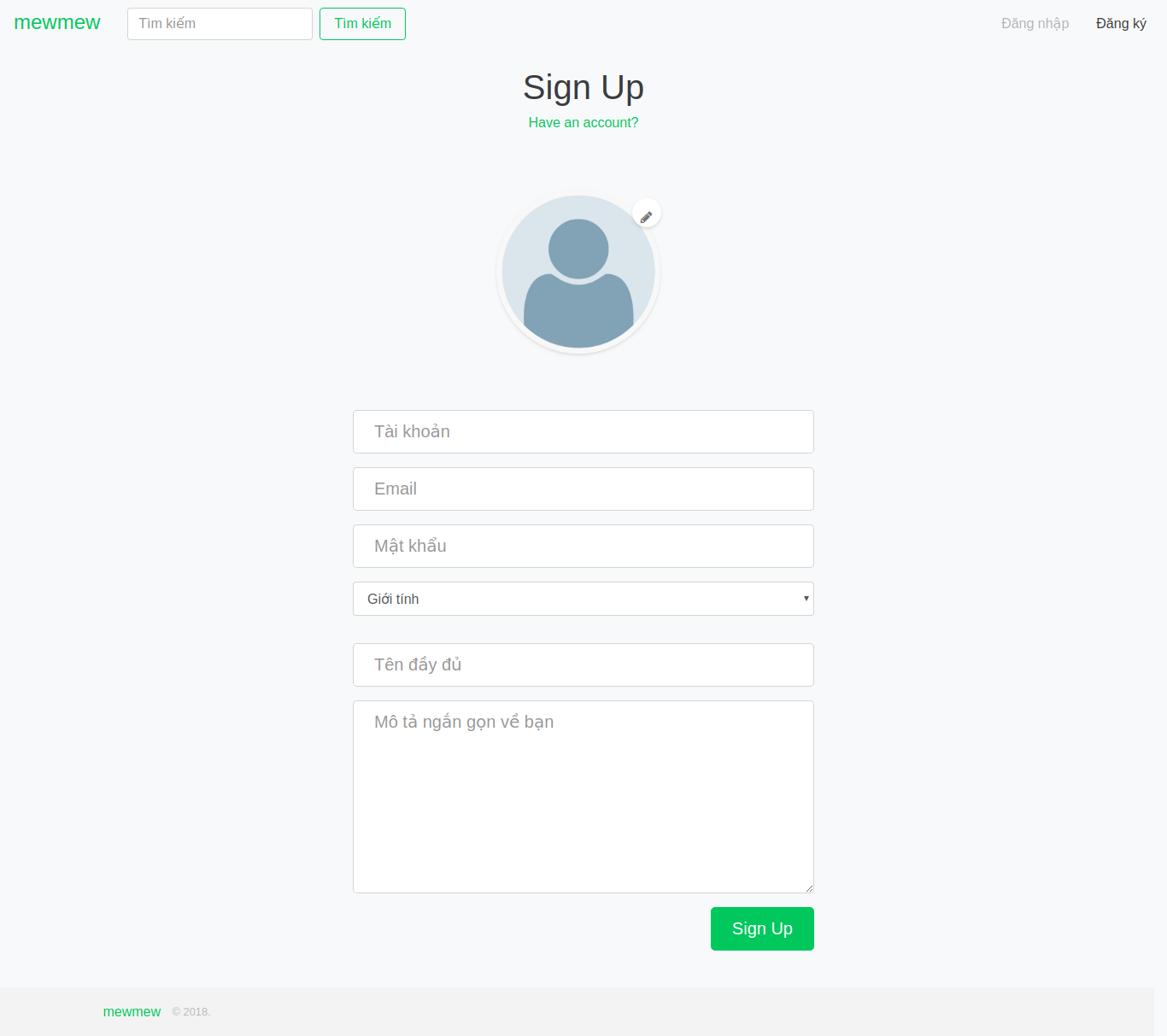
Mục đích: Cho khách đăng nhập vào trang web để sử dụng đầy đủ chức năng.



Hình 3.5.2. Trang đăng nhập

### 3.5.3. Trang đăng ký

Mục đích: Cho khách đăng ký tài khoản.

