**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

BÁO CÁO THỰC TẬP CƠ SỞ NGÀNH

**ĐỀ TÀI: XÂY DỰNG WEBSITE GIỚI THIỆU ĐỊA ĐIỂM DU LỊCH**

**Giảng viên hướng dẫn: VŨ VIỆT THẮNG**

**Lớp: 20222IT6055002**

**Nhóm: 01**

**Nhóm sinh viên thực hiện: Mã sinh viên:**

1. **Nguyễn Hà Đức Minh 2021605748**
2. **Bùi Huy Nam 2021603337**
3. **Hà Thị Huệ 2021607262**
4. **Lương Thu Hường 2021603215**
5. **Tạ Thị Tuyết Mai 2021605461**

**Hà Nội 2023**

MỤC LỤC

[CHƯƠNG 1: TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU 5](#_Toc135508975)

[I. Giới thiệu về ngôn ngữ lập trình JavaScript, HTML, CSS và phần mềm Visual Studio Code 5](#_Toc135508976)

[1. Ngôn ngữ lập trình JavaScript 5](#_Toc135508977)

[2. Ngôn ngữ HTML và CSS 8](#_Toc135508978)

[2.1. Định nghĩa về HTML và CSS 8](#_Toc135508979)

[2.2. Cấu trúc cơ bản của HTML 8](#_Toc135508980)

[2.3. Các thẻ cơ bản trong HTML 9](#_Toc135508981)

[2.4. Bố cục và cấu trúc của một đoạn CSS 11](#_Toc135508982)

[2.5. Cách nhúng CSS vào website 12](#_Toc135508983)

[3. Phần mềm Visual Studio Code 13](#_Toc135508984)

[II. Giới thiệu về cơ sở dữ liệu Mysql 18](#_Toc135508985)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 22](#_Toc135508986)

[I. Khảo sát hệ thống 22](#_Toc135508987)

[1. Khảo sát sơ bộ 22](#_Toc135508988)

[2. Tài liệu đặc tả người dùng 22](#_Toc135508989)

[2.1. Khảo sát chi tiết 22](#_Toc135508990)

[2.2. Yêu cầu chức năng: 24](#_Toc135508991)

[2.3. Yêu cầu phi chức năng: 24](#_Toc135508992)

[II. Phân tích hệ thống 25](#_Toc135508993)

[1. Biểu đồ use case 25](#_Toc135508994)

[2. Mô tả chi tiết các use case 26](#_Toc135508995)

[2.1. Mô tả use case Đăng ký tài khoản 26](#_Toc135508996)

[2.2. Mô tả use case Đăng nhập 27](#_Toc135508997)

[2.3. Mô tả use case Tìm kiếm địa điểm 28](#_Toc135508998)

[2.4. Mô tả use case Đặt chỗ trực tuyến 29](#_Toc135508999)

[2.5. Mô tả use case Quản lý tài khoản 30](#_Toc135509000)

[2.6. Mô tả use case Tổng hợp số lượng khách hàng 31](#_Toc135509001)

[3. Biểu đồ trình tự 32](#_Toc135509002)

[3.1. Use case Đăng ký tài khoản 32](#_Toc135509003)

[3.2. Use case Đăng nhập 33](#_Toc135509004)

[3.3. Use case Tìm kiếm địa điểm 34](#_Toc135509005)

[3.4. Use case Đặt chỗ trực tuyến 35](#_Toc135509006)

[3.5. Use case Quản lý tài khoản 36](#_Toc135509007)

[3.6. Use case Tổng hợp số lượng khách hàng 37](#_Toc135509008)

[4. Mô hình hoá dữ liệu của hệ thống 38](#_Toc135509009)

[4.1. Các yêu cầu về dữ liệu 38](#_Toc135509010)

[4.2. Biểu đồ thực thể liên kết mức logic 39](#_Toc135509011)

[4.3. Biểu đồ thực thể liên kết mức vật lý 39](#_Toc135509012)

[5. Thiết kế cơ sở dữ liệu 40](#_Toc135509013)

[5.1. Bảng phòng 40](#_Toc135509014)

[5.2. Bảng Khách hàng 40](#_Toc135509015)

[5.3. Bảng Đặt phòng 41](#_Toc135509016)

[5.4. Bảng Địa điểm 41](#_Toc135509017)

[5.5. Bảng Tài khoản 42](#_Toc135509018)

[5.6. Bảng Admin 42](#_Toc135509019)

[5.7. Bảng báo cáo 42](#_Toc135509020)

[5.8. Mô hình cơ sở dữ liệu mức vật lý 42](#_Toc135509021)

DANH MỤC HÌNH ẢNH

[Hình 1: Ngôn ngữ Javascript 7](#_Toc135511114)

[Hình 2: Phần mềm Visual Studio 14](#_Toc135511115)

[Hình 3: Ngôn ngữ lập trình C# 15](#_Toc135511116)

[Hình 4: Kho lưu trữ an toàn 16](#_Toc135511117)

[Hình 5: Intellisense 17](#_Toc135511118)

[Hình 6 : Visual Studio 2022 18](#_Toc135511119)

[Hình 7 : Sơ đồ usecase 26](#_Toc135511120)

[Hình 8: Use case Đăng kí tài khoản 33](#_Toc135511121)

[Hình 9 : Use case Đăng nhập 34](#_Toc135511122)

[Hình 10: Use case Tìm kiếm địa điểm 35](#_Toc135511123)

[Hình 11: Use case Đặt chỗ trực tuyến 36](#_Toc135511124)

[Hình 12: Use case Quản lý tài khoản 37](#_Toc135511125)

[Hình 13: Use case Tổng hợp số lượng khách hàng 38](#_Toc135511126)

[Hình 14: Biểu đồ thực thể liên kết mức logic 39](#_Toc135511127)

[Hình 15: Biểu đồ thực thể mức vật lý 40](#_Toc135511128)

[Hình 16: Bảng phòng 40](#_Toc135511129)

[Hình 17: Bảng khách hàng 41](#_Toc135511130)

[Hình 18: Bảng Đặt phòng 41](#_Toc135511131)

[Hình 19: Bảng Địa điểm 41](#_Toc135511132)

[Hình 20: Bảng Tài khoản 42](#_Toc135511133)

[Hình 21: Bảng Admin 42](#_Toc135511134)

[Hình 22: Bảng Báo cáo 42](#_Toc135511135)

[Hình 23: Mô hình cơ sở dữ liệu mức vật lý 43](#_Toc135511136)

LỜI MỞ ĐẦU

Ngày nay, công nghệ thông tin đã có những bước phát triển mạnh mẽ trên mọi lĩnh vực, hoạt động đời sống xã hội, đã và đang dần trở thành nhu cầu rất cần thiết và không thể thiếu trong cuộc sống con người, trong bất kì một tổ chức, lĩnh vực nào. Công nghệ thông tin là nhu cầu cần thiết đối với các công ty, doanh nghiệp, các tổ chức kinh tế xã hội…cũng như mỗi cá nhân chúng ta.

Đứng trước vai trò của công nghệ thông tin, rất nhiều các ứng dụng, website quản lý doanh nghiệp, quản lý cá nhân ra đời đã ra đời và thực sự chiếm được lòng tin của khách hàng. Tuy nhiên để cho ra đời một sản phẩm có chất lượng thì người thiết kế cần đáp ứng được một số điều kiện sau: độ chính xác cao, lưu trữ dữ liệu thuận tiện, dễ dàng tra cứu, đáp ứng tốt các nhu cầu mới phát sinh, cho hiệu quả cao, phù hợp với nhu cầu, điều kiện người dùng. Trong thế giới ngày càng phát triển , nhu cầu đi du lịch của mọi người ngày càng cao. Nhưng câu hỏi luôn đặt ra là đi đâu , đến đâu thì thoải mái , đi đâu để ngắm được cảnh đẹp .Theo thống kê ngành du lịch ngày càng đóng vai trò quan trọng về mọi mặt , đặc biệt trong mức độ tăng trưởng GDP cho đất nước. Bất kể khu vực nào có tiềm năng về du lịch đều được nhà nước , công ty du lịch chú trọng đầu tư và phát triển để trở thành khu du lịch thu hút quan khách trong và ngoài nước.

Trong xu thế đó, để tiếp cận và góp phần đẩy mạnh sự phổ biến của website quảng cáo ở Việt Nam, giải quyết vấn đề của khách du lịch , nhóm đã quyết định thực hiện đề tài: “Xây dựng website giới thiệu địa điểm du lịch”. Với mong muốn đáp ứng nhu cầu khách du lịch trong và ngoài nước: chọn được địa điểm du lịch yêu thích cho bản thân , là cầu nối giữa công ty du lịch và khách du lịch, giúp nâng cao chất lượng cuộc sống.

Nhóm thực hiện!

# CHƯƠNG 1: TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU

## I. Giới thiệu về ngôn ngữ lập trình JavaScript, HTML, CSS và phần mềm Visual Studio Code

### 1. Ngôn ngữ lập trình JavaScript

**JavaScript** là ngôn ngữ lập trình website phổ biến hiện nay, nó được tích hợp và nhúng vào HTML giúp website trở nên sống động hơn. JavaScript đóng vai trò như là một phần của trang web, thực thi cho phép Client-side script từ phía người dùng cũng như phía máy chủ (Nodejs) tạo ra các trang web động.

[***JavaScript***](https://www.javascript.com/) là một ngôn ngữ lập trình thông dịch với khả năng hướng đến đối tượng. Là một trong 3 ngôn ngữ chính trong lập trình web và có mối liên hệ lẫn nhau để xây dựng một website sống động, chuyên nghiệp:

* HTML: Hỗ trợ trong việc xây dựng layout, thêm nội dung dễ dàng trên website.
* CSS: Hỗ trợ việc định dạng thiết kế, bố cục, style, màu sắc,…
* JavaScript: Tạo nên những nội dung “động” trên website. Cùng tìm hiểu rõ hơn ở phần dưới đây.
* JS là viết tắt của JavaScript, khi có JS bạn sẽ hiểu đó đang nói đến JavaScript.

Nhiệm vụ của Javascript là xử lý những đối tượng HTML trên trình duyệt. Nó có thể can thiệp với các hành động như thêm / xóa / sửa các thuộc tính CSS và các thẻ HTML một cách dễ dàng. Hay nói cách khác, Javascript là một ngôn ngữ lập trình trên trình duyệt ở phía client. Tuy nhiên, hiện nay với sự xuất hiện của NodeJS đã giúp cho Javascript có thể làm việc ở backend.

***\* Lịch sử phát triển của Javascript***

- Brendan Eich chính là người đã phát triển JS tại Netscape với tiền thân là Mocha. - Sau đó, Mocha được đổi thành LiveScript và cuối cùng mới đổi thành JavaScript.

- Năm 1998, JavaScript với phiên bản mới nhất là ECMAScript 2 phát hành và đến năm 1999 thì ECMAScript 3 được ra mắt.

- Năm 2016, ứng dụng JavaScript đã đạt kỷ lục lên tới 92% website sử dụng, đồng thời cũng được đánh giá là một công cụ cực kỳ quan trọng đối với lập trình viên.

***\*Một số ưu điểm nổi bật của ngôn ngữ lập trình JS như sau:***

* Chương trình rất dễ học.
* Những lỗi Javascript rất dễ để phát hiện, từ đó giúp bạn sửa lỗi một cách nhanh chóng hơn.
* Những trình duyệt web có thể dịch thông qua HTML mà không cần sử dụng đến một compiler.
* JS có thể hoạt động ở trên nhiều nền tảng và các trình duyệt web khác nhau.
* Được các chuyên gia đánh giá là một loại ngôn ngữ lập trình nhẹ và nhanh hơn nhiều so với các ngôn ngữ lập trình khác.
* JS còn có thể được gắn trên một số các element hoặc những events của các trang web.
* Những website có sử dụng JS thì chúng sẽ giúp cho trang web đó có sự tương tác cũng như tăng thêm nhiều trải nghiệm mới cho người dùng.
* Người dùng cũng có thể tận dụng JS với mục đích là để kiểm tra những input thay vì cách kiểm tra thủ công thông qua hoạt động truy xuất database.
* Giao diện của ứng dụng phong phú với nhiều thành phần như Drag and Drop, Slider để cung cấp đến cho người dùng một Rich Interface (giao diện giàu tính năng).
* Giúp thao tác với người dùng phía Client và tách biệt giữa các Client với nhau.

***\*Một số nhược điểm nổi bật của ngôn ngữ lập trình JS như sau:***

Bên cạnh những ưu điểm kể trên thì JS vẫn có những nhược điểm riêng tương tự như các ngôn ngữ lập trình khác hiện nay. Cụ thể:



Hình : Ngôn ngữ Javascript

* JS Code Snippet khá lớn.
* JS dễ bị các hacker và scammer khai thác hơn.
* JS cũng không có khả năng đa luồng hoặc đa dạng xử lý.
* Có thể được dùng để thực thi những mã độc ở trên máy tính của người sử dụng.
* Những thiết bị khác nhau có thể sẽ thực hiện JS khác nhau, từ đó dẫn đến sự không đồng nhất.
* Vì tính bảo mật và an toàn nên các Client-Side Javascript sẽ không cho phép đọc hoặc ghi các file.
* JS không được hỗ trợ khi bạn sử dụng ở trong tình trạng thiết bị được kết nối mạng.

### 2. Ngôn ngữ HTML và CSS

#### 2.1. Định nghĩa về HTML và CSS

HTML (HyperText Markup Language) : là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web, nghĩa là các mẩu thông tin được trình bày trên World Wide Web.

CSS (Cascading Style Sheets) : định nghĩa về cách hiển thị của một tài liệu HTML. CSS đặc biệt hữu ích trong việc thiết kế Web. Nó giúp cho người thiết kế dễ dàng áp đặt các phong cách đã được thiết kế lên bất kì page nào của website một cách nhanh chóng, đồng bộ.

#### 2.2. Cấu trúc cơ bản của HTML

Có thể dùng notepad hoặc text editor để soạn thảo 1 tài liệu HTML. Ta cùng làm 1 ví dụ nhỏ về HTML nhé:

B1: Mở notepad hoặc text editor

B2: Nhập đoạn HTML sau:

<!DOCTYPE html>

<html>

<head>

    <title>Page Title</title>

</head>

<body>

    <h1>My First Heading</h1>

    <p>My first paragraph</p>

</body>

</html>

B3: Save file với định dạng .html hoặc .htm

B4: Thử xem kết quả ra là gì nhé. Bằng cách mở file bằng 1 trình duyệt bất kỳ. Ví dụ: Chrome, IE, Firefox, Safari.

Cấu trúc cơ bản của HTML:

* Một tài liệu HTML luôn được gói trong cặp tag <html> và </html>
* Cặp tag <body> và </body> sẽ là nơi mô tả những gì có thể nhìn thấy của trang.

Như vậy, một trang web viết bằng html sẽ có cấu trúc cơ bản như sau:

<html>

   <body>

         Phần trình bày nội dung

   </body>

</html>

Ngoài phần body còn có phần header, được viết bởi cặp tag <head></head>. Nếu bạn sử dụng cặp tag này, bạn bắt buộc phải viết thêm một cặp tag nữa, đó là <title></title>. Giữa <title> và </title> là tên của trang web được hiển thị phía trên cùng của menubar. Như vậy một trang web với lúc này sẽ có cấu trúc như sau:

<head>

    <title>Tiêu đề của trang web</title>

</head>

<body>

    Phần trình bày nội dung

</body>

</html>

#### 2.3. Các thẻ cơ bản trong HTML

Một tài liệu HTML được tạo nên từ các cặp thẻ html

* Thẻ được bắt đầu bằng dấu < và kết thúc bằng dấu >
* Tên thẻ nằm giữa cặp dấu <>
* Cặp thẻ được tạo nên từ thẻ mở và thẻ đóng
* Nội dung của thẻ sẽ được nằm giữa thẻ đóng và thẻ mở
* Cũng có một số thẻ chỉ có thẻ mở mà không có thẻ đóng.

A.Các thẻ tiêu đề (HTML Headings):

* Thường được sử dụng để thế hiện cho tiêu đề của bài viết, bản tin hay các mục nhấn mạnh
* Bao gồm các thẻ từ <h1> đến <h6>
* Font chữ của nội dung trong các thẻ giảm từ <h1> đến <h6>

B. Đoạn văn bản trong html (HTML Paragraphs):

* Nội dung văn bản được thể hiện trong cặp thẻ <p></p>

C. Liên kết (HTML Links):

* Ta dùng cặp thẻ <a></a> để làm công việc liên kết các trang web với nhau
* Thuộc tính của thẻ <a> gồm:
* href: qui định địa chỉ mà url trỏ tới

target: qui định liên kết sẽ được mở ra ở đâu  
D. Xuống dòng (HTML Line Breaks):

* Ta dùng thẻ <br /> để xuống dòng trong một đoạn văn bản

E. HTML Lines (<hr />):

* Ta dùng thẻ <hr /> để tạo một đường kẻ ngang trong trang HTML

F. HTML images (Hình ảnh):

* Ta dùng thẻ <img> để chèn ảnh vào trang web
* Thuộc tính của thẻ <img> gồm:
  + src: chỉ ra đường dẫn file ảnh
  + alt: để mô tả nội dung sẽ hiển thị khi đường dẫn tới file ảnh không tồn tại
  + title=”Tiêu đề”: nội dung hiển thị khi đưa trỏ chuột lên hình.
  + width, height: độ rộng và độ cao của file được tính bằng excel, nếu không có width và height thì mặc định sẽ lấy kích thước gốc của file

G. Các thẻ định dạng text (HTML Text Formatting):

* <b> (bold): Chữ In đậm
* <u> (Underline): Chữ gạch chân
* <i> (italic): Chữ in nghiêng
* <big> (Big): Chữ lớn hơn
* <sub> (Subscript) Chỉ số dưới, ví dụ: H2O
* <sup> (Superscript): Chỉ số trên, ví dụ: x2y
* <strong> In đậm (nhấn mạnh <b>)

#### Bố cục và cấu trúc của một đoạn CSS

❖ Bố cục của một đoạn CSS

Bố cục của một đoạn CSS chủ yếu dựa vào hình hộp với mỗi hộp chiếm

những khoảng trống trên trang web với các thuộc tính chính như:

* Padding: Là các không gian xung quanh nội dung (ví dụ: không gian
* xung quanh đoạn văn bản).
* Border: Là các đường nằm ngoài phần đệm.
* Margin: Là khoảng cách bao quanh phía ngoài của phần tử.

❖ Cấu trúc của một đoạn CSS

* Thông thường, một đoạn CSS sẽ bao gồm các phần:vùng chọn {thuộc tính: giá trị; thuộc tính: giá trị;….. }
* Đoạn CSS sẽ được khai báo bằng vùng chọn, các thuộc tính, giá trị nằm trong dấu ngoặc nhọn. Mỗi thuộc tính là một giá trị riêng ở dạng số, hoặc chính là tên của các giá trị đã có trong danh sách của CSS.
* Quy tắc khai báo đó chính là: thuộc tính và giá trị cần cách nhau bằng dấu hai chấm, mỗi dòng khai báo thuộc tính cần có dấu chấm phẩy cuối cùng. Các thuộc tính không bị giới hạn ở một vùng chọn.

Trong đó:

* Bộ chọn (Selector): mẫu để chọn phần tử HTML mà bạn muốn định nghĩa phong cách. Bạn có thể áp dụng các selector cho các trường hợp sau:
* Tất cả những phần tử được định dạng theo một dạng cụ thể nào đó, ví dụ phần tử tiêu đề h2.
* Thuộc tính id, class của phần tử.
* Các phần tử có mối liên quan với các phần tử khác trong hệ thống cây phân cấp tài liệu.
* Khai báo (Declaration): Khối khai báo có thể chứa một hoặc nhiều khai báo và chúng được phân tách với nhau bằng dấu chấm phẩy. Mỗi khai báo lại bao gồm tên &amp; giá trị đặc tính CSS, dược phân tách với nhau bằng dấu phẩy. Quy tắc khai báo CSS là chúng luôn phải kết thúc bằng dấu chấm phẩy, và khối khai báo phