ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA CÔNG NGHỆ THÔNG TIN

Tel. (84-511) 3736949, Fax. (84-511) 3842771

Website: [itf.dut.edu.vn](mailto:itf.ud.edu.vn), E-mail: cntt@dut.udn.vn



ĐỒ ÁN TỐT NGHIỆP KỸ SƯ

NGÀNH CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI :**

XÂY DỰNG HỆ THỐNG

GIAO TIẾP THỜI GIAN THỰC

SINH VIÊN : Trần Thông Thành Luân

MSV : 102140079

LỚP : 14T2

CBHD : ThS. Mai Văn Hà

Đà Nẵng, 05/2018

LỜI CẢM ƠN

Trên thực tế, không có thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù là trực tiếp hay gián tiếp của người khác. Trong suốt thời gian năm năm qua từ khi bắt đầu học tập ở giảng đường đại học đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý thầy cô, gia đình và bạn bè.

Đầu tiên em xin gửi lời cảm ơn chân thành nhất đến tất cả các thầy cô khoa Công nghệ Thông tin – Trường Đại học Bách khoa Đà Nẵng đã tận tình dạy bảo và dìu dắt em trong suốt năm năm học vừa qua tại trường.

Em xin chân thành cảm ơn thầy - ThS. Mai Văn Hà, người đã tận tình hướng dẫn, cũng như đã tạo mọi điều kiện để em có thể hoàn thành đồ án tốt nghiệp này. Nếu như không nhờ những lời hướng dẫn, những đánh giá và chỉ bảo sâu sắc của thầy thì em nghĩ việc hoàn thành đồ án tốt nghiệp của em sẽ khó khăn hơn rất nhiều. Một lần nữa em xin chân thành cảm ơn thầy.

Cảm ơn tập thể 14T2 cũng như gia đình đã luôn sát cánh bên cạnh và là nguồn động lực vô cùng lớn giúp cho em không ngừng nỗ lực trong năm năm qua.

Đồ án được thực hiện trong khoảng thời gian ba tháng. Bước đầu đi vào thực tế, tìm hiểu lĩnh vực ứng dụng web, với kiến thức còn hạn chế và nhiều bỡ ngỡ. Do vậy, không thể tránh khỏi những thiếu sót, em rất mong nhận được những ý kiến đóng góp quý báu của quý thầy cô và các bạn để kiến thức trong lĩnh vực này của em được hoàn thiện hơn.

Sau cùng, em xin kính chúc quý thầy cô trong Khoa Công nghệ Thông tin và thầy Mai Văn Hà thật dồi dào sức khỏe, niềm tin để tiếp tục thực hiện sứ mệnh cao đẹp của mình là truyền đạt kiến thức cho thế hệ mai sau.

Trân trọng!

Đà Nẵng, ngày ...., tháng ...., năm 2018

Trần Thông Thành Luân

LỜI CAM ĐOAN

Tôi xin cam đoan :

1. Nội dung trong đồ án này là do tôi thực hiện dưới sự hướng dẫn trực tiếp của ThS. Mai Văn Hà.
2. Các tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.
3. Nếu có những sao chép không hợp lệ, vi phạm, tôi xin chịu hoàn toàn trách nhiệm.

Sinh viên thực hiện

Trần Thông Thành Luân

NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN

........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN

........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

**TÓM TẮT**

Tên đề tài: Xây dựng Hệ thống giao tiếp thời gian thực.

Sinh viên thực hiện: Trần Thông Thành Luân

Số thẻ SV: 102140079 Lớp: 14T2

Người hướng dẫn: ThS. Mai Văn Hà

Hệ thống bao gồm:

- Website giao tiếp thời gian thực thông qua Video Call và Chat.

Các chức năng chính:

- Gọi video call nhiều người trong một phòng mà không cần đăng nhập.

- Mọi người có thể giao tiếp với nhau thông qua Chat.

- Chia sẻ màn hình Screen của mình cho mọi người.

- Người dùng có thể thiết lập các cài đặt về camera hoặc audio của chính mình.

|  |  |
| --- | --- |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | **CỘNG HÒA XÃ HÔI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc |

**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

Họ tên sinh viên: Trần Thông Thành Luân Số thẻ sinh viên: 102140079

Lớp: 14T2 Khoa: Công nghệ thông tin Ngành: Công nghệ thông tin

1. *Tên đề tài đồ án:*Xây dựng hệ thống giao tiếp thời thực

*Đề tài thuộc diện:*  *Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện*

1. *Các số liệu và dữ liệu ban đầu:*

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

1. *Nội dung các phần thuyết minh và tính toán:*

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

1. *Các bản vẽ, đồ thị ( ghi rõ các loại và kích thước bản vẽ ):*

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

1. *Họ tên người hướng dẫn:* ThS. Mai Văn Hà
2. *Ngày giao nhiệm vụ đồ án:*  *18/02/2018*
3. *Ngày hoàn thành đồ án: 21/05/2018*

|  |  |
| --- | --- |
|  | *Đà Nẵng, ngày tháng năm 201* |
| **Trưởng Bộ môn** …………………….. | **Người hướng dẫn** |

MỤC LỤC

**LỜI CẢM ƠN** ......................................................................................................... i

**LỜI CAM ĐOAN** .................................................................................................. iii

**MỤC LỤC** ................................................................................................................vi

**DANH SÁCH HÌNH ẢNH** ..................................................................................... x

**DANH SÁCH CÁC BẢNG** ................................................................................... xi

**DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT** ............................................. xii

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 16](#_Toc514525602)

[1.1. Tổng quan về ngôn ngữ Ruby và Framework Ruby on Rails 16](#_Toc514525603)

[1.1.1. Lịch sử ngôn ngữ Ruby 16](#_Toc514525604)

[1.1.2. Đặc tính của Ruby 16](#_Toc514525605)

[1.1.3. Framework Ruby on Rails 17](#_Toc514525606)

[1.1.4. Thư viện jQuery 18](#_Toc514525607)

[1.1.5. Framework Bootstrap 18](#_Toc514525608)

[1.1.6. Hệ quản trị cơ sở dữ liệu MySQL 19](#_Toc514525609)

[1.1.7. Mô hình MVC trong Ruby on Rails 19](#_Toc514525610)

[1.1.8. Môi trường lập trình Sublime Text 20](#_Toc514525611)

[1.2. Tìm hiểu Google map api 20](#_Toc514525612)

[1.2.1. Giới thiệu về Google map api 20](#_Toc514525613)

[1.3. Ứng dụng bản đồ trong Ruby on Rails thông qua gem Geocoder 22](#_Toc514525614)

[1.3.1. Giới thiệu gem Geocoder 22](#_Toc514525615)

[1.3.2. Sử dụng gem Geocoder 23](#_Toc514525616)

[1.4. Kết chương 23](#_Toc514525617)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 24](#_Toc514525618)

[2.1. Giới thiệu 24](#_Toc514525619)

[2.2. Phân tích và đặc tả yêu cầu 24](#_Toc514525620)

[2.2.1. Giới thiệu 24](#_Toc514525621)

[2.2.2. Mô tả chung 25](#_Toc514525622)

[2.2.3. Yêu cầu giao diện 26](#_Toc514525623)

[2.2.4. Yêu cầu phi chức năng 26](#_Toc514525624)

[2.3. Thiết kế hệ thống 27](#_Toc514525625)

[2.3.1. Các tác nhân trong hệ thống 27](#_Toc514525626)

[2.3.2. Các biểu đồ 27](#_Toc514525627)

[2.3.3. Thiết kế cơ sở dữ liệu 41](#_Toc514525628)

[2.4. Kết chương 49](#_Toc514525629)

[CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 50](#_Toc514525630)

[3.1. Triển khai hệ thống 50](#_Toc514525631)

[3.1.1. Môi trường triển khai 50](#_Toc514525632)

[3.1.2. Cài đặt môi trường 50](#_Toc514525633)

[3.2. Kết quả thực nghiệm 51](#_Toc514525634)

[3.2.1. Giao diện người dùng 51](#_Toc514525635)

[3.2.2. Giao diện quản trị dành cho vai trò admin 59](#_Toc514525636)

[3.3. Nhận xét và đánh giá kết quả 61](#_Toc514525637)

[3.4. Kết chương 62](#_Toc514525638)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 63](#_Toc514525639)

[TÀI LIỆU THAM KHẢO 65](#_Toc514525640)

DANH SÁCH HÌNH ẢNH

*[Hình 1.1: Mô hình MVC](#_Toc514525481)* [20](#_Toc514525481)

*[Hình 1.2: Bản đồ google maps](#_Toc514525482)* [22](#_Toc514525482)

*[Hình 1.3: Chỉ đường thông qua google maps](#_Toc514525483)* [23](#_Toc514525483)

*[Hình 2.1: Use case Tổng quát](#_Toc514525484)* [28](#_Toc514525484)

*[Hình 2.2: Usecase Người dùng](#_Toc514525485)* [29](#_Toc514525485)

*[Hình 2.3: Usecase Thành viên](#_Toc514525486)* [31](#_Toc514525486)

*[Hình 2.4: Usecase quản trị viên](#_Toc514525487)* [34](#_Toc514525487)

*[Hình 2.5: Biểu đồ tuần tự chức năng đăng nhập](#_Toc514525488)* [36](#_Toc514525488)

*[Hình 2.6: Biểu đồ tuần tự chức năng đăng ký](#_Toc514525489)* [36](#_Toc514525489)

*[Hình 2.7: Biểu đồ tuần tự chức năng tìm kiếm](#_Toc514525490)* [37](#_Toc514525490)

*[Hình 2.8: Biểu đồ tuần tự chức năng đăng ký địa điểm](#_Toc514525491)* [38](#_Toc514525491)

*[Hình 2.9: Biểu đồ tuần tự chức năng phê duyệt địa điểm](#_Toc514525492)* [38](#_Toc514525492)

*[Hình 2.10: Biểu đồ tuần tự chức năng đặt món](#_Toc514525493)* [39](#_Toc514525493)

*[Hình 2.11: Biểu đồ tuần tự chức năng đánh giá địa điểm](#_Toc514525494)* [39](#_Toc514525494)

*[Hình 2.12: Biểu đồ hoạt động đăng nhập](#_Toc514525495)* [40](#_Toc514525495)

*[Hình 2.13: Biểu đồ hoạt động đăng ký tài khoản](#_Toc514525496)* [40](#_Toc514525496)

*[Hình 2.14: Biểu đồ hoạt động chức năng tìm kiếm](#_Toc514525497)* [41](#_Toc514525497)

*[Hình 2.15: Biểu đồ hoạt động chức năng tạo mới địa điểm](#_Toc514525498)* [41](#_Toc514525498)

*[Hình 2.16: Biểu đồ hoạt động chức năng đánh giá địa điểm](#_Toc514525499)* [42](#_Toc514525499)

*[Hình 2.17: Biểu đồ hoạt động chức năng đặt món](#_Toc514525500)* [42](#_Toc514525500)

*[Hình 2.18: Cơ sở dữ liệu của hệ thống](#_Toc514525501)* [44](#_Toc514525501)

*[Hình 3.1: Giao diện trang chủ](#_Toc514525502)* [52](#_Toc514525502)

*[Hình 3.2: Giao diện trang chủ (tiếp)](#_Toc514525503)* [53](#_Toc514525503)

*[Hình 3.3: Giao diện trang chi tiết địa điểm](#_Toc514525504)* [53](#_Toc514525504)

*[Hình 3.4: Giao diện bản đồ địa điểm](#_Toc514525505)* [54](#_Toc514525505)

*[Hình 3.5: Hiển thị các món ăn địa điểm phục vụ](#_Toc514525506)* [54](#_Toc514525506)

*[Hình 3.6: Giao diện đăng nhập](#_Toc514525507)* [55](#_Toc514525507)

*[Hình 3.7: Thành viên có thể đặt món](#_Toc514525508)* [55](#_Toc514525508)

*[Hình 3.8: Xác nhận đơn hàng](#_Toc514525509)* [56](#_Toc514525509)

*[Hình 3.9: Giao diện quản lý đơn hàng](#_Toc514525510)* [56](#_Toc514525510)

*[Hình 3.10: Chức năng đánh giá địa điểm](#_Toc514525511)* [57](#_Toc514525511)

*[Hình 3.11: Thêm hình ảnh địa điểm trong đánh giá](#_Toc514525512)* [57](#_Toc514525512)

*[Hình 3.12: Hiển thị bình luận của thành viên](#_Toc514525513)* [57](#_Toc514525513)

*[Hình 3.13: Xem các hình ảnh về địa điểm](#_Toc514525514)* [58](#_Toc514525514)

*[Hình 3.14: Chức năng thêm địa điểm dành cho thành viên](#_Toc514525515)* [58](#_Toc514525515)

*[Hình 3.15: Thêm thông tin cho địa điểm](#_Toc514525516)* [59](#_Toc514525516)

*[Hình 3.16: Chức năng cập nhật thông tin thành viên](#_Toc514525517)* [59](#_Toc514525517)

*[Hình 3.17: Lọc địa điểm xung quanh khu vực theo khoảng cách](#_Toc514525518)* [60](#_Toc514525518)

*[Hình 3.18: Trang chủ giao diện admin](#_Toc514525519)* [60](#_Toc514525519)

*[Hình 3.19: Giao diện quản lý địa điểm](#_Toc514525520)* [61](#_Toc514525520)

*[Hình 3.20: Chức năng thêm địa điểm](#_Toc514525521)* [61](#_Toc514525521)

*[Hình 3.21: Chức năng quản lý người dùng](#_Toc514525522)* [62](#_Toc514525522)

*[Hình 3.22: Chức năng thêm mới người dùng](#_Toc514525523)* [62](#_Toc514525523)

DANH SÁCH CÁC BẢNG

*[Bảng 2.1: Chức năng xem toàn bộ địa điểm](#_Toc514525581)* [28](#_Toc514525581)

*[Bảng 2.2: Chức năng xem thông tin địa điểm](#_Toc514525582)* [31](#_Toc514525582)

*[Bảng 2.3: Chức năng đánh giá địa điểm](#_Toc514525583)* [31](#_Toc514525583)

*[Bảng 2.4: Chức năng đặt món](#_Toc514525584)* [32](#_Toc514525584)

*[Bảng 2.5: Chức năng quản lý người dùng](#_Toc514525585)* [33](#_Toc514525585)

*[Bảng 2.6: Chức năng quản lý địa điểm](#_Toc514525586)* [34](#_Toc514525586)

*[Bảng 2.7: Bảng Users](#_Toc514525587)* [44](#_Toc514525587)

*[Bảng 2.8: Bảng Places](#_Toc514525588)* [44](#_Toc514525588)

*[Bảng 2.9: Bảng UserRatings](#_Toc514525589)* [45](#_Toc514525589)

*[Bảng 2.10: Bảng UserRatingImages](#_Toc514525590)* [45](#_Toc514525590)

*[Bảng 2.11: Bảng PlaceCollections](#_Toc514525591)* [46](#_Toc514525591)

*[Bảng 2.12: Bảng Orders](#_Toc514525592)* [46](#_Toc514525592)

*[Bảng 2.13: Bảng Carts](#_Toc514525593)* [47](#_Toc514525593)

*[Bảng 2.14: Bảng Districts](#_Toc514525594)* [47](#_Toc514525594)

*[Bảng 2.15: Bảng Provices](#_Toc514525595)* [47](#_Toc514525595)

*[Bảng 2.16: Bảng PlacesCategories](#_Toc514525596)* [47](#_Toc514525596)

*[Bảng 2.17: Bảng Foods](#_Toc514525597)* [48](#_Toc514525597)

*[Bảng 2.18: Bảng FoodCategories](#_Toc514525598)* [48](#_Toc514525598)

*[Bảng 2.19: Bảng UserSettings](#_Toc514525599)* [48](#_Toc514525599)

*[Bảng 2.20: Bảng Notifications](#_Toc514525600)* [49](#_Toc514525600)

*[Bảng 3.1: Đánh giá kết quả](#_Toc514525601)* [61](#_Toc514525601)

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| **Từ viết tắt** | **Diễn giải** |
| IP | Internet Protocol |
| API | Application Programming Interface |
| URL | Uniform Resource Locator |
| MVC | Model – View - Controller |

MỞ ĐẦU

**1.**  **Tổng quan về đề tài**

WebRTC là một web API có khả năng hỗ trợ trình duyệt (Browser) giao tiếp với nhau thông qua VideoCall, VoiceCall hay transfer data “Peer-to-Peer” (P2P) mà không cần browser phải cài thêm plugins hay phần mềm hỗ trợ nào từ bên ngoài.

Nói cách khác, WebRTC là tập hợp các tiêu chuẩn và giao thức cho phép các trình duyệt Web thực hiện trực tiếp các tính năng truyền thông đa phương tiện thời gian thực như Video Call, Chat, truyền dữ liệu bằng các API Javacript.

Hiện nay WebRTC đang được chuẩn hóa ở cấp độ API của W3C và cấp độ giao thức của IETF, những thứ này đang được hỗ trợ bởi các trình duyệt Google Chrome, Mozilla Firefox, Opera trên PC và Android.

**2. Mục đích và ý nghĩa của đề tài**

**2.1. Mục đích**

Đồ án tập trung tìm hiểu về công nghệ WebRTC, các APIs của trình duyệt, các giao thức được WebRTC sử dụng để có thể chia sẻ và truyền dữ liệu trực tiếp thời gian thực giữa các trình duyệt trong môi trường mạng. Đồ án cũng phân tích yêu cầu tính chất “thời gian thực” khi truyền dữ liệu media và cách thức WebRTC đang được xây dựng để giải quyết, cũng như cách thức vượt NAT, Firewall để thiết lập kết nối Peer to Peer.

**2.2. Ý nghĩa**

Hệ thống cho phép người dùng có thể gọi Video call cho nhiều người trong cùng một phòng mà không cần đăng nhập hoặc cài đặt thêm gì cả. Ngoài ra, người dùng còn có thể chia sẻ màn hình của mình cho những người khác. Màn hình được chia sẻ có thể là toàn bộ screen hoặc là một phần của screen tùy theo nhu cầu người sử dụng.

Trong khi meeting, người dùng có thể trao đổi với nhau thông qua cách gửi tin nhắn chat. Hệ thống sẽ lưu lại lịch sử chat để những người mới vào phòng cũng có thể xem được những đoạn hội thoại trước đó.

**3. Phương pháp thực hiện**

Công nghệ áp dụng: WebRTC, React - Redux, Node.js, socket.io, styled-components

Công cụ thực hiện: VSCode.

Ngôn ngữ lập trình: JavaScript, HTML, CSS.

Thư viện bổ trợ: Antd

**3. Bố cục của đồ án**

Đồ án bao gồm các nội dung sau:

Mở đầu

Chương 1: Cơ sở lý thuyết

Chương 2: Phân tích thiết kế hệ thống

Chương 3: Triển khai và đánh gia kết quả

Kết luận và hướng phát triển.

# CƠ SỞ LÝ THUYẾT

## 1. Giới thiệu về công nghệ WebRTC

* 1. **Khái niệm WebRTC**

Theo **Arin Sime** - CEO của công ty AgilityFeat có trụ sở tại Charlottesville, bang Virginia, Hoa Kỳ, người đã đưa ra định nghĩa rất ngắn gọn về WebRTC như sau:

**“WebRTC là một tiêu chuẩn HTML5 cho mã hóa peer-to-peer (P2P) trao đổi video, âm thanh và dữ liệu trực tiếp giữa hai máy tính”**.

Các tính năng nổi bật nhất của WebRTC đó là được tích hợp trực tiếp vào trình duyệt, chúng ta có thể dùng HTML5 và JavaScript để sử dụng nó.

WebRTC không chỉ là một sản phẩm hay một hàm API duy nhất. Nó là một tập hợp rất nhiều các hàm, có thể được lập trình viên sử dụng cho nhiều mục đích khác nhau. Có hàm chỉ để làm những việc đơn giản như đòi quyền truy cập vào camera và microphone của máy tính, có hàm phức tạp hơn thì để thiết lập kết nối giữa hai người dùng với nhau, có hàm còn dùng để chia sẻ màn hình với người dùng khác. Và có hàm để kết nối (Gọi) video, đây là chức năng quan trọng nhất của WebRTC.

Tuy nhiên, tất cả các hàm lập trình nằm trong bộ API có đặc điểm chung là thực thi hầu hết các tác vụ theo thời gian thực. Và nó không chỉ được dùng cho việc gọi video giữa hai trình duyệt mà còn có thể thực hiện nhiều tác vụ khác, để hai hoặc nhiều người dùng liên lạc với nhau. WebRTC được hỗ trợ chính thức bởi Google, Mozilla, Opera cùng nhiều hãng khác.

* 1. **Đặc điểm của WebRTC**
     1. **Chuẩn Video**

WebRTC chưa có chuẩn video thống nhất. Đặc điểm nổi bật nhất của WebRTC là khả năng kết nối video từ trình duyệt web của người sử dụng. Nhưng hiện nay, các hãng trình duyệt cũng chưa thống nhất với nhau là chuẩn video nào sẽ được dùng cho WebRTC.

Google và Mozilla thì muốn dùng VP8 hoặc VP9, một codec video do chính Google phát triển theo mô hình mã nguồn mở. Thực tế cho thấy, đối với trình duyệt web Chrome 50 trở lên, codec VP9 tốt hơn. Nó hỗ trợ cho chất lượng tốt hơn trong truyền tải phương tiện truyền thông nhờ khả năng dự phòng (RED), cơ chế FEC tương tác với ước lượng băng thông và kiểm soát các thông số (NACK, POI, RTX, remb, ...) trong trình duyệt, khả năng tương tác với các trình duyệt khác tốt hơn VP8. Do đó, VP9 cho độ phân giải cao hơn, giảm thiểu các lỗi (nhờ hỗ trợ cho Error Correction Forward).

Trong khi đó, Microsoft và một số công ty khác thì muốn đề xuất sử dụng H.264 hoặc H.265 cho WebRTC, vốn đang là codec được dùng phổ biến nhất hiện nay trên Internet. Và tất nhiên, H.264 lại thuộc quyền sở hữu của Hiệp hội MPEG LA và phải trả phí bản quyền để sử dụng.

Hiện nay, nhóm làm việc của Google gồm Niklas Blum, Justin Uberti và Per Ahgren đã thực hiện được những cải tiến về hiệu suất, đó là trình duyệt Chrome đã có thể bớt thời gian kết nối trung bình từ 2 giây xuống còn 650ms cho một kết nối 1Mbps. Họ cũng đang làm việc để mang lại VP9 – codec video đến các thiết bị di động, nhờ đó sẽ cho phép đạt chất lượng video cao với yêu cầu băng thông ít hơn.

* + 1. **Tính an toàn, bảo mật của công nghệ WebRTC**
       1. **Tính an toàn của WebRTC**

- Mã hóa là bắt buộc đối với tất cả thành phần WebRTC, bao gồm các cơ chế báo hiệu (Signaling)

- Truy xuất camera và microphone phải được cấp quyền rõ ràng và khi camera hoặc microphone đang hoạt động thì phải được thể hiện ra ở phía giao diện người dùng.

- Thành phần của WebRTC chạy trong sandbox của trình duyệt, các thành phần không yêu cầu cài đặt riêng và được cập nhật bất cứ khi nào trình duyệt được cập nhật. Sandbox là một kỹ thuật qua trọng trong lĩnh vực bảo mật có tác dụng cô lập các ứng dụng, ngăn chặn các phần mềm độc hại để chú không thể làm hỏng hệ thống máy tính, hay cài cắm các mã độc nhằm ăn cắp thông tin cá nhân của người sử dụng.

* + - 1. **Tính bảo mật của WebRTC**

Công nghệ WebRTC sử dụng giao thức bảo mật DTLS và SRTP:

- DTLS (Datagram Transport Layer Security) cho phép các ứng dụng được thiết kế để ngăn chặn việc nghe trộm, giả mạo, hoặc tin nhắn giả mạo. Các gói giao thức DTLS đảm bảo các ứng dụng khô bị chậm trễ kết hợp với việc xử lý khi mất gói tin và dữ liệu.

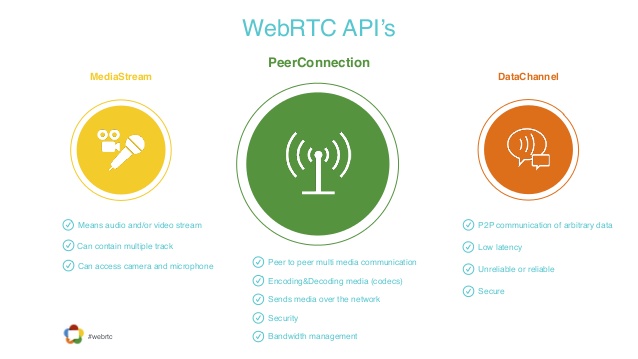
- Giao thức SRTP (Secure Real-time Trn sport Protocol) cung cấp tính bảo mật, tính toàn vẹn và xác thực thông báo trong môi trường truyền thông (tiếng nói và hình ảnh). Bằng việc sử dụng thuật toán dẫn xuất khoá nguyên thuỷ, SRTP có khả năng tối thiểu hoá tính toán và sử dụng tài nguyên để sinh các khoá mật mã qua cơ chế quả lý khoá ngoài.

* + - 1. **WebRTC không phải là một plugin**

- Thành phần của WebRTC chạy trong sandbox của trình duyệt, các thành phần không yêu cầu cài đặt riêng và được cập nhật bất cứ khi nào trình duyệt được cập nhật.

- Sandbox là một kỹ thuật qua trọng trong lĩnh vực bảo mật có tác dụng cô lập các ứng dụng, ngăn chặn các phần mềm độc hại để chú không thể làm hỏng hệ thống máy tính, hay cài cắm các mã độc nhằm ăn cắp thông tin cá nhân của người sử dụng.

* 1. **Các API của WebRTC**



WebRTC có ba loại hàm API thường được dùng.

* + - 1. **Media Capture and Stream**

API **Media Capture & Streams**, thường được gọi là Media Stream API hoặc Stream API, là 1 API hỗ trợ các phương thức để làm việc với những luồng (Stream) dữ liệu âm thanh hay hình ảnh.

Các API này cho phép bạn truy xuất vào các thiết bị đầu vào, chẳng hạn như microphone hay web camera và lấy ra luồng media từ các thiết bị đó.

Phương thức **getUserMedia**() của **MediaDevices** nhắc nhở người dùng cấp quyền để sử dụng đầu vào media, nó tạo ra 1 MediaStream kèm với các track chứa kiểu request của media. Luồng đó có thể chứa 1 track video (được tạo ra bởi phần cứng hay nguồn video ảo chẳng hạn như camera, thiết bị ghi video, dịch vụ chia sẻ màn hình,...), 1 track audio (được tạo ra bởi nguồn ghi vật lý hoặc nguồn ghi ảo như microphone, bộ chuyển A/D...) và có thể có cả những loại track khác.

**getUserMedia()** sẽ trả về 1 Promise với resolve là object MediaStream. Nếu người dùng từ chối cấp quyền hoặc là media phù hợp không tồn tại thì promise đó sẽ bị reject với lỗi là PermissionDeniedError hoặc NotFoundError.

Mô hình singleton MediaDevice có thể được truy xuất thông qua object navigator như sau:

navigator.mediaDevices.getUserMedia(constraints)

.then(function(stream) {

/\* use the stream \*/

})

.catch(function(err) {

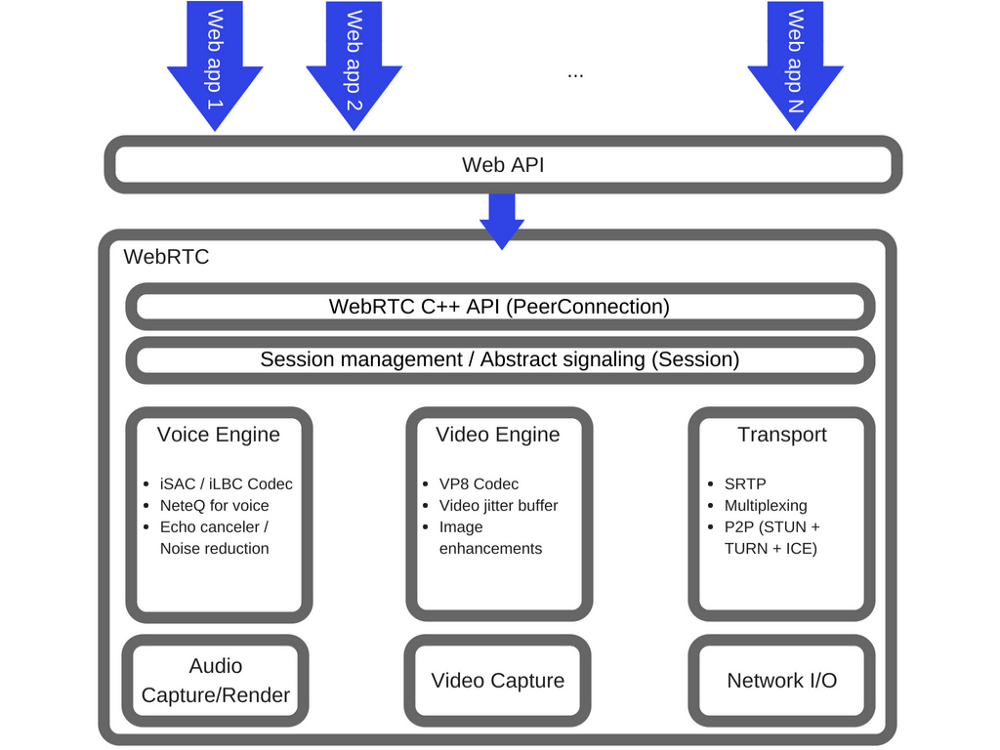
/\* handle the error \*/

});

Ta cần truyền object constraints để nói cho API loại stream nào để trả về. Ta có thể cấu hình tất cả những thứ liên quan đến camera và microphone như: Camera trước / sau, tần suất khung hình, độ phân giải,...

* + - 1. **RTCPeerConection**

RTCPeerConnection thể hiện 1 kết nối WebRTC giữa máy tính local và remote. Nó cung cấp các phương thức để kết nối đến peer từ xa, duy trì, kiểm soát kết nối và ngắt kết nối một khi nó không cần dùng tới nữa.



Các mã codec và giao thức được dùng bởi WebRTC thực hiện 1 lượng lớn công việc dể làm cho giao tiếp real-time hoạt động được, kể cả với các mạng không đáng tin cậy:

- Che dấu mất gói tin (Package loss concealment - PLC, là 1 kỹ thuật dùng để che đậy ảnh hưởng của tình trạng mất gói tin trong giao tiếp VoIP)

- Hủy bỏ phản hồi (Echo cancellation - trong mạng máy tính thì "echo" được hiểu là quá trình gửi trả về gói tin đã gửi đi)

- Khả năng thích ứng băng thông

- Bộ đệm jitter động (jitter buffer - Trong kỹ thuật VoIP thì jitter buffer là vùng dữ liệu chia sẻ nơi mà các gói tin âm thanh có thể được thu thập, lưu trữ và gửi đi đến bộ xử lý âm thanh)

- Tự động chiếm quyền kiểm soát

- Giảm nhiễu và xóa nhiễu

- "Dọn dẹp" hình ảnh

* + - 1. **RTCDataChannel**

Cũng như hình ảnh và âm thanh, WebRTC cũng hỗ trợ giao tiếp real-time cho các loại dữ liệu khác. API RTCDataChannel cho phép peer-to-peer trao đổi các dữ liệu tùy ý.

Có rất nhiều trường hợp sử dụng API này, ví dụ: Dùng trong các game, Ứng dụng chat real-time, Truyền file, Các mạng phi tập trung,...

API cũng có nhiều tính năng để tận dụng tối đa RTCPeerConnection và kích hoạt sức mạnh cũng như sự linh động của giao tiếp peer-to-peer:

- Tận dụng cài đặt phiên của RTCPeerConnection

- Đa kênh đồng thời với khả năng phân chia mức độ ưu tiên

- Ngữ cảnh vận chuyển đáng tin cậy và không đáng tin cậy.

- Tích hợp sẵn bảo mật (DTLS) và kiểm soát tắc nghẽn.

Sự giao tiếp diễn ra trực tiếp giữa các trình duyệt, vì thế RTCDataChannel có thể nhanh hơn nhiều so với WebSocket kể cả khi cần đến 1 server chuyển tiếp (TURN).

* 1. **Giao tiếp Peer-to-peer**

Để có thể giao tiếp lẫn nhau thông qua trình duyệt web, mỗi trình duyệt của user phải thực hiện những bước sau đây:

- Đồng ý để bắt đầu giao tiếp

- Biết cách xác định vị trí của đối tượng

- Vượt qua an ninh và tưởng lửa bảo vệ

- Chuyển giao tất cả các giao tiếp đa phương tiện theo real-time

Một trong số những thách thức lớn nhất liên quan đến các giao tiếp P2P dựa trên trình duyệt là làm sao để biết vị trí & thiết lập một kết nối socket mạng (Network socket connection) với một trình duyệt khác để vận chuyển dữ liệu hai chiều. Ta sẽ xem xét những khó khăn liên quan đến việc thiết lập kết nối này.

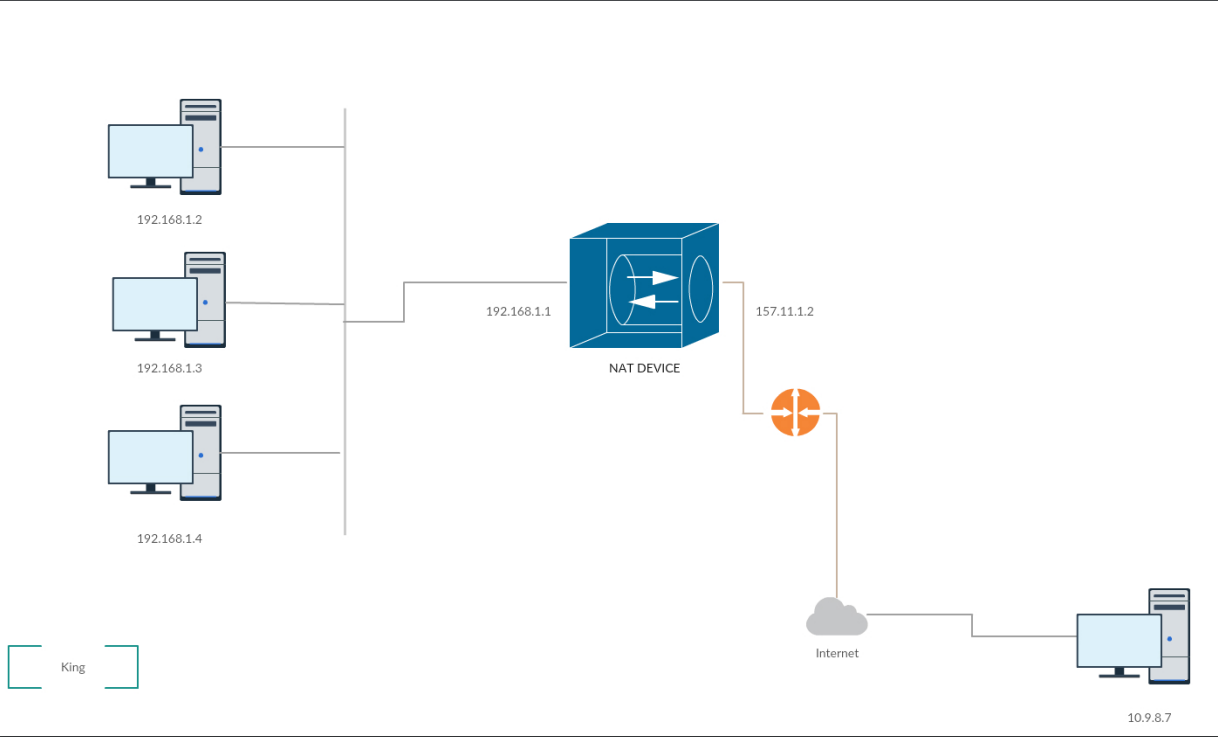
Khi WebApp cần dữ liệu hoặc tài nguyên, nó sẽ lấy về từ server thông qua các request. Tuy nhiên, nếu muốn tạo ra một ứng dụng video chat bằng cách kết nối trực tiếp đến trình duyệt của người khác thì đây lại là một vấn đề. Ta không biết địa chỉ cụ thể của họ bởi vì trình duyệt của người kia không phải là một web server.

* 1. **Tường lửa và NAT Traversal**

IMG_256

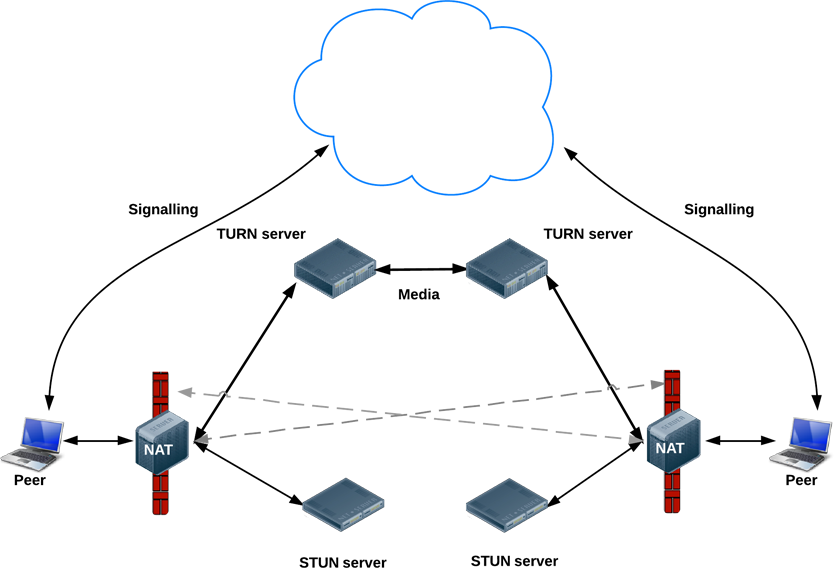
Thường thì máy tính của người dùng khôngcó địa chỉ IP public tĩnh. Lý do là máy tính của người dùng phải ở đằng sau tường lửa và thiết bị NAT (Network access translation - Bộ phiên dịch truy cập mạng).

Thiết bị NAT sẽ dịch địa chỉ IP cá nhân từ bên trong tường lửa thành địa chỉ IP công khai (Public-facing IP). Ta cần các thiết bị NAT để bảo mật và giải quyết sự giới hạn của IPv4 đối với những địa chỉ IP công khai sẵn có. Đó là lý do tại sao WebApp không nên giả định rằng thiết bị hiện tại có 1 địa chỉ IP public tĩnh.



Nếu như bạn đang ở trong môi trường công cộng và kết nối vào mạng WiFi, máy tính của bạn sẽ được gán 1 địa chỉ IP mà nó chỉ tồn tại đằng sau NAT. Giả sử IP là **172.0.23.4**, tuy nhiên, với thế giới bên ngoài, địa chỉ IP của bạn có thể mang giá trị khác, ví dụ **164.53.27.98**. Vì vậy, thế giới bên ngoài sẽ thấy các request của bạn đến từ địa chỉ **164.53.27.98** nhưng thiết bị NAT sẽ đảm bảo các response cho những request (Được gửi từ máy của bạn) sẽ trả về đúng chỗ là **172.0.23.4**. Nhờ vào các bảng ánh xạ (Mapping table). Ngoài địa chỉ IP thì cổng (Port) cũng là điều kiện cần thiết cho các giao tiếp mạng.

Do có sự tham gia của các thiết bị NAT, trình duyệt của người dùng cần tìm được địa chỉ IP của máy tính có trình duyệt mà họ muốn giao tiếp.



Đến đây thì ta lại phải nhờ đến các server **STUN** (**S**ession **T**raversal **U**tilities for **N**AT - Tiện ích truyền tải theo phiên cho NAT) và **TURN** (**T**raversal **U**sing **R**elays around **N**AT - Truyền tải sử dụng điểm chuyển tiếp vòng quanh NAT). Để các công nghệ WebRTC hoạt động được, đầu tiên một request hỏi địa chỉ IP public của bạn sẽ được gửi đến server STUN. Ta cứ nghĩ theo hướng kiểu như máy tính đang tạo truy vấn đến 1 server từ xa để hỏi về địa chỉ IP mà server đó nhận câu truy vấn là bao nhiêu. Server từ xa sẽ trả về địa chỉ IP Public mà nó thấy.

Cũng giống như STUN tuy nhiên TURN hỗ trợ cả giao thức TCP làm giao thức truyền tải. TURN bổ xung cho hạn chế của STUN là hỗ trợ Symmetric NAT. Dữ liệu thay vì được gửi trực tiếp tới các peer thì các peer sẽ gửi dữ liệu tới các TURN server và TURN server sẽ đóng vai trò trung gian vận chuyển gói tin. Điều này nâng cao giúp chất lượng dịch vụ của ứng dụng mà còn đảm bảo an toàn thông tin khi truyền dẫn. Bất lợi của TURN là chi phí sử dụng lớn, vì sẽ có một lưu lượng băng thông lớn được sử dụng.

Giả sử tiến trình này hoạt động bình thường và ta nhận được địa chỉ IP public của mình cũng như số port. Sau đó ta có thể nói với những peer ngang hàng khác làm thế nào để kết nối trực tiếp đến mình. Những peer này cũng có thể làm cùng một việc là sử dụng STUN & server TURN và nói cho ta biết nên liên lạc đến địa chỉ nào.

* 1. **ICE (Interactive Communication Establishment)**

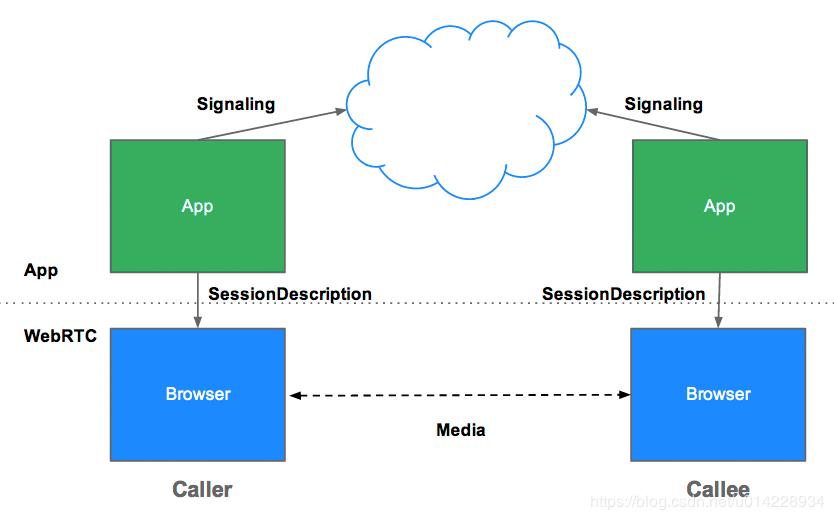
ICE là một giao thức được dùng để thiết lập phiên media dựa trên UDP đi qua NAT một cách nhanh nhất. ICE sẽ tìm đường tốt nhất để kết nối giữa các peer, nó thử tất cả khả năng có thể kết nối một cách song song và lựa chọn con đường hiệu quả nhất.

ICE bao gồm: Những metadata cần thiết, các định tuyến mạng (Địa chỉ IP và port) và các thông tin media thường dùng để giao tiếp với mỗi peer.

Đầu tiên ICE sẽ cố gắng tạo ra một kết nối bằng cách sử dụng địa chỉ thu được từ hệ điều hành và card mạng của thiết bị, nếu không thành công (Có thể do thiết bị ở đằng sau NAT) thì ICE sẽ lấy Public IP của thiết bị bằng cách sử dụng máy chủ STUN, nếu không thành công nữa thì nó sẽ chuyển lưu lượng mạng qua một máy chủ chuyển tiếp là TURN.

Việc sử dụng server TURN về cơ bản là dùng một server hoạt động như người trung gian và nó chuyển tiếp bất kỳ dữ liệu nào truyền qua lại giữa các peer. Lưu ý rằng đây không phải là giao tiếp peer-to-peer thực thụ trong đó các peer truyền dữ liệu hai chiều trực tiếp đến với nhau. Mỗi peer sẽ không cần phải biết làm thế nào để liên lạc và truyền dữ liệu đến bên kia. Thay vào đó, nó cần biết server TURN public nào để gửi và nhận dữ liệu đa phương tiện theo thời gian thực xuyên suốt phiên giao tiếp.

* 1. **Signaling, Sessions, Protocols**



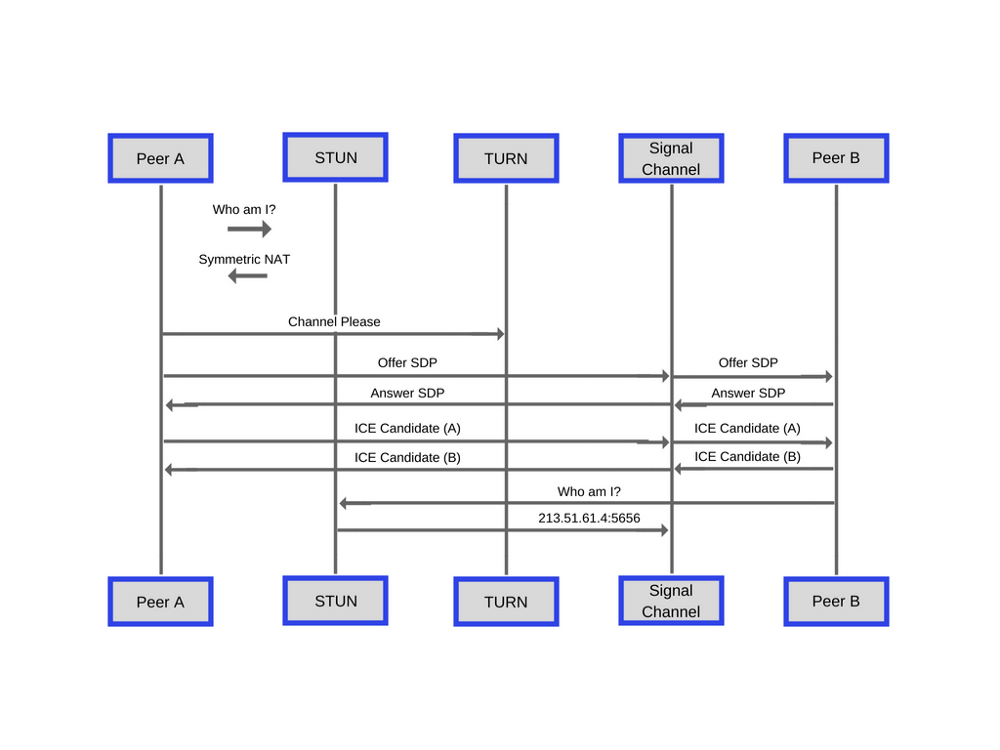
Signaling dựa trên 1 chuẩn JSEP (**J**avascript **S**ession **E**stablishment **P**rotocol - Giao thức thiết lập phiên của Javascript). Signaling bao gồm cả khám phá mạng (Network discovery), NAT Traversal, tạo và quản lý phiên, bảo mật giao tiếp, siêu dữ liệu (Metadata) và phối hợp khả năng của media, xử lý lỗi.

Để kết nối có thể hoạt động, peer thu nhập những điều kiện về local media cho metadata (Ví dụ: Chuẩn màn hình, kiểu codec,...) và gom góp các địa chỉ mạng có thể có cho host của ứng dụng. Cơ chế signaling dùng để truyền tới / lui những thông tin quan trọng này không được định nghĩa trong API của WebRTC.

Signaling không được quy định bởi chuẩn WebRTC và nó không được triển khai bằng WebRTC API để cho phép sử dụng một cách linh động các công nghệ và giao thức cần thiết (XMPP, SIP, Socket,...)

Giao thức signaling được chọn phải hoạt động với 1 giao thức ở tầng ứng dụng gọi là **S**ession **D**escription **P**rotocol (SDP - Giao thức mô tả phiên), giao thức này được sử dụng trong trường hợp của WebRTC. Tất cả các metadata đa phương tiện cụ thể được truyền đi bằng giao thức SDP này.

Bất kỳ peer nào khi giao tiếp với một peer khác đều sinh ra một tập các ứng viên giao thức Interactive Connectivity Establishment (ICE Candicate). Những ứng viên này biểu diễn 1 bộ kết hợp của địa chỉ IP, port, giao thức giao vận được dùng. Lưu ý rằng một máy tính có thể có nhiều giao diện mạng (Không dây, có dây,...), vì thế có thể được gán nhiều địa chỉ IP cho mỗi giao diện.



Một khi 2 hoặc nhiều peer đã cùng kết nối vào 1 "kênh" thì những peer đã có thể giao tiếp và trao đổi thông tin phiên. Về cơ bản, peer khởi tạo ban đầu sẽ gửi 1 "lời đề nghị" (CreateOffer) sử dụng một giao thức signaling chẳng hạn như Session Initiation Protocol (SIP - Giao thức khởi tạo phiên) và SDP. Người khởi tạo sẽ chờ để nhận được "câu trả lời" (CreateAnswer) từ bất kỳ người nhận nào đã kết nối với "kênh". Khi đã nhận được câu trả lời, 1 tiến trình diễn ra để xác định và trao đổi giao thức ứng viên ICE tốt nhất mà mỗi peer thu thập được.

### Lịch sử ngôn ngữ Ruby

Ruby là một ngôn ngữ lập trình hướng đối tượng được sử dụng để tạo ra các loại ứng dụng web và di động. Ruby được tạo ra bởi Yukihiro "Matz" Matsumoto từ 24 tháng 2 năm 1993 và đưa ra bản chính thức vào năm 1995.

Tháng 9 năm 2005, phiên bản ổn định mới nhất là 1.8.3. Ruby 1.9 (với một số thay đổi) cũng đang trong quá trình phát triển.

Hiện tại phiên bản mới nhất đang là Ruby 2.5.1.

Nguồn gốc của cái tên "Ruby" là từ một phiên chat online giữa Matsumoto và Ishitsuka Keiji vào ngày 24 tháng 2 năm 1993, trước khi viết code cho ngôn ngữ này. Ban đầu, "Coral" và "Ruby" là hai cái tên được đề xuất. Matsumoto chọn cái cuối cùng trong email gửi đến Ishitsuka. Sau này Matz cũng bất ngờ khi phát hiện ra Pearl là viên đá quý tượng trưng cho những người sinh tháng 6, còn Ruby thì tượng trưng cho những người sinh tháng 7. Anh cho rằng cái tên Ruby như thế là phù hợp vì Ruby kế thừa và phát triển nhiều đặc tính từ Perl.

### Đặc tính của Ruby

Ruby là một ngôn ngữ hướng đối tượng: mỗi bit dữ liệu đều là một đối tượng (object), bao gồm các kiểu dữ liệu mà đối với các ngôn ngữ khác, chúng là kiểu cơ bản (primitive) như integer. Mỗi hàm (function) là một phương thức (method). Tên biến (variables) chính là tham chiếu (references) đến các đối tượng, bản thân nó không phải là đối tượng.

Ruby hỗ trợ kế thừa (inheritance) với dynamic dispatch, mixin và singleton method (thuộc về, và để định nghĩa cho, một instance đơn hơn là định nghĩa dành cho lớp). Mặc dù Ruby không hỗ trợ đa kế thừa, các lớp vẫn có thể được đưa vào các module dưới dạng các mixins. Cú pháp dạng thủ tục (procedural syntax) vẫn còn được hỗ trợ, có vẻ như là ngoài tầm vực của mọi đối tượng, nhưng thực sự là thuộc một thể hiện của class Object tên là 'main'. Vì class này là cha của mọi class khác, nó trở trên ẩn đối với mọi lớp và đối tượng.

Ruby cung cấp một hệ cú pháp đơn giản, dễ đọc, và tương tự như ngôn ngữ dùng để giao tiếp hằng ngày. Ngôn ngữ này giúp lập trình viên dễ dàng hơn trong việc lập trình bởi cú pháp ngắn gọn, ít các dấu ngoặc hay chấm phẩy.

### Framework Ruby on Rails

Ruby on Rails là một framework được viết bằng ngôn ngữ Ruby theo giấy phép MIT License. Ruby on Rails hoạt động theo mô hình MVC, cung cấp các cấu trúc dữ liệu mặc định cho cơ sở dữ liệu, cung cấp dịch vụ web và các trang web. Cho phép server trả về theo nhiều định dạng khác nhau như xml để truyền dữ liệu, json cũng như html, css và javascript để hiện thị giao diện tới người dùng.

Ngoài MVC, Rails nhấn mạnh việc sử dụng các mẫu và mô hình kỹ thuật phần mềm nổi tiếng khác, bao gồm quy ước về cấu hình (CoC), không lặp lại chính mình (DRY) và mẫu bản ghi hoạt động

Sự nổi lên của Ruby on Rails trong những năm 2000 đã ảnh hưởng rất lớn đến phát triển ứng dụng web, thông qua các tính năng sáng tạo như sáng tạo bảng cơ sở dữ liệu liền mạch và cấu trúc để cho phép phát triển ứng dụng nhanh chóng.

David Heinemeier Hansson thiết kế Ruby on Rails từ công việc của mình lúc làm project manager tại Basecamp. Hansson ra mắt phiên bản đầu tiên của Ruby on Rails vào tháng 7 năm 2004 dưới dạng mã nguồn mở nhưng đến tháng 2 năm 2005 mới chia sẻ đầy đủ về framwork này. Tháng 8 năm 2006 framework này đạt được mốc quan trọng khi công ty Apple công bố rằng sẽ đưa Ruby on Rails tới Mac OS X v10.5 "Leopard", được phát hành vào tháng 10 năm 2007.

Rails phiên bản 2.3 đã được phát hành vào ngày 15 tháng 3 năm 2009, với những phát triển mới trong các khung giao diện, công cụ, Rack và các mẫu biểu mẫu lồng nhau. Các mẫu cho phép nhà phát triển tạo ra một ứng dụng khung với các cấu hình và các Gem tùy chỉnh. Engine này cung cấp cho nhà phát triển khả năng sử dụng lại các phần ứng dụng hoàn chỉnh với các router, đường dẫn url và các model. Giao diện máy chủ web Rack và Metal cho phép một người viết các đoạn mã được tối ưu hóa xung quanh Action Controller.

Sau các phiên bản nối tiếp, Rails 5.1 được phát hành vào ngày 27 tháng 4 năm 2017, giới thiệu các thay đổi tích hợp JavaScript (quản lý các phụ thuộc JavaScript từ NPM qua Yarn, biên dịch JavaScript tùy chọn bằng Webpack và viết lại Rails UJS để sử dụng Vanila JavaScript thay vì phụ thuộc vào jQuery), kiểm tra hệ thống sử dụng Capybara, mã hóa ký tự, các params truyền lên, xử lý router và trình trợ giúp form\_with hợp nhất thay thế form\_tag / form\_for trợ giúp giúp lập trình viên dễ dàng làm việc với các form hơn.

Với việc hỗ trợ đơn giản hóa các thao tác của lập trình viên, Ruby on Rails được nhiều lập trình viên yêu thích. Ruby on Rails là một trong những lựa chọn hàng đầu cho việc phát triển website với đầy đủ các chức năng trong thời gian ngắn nhất, tốn ít chi phí nhất.

### Thư viện jQuery

jQuery là một thư viện kiểu mới của JavaScript, được tạo bởi John Resig vào năm 2006 với một phương châm: *Write less, do more - Viết ít hơn, làm nhiều hơn*.

jQuery làm đơn giản hóa việc truyền tải HTML, xử lý sự kiện, tạo hiệu ứng động và tương tác Ajax. Với jQuery, khái niệm Rapid Web Development đã không còn quá xa lạ.

jQuery là một bộ công cụ tiện ích JavaScript làm đơn giản hóa các tác vụ đa dạng với việc viết ít code hơn. Dưới đây liệt kê một số tính năng tối quan trọng được hỗ trợ bởi jQuery:

* Thao tác DOM − jQuery giúp dễ dàng lựa chọn các phần tử DOM để duyệt một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là Sizzle.
* Xử lý sự kiện − jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler.
* Hỗ trợ AJAX − jQuery giúp bạn rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX.
* Hiệu ứng − jQuery đi kèm với rất nhiều các hiệu ứng đa dạng và đẹp mắt mà bạn có thể sử dụng trong các Website của mình.
* Gọn nhẹ − jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped).
* Được hỗ trợ hầu hết bởi các trình duyệt hiện đại − jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+
* Cập nhật và hỗ trợ các công nghệ mới nhất − jQuery hỗ trợ CSS3 Selector và cú pháp XPath cơ bản.

### Framework Bootstrap

Bootstrap là một framework cho phép thiết kế website reponsive nhanh hơn và dễ dàng hơn.

Bootstrap là bao gồm các HTML templates, CSS templates và Javascript tao ra những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm các plugin Javascript trong nó. Giúp cho việc thiết kế reponsive của bạn dễ dàng hơn và nhanh chóng hơn.

Bootstrap là dược phát triển bởi Mark Otto và Jacob Thornton tại Twitter. Nó được xuất bản như là một mã nguồn mở vào tháng 8 năm 2011 trên GitHub.

Bootstrap được sử dụng đơn giản bằng cách khai báo các class trong các thẻ html của trang web.

Framework này tương thích với hầu hết các trình duyệt hiện nay(Chrome, Firefox, Internet Explorer, Safari, and Opera).

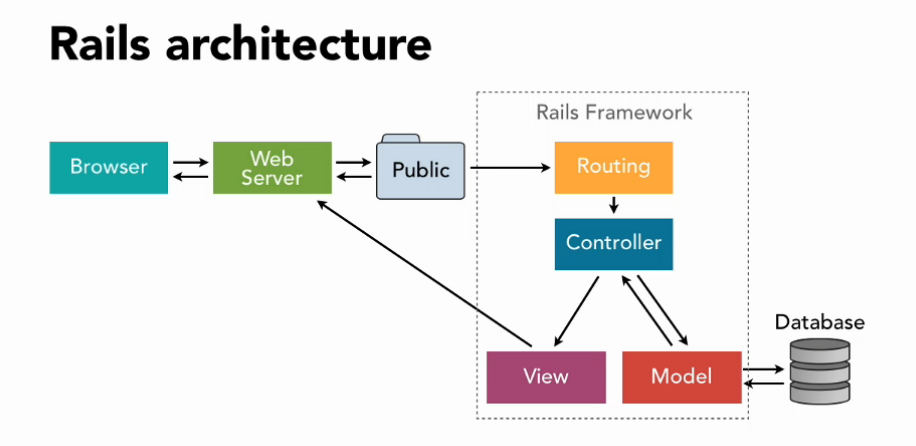
### Hệ quản trị cơ sở dữ liệu MySQL

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,...

MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

MySQL được sử dụng cho việc bổ trợ NodeJs, PHP, Perl, và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng NodeJs, PHP hay Perl,...

### Mô hình MVC trong Ruby on Rails



*Hình 1.1: Mô hình MVC*

* **Model**:

Có nhiệm vụ thao tác với cơ sở dữ liệu, nghĩa là nó sẽ chứa tất cả các hàm, các phương thức truy vấn trực tiếp với dữ liệu và controller sẽ thông qua các hàm, phương thức đó để lấy dữ liệu rồi gửi qua View.

* **View**:

Có nhiệm vụ tiếp nhận dữ liệu từ controller và chuyển đổi nội dung sang các đoạn mã HTML, view đảm nhận các công việc liên quan các thành phần giao diện.

* **Controller**:

Đóng vài trò trung gian giữa Model và View. Nó có nhiệm vụ tiếp nhận yêu cầu từ client sau đó xử lý request, load model tương ứng và gửi dữ liệu qua view tương ứng rồi trả kết quả về cho trình duyệt.

Khi người dùng thao tác trên trình duyệt như nhập địa chỉ trên Url, kích vào một đường link hay gửi một form nào đó, trình duyệt sẽ gửi một yêu cầu lên server, và ở server, ứng dụng Rails xử lý yêu cầu đó. Router sẽ xác minh yêu cầu đó có tương ứng với controller nào không. Nếu yêu cầu đó tồn tại, các thông số sẽ được xử lý trong controller tương ứng. Rails cung cấp đầy đủ các phương thức của HTTP như GET, POST, PATCH, PUT hay DELETE.

Tại Controller, các thao tác với Model sẽ xảy ra ở đây như gọi dữ liệu, thêm xóa hay cập nhật các đối tượng. Controller là nơi trung gian để tiếp nhận và xử lý các yêu cầu của người dùng. Các yêu cầu sẽ được xử lý, sau đó các thay đổi nếu có sẽ được cập nhật vào cơ sở dữ liệu thông qua Model. Model trong Rails là nơi xảy ra các thao tác với cơ sở dữ liệu như cập nhật, thêm, sửa xóa các bản ghi thông qua sự hỗ trợ của ORM. ORM trong Rails là mặc định là Active Record, nơi cung cấp các phương thức để thao tác với cơ sở dữ liệu thông qua các hàm được hỗ trợ sẵn. Ví dụ muốn xóa một bản ghi, chúng ta sử dụng hàm delete, cập nhật có phương thức update, update\_attributes... Thông qua các phương thức đó, Active Record sẽ tạo các câu truy vấn tương ứng lên cơ sở dữ liệu và trả về kết quả nếu cần.

Sau các thao tác lên cơ sở dữ liệu, Controller sẽ trả lời browser thông qua response trả về kèm dữ liệu từ database nếu có thông qua View.

### Môi trường lập trình Sublime Text

Sublime text là một công cụ lập trình nhẹ nhàng, giao diện khá đơn giản tuy vậy được đánh giá là một công cụ rất mạnh, các plugin đa dạng, hỗ trợ rất nhiều ngôn ngữ lập trình. Sublime text khởi động rất nhanh và có thể chạy tốt trên các máy có cấu hình thấp, hỗ trợ rất nhiều phím tắt tiện lợi cho việc lập trình, đặc biệt hỗ trợ tốt cho việc lập trình web.

## Tìm hiểu Google map api

### Giới thiệu về Google map api

Google Map là một dịch vụ ứng dụng và công nghệ bản đồ trực tuyến trên web và application miễn phí được cung cấp bởi Google, hỗ trợ nhiều dịch vụ khác của Google nổi bật là dẫn đường. Nó cho phép hiển thị bản đồ đường sá, đường đi cho xe đạp, xe máy, cho người đi bộ và xe hơi, và những địa điểm kinh doanh trong khu vực cũng như khắp nơi trên thế giới.

Google cung cấp api cho các lập trình viên sử dụng trong các ứng dụng di động cũng như các ứng dụng web thông qua key đã đăng ký.

Map API là một phương thức cho phép 1 website B sử dụng dịch vụ bản đồ của website A (gọi là Map API) và nhúng vào website của mình (site B). Site A ở đây là google map, site B là các website cá nhân hoặc tổ chức muốn sử dụng dịch vụ của google (di chuột, zoom, đánh dấu trên bản đồ, chỉ đường…).

Các ứng dụng xây dựng trên maps được nhúng vào trang web cá nhân thông qua các thành phần trong trang do vậy việc sử dụng API google rất dễ dàng.

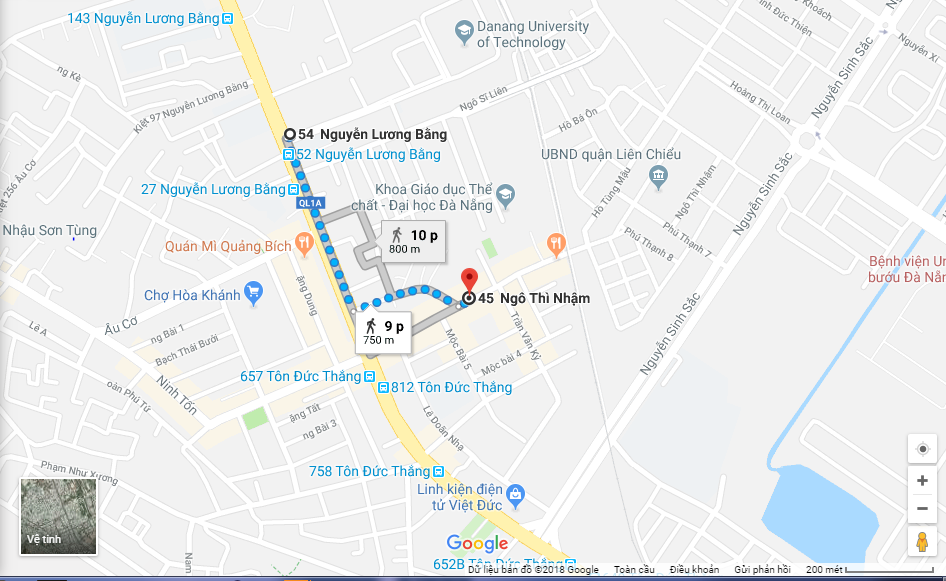
Các dịch vụ hoàn toàn miễn phí với việc xây dựng một ứng dụng nhỏ. Trả phí nếu đó là việc sử dụng cho mục đích kinh doanh, doanh nghiệp.

Khả năng hiện thị địa điểm qua api của Google maps:



*Hình 1.2: Bản đồ google maps*

Khả năng chỉ đường dùng api của Google:



*Hình 1.3: Chỉ đường thông qua google maps*

Google map api cung cấp nhiều tính năng liên quan việc điều hướng, định vị, phân loại địa điểm với chất lượng tốt, độ chính xác khá cao và miễn phí, trong đó một vài chức năng cơ bản như sau:

* Chỉ đường.
* Định vị vị trí qua tọa độ, tìm kiếm địa điểm
* Lọc các địa điểm, các nhà hàng, atm, tạp hóa...
* Hình ảnh vệ tinh trên toàn thế giới.

Thông qua việc sử dụng google map api, các lập trình viên có thể đưa bản đồ của google lên ứng dụng, website của mình với các tính năng miễn phí và có độ tin cậy cao.

## Ứng dụng bản đồ trong Ruby on Rails thông qua gem Geocoder

### Giới thiệu gem Geocoder

Geocoder là một giải pháp mã hóa địa lý trong Ruby. Nó có một vài tính năng cụ thể như sau:

* Chuyển đổi địa điểm thành tọa độ địa lý và ngược lại
* Tìm địa điểm gần vị trí cho trước
* Chuyển đổi địa chỉ IP thành vị trí

Để sử dụng gem này, cần cung cấp key đã đăng ký với Google tới server api của Google thông qua đường dẫn để có thể dùng các hàm liên quan đến bản đồ.

### Sử dụng gem Geocoder

Geocoder trả về tọa độ địa điểm thông qua tên đường, quận huyện, tỉnh thành hay quốc gia cũng như theo địa chỉ ip của người dùng.

Để có thể tìm được vị trí, cần cung cấp cho Geocoder một địa chỉ ip hoặc một tên đường, quận huyện tương ứng, hoặc một tọa đồ vật lý trên bản đồ thế giới.

Trong phạm vi đồ án, hệ thống sử dụng tên đường và địa chỉ quận huyện, thành phố để tìm tọa độ tương ứng, sau đó dựa vào tọa độ đó hiển thị lên bản đồ.

Geocoder hỗ trợ tính toán khoảng cách tới địa điểm được chia sẻ trên hệ thống thông qua hàm distance\_from(tọa độ điểm cần tính toán).

Để hiển thị trên bản đồ lên giao diện chúng ta cần sử dụng Javascript và thông qua phương thức draw\_map, bản đồ sẽ được tải ở website.

## Kết chương

Qua những nội dung đã nêu ở chương 1, chúng ta đã có một cái nhìn tổng quát về mặt cơ sở lý thuyết cũng như những công nghệ được áp dụng trong đồ án.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Giới thiệu

Hệ thống cung cấp giao diện web, người dùng có thể dễ dàng tiếp cận, tìm kiếm các địa điểm ẩm thực có trên hệ thống, lọc các kết quả theo danh mục, theo địa chỉ cụ thể.

Các chức năng cơ bản của hệ thống như sau:

* Chức năng chia sẻ cảm nhận, đăng hình ảnh về địa điểm
* Người dùng được cung cấp thông tin cụ thể cũng như trực quan về địa điểm như các món ăn, địa chỉ cụ thể, dẫn đường đến địa điểm
* Tìm kiếm các địa điểm xung quanh khu vực người dùng truy cập, các địa điểm được đặt hàng nhiều cũng như các địa điểm được người dùng đánh giá tốt
* Người dùng cũng có thể thêm mới địa điểm đã ghé thăm nếu chưa có trên hệ thống. Sau khi được phê duyệt thì địa điểm đó được cập nhật và hiển thị với người dùng truy cập trang web
* Chức năng cho chủ địa điểm, người dùng được phê duyệt làm chủ địa điểm nếu được quản trị viên chấp nhận. Người dùng là chủ địa điểm có thể quản lý địa điểm đó như thêm món ăn, cập nhật tình trạng của địa điểm, quản lý các đơn hàng người dùng đặt
* Quản trị hệ thống có thể thao tác với mọi địa điểm trên hệ thống, phê duyệt yêu cầu tạo địa điểm, cũng như các chức năng thêm sửa xóa địa điểm
* Chức năng quản lý người dùng trong hệ thống
* Cung cấp các tính năng thống kê

## Phân tích và đặc tả yêu cầu

### Giới thiệu

* + - 1. *Mục đích*

Hệ thống được xây dựng nhằm mục đích tao ra nơi mà mọi người có một kênh thông tin về các địa điểm ẩm thực, người dùng có thể tìm kiếm các địa điểm, có thể chia sẻ cảm nhận cũng như được chỉ dẫn về địa điểm.

* + - 1. *Đối tượng*

Hệ thống phục vụ người dùng địa phương có nhu cầu tìm kiếm thông tin về các địa điểm ẩm thực, khách du lịch đến từ các nơi khi cần tìm kiếm địa điểm để khám phá về ẩm thực ở khu vực đó.

### Mô tả chung

* + - 1. *Tổng quan:*

Hiện tại xã hội đang phát triển rất nhanh, kéo theo đó là nhiều nhu cầu của xã hội tăng cao, trong đó nhu cầu ăn uống, thưởng thức các món ăn ngon là một nhu cầu không thể thiếu, hệ thống chia sẻ và chỉ dẫn ẩm thực được tạo ra để đáp ứng nhu cầu đó.

Hệ thống ra đời và ứng dụng thực tế có thể giúp người dùng có thể tra cứu thông tin về địa điểm cũng như để lựa chọn được địa điểm ưng ý trước khi tới địa điểm để thưởng thức ẩm thực tại địa điểm đó.

* + - 1. *Chức năng*

Hệ thống được xây dựng với mục đích chia sẻ và chỉ dẫn địa điểm ẩm thực nên sẽ có các chức năng chính được mô tả như sau:

* Hiển thị các địa điểm ẩm thực, các địa điểm xung quanh vị trí của người dùng.
* Sau khi trải nghiệm tại địa điểm, người dùng có thể gửi thông tin tạo mới địa điểm.
* Hiển thị thông tin chi tiết về địa điểm, đánh giá của người dùng đã trải nghiệm tại địa điểm, hình ảnh mà người dùng cung cấp tại địa điểm.
* Đánh giá địa điểm qua các loại tiêu chí khác nhau, tải hình ảnh của địa điểm lên hệ thống.
* Được chỉ dẫn đường đi từ địa chỉ của người dùng hoặc địa chỉ đang truy cập vào hệ thống đến địa điểm đang tra cứu.
* Địa điểm có thể thông qua hệ thống để cung cấp dịch vụ ship đồ ăn/ đồ uống đến khách hàng.
* Người dùng có nhu cầu đặt món có thể đặt các món ăn, người dùng xem được thông tin về giá cả cũng như các thông tin khác về đơn hàng đã đặt.
* Người dùng là chủ của địa điểm có thể quản lý các đơn hàng, cập nhật các thông tin về địa điểm của mình.
  + - 1. *Lớp người dùng và đặc điểm*

Hệ thống phân biệt bốn đối tượng người dùng là khách vãng lai, người có tài khoản, chủ địa điểm và người quản trị hệ thống.

Bốn đối tượng người dùng này sẽ có các chức năng khác nhau tùy theo cấp độ. Khách ghé thăm website sẽ chỉ có thể xem các thông tin cơ bản, người dùng có tài khoản có thể gửi thông tin địa điểm mới, thêm đánh giá cũng như lưu các địa điểm ưa thích. Chủ địa điểm có thêm chức năng quản lý địa điểm, và người quản trị hệ thống là người có toàn bộ quyền lên hệ thống như thêm sửa xóa hay khóa địa điểm, cũng như người dùng khác.

Ngoài ra quản trị viên có thêm một số chức năng như thống kê số liệu...

* + - 1. *Môi trường hoạt động*

Hệ thống hoạt động trên môi trường web, có thể chạy trên các trình duyệt web có kết nối internet.

### Yêu cầu giao diện

* + - 1. *Giao diện người dùng*

Giao diện dễ nhìn, trực quan và hiển thị được đầy đủ thông tin nhất. Giao diện chia thành nhiều khu vực, nhưng phải đảm bảo sự thống nhất và không gây rối mắt khi thao tác.

* + - 1. *Giao diện quản lý của thành viên*

Giao diện phải có đầy đủ các phím chức năng, trực quan và dễ dàng sử dụng nhất. Hiển thị đầy đủ thông tin nhưng không rườm rà phức tạp.

* + - 1. *Giao diện quản trị viên*

Giao diện quản trị viên phải cung cấp cái nhìn tổng quan nhất của hệ thống như danh sách các địa điểm, danh sách người dùng và đầy đủ các phím chức năng.

### Yêu cầu phi chức năng

* + - 1. *Hiệu suất*

Hệ thống phải đạt độ ổn định tương đối, đáp ứng tốc độ tải trang nhất định với lượng truy cập không lớn. Hệ thống đảm bảo được sự tối ưu trong việc tải các thành phần của trang, không gây lãng phí tài nguyên.

* + - 1. *Yêu cầu an toàn*

Hệ thống hoạt động độc lập, không phụ thuộc vào bất kỳ phần mềm nào, không ảnh hưởng đến các phần mềm khác.

* + - 1. *Yêu cầu bảo mật*

Hệ thống đảm bảo các thông tin người dùng phải được bảo mật nhất, không sử dụng các thư viện Javascript không rõ nguồn gốc.

Mật khẩu người dùng không được truyền dưới dạng plain-text. Mật khẩu lưu xuống cơ sở dữ liệu phải được mã hóa một chiều và không thể dịch ngược.

* + - 1. *Yêu cầu chất lượng*

Hệ thống phải đảm bảo chất lượng tốt, hoạt động tốt, ít các lỗi cơ bản. Tiết kiệm được dung lượng web, giảm băng thông. Chỉ sử dụng các thư viện cần thiết và sử dụng đúng chỗ.

## Thiết kế hệ thống

### Các tác nhân trong hệ thống

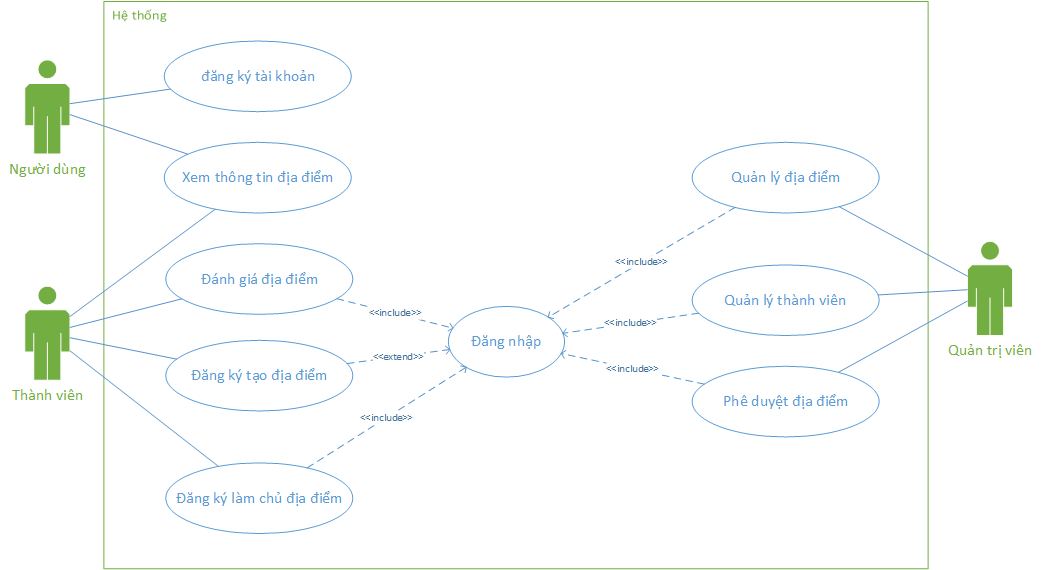
Hệ thống có các tác nhân sau:

* Khách vãng lai ghé thăm website
* Người dùng có tài khoản
* Người dùng là chủ địa điểm
* Quản trị viên hệ thống
* Các địa điểm ăn uống

### Các biểu đồ

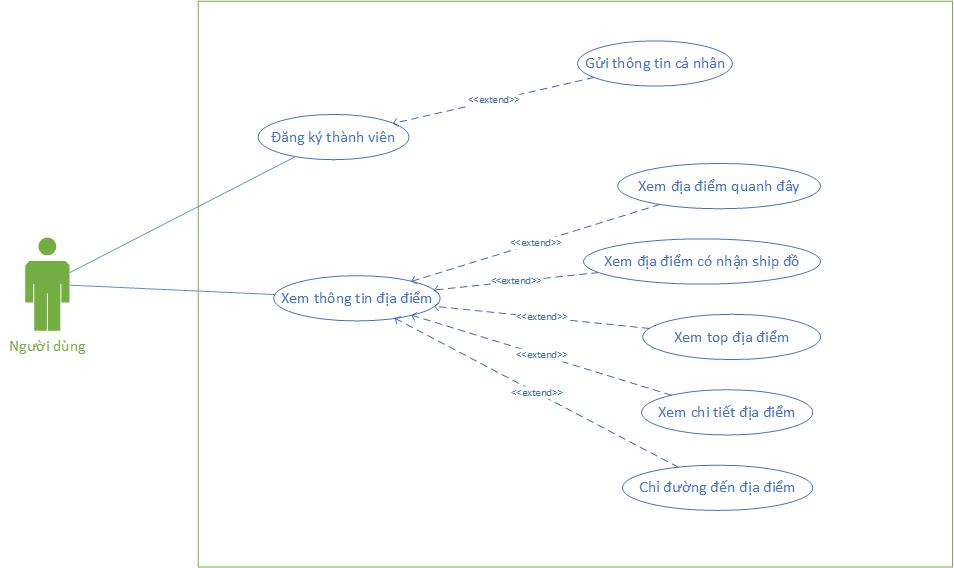
* + - 1. *Biểu đồ ca sử dụng*

1. *Biểu đồ usecase tổng quát*



*Hình 2.1: Use case Tổng quát*

1. *Biểu đồ usecase người dùng*



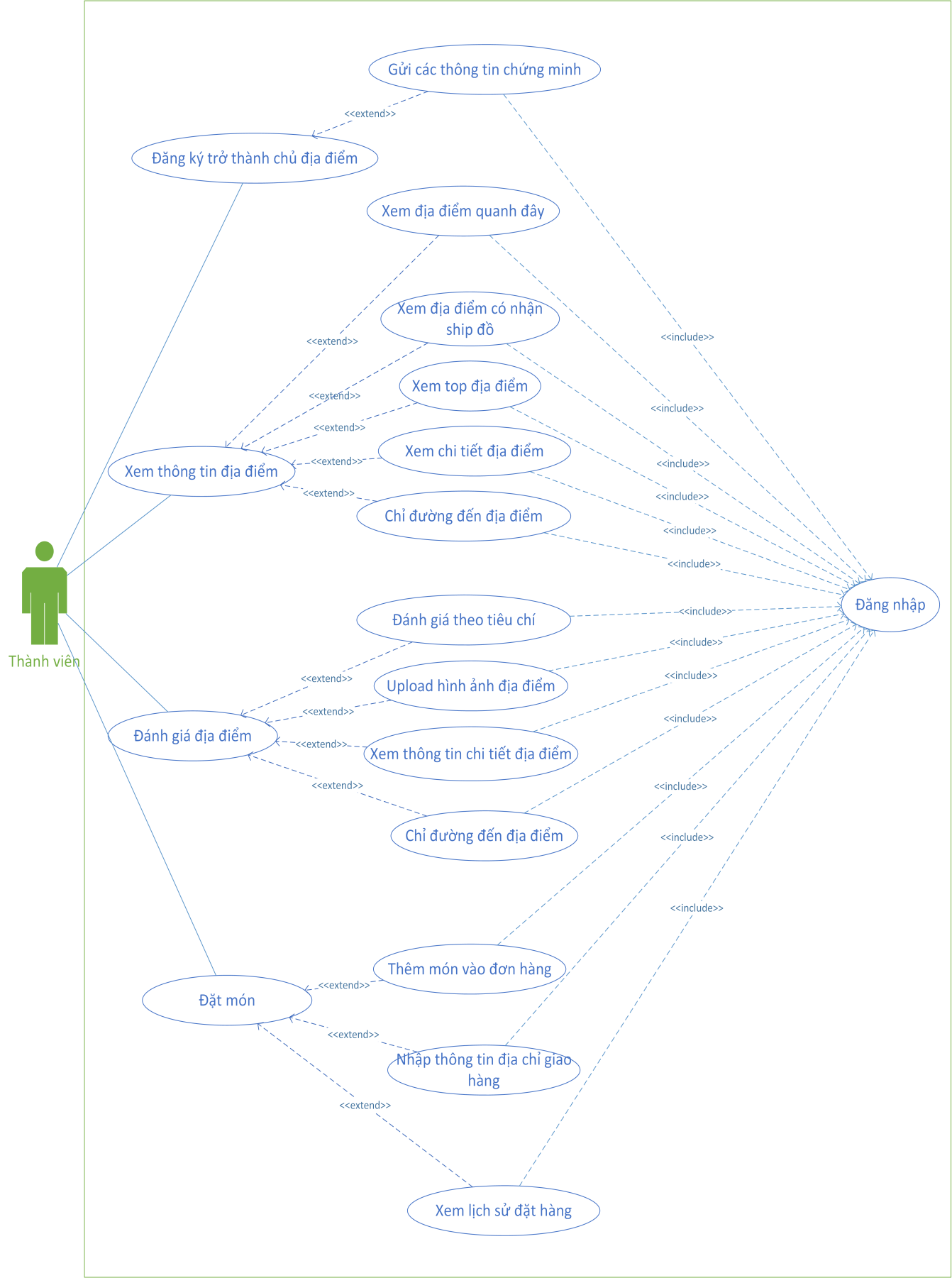
*Hình 2.2: Usecase Người dùng*

*Bảng 2.1: Chức năng xem toàn bộ địa điểm*

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Xem toàn bộ địa điểm | |
| **Độ ưu tiên** | Trung bình. | |
| **Kích hoạt** | Truy cập đường dẫn /places hoặc trang chủ | |
| **Điều kiện trước** | Truy cập trang web | |
| **Mô tả xử lý** | Hệ thống sẽ tải những địa điểm đã được phê duyệt, địa điểm quanh khu vực, địa điểm được đặt nhiều, địa điểm top bình chọn, kèm thông tin các bảng liên quan khác. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Chọn Khám phá, địa điểm. | Hiển thị màn hình địa điểm. |
| Lọc kết quả tìm kiếm. | Truy vấn cơ sở dữ liệu và trả về kết quả tương ứng. |
| Click vào địa điểm | Chuyển hướng đến trang chi tiết |

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Lọc địa điểm quanh khu vực | |
| **Độ ưu tiên** | Cao | |
| **Kích hoạt** | Truy cập đường dẫn /nearby\_places hoặc trang chủ | |
| **Điều kiện trước** | Truy cập trang web | |
| **Mô tả xử lý** | Hệ thống sẽ tải những địa điểm đã được phê duyệt, địa điểm quanh khu vực kèm thông tin liên quan từ các bảng khác. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Chọn khám phá, địa điểm. | Hiển thị màn hình chi tiết địa điểm. |
| Lọc kết quả tìm kiếm. | Truy vấn cơ sở dữ liệu và trả về kết quả tương ứng. |
| Click vào địa điểm | Chuyển hướng đến trang chi tiết |

1. *Biểu đồ usecase thành viên*



*Hình 2.3: Usecase Thành viên*

*Bảng 2.2: Chức năng xem thông tin địa điểm*

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Xem thông tin địa điểm | |
| **Độ ưu tiên** | Cao | |
| **Kích hoạt** | Truy cập đường dẫn /places/place\_id | |
| **Điều kiện trước** | Truy cập trang web | |
| **Mô tả xử lý** | Hệ thống sẽ tải những địa điểm đã được phê duyệt kèm thông tin các bảng liên quan khác. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Chọn Khám phá, địa điểm. | Hiển thị màn hình địa điểm. |
| Lọc kết quả tìm kiếm. | Truy vấn cơ sở dữ liệu và trả về kết quả tương ứng. |

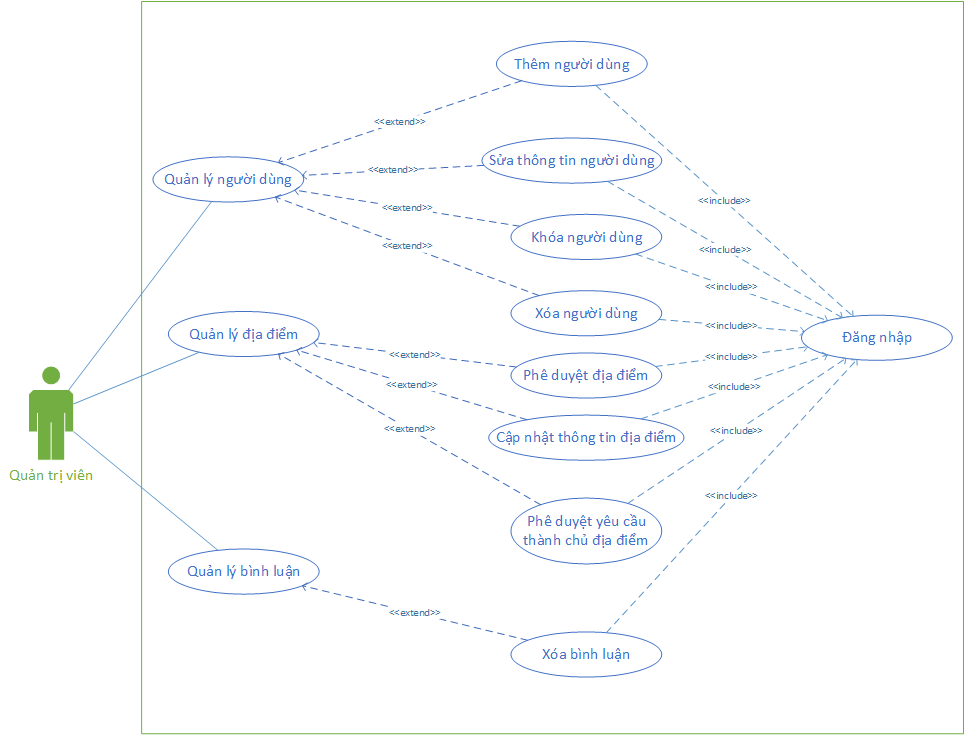
*Bảng 2.3: Chức năng đánh giá địa điểm*

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Đánh giá địa điểm | |
| **Độ ưu tiên** | Cao | |
| **Kích hoạt** | Truy cập đường dẫn chi tiết địa điểm /places/:place\_id | |
| **Điều kiện trước** | Truy cập trang web, đăng nhập tài khoản thành viên | |
| **Mô tả xử lý** | Hệ thống sẽ tải các đánh giá của địa điểm, người dùng có thể xem ảnh địa điểm qua các đánh giá, thêm đánh giá cho địa điểm theo các tiêu chí. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Chọn sao cho các tiêu chí mà cá nhân đánh giá cho địa điểm | Cập nhật điểm số trong form bằng jquery |
| Gửi đánh giá | Gửi params lên controller xử lý |

*Bảng 2.4: Chức năng đặt món*

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Chức năng đặt món | |
| **Độ ưu tiên** | Trung bình | |
| **Kích hoạt** | Truy cập đường dẫn chi tiết địa điểm /places/:place\_id | |
| **Điều kiện trước** | Truy cập trang web, đăng nhập tài khoản thành viên | |
| **Mô tả xử lý** | Hệ thống sẽ tải thông tin các món ăn của địa điểm, người dùng có thể chọn món, tăng giảm số lượng đặt và xác nhận đặt hàng. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Chọn thêm món ăn | Cập nhật đơn hàng trên controller, thêm món ăn trên view order dùng jquery |
| Xác nhận đặt hàng | Gửi yêu cầu đặt món, đơn hàng được duyệt bởi chủ cửa hàng |

1. *Biểu đồ usecase quản trị viên*



*Hình 2.4: Usecase quản trị viên*

*Bảng 2.5: Chức năng quản lý người dùng*

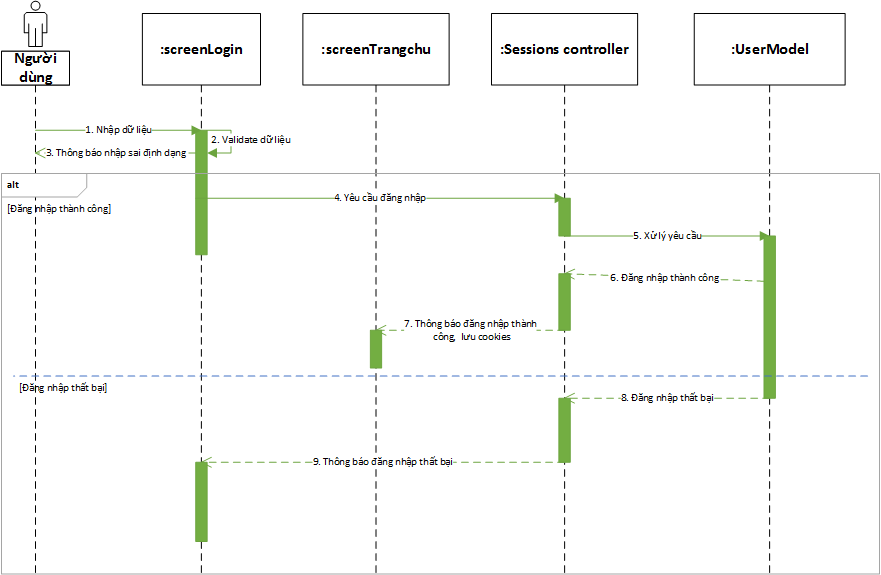
|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Chức năng quản lý người dùng | |
| **Độ ưu tiên** | Thấp | |
| **Kích hoạt** | Truy cập đường dẫn admin/users | |
| **Điều kiện trước** | Truy cập trang web, đăng nhập bằng tài khoản admin | |
| **Mô tả xử lý** | Hệ thống sẽ tải thông tin người dùng trong hệ thống. Các button có tính chất quản lý như thêm, xóa, sửa người dùng, block người dùng. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Click thêm người dùng | Chuyển tới trang thêm người dùng |
| Xóa người dùng | Xử lý yêu cầu xóa người dùng |
| Click chỉnh sửa người dùng | Chuyển đến trang chỉnh sửa người dùng |
| Block người dùng | Xử lý yêu cầu khóa người dùng |

*Bảng 2.6: Chức năng quản lý địa điểm*

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | Chức năng quản lý địa điểm | |
| **Độ ưu tiên** | Cao | |
| **Kích hoạt** | Truy cập đường dẫn admin/places | |
| **Điều kiện trước** | Truy cập trang web, đăng nhập bằng tài khoản admin | |
| **Mô tả xử lý** | Hệ thống sẽ tải thông tin các địa điểm trong hệ thống. Các button có tính chất quản lý như thêm, xóa, sửa địa điểm, phê duyệt địa điếm. | |
| **Điều kiện sau** | Những thay đổi sẽ được thực hiện. | |
| **Xử lý ngoại lệ** | Nếu người dùng không yêu cầu gì thêm có thể thoát khỏi trang web. | |
| **Kịch bản** | **Tác nhân** | **Hệ thống** |
| Click thêm địa điểm | Chuyển tới trang thêm địa điểm |
| Xóa địa điểm | Xử lý yêu cầu xóa địa điểm |
| Click chỉnh sửa địa điểm | Chuyển đến trang chỉnh sửa địa điểm |
| Phê duyệt địa điểm, chấp nhận/ từ chối | Xử lý yêu cầu phê duyệt địa điểm |

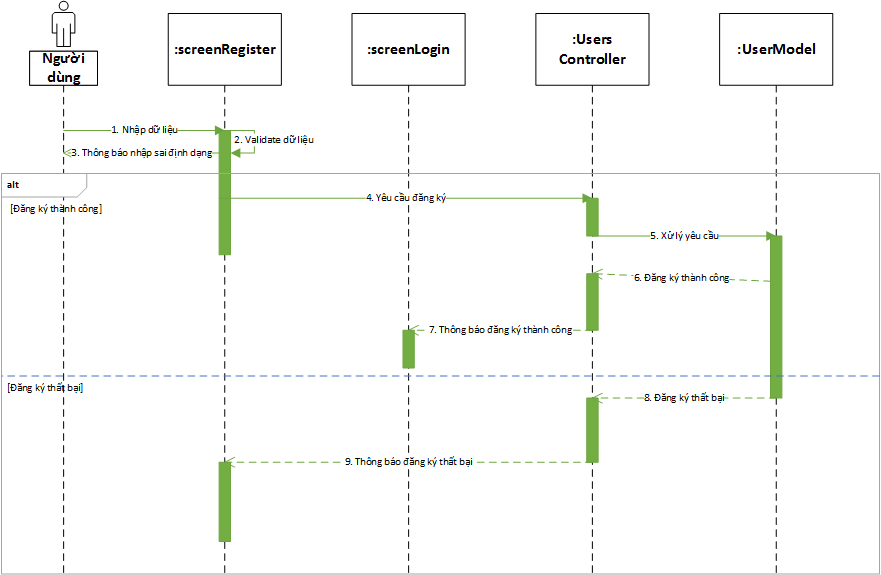
* + - 1. *Biểu đồ tuần tự*

1. *Chức năng đăng nhập*



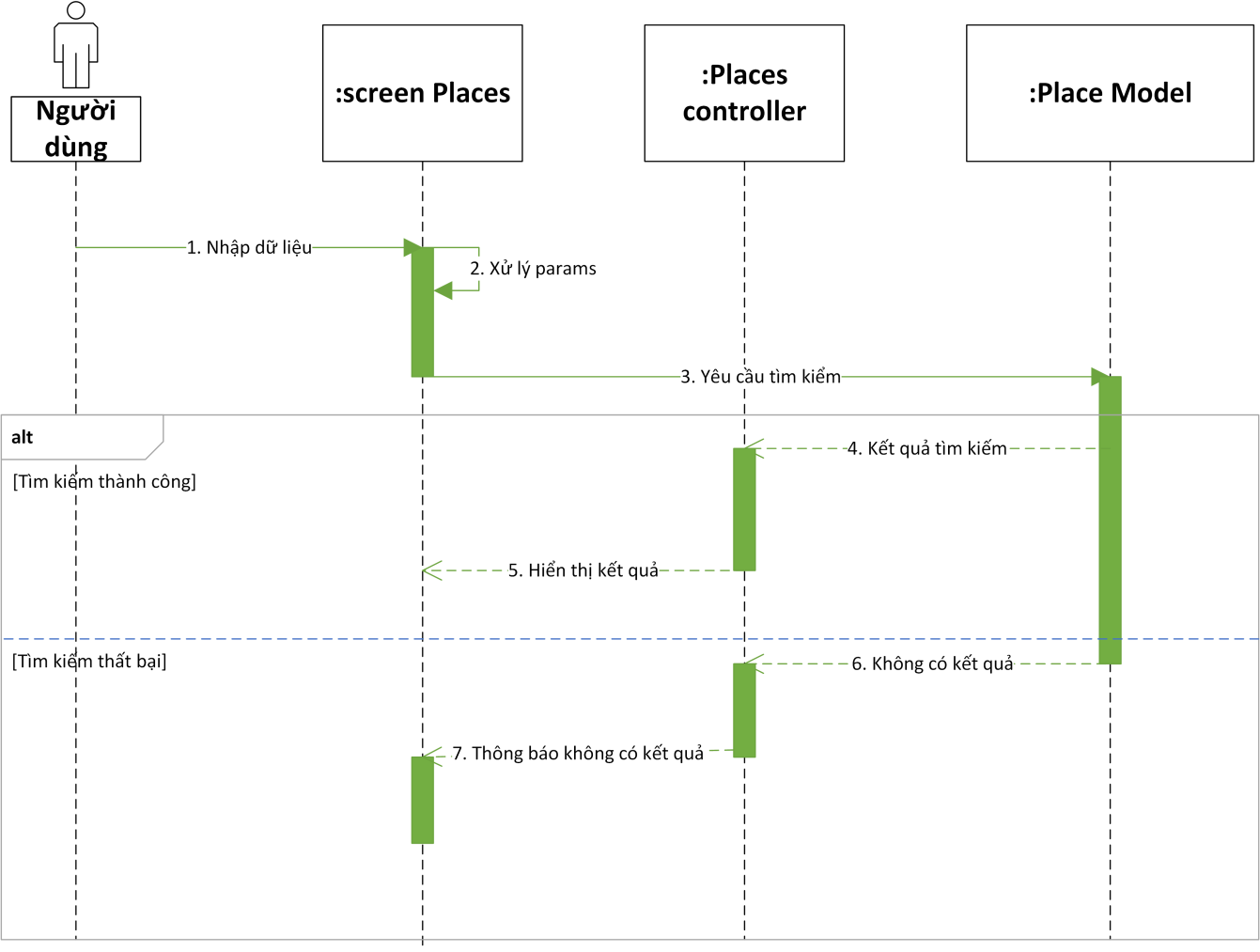
*Hình 2.5: Biểu đồ tuần tự chức năng đăng nhập*

1. *Chức năng đăng ký thành viên*



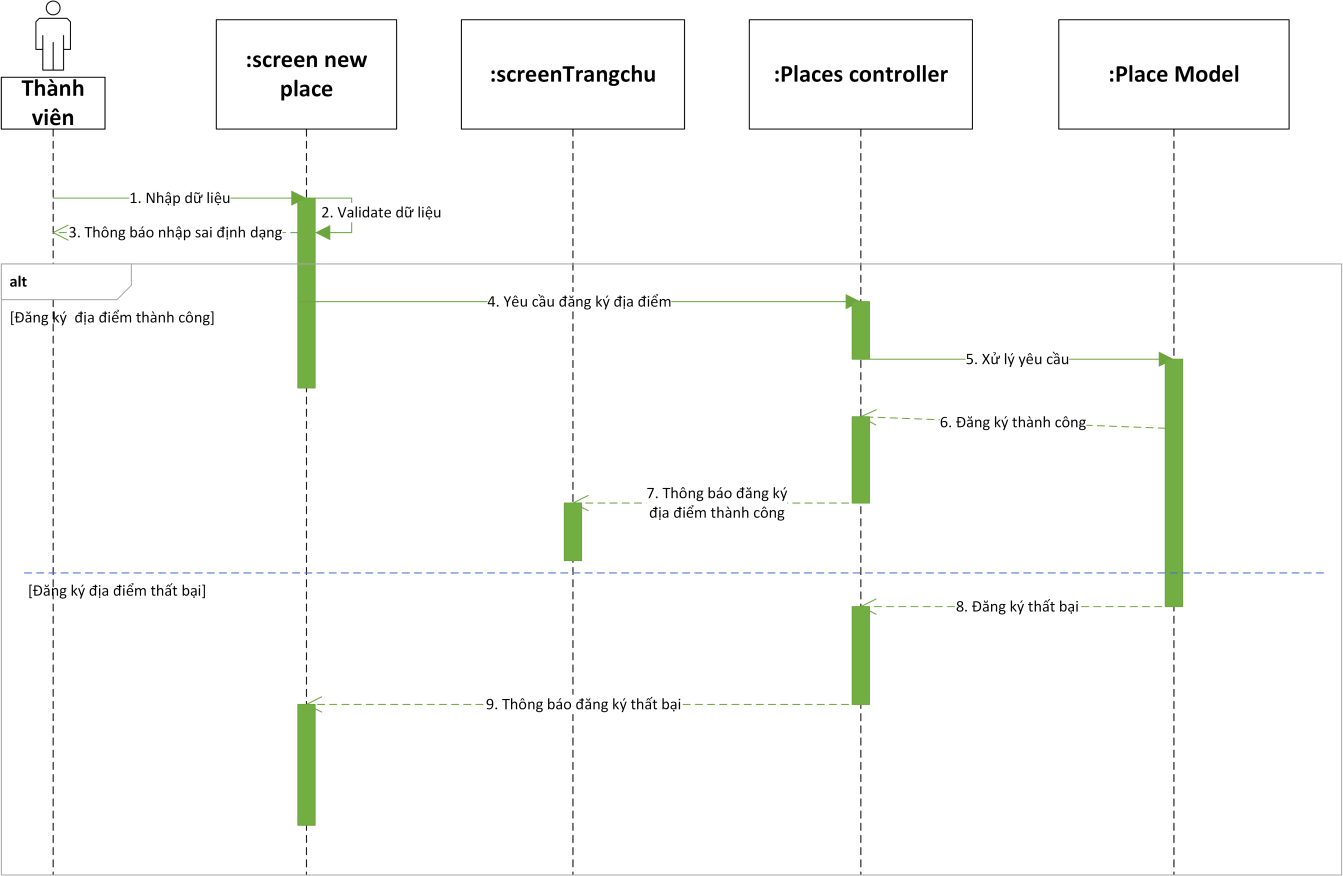
*Hình 2.6: Biểu đồ tuần tự chức năng đăng ký*

1. *Chức năng tìm kiếm địa điểm*



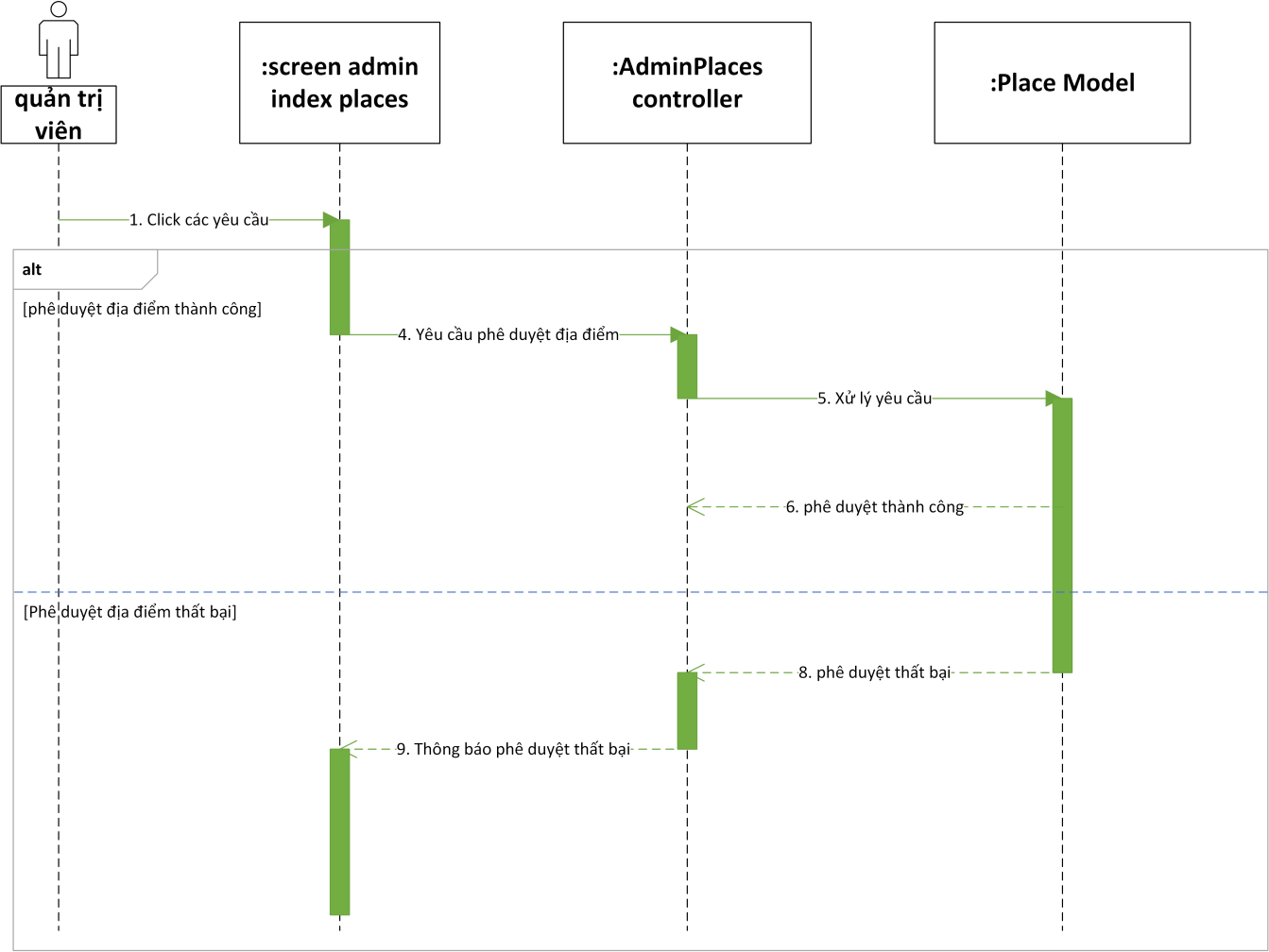
*Hình 2.7: Biểu đồ tuần tự chức năng tìm kiếm*

1. *Chức năng đăng ký địa điểm*



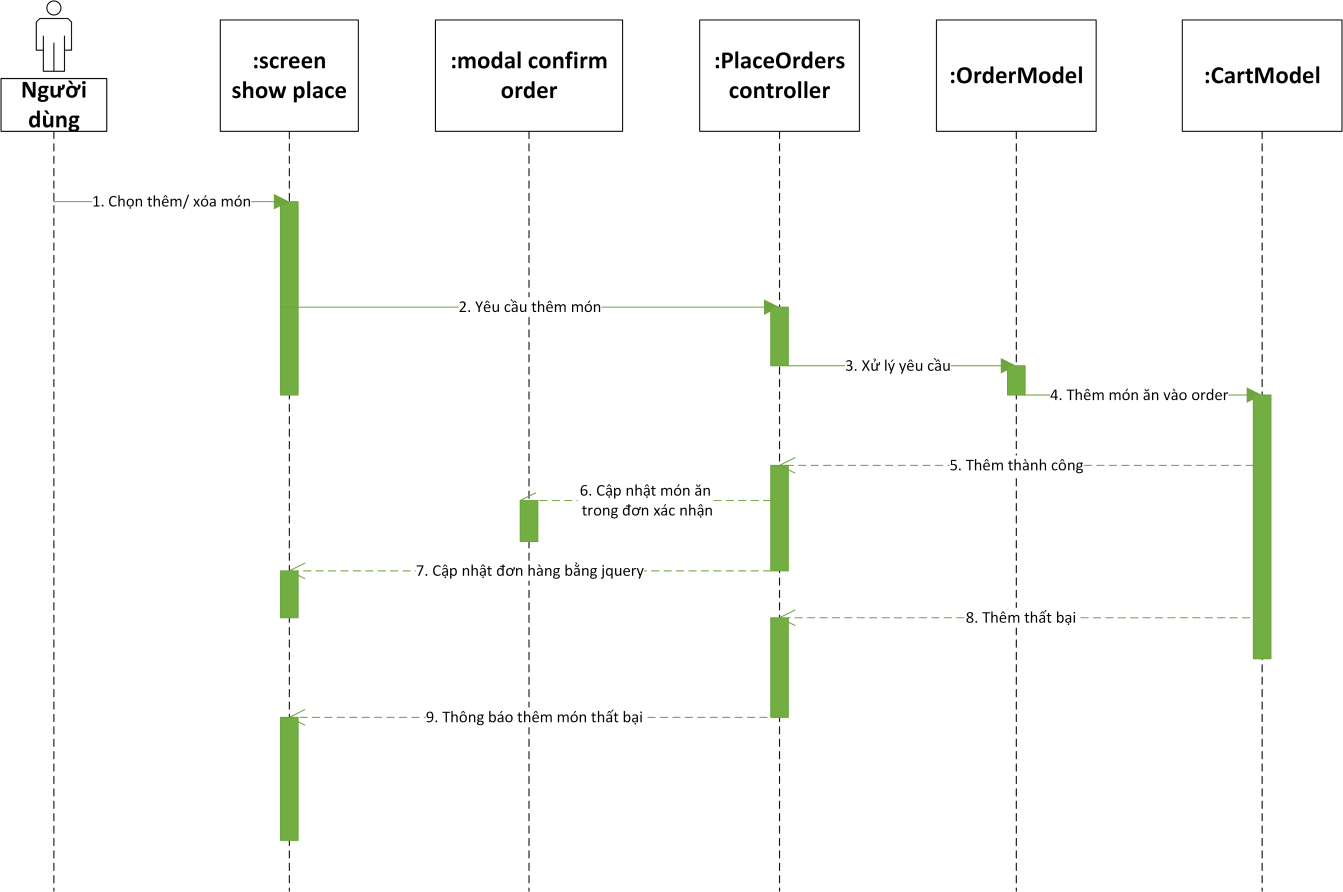
*Hình 2.8: Biểu đồ tuần tự chức năng đăng ký địa điểm*

1. *Chức năng phê duyệt địa điểm*



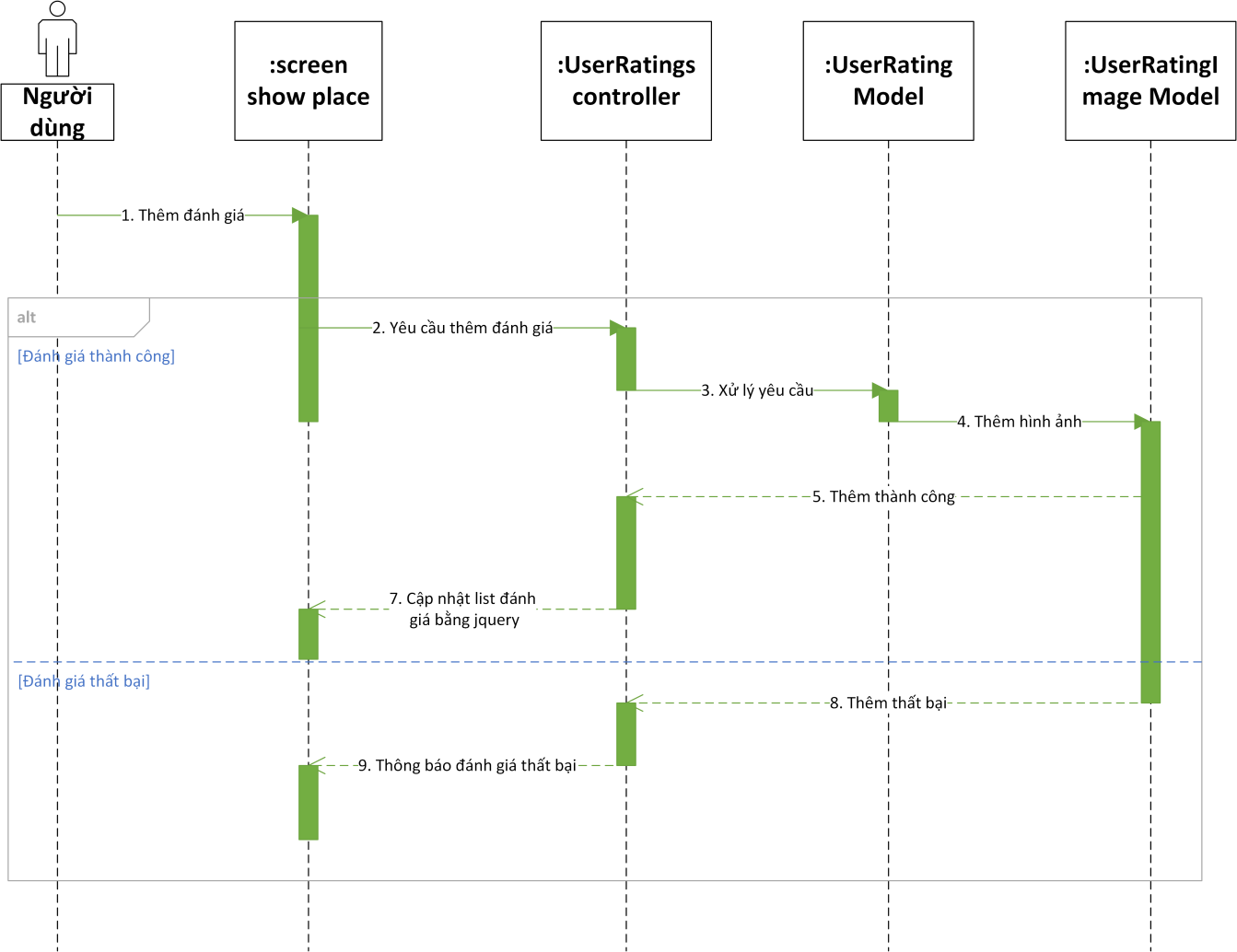
*Hình 2.9: Biểu đồ tuần tự chức năng phê duyệt địa điểm*

1. *Chức năng đặt món*



*Hình 2.10: Biểu đồ tuần tự chức năng đặt món*

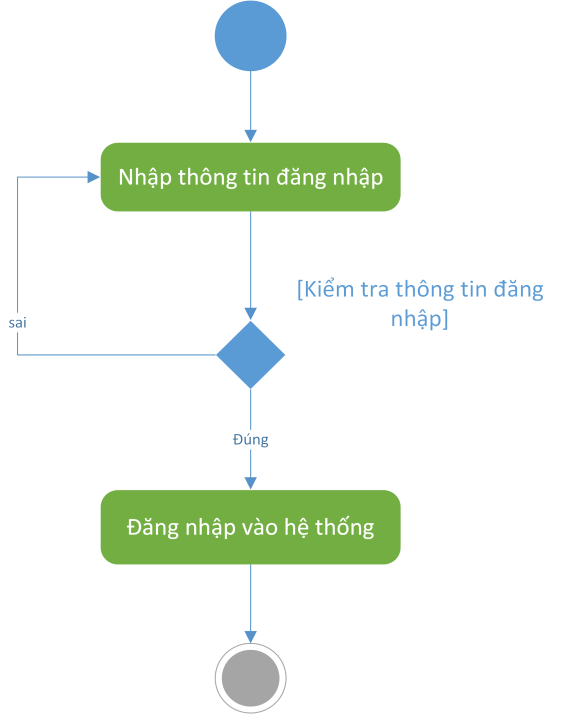
1. *Chức năng đánh giá địa điểm*



*Hình 2.11: Biểu đồ tuần tự chức năng đánh giá địa điểm*

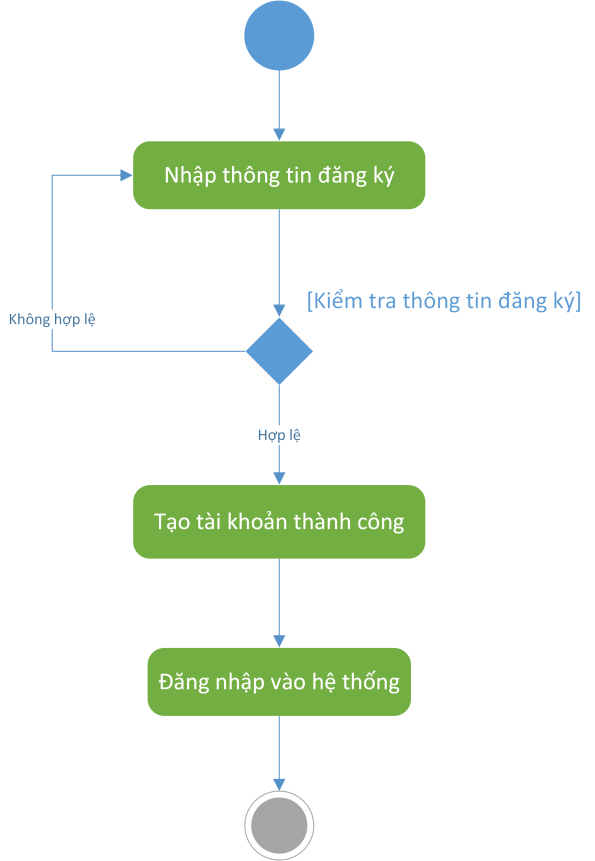
* + - 1. *Biểu đồ hoạt động*

1. *Chức năng đăng nhập*

**

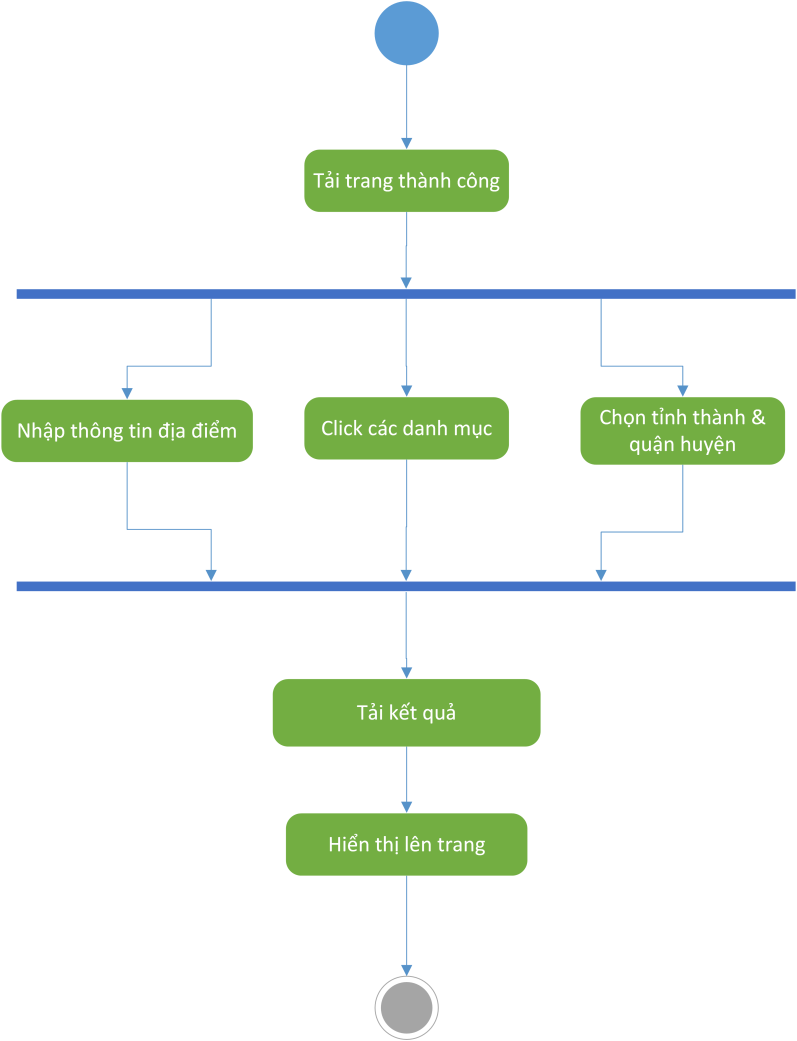
*Hình 2.12: Biểu đồ hoạt động đăng nhập*

1. *Chức năng đăng ký thành viên*

**

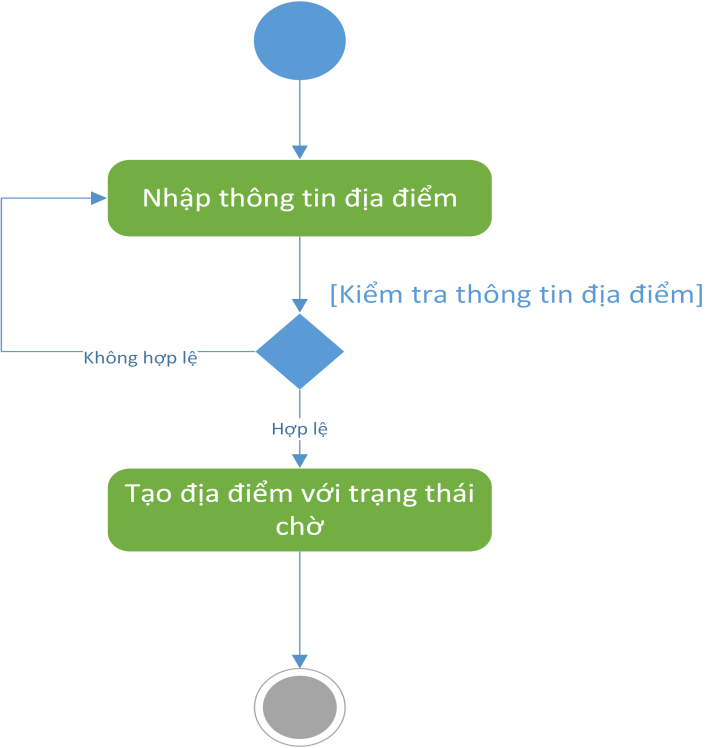
*Hình 2.13: Biểu đồ hoạt động đăng ký tài khoản*

1. *Chức năng tìm kiếm địa điểm*

**

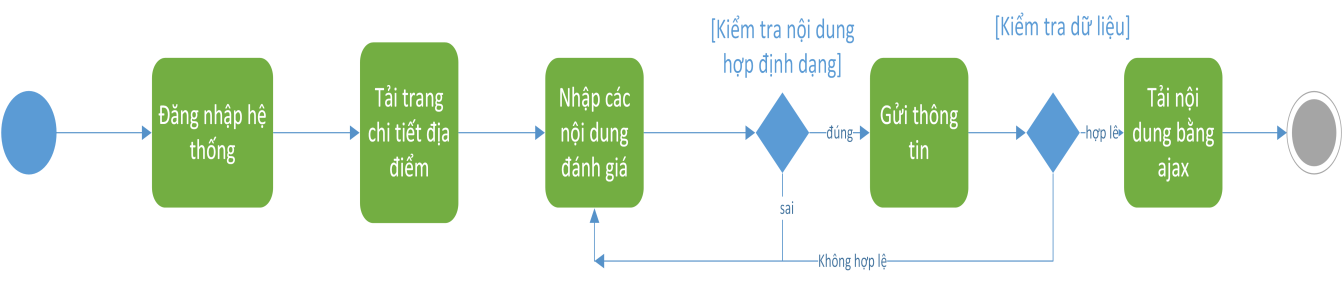
*Hình 2.14: Biểu đồ hoạt động chức năng tìm kiếm*

1. *Chức năng đăng ký tạo mới địa điểm*

**

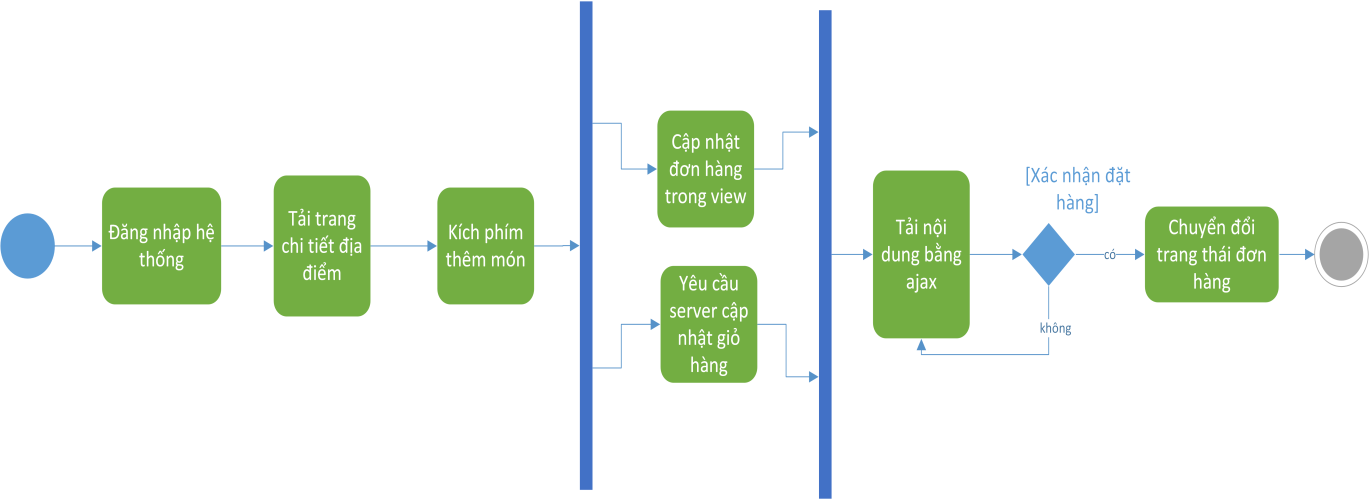
*Hình 2.15: Biểu đồ hoạt động chức năng tạo mới địa điểm*

1. *Chức năng đánh giá địa điểm*

**

*Hình 2.16: Biểu đồ hoạt động chức năng đánh giá địa điểm*

1. *Chức năng đặt món*

**

*Hình 2.17: Biểu đồ hoạt động chức năng đặt món*

### Thiết kế cơ sở dữ liệu

* + - 1. *Mô tả cơ sở dữ liệu*

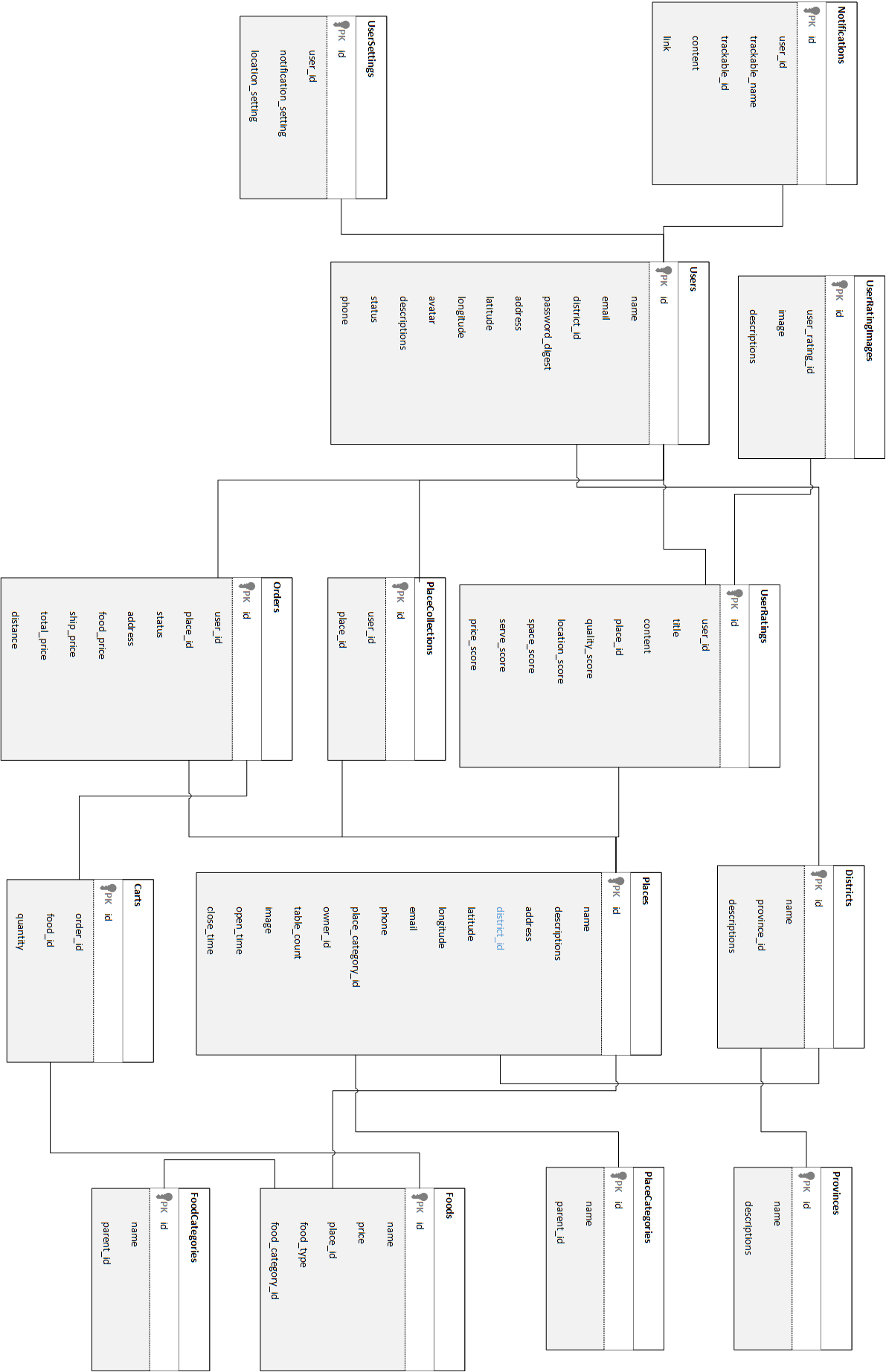
Cơ sở dữ liệu được thiết kế sao cho tối ưu nhất, thể hiện mối quan hệ giữa các bảng, tránh việc dư thừa dữ liệu.

Liên hệ qua thực tế, các bảng trong cơ sở dữ liệu cần phải lưu các đối tượng liên quan đến yêu cầu lưu trữ của đồ án.

Cơ sở dữ liệu của hệ thống sẽ gồm các bảng sau:

* Users
* UserSettings
* Notifications
* Districts
* Provinces
* UserRatings
* UserRatingImages
* PlaceCollecctions
* Orders
* Carts
* Places
* PlaceCategories
* PlaceFoods
* Foods
* FoodCategories

Mối quan hệ giữa các bảng được thể hiện thông qua hình sau:



*Hình 2.18: Cơ sở dữ liệu của hệ thống*

* + - 1. *Nội dung các bảng dữ liệu*

1. Bảng Places (Địa điểm)

*Bảng 2.7: Bảng Users*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã người dùng | Int | 255 | Khóa chính |
| Name | Họ và tên | Varchar | 255 |  |
| Email | Địa chỉ email | Varchar | 255 |  |
| Password\_digest | Mật khẩu đã mã hóa | Varchar | 255 |  |
| Address | Địa chỉ người dùng | Varchar | 255 |  |
| Descriptions | Mô tả về người dùng | Varchar | 255 |  |
| Phone | Số điện thoại | Varchar | 255 |  |
| District\_id | Id quận huyện | Float | 255 | Khóa phụ |
| Latitude | Vỹ độ | Float | 255 |  |
| Longitude | Kinh độ | Float | 255 |  |
| Status | Trạng thái người dùng | Int | 255 |  |
| Avatar | Hình ảnh đại diện | Varchar | 255 |  |

1. Bảng Places (Địa điểm)

*Bảng 2.8: Bảng Places*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã địa điểm | Int | 255 | Khóa chính |
| Name | Họ và tên | Varchar | 255 |  |
| Email | Địa chỉ email | Varchar | 255 |  |
| Address | Địa chỉ địa điểm | Varchar | 255 |  |
| Descriptions | Mô tả về địa điểm | Varchar | 255 |  |
| Phone | Số điện thoại liên hệ | Varchar | 255 |  |
| District\_id | Id quận huyện | Int | 255 | Khóa phụ |
| User\_id | Id chủ địa điểm | Int | 255 | Khóa phụ |
| Latitude | Vỹ độ | Float | 255 |  |
| Longitude | Kinh độ | Float | 255 |  |
| Status | Trạng thái địa điểm | Int | 255 |  |
| Image | Hình ảnh địa điểm | Varchar | 255 |  |
| Table\_count | Số lượng bàn phục vụ | Int | 255 |  |
| Open\_time | Giờ mở cửa | DateTime |  |  |
| Close\_time | Giờ đóng cửa | DateTime |  |  |
| Ship\_price | Chi phí ship trên km | Float | 255 |  |

1. Bảng UserRatings (Đánh giá địa điểm)

*Bảng 2.9: Bảng UserRatings*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã đánh giá | Int | 255 | Khóa chính |
| User\_id | Id người dùng | Int | 255 | Khóa phụ |
| Place\_id | Id địa điểm | Int | 255 | Khóa phụ |
| Score\_space | Điểm không gian | Int | 255 |  |
| Score\_location | Điểm vị trí | Int | 255 |  |
| Score\_serve | Điểm phục vụ | Int | 255 |  |
| Score\_quality | Điểm chất lượng | Int | 255 |  |
| Score\_price | Điểm về giá cả | Int | 255 |  |
| Title | Tiêu đề đánh giá | Varchar | 255 |  |
| Content | Nội dung đánh giá | Varchar | 255 |  |

1. Bảng UserRatingImages (Hình ảnh đánh giá)

*Bảng 2.10: Bảng UserRatingImages*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã đánh giá | Int | 255 | Khóa chính |
| User\_id | Id người dùng | Int | 255 | Khóa phụ |
| Place\_id | Id địa điểm | Int | 255 | Khóa phụ |
| Image | Hình ảnh | Varchar | 255 |  |
| Description | Mô tả hình ảnh | Varchar | 255 |  |

1. Bảng PlaceCollections (Địa điểm đã lưu)

*Bảng 2.11: Bảng PlaceCollections*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã bộ sưu tập được lưu | Int | 255 | Khóa chính |
| User\_id | Id người dùng | Int | 255 | Khóa phụ |
| Place\_id | Id địa điểm | Int | 100 | Khóa phụ |

1. Bảng Orders (Bảng đơn hàng)

*Bảng 2.12: Bảng Orders*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã đơn hàng | Int | 255 | Khóa chính |
| User\_id | Id người dùng | Int | 255 | Khóa phụ |
| Place\_id | Id địa điểm | Int | 100 | Khóa phụ |
| Total\_price | Tổng tiền | Float | 255 |  |
| Ship\_price | Phí ship | Float | 255 |  |
| Status | Trang thái đơn hàng | Int | 255 |  |
| Payment\_status | Trạng thái thanh toán | Int | 255 |  |
| Address | Địa chỉ nhận hàng | Varchar | 255 |  |
| Distance | Khoảng cách | Float | 255 |  |

1. Bảng Carts (Giỏ hàng)

*Bảng 2.13: Bảng Carts*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã giỏ hàng | Int | 255 | Khóa chính |
| Order\_id | Id Đơn hàng | Int | 255 | Khóa phụ |
| Food\_id | Id món hàng | Int | 255 | Khóa phụ |
| Quantity | Số lượng sản phẩm | Int | 255 |  |

1. Bảng Districts (Quận huyện)

*Bảng 2.14: Bảng Districts*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã Quận huyện | Int | 255 | Khóa chính |
| Province\_id | Id Tỉnh thành | Int | 255 | Khóa phụ |
| Description | Mô tả quận huyện | Varchar | 255 |  |

1. Bảng Provinces (Tình thành)

*Bảng 2.15: Bảng Provices*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã Tỉnh thành | Int | 255 | Khóa chính |
| Description | Mô tả Tỉnh thành | Int | 255 |  |

1. Bảng PlaceCategories (Danh mục địa điểm)

*Bảng 2.16: Bảng PlacesCategories*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã danh mục địa điểm | Int | 255 | Khóa chính |
| Parent\_id | Id danh mục cha | Int | 255 | Khóa phụ |
| Name | Tên danh mục | Varchar | 255 |  |

1. Bảng Foods (Món ăn/đồ uống)

*Bảng 2.17: Bảng Foods*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã món ăn | Int | 255 | Khóa chính |
| Parent\_id | Id địa điểm | Int | 255 | Khóa phụ |
| Food\_category\_id | Id loại món ăn | Int | 255 | Khóa phụ |
| Name | Tên món ăn | Varchar | 255 |  |
| Price | Giá món ăn | Int | 255 |  |
| Food\_type | Loại món ăn | Int | 11 |  |

1. Bảng FoodCategories (Danh mục món ăn/đồ uống)

*Bảng 2.18: Bảng FoodCategories*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã danh mục món ăn | Int | 255 | Khóa chính |
| Parent\_id | Id danh mục cha | Int | 255 | Khóa phụ |
| Name | Tên danh mục | Varchar | 255 |  |

1. Bảng UserSettings (Cài đặt người dùng)

*Bảng 2.19: Bảng UserSettings*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã cài đặt người dùng | Int | 255 | Khóa chính |
| User\_id | Mã người dùng | Int | 255 | Khóa phụ |
| Notification\_setting | Lưu cài đặt về thông báo | Varchar | 255 |  |
| Location\_setting | Lưu cài đặt về vị trí | Varchar | 255 |  |

1. Bảng Notifications (Thông báo)

*Bảng 2.20: Bảng Notifications*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột** | **Giải thích** | **Kiểu dữ liệu** | **Maxlength** | **Ghi chú** |
| id | Mã thông báo | Int | 255 | Khóa chính |
| User\_id | Id người dùng | Int | 255 |  |
| Trackable\_id | Mã đối tượng thông báo | Int | 255 | Khóa phụ |
| Trackable\_name | Tên đối tượng thông báo | Varchar | 255 |  |
| Content | Nội dung thông báo | Varchar | 255 |  |
| Link | Đường dẫn đến thông tin | Varchar | 255 |  |

## Kết chương

Qua chương này chúng ta đã nắm rõ được bài toán mà hệ thống phải giải quyết, thông qua các bước phân tích, cấu trúc hệ thống, các luồng xử lý trong hệ thống được biểu diễn rõ ràng và chi tiết

# TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

## Triển khai hệ thống

### Môi trường triển khai

Hệ thống triển khai trên nền web, chạy được trên các trình duyệt có kết nối internet như Chrome, Firefox, Internet Explorer ...

* Công nghệ sử dụng: Ruby on Rails
* Ngôn ngữ lập trình: Ruby, Javascript, Html, Css.
* Framework sử dụng: Rails framework.
* Thư viện, framework bổ trợ: Bootstrap, JQuery, Font Awesome, Light box, Google api ...

### Cài đặt môi trường

* + - 1. *Cài đặt Ruby và Rails*
* Cài đặt Ruby:

Chạy lệnh $ sudo apt-get install ruby-full



Kiểm tra cài đặt bằng lệnh: ruby --version

* Cài đặt Rails:

Sau khi đã cài đặt Ruby thành công thì mặc định Ruby đã cung cấp luôn cho chúng ta chương trình cài gem.

Cài đặt gem Rails:



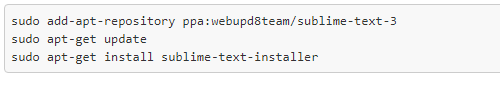
Kiểm tra cài đặt bằng lệnh: rails --version

* + - 1. *Cài đặt hệ quản trị cơ sở dữ liệu mysql*
* Cài đặt hệ quản trị csdl thông qua gem mysql:



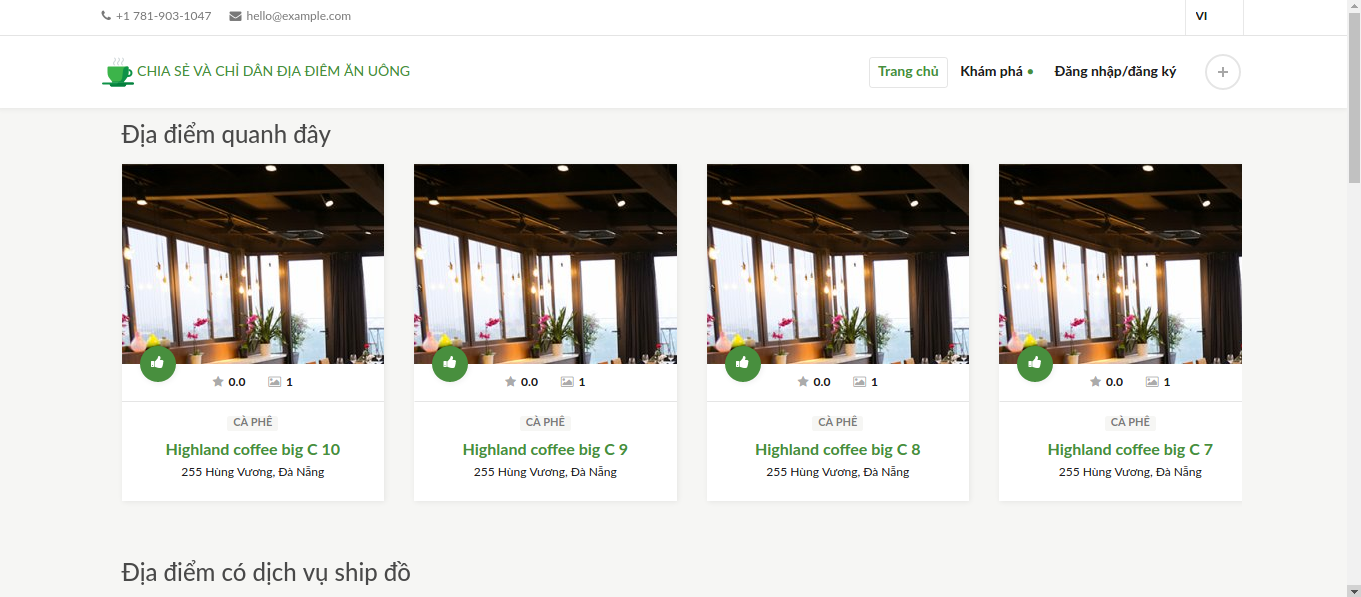
Kiểm tra cài đặt các gem bằng lệnh: gem list

* + - 1. *Cài đặt công cụ lập trình sublime text 3*
* Cài đặt sublime text:

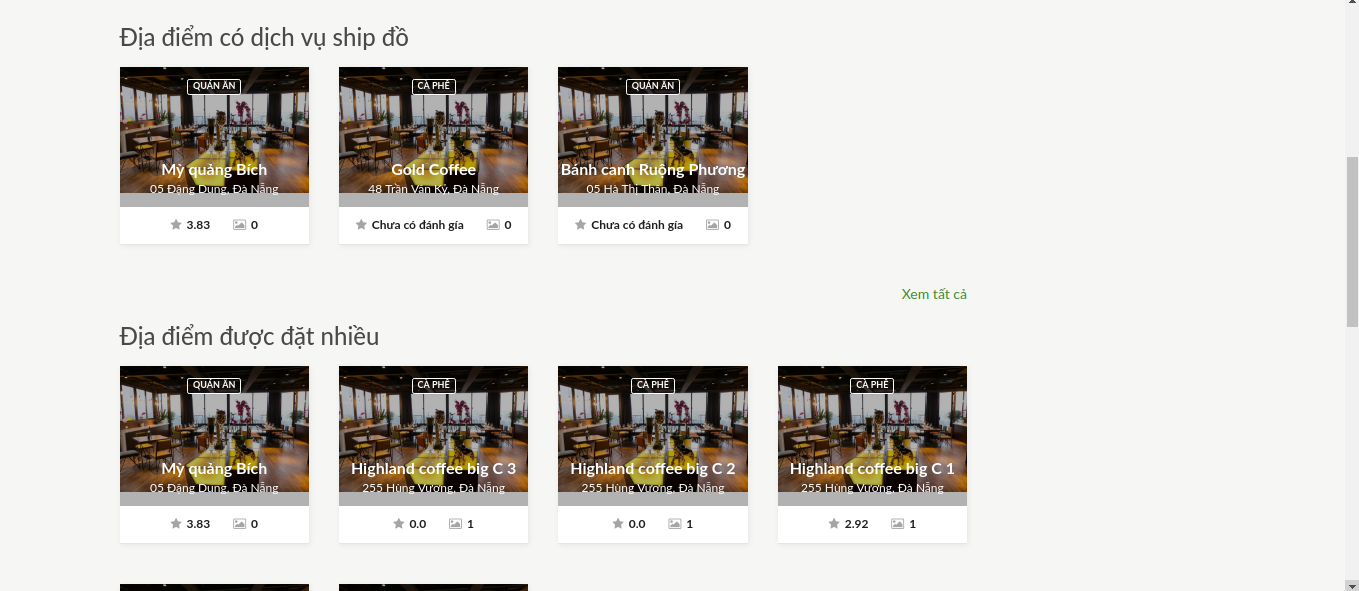


## Kết quả thực nghiệm

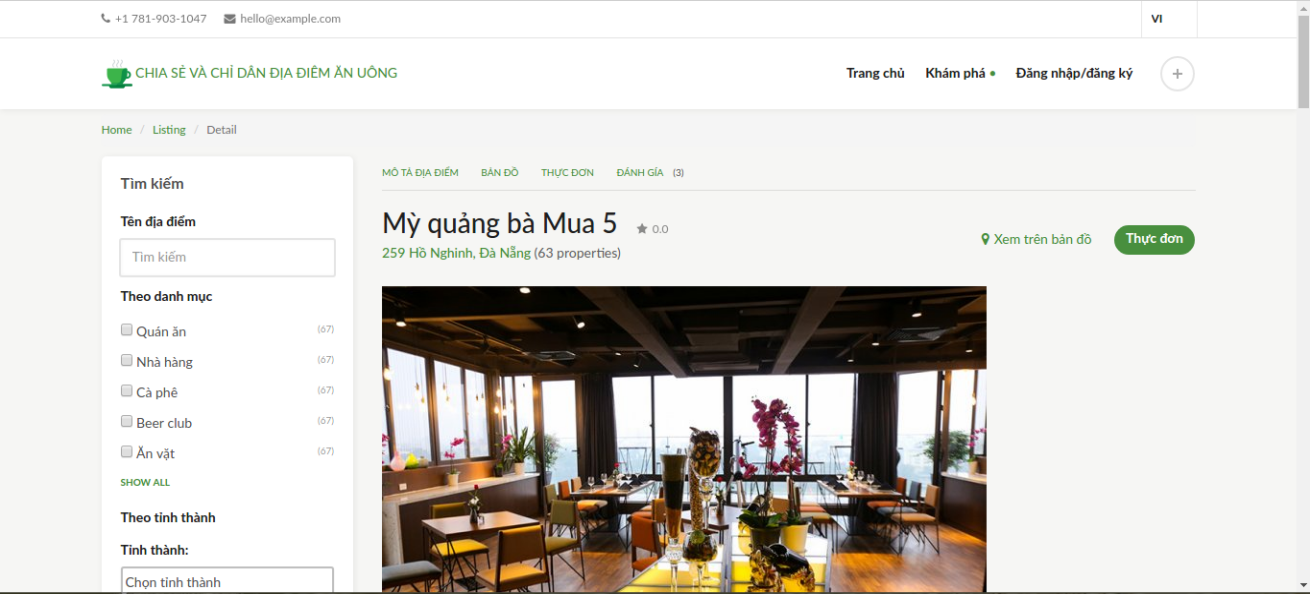
### Giao diện người dùng



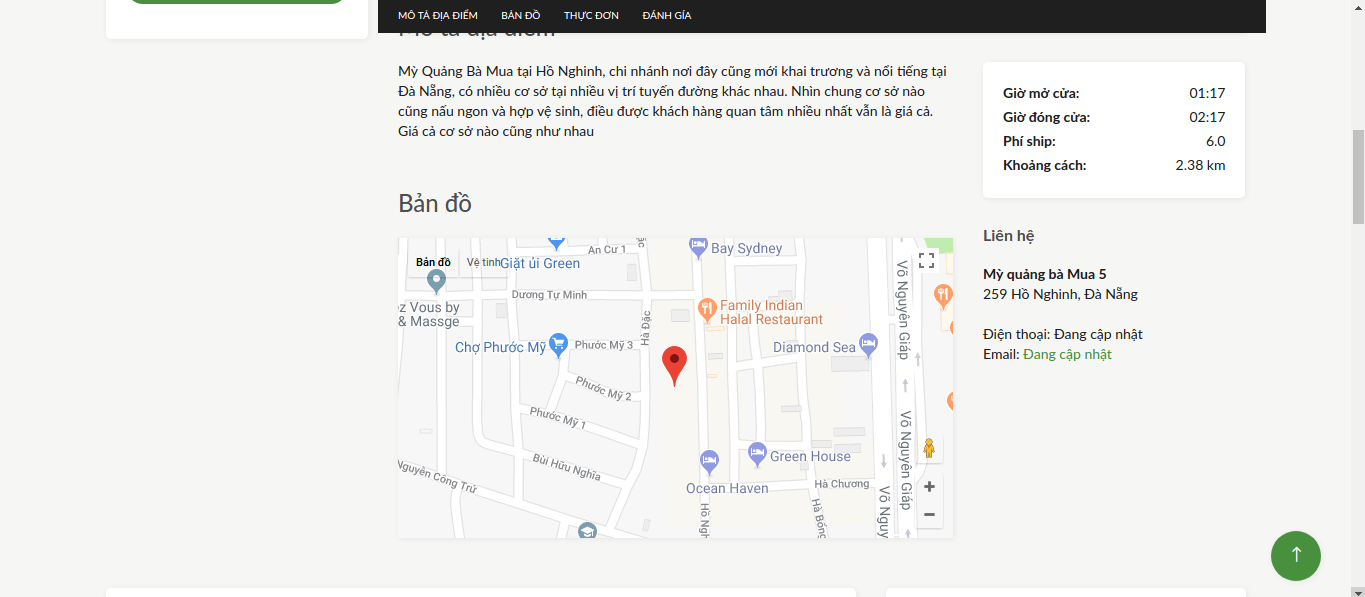
*Hình 3.1: Giao diện trang chủ*



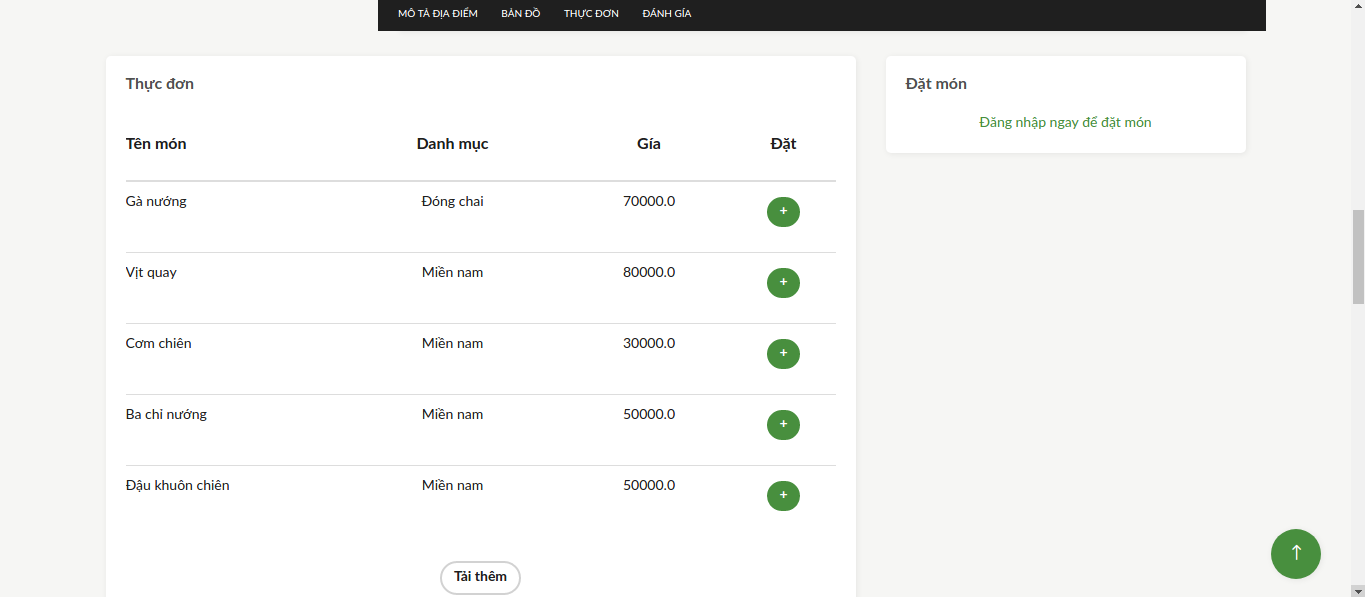
*Hình 3.2: Giao diện trang chủ (tiếp)*



*Hình 3.3: Giao diện trang chi tiết địa điểm*



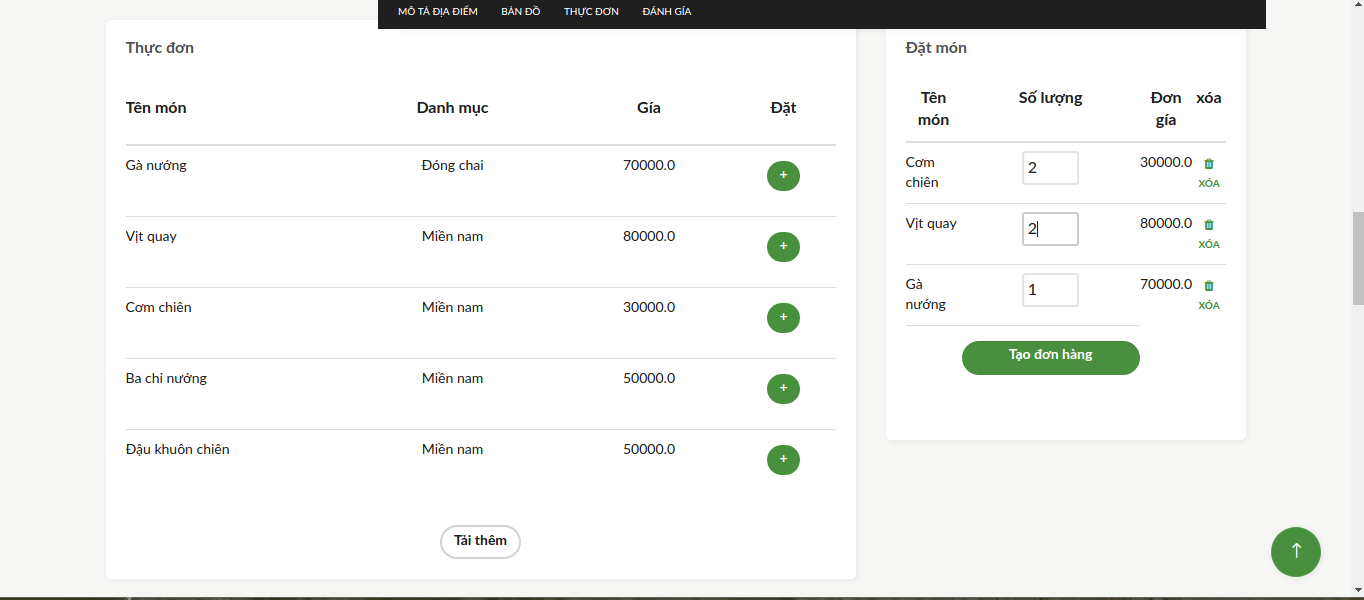
*Hình 3.4: Giao diện bản đồ địa điểm*



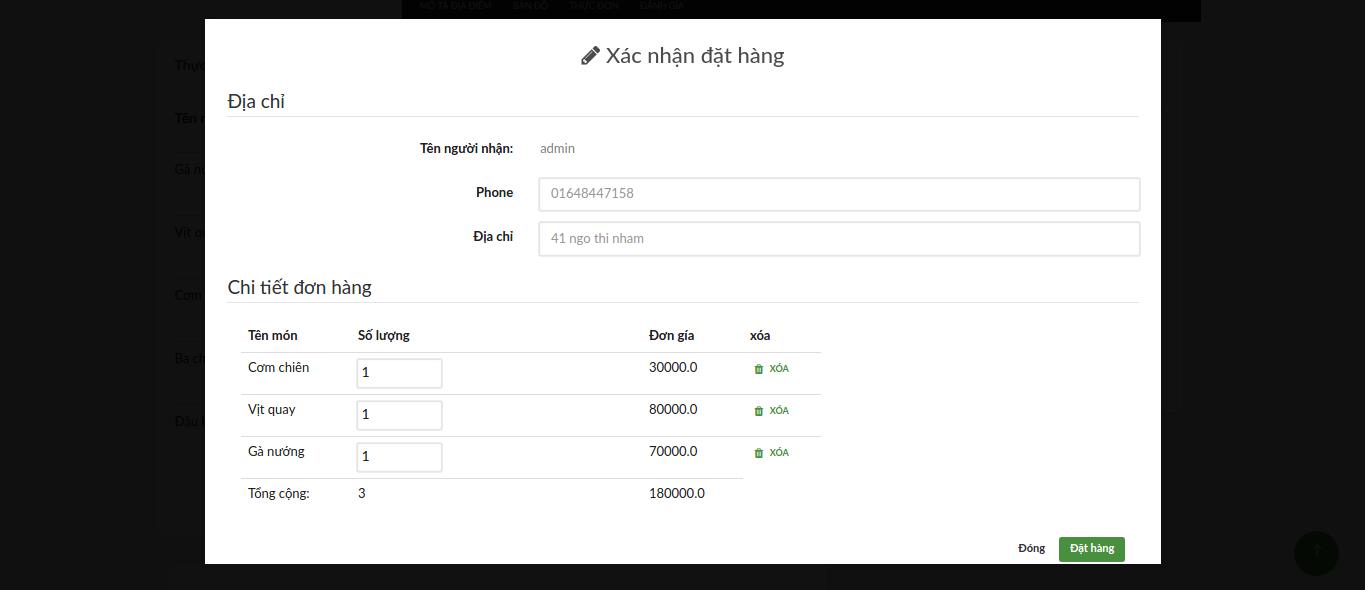
*Hình 3.5: Hiển thị các món ăn địa điểm phục vụ*



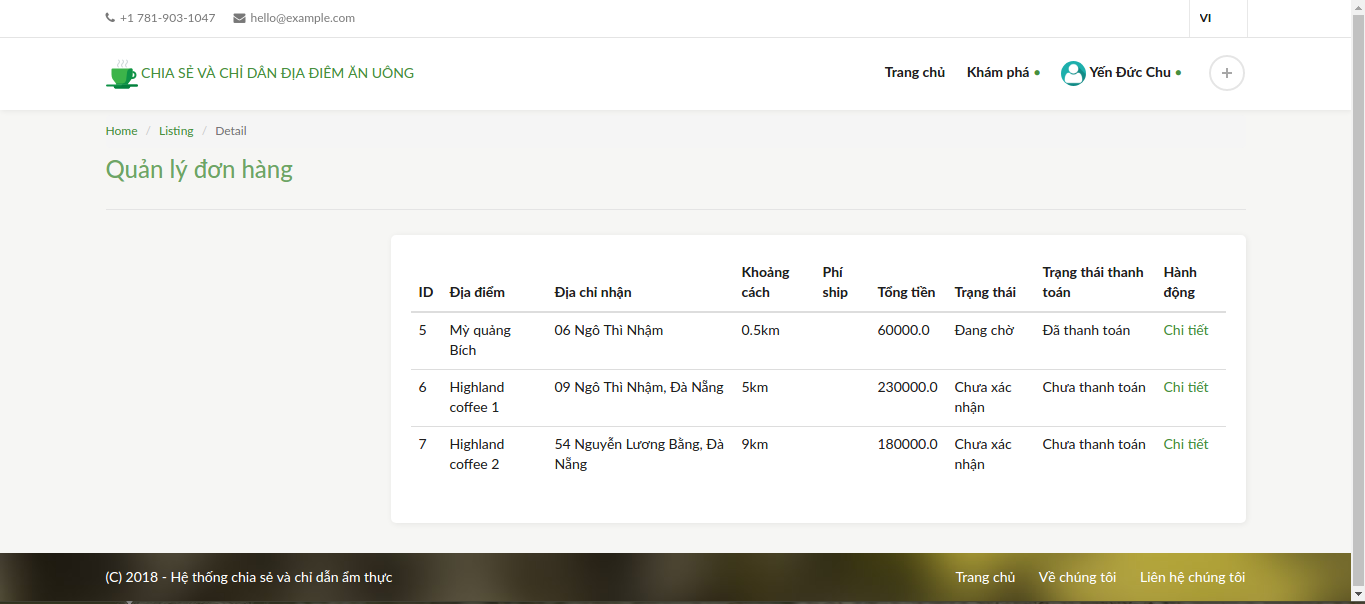
*Hình 3.6: Giao diện đăng nhập*



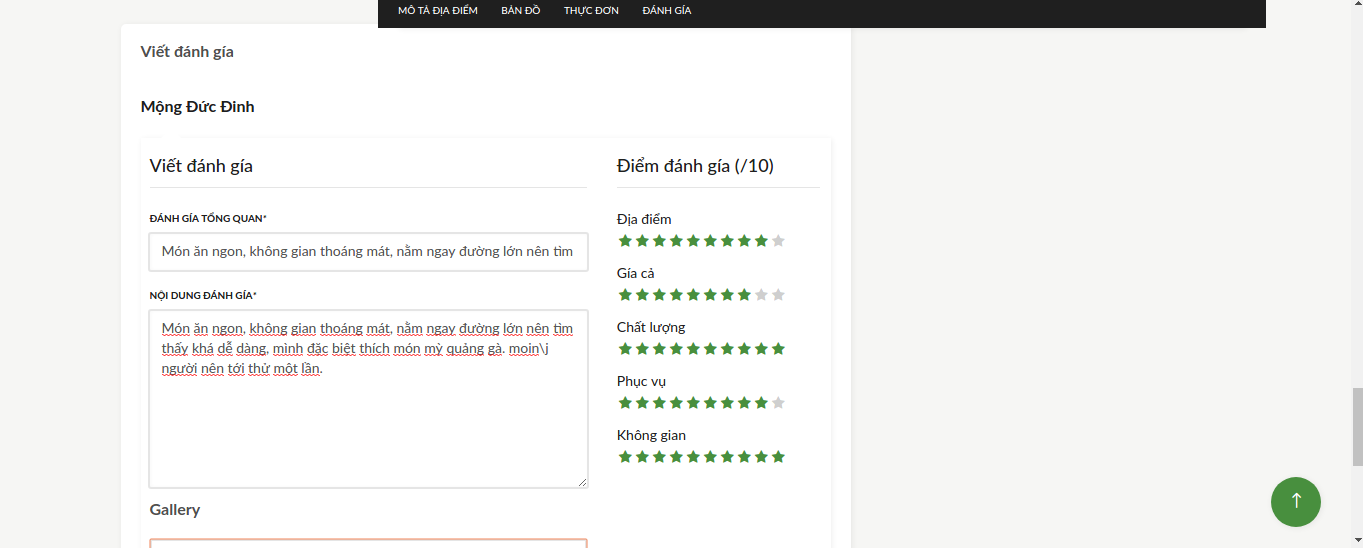
*Hình 3.7: Thành viên có thể đặt món*



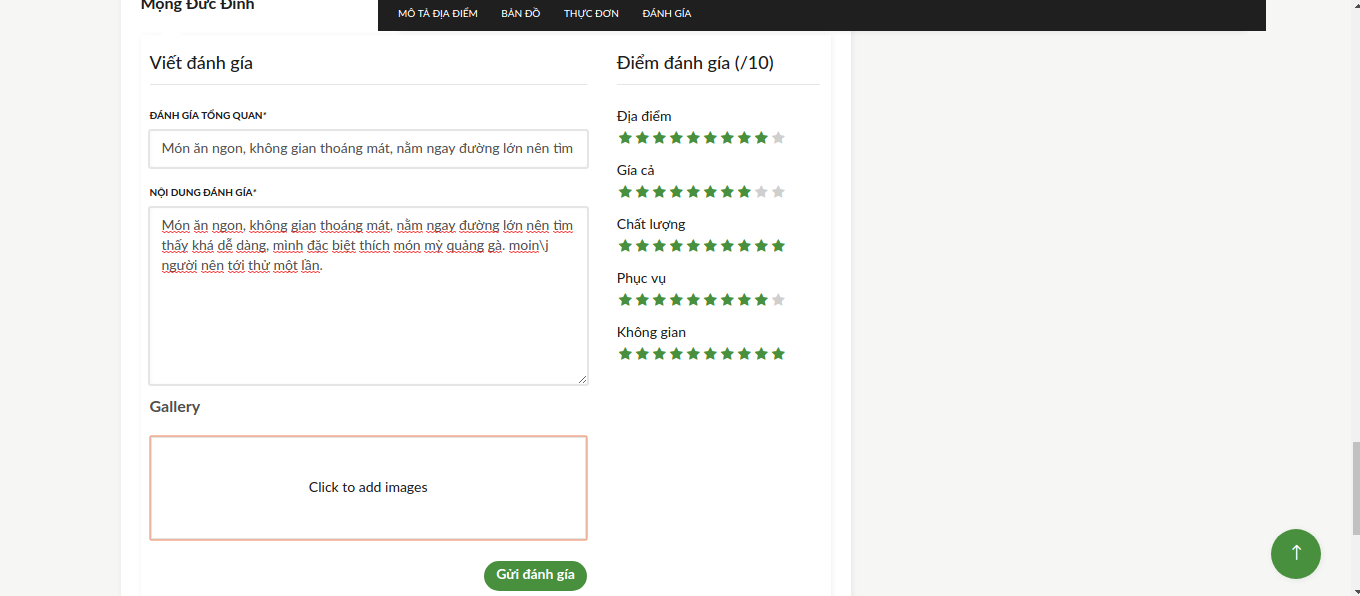
*Hình 3.8: Xác nhận đơn hàng*



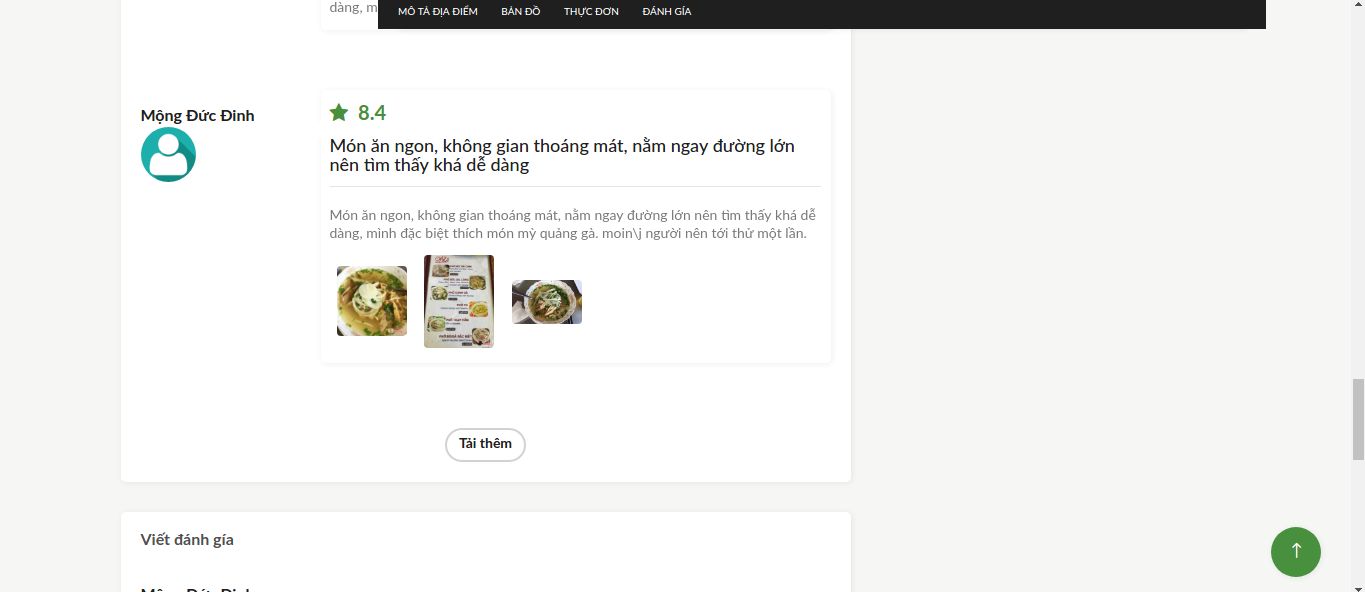
*Hình 3.9: Giao diện quản lý đơn hàng*



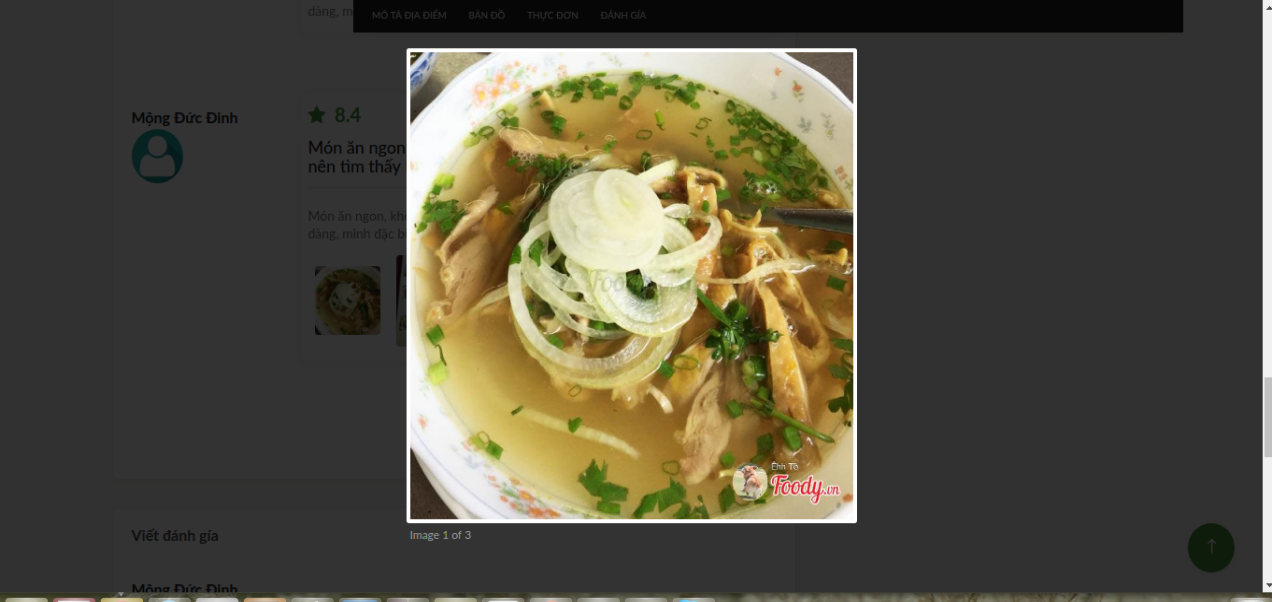
*Hình 3.10: Chức năng đánh giá địa điểm*



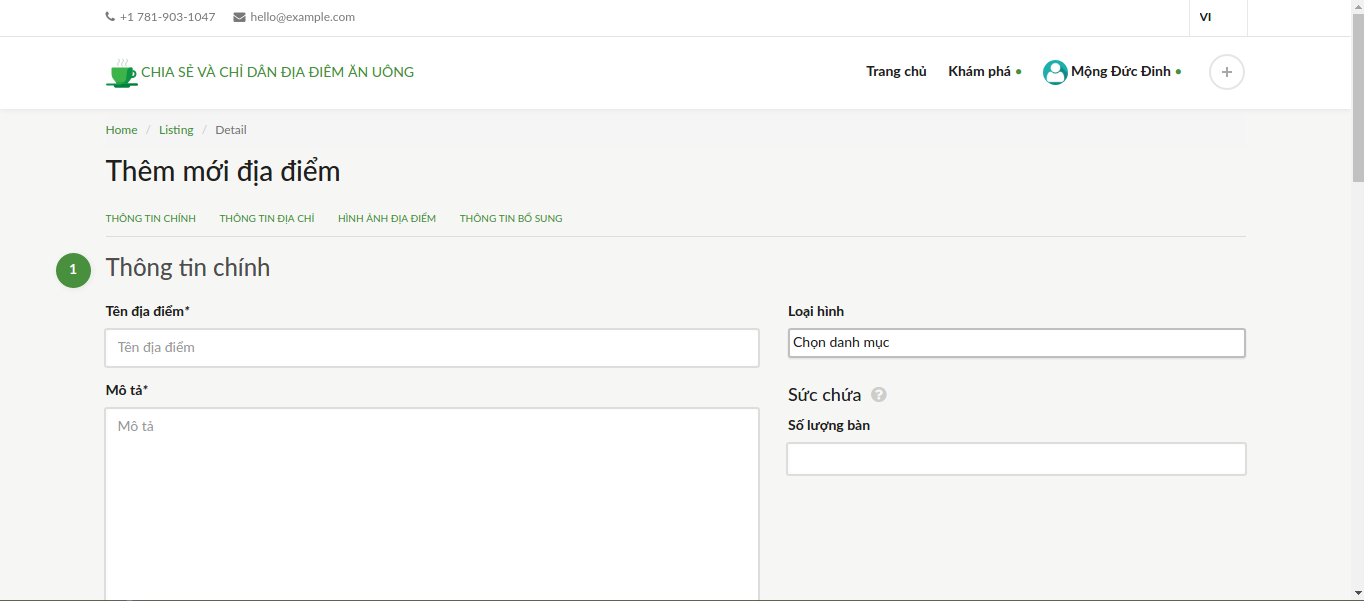
*Hình 3.11: Thêm hình ảnh địa điểm trong đánh giá*



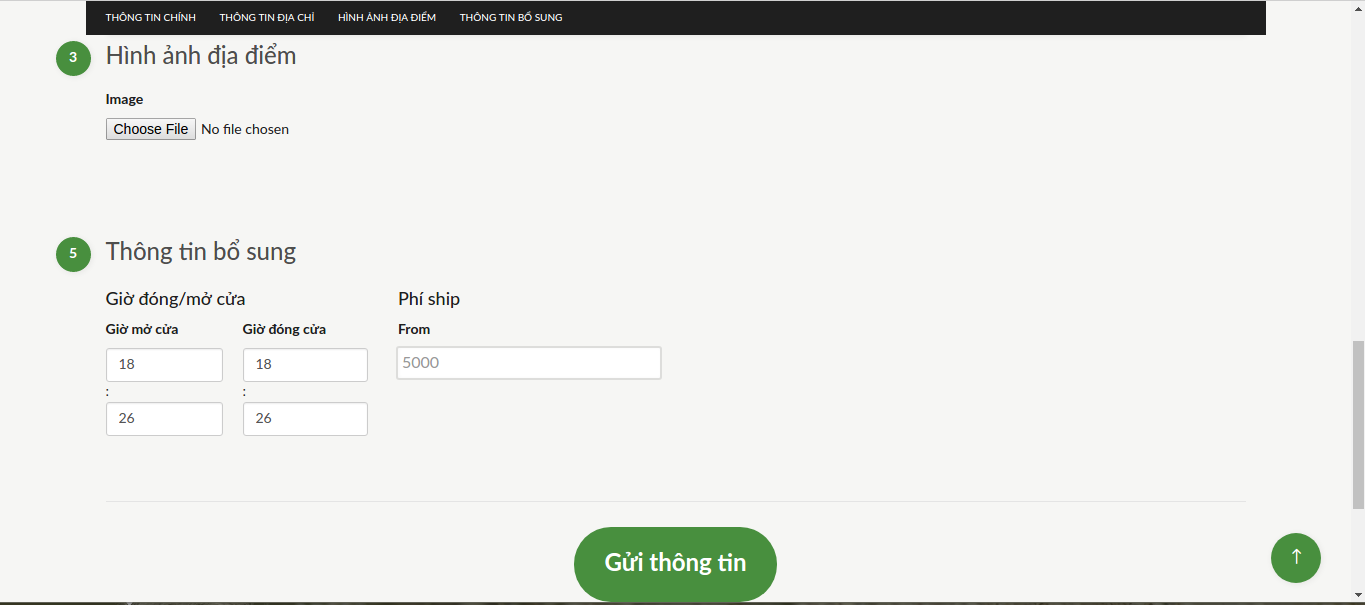
*Hình 3.12: Hiển thị bình luận của thành viên*



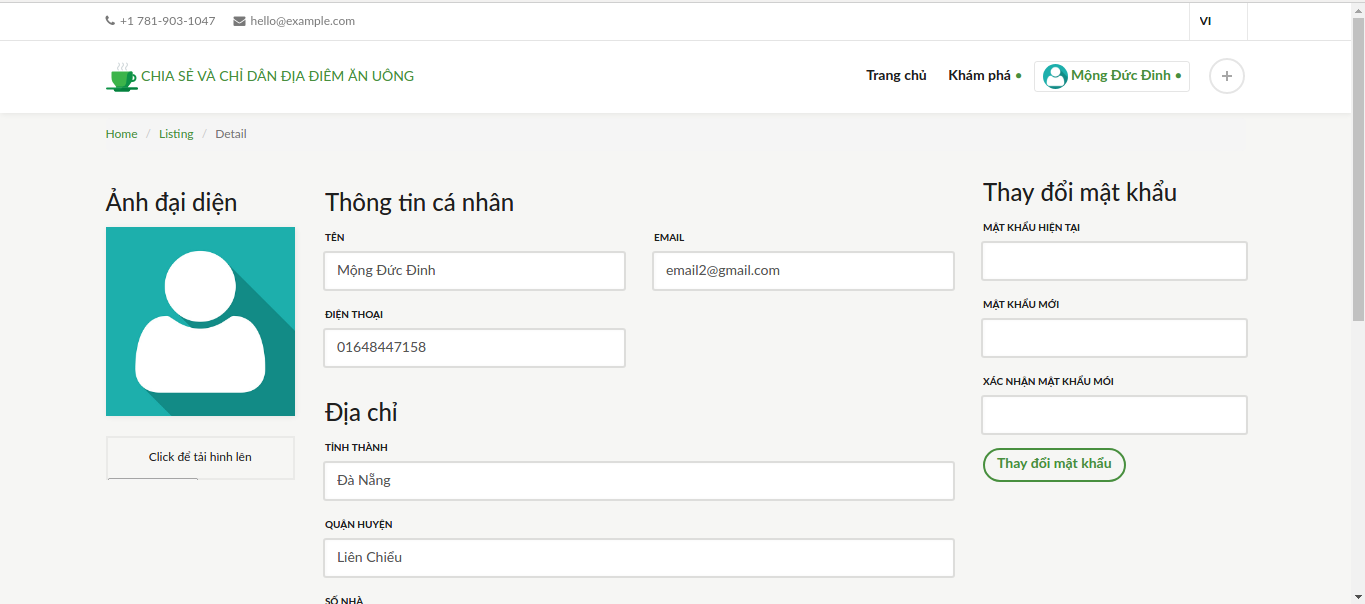
*Hình 3.13: Xem các hình ảnh về địa điểm*



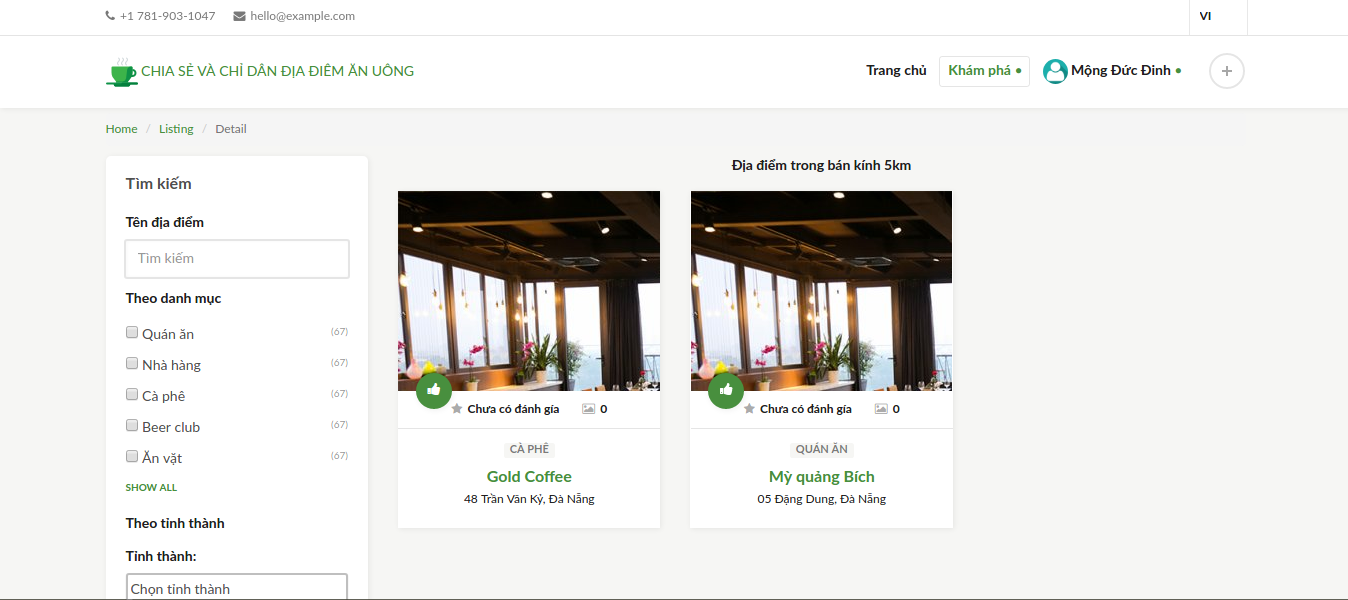
*Hình 3.14: Chức năng thêm địa điểm dành cho thành viên*



*Hình 3.15: Thêm thông tin cho địa điểm*

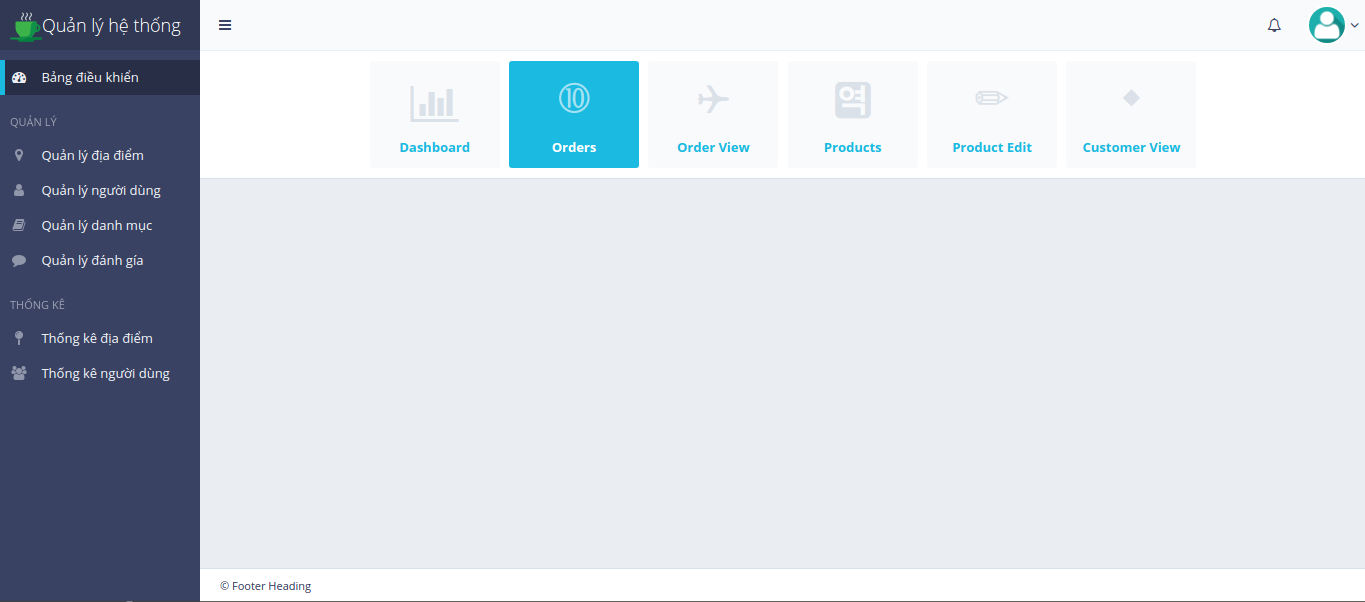


*Hình 3.16: Chức năng cập nhật thông tin thành viên*

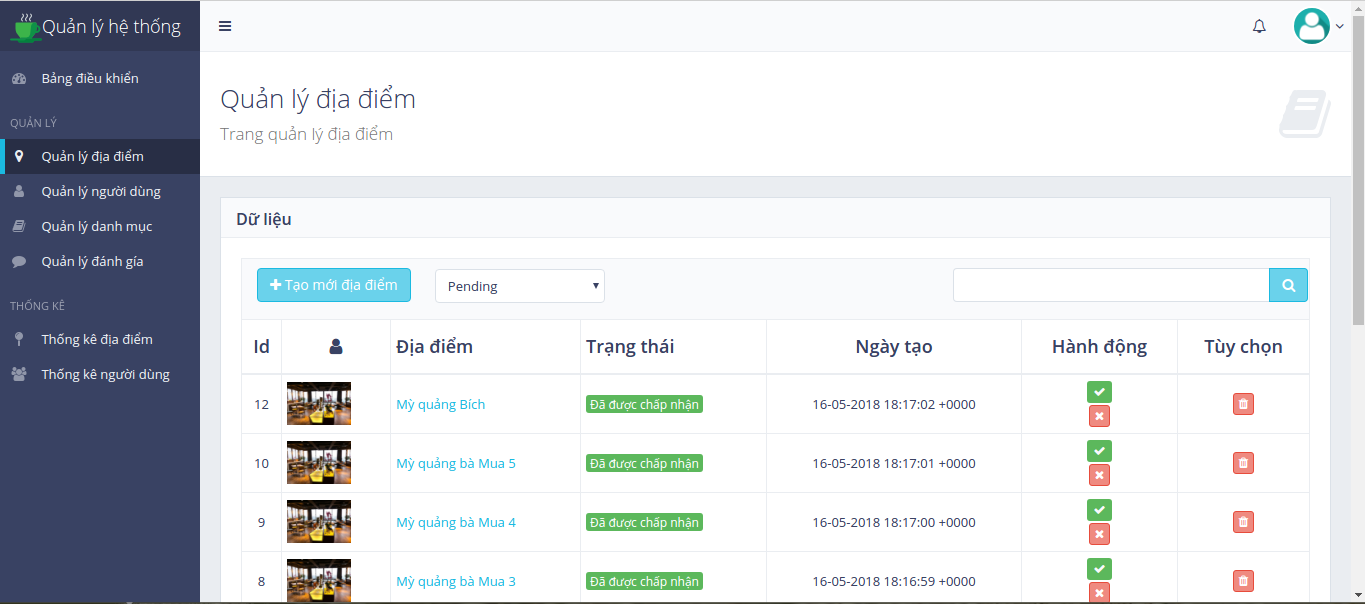


*Hình 3.17: Lọc địa điểm xung quanh khu vực theo khoảng cách*

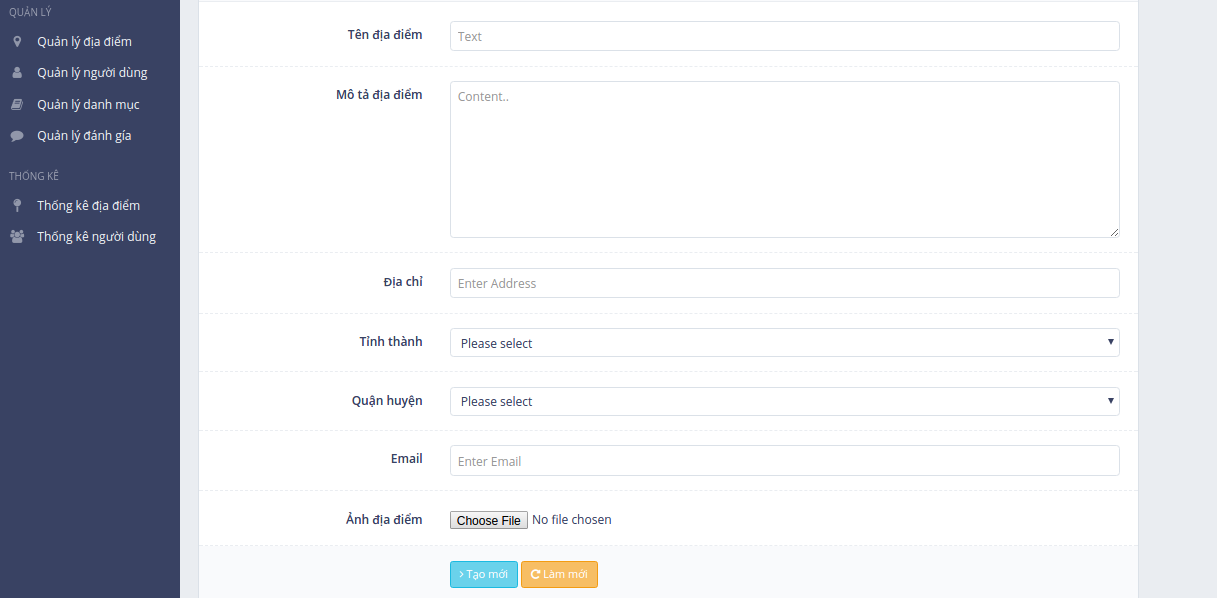
### Giao diện quản trị dành cho vai trò admin



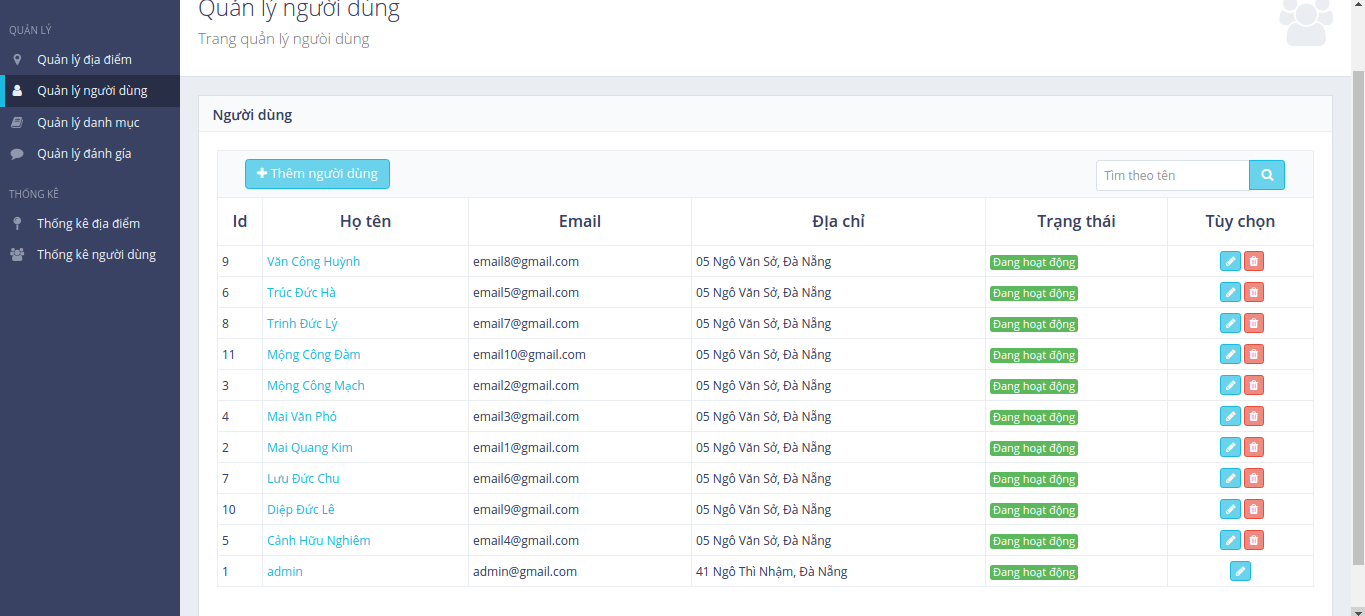
*Hình 3.18: Trang chủ giao diện admin*



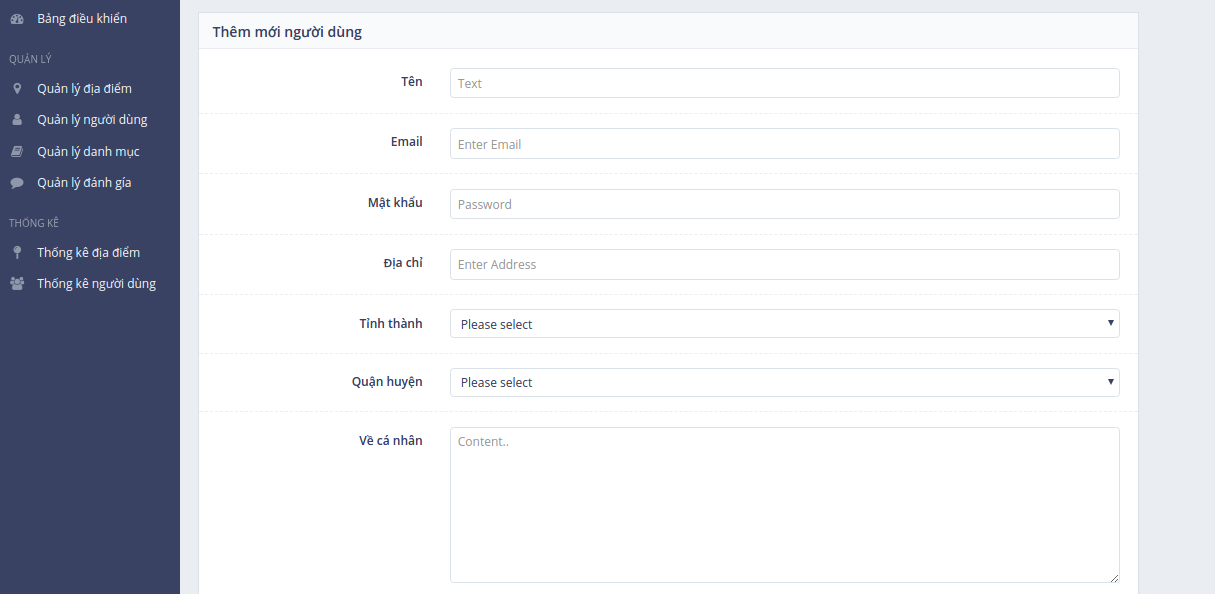
*Hình 3.19: Giao diện quản lý địa điểm*



*Hình 3.20: Chức năng thêm địa điểm*



*Hình 3.21: Chức năng quản lý người dùng*



*Hình 3.22: Chức năng thêm mới người dùng*

## Nhận xét và đánh giá kết quả

*Bảng 3.1: Đánh giá kết quả*

|  |  |  |
| --- | --- | --- |
| **Chức năng** | **Tình trạng** | **Mô tả** |
| Đăng nhập, đăng xuất | Hoàn thành |  |
| Đăng ký | Hoàn thành |  |
| Xác thực email | Hoàn thành |  |
| Tạo mới địa điểm | Hoàn thành |  |
| Xem chỉ dẫn tới địa điểm | Hoàn thành |  |
| Đánh giá, bình luận địa điểm | Hoàn thành |  |
| Tìm kiếm địa điểm | Hoàn thành |  |
| Lọc các địa điểm trong khu vực | Hoàn thành |  |
| Đặt món online | Hoàn thành |  |
| Quản lý người dùng | Hoàn thành |  |
| Quản lý địa điểm | Hoàn thành |  |
| Thống kê hệ thống | Hoàn thành |  |

## Kết chương

Chương này đã cung cấp được các bước triển khai thực tế của đồ án, các hình ảnh về sản phẩm lúc hoạt động thực tế.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. **Kết quả đạt được**

Trong thời gian tìm hiểu, nghiên cứu cơ sở lý thuyết và triển khai ứng dụng công nghệ, đồ án đã đạt được những kết quả sau:

1. *Về mặt lý thuyết, đồ án đã đạt được:*

* Hiểu được lý thuyết cơ bản của lập trình web.
* Các bước triển khai hệ thống để hoàn thiện một sản phầm trên thực tế.
* Tìm hiểu nhu cầu của người dùng về lĩnh vực ẩm thực.
* Phân tích thiết kế hệ thống, các luồng hoạt động cũng như các ca hoạt động cho một ứng dụng web nhiều chức năng.
* Cách ứng dụng framework vào một dự án.

1. *Về mặt thực tiễn ứng dụng, đồ án đã đạt được:*

Cung cấp được một kênh thông tin để có thể chia sẻ và chỉ dẫn các địa điểm ẩm thực, các chức năng cơ bản và cần thiết cho một hệ thống.

Hệ thống đã hoàn thiện các chức năng như:

* Đăng nhập, đăng xuất
* Đăng ký
* Xác thực email
* Tạo mới địa điểm
* Xem chỉ dẫn tới địa điểm
* Đánh giá, bình luận địa điểm
* Tìm kiếm địa điểm
* Lọc các địa điểm trong khu vực
* Đặt món online
* Quản lý người dùng
* Quản lý địa điểm
* Thống kê hệ thống

1. **Những điều chưa đạt được**

* Giao diện người dùng còn khá đơn giản.
* Các chức năng như thông báo còn chưa hoàn thiện.
* Các cài đặt, chức năng cá nhân cho thành viên còn ít.

1. **Hướng nghiên cứu và phát triển**

Một số số hướng nghiên cứu và phát triển của đề tài như sau:

* Cải thiện giao diện người dùng
* Hoàn thiện các tính năng đang có của hệ thống
* Bổ sung chức năng quảng bá địa điểm, thông báo về địa điểm có khuyến mãi cho thành viên
* Thêm các phương thức thanh toán cho chức năng đặt món
* Bổ sung chức năng đặt bàn và thêm bình luận video cho địa điểm

TÀI LIỆU THAM KHẢO

[1] Trang chủ Ruby: <https://www.ruby-lang.org/vi/>

[2] Trang chủ Ruby on Rails: <http://rubyonrails.org/>

[3] Trang Stackoverflow: <http://stackoverflow.com/>

[4] Trang chia sẻ kiến thức công nghệ Viblo: https://viblo.asia /

[5] Ruby on Rails Tutorial (Rails 5): [https://www.railstutorial.org/book /](https://www.railstutorial.org/book%20/)

[6] Google maps Javascript api: <https://developers.google.com/maps/documentation/javascript/directions>

<https://developers.google.com/maps/documentation/javascript/examples/map-geolocation>

[7] Wikipedia: https://wikipedia.org/