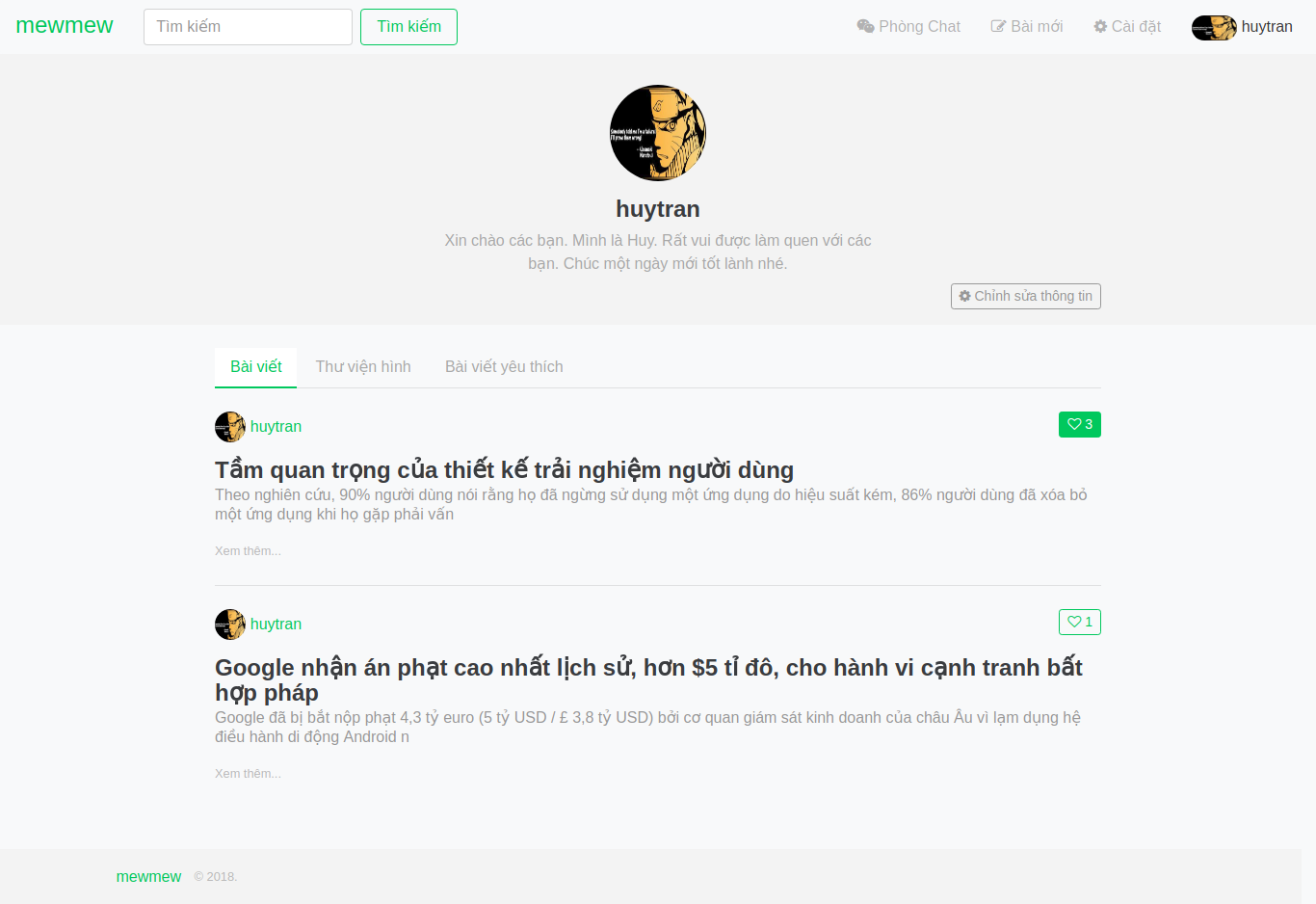
****

Hình 3.5.3. Trang đăng ký

### 3.5.4. Trang Thông tin tài khoản

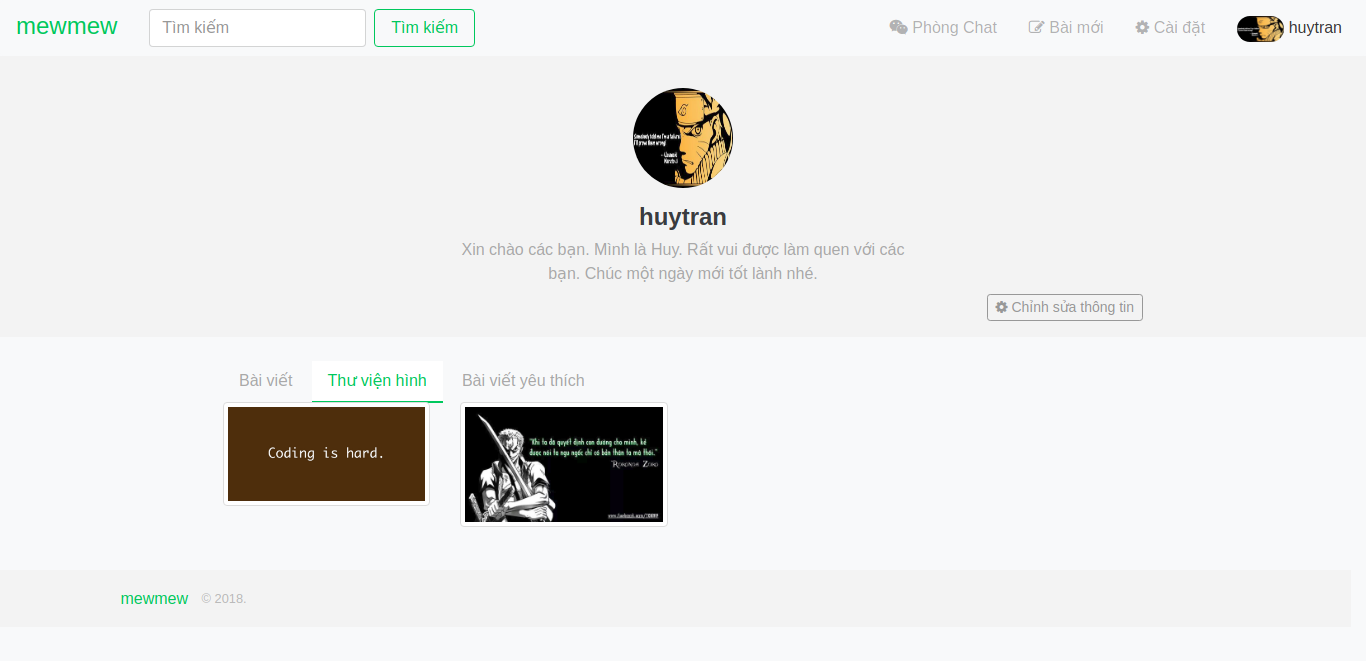
Mục đích: Xem trang cá nhân của một người dùng với 3 tùy chọn:

* Xem bài viết

****

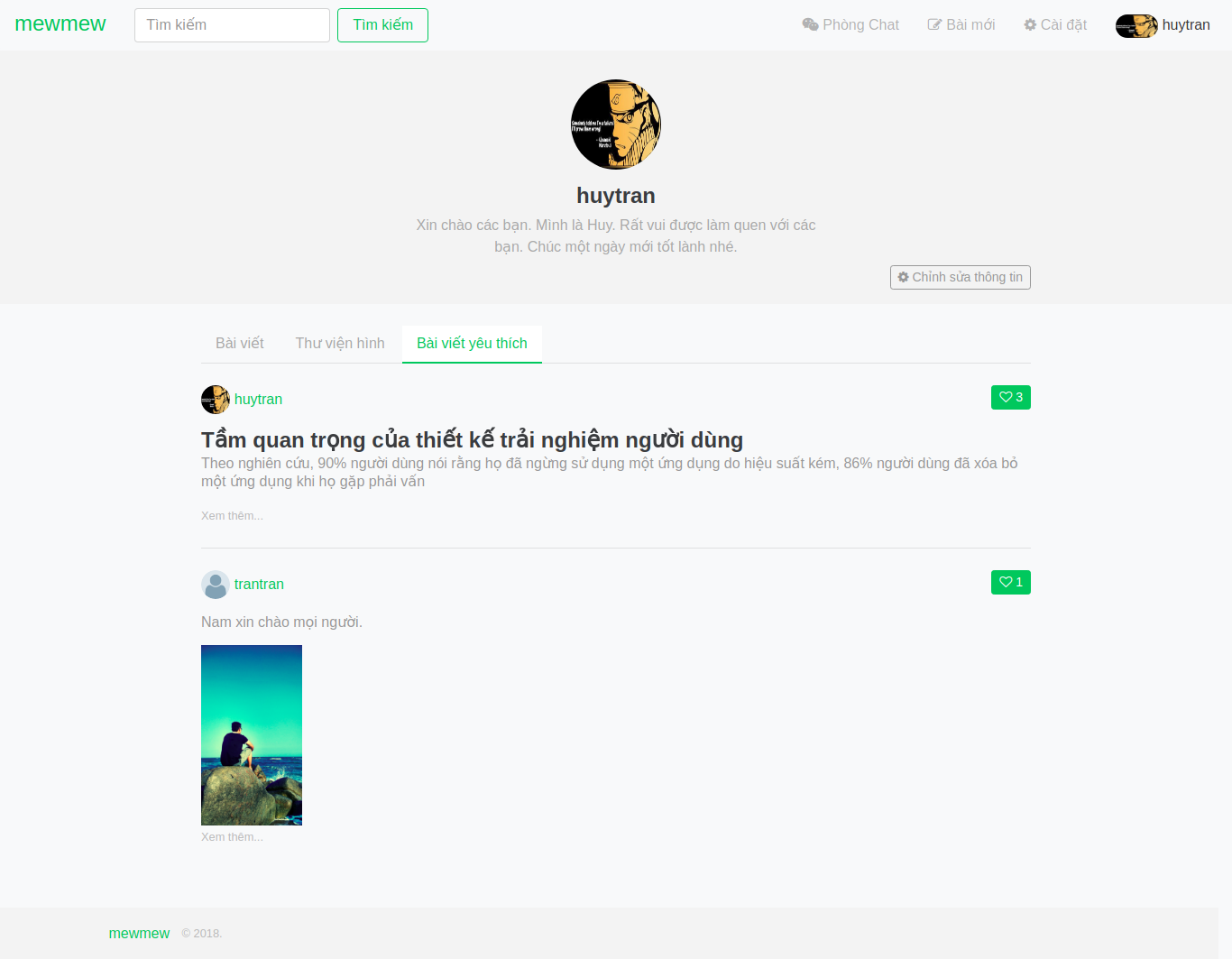
Hình 3.5.4.a. Trang Thông tin tài khoản (Xem bài viết)

* + - Thư viện hình



Hình 3.5.4.b. Trang thông tin tài khoản (Thư viện hình)

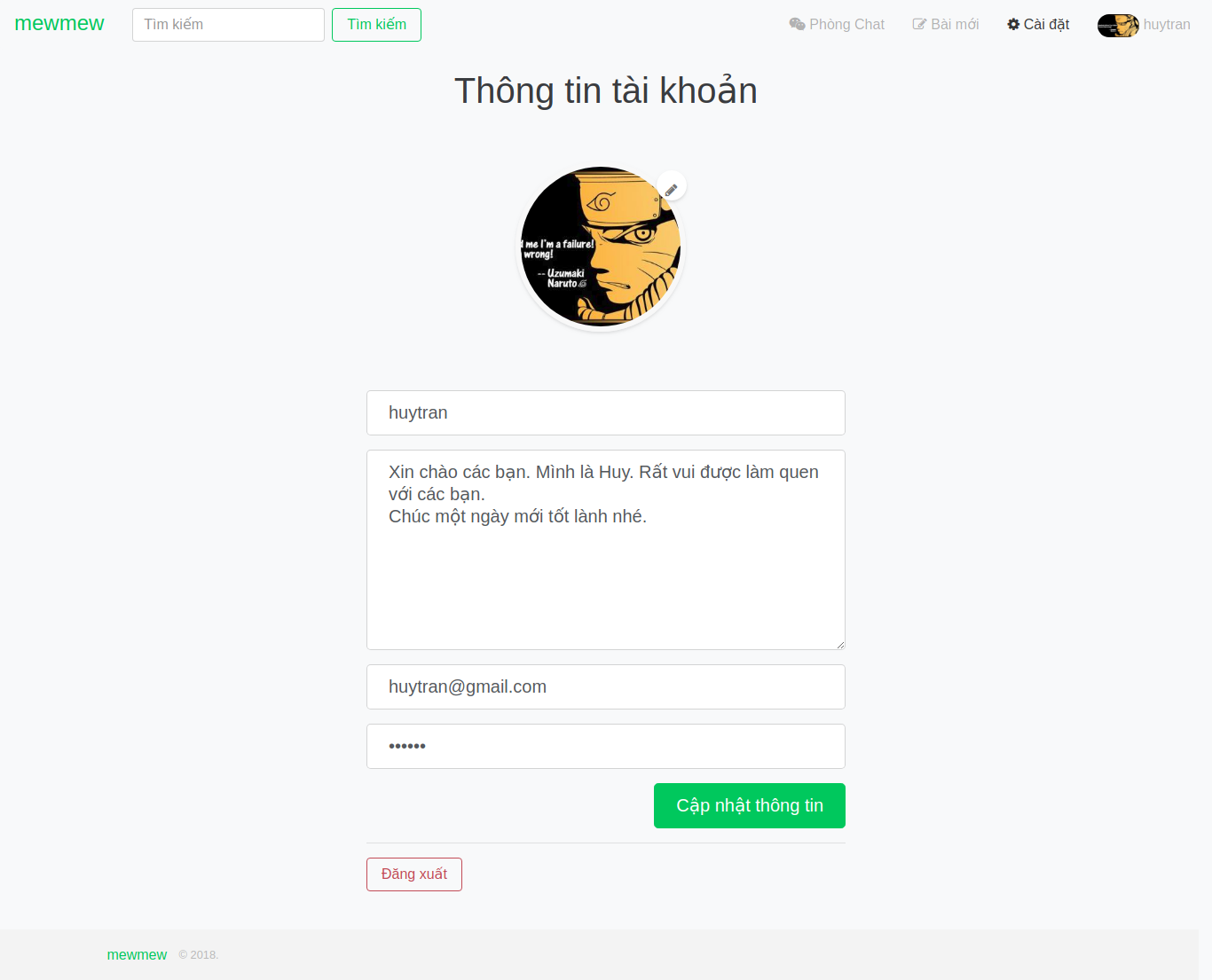
* + - Bài viết yêu thích



Hình 3.5.4.c. Trang thông tin tài khoản (Bài viết yêu thích)

### 3.5.5. Trang cài đặt tài khoản

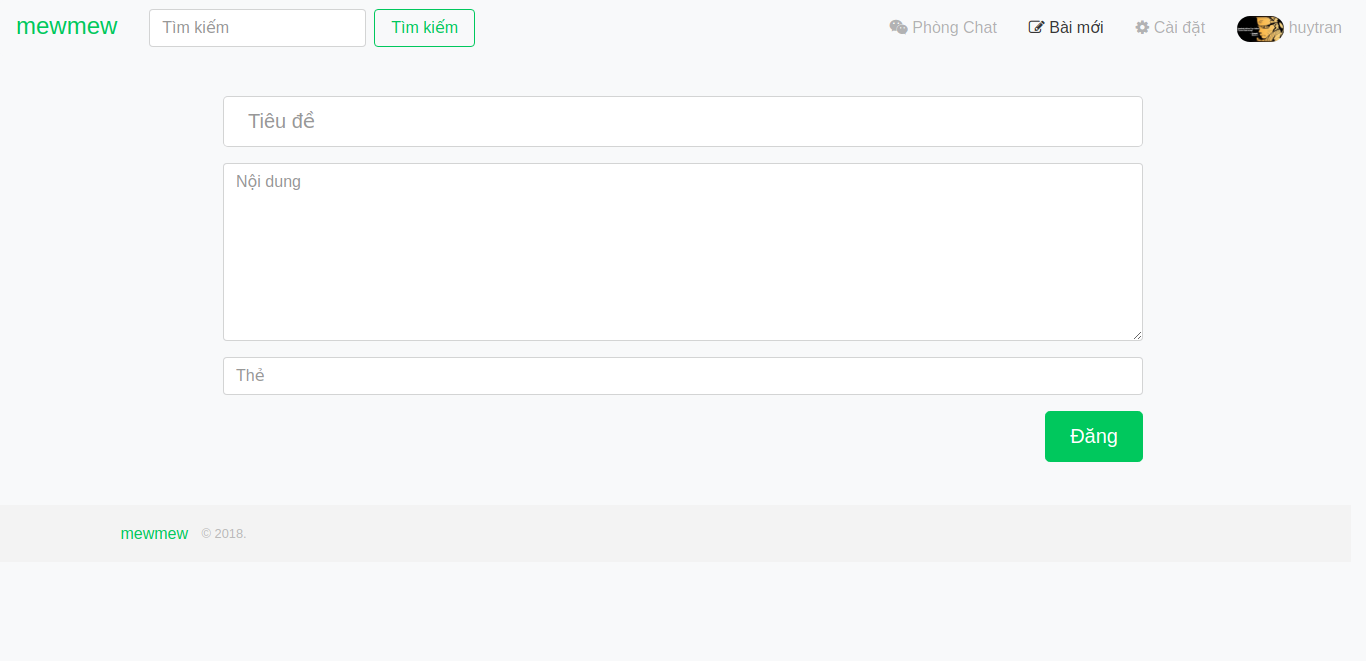
Mục đích: Cho người dùng cập nhật thông tin tài khoản và có thể đăng xuất tài khoản.

****

Hình 3.5.5. Trang cài đặt tài khoản

### 3.5.6. Trang bài viết mới

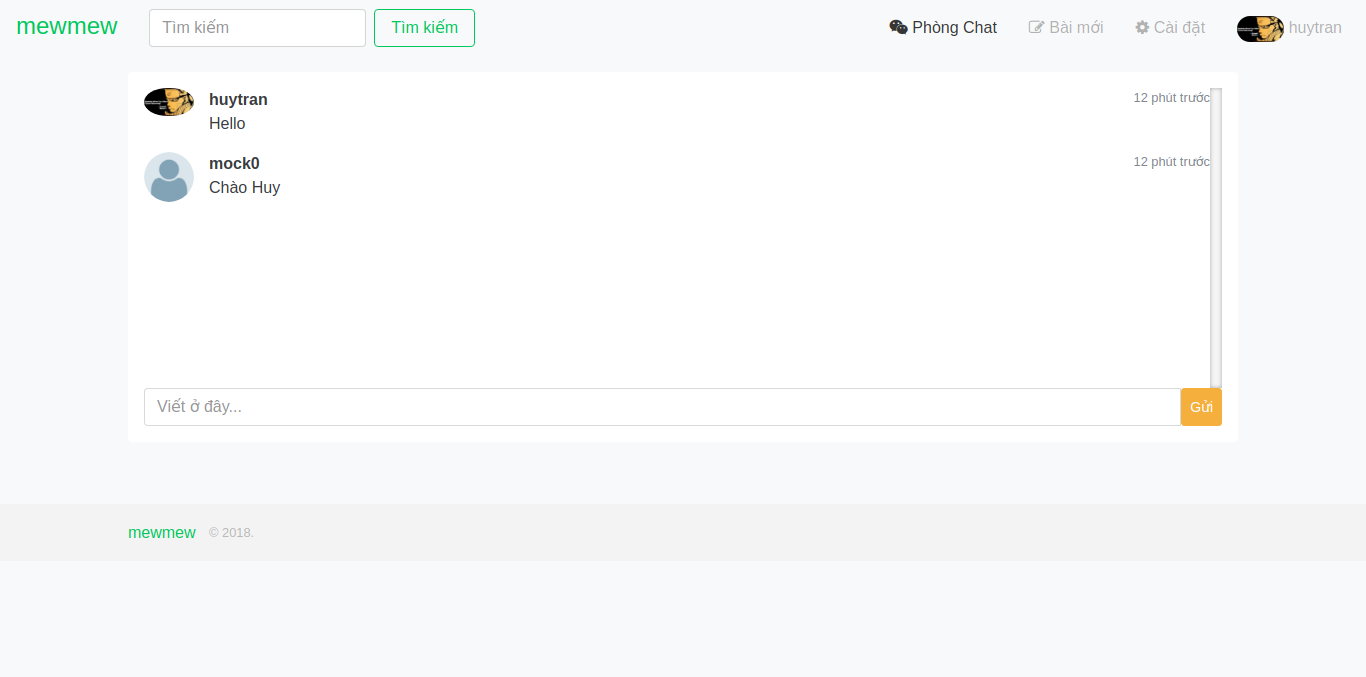
Mục đích: Cho người dùng đăng bài viết mới.



Hình 3.5.6. Trang bài viết mới

### 3.5.7. Phòng chat

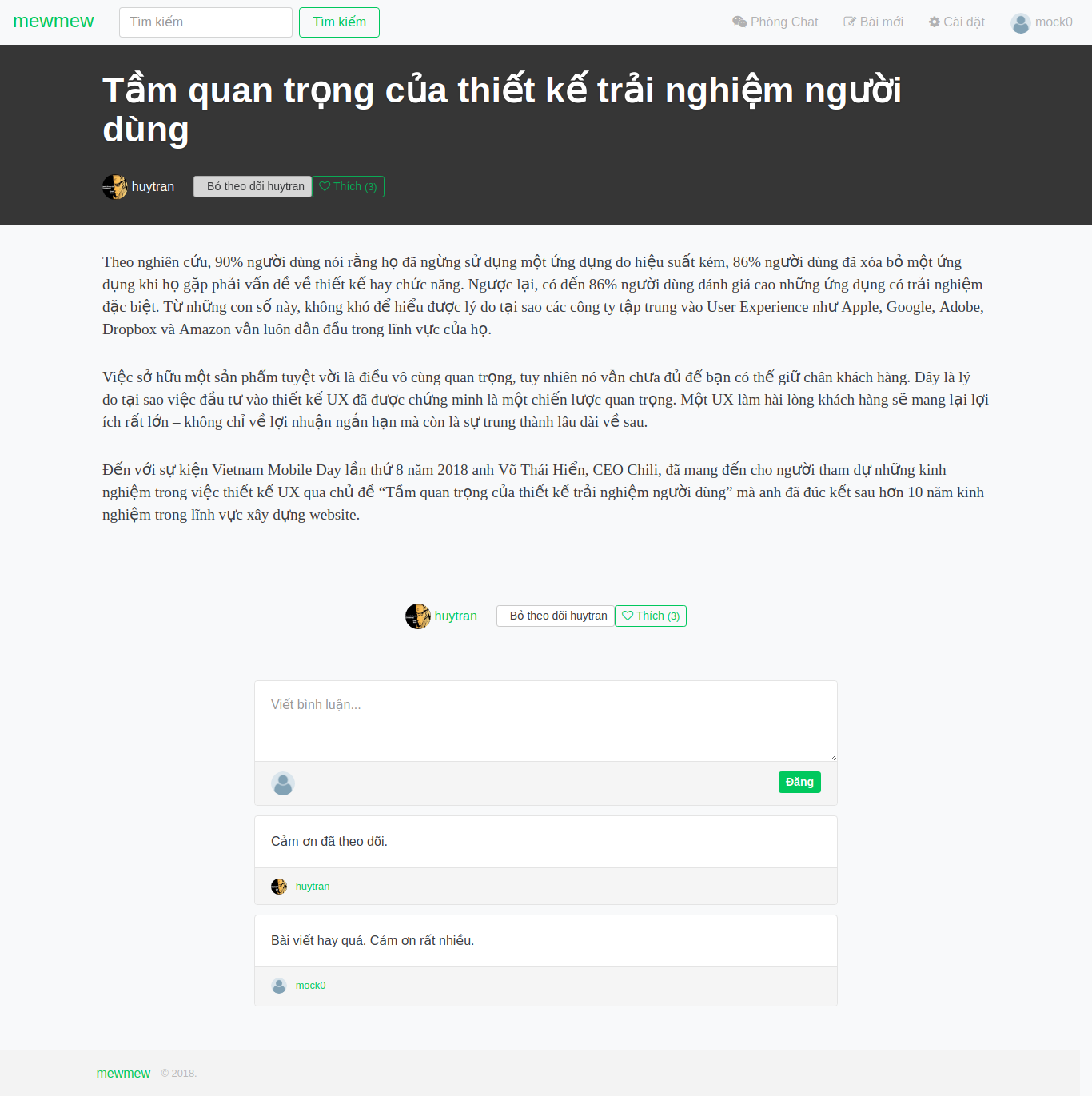
Mục đích: Cho người dùng tham gia phòng chat, trò chuyện với nhau.



Hình 3.5.7. Trang phòng chat

### 3.5.8. Bài viết

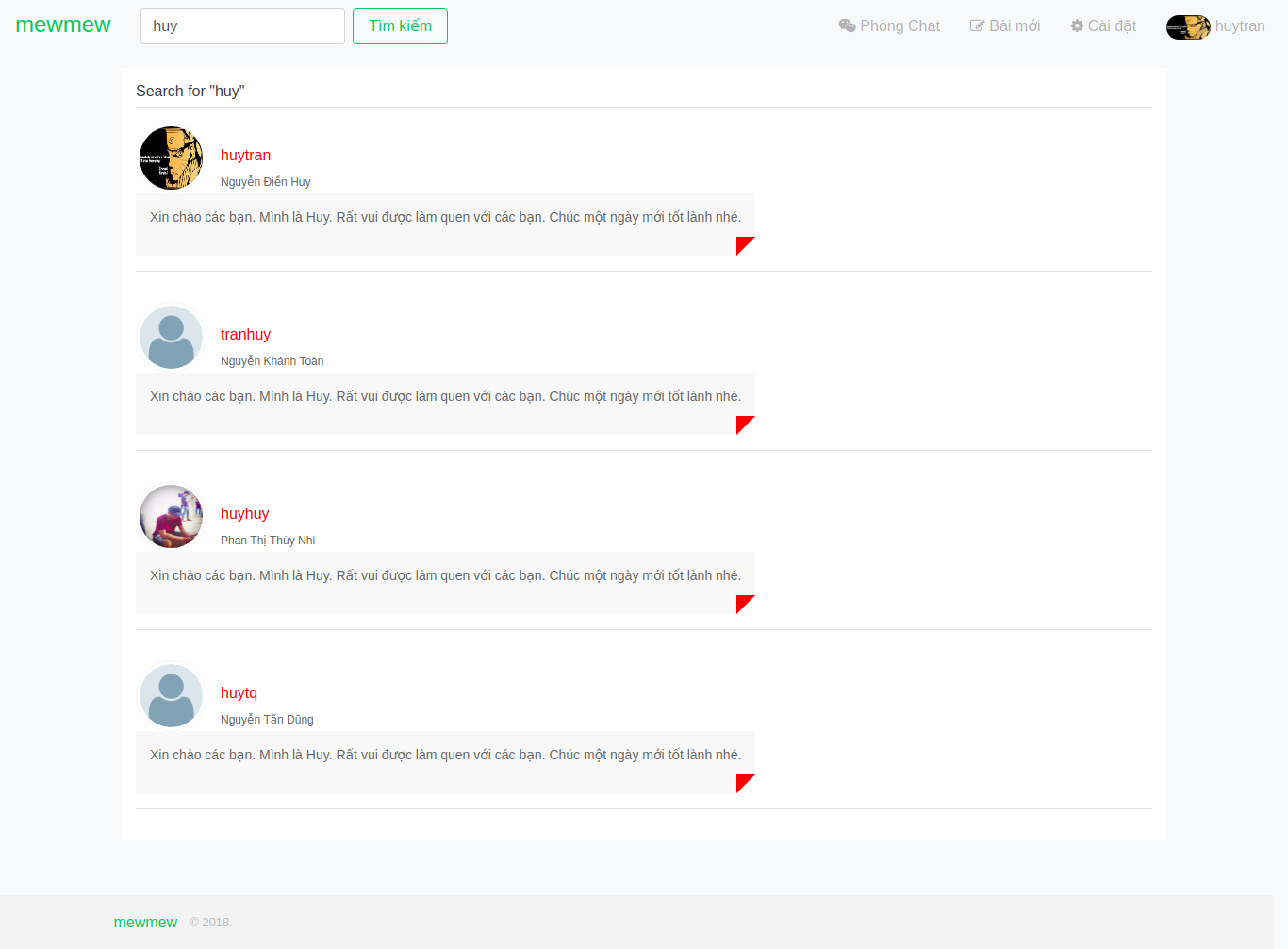
Mục đích: Xem một bài viết, người dùng có quyền Thích bài viết hoặc đăng Bình luận về bài viết.



Hình 3.5.8. Trang bài viết

### 3.5.9. Tìm kiếm

Mục đích: Tìm kiếm người dùng bằng tên đầy đủ hoặc tên tài khoản.



Hình 3.5.9. Trang tìm kiếm

# KẾT LUẬN

## Kết quả đạt được

Website về cơ bản đã thỏa mãn được các yêu cầu của một website mạng xã hội cần có, bao gồm Tạo tài khoản, đăng nhập, xem bài viết, thích và bình luận bài viết.

Các chức năng yêu cầu của website như thiết kế ban đầu đã được hoàn thành.

Trong quá trình làm đồ án, bản thân em cũng đã học hỏi và tìm hiểu được rất nhiều các công nghệ mới cũng như áp dụng vào thực tế đề tài nhiều công nghệ như đã trình bày ở trong báo cáo.

## Hạn chế

Tuy nhiên, do thời gian ngắn và số lượng công việc cũng như tìm hiểu khá nhiều công nghệ cho nên việc xảy ra nhiều lỗi làm cho quá trình hoàn thành bị chậm. Bên cạnh đó một số các chức năng dự kiến thêm vào vẫn chưa được triển khai như chia sẻ bài viết, chat giữa các Thành viên…

## Hướng phát triển

* Hoàn thiện các chức năng còn hạn chế.
* Về mặt đối tượng sử dụng, có thể thêm đối tượng Admin vào để quản lý tất cả dữ liệu bài đăng và quản lý thông tin các Thành viên.
* Về mặt giao diện, chỉnh sửa các phần hiển thị cho dễ nhìn, đẹp hơn.
* Phát triển thêm nhiều tính năng hỗ trợ người dùng một cách thuận tiện nhất.

# DANH MỤC TÀI LIỆU THAM KHẢO

**Danh mục các Website tham khảo:**

**1. Hướng dẫn cài đặt công cụ, môi trường:** [**https://o7planning.org**](https://o7planning.org)

**2. Tài liệu Angular:** [**https://angular.io/docs**](https://angular.io/docs)

**3. Tài liệu Spring Framework:** [**https://spring.io/guides**](https://spring.io/guides)

**4. Diễn đàn trao đổi, thảo luận:** [**https://stackoverflow.com**](https://stackoverflow.com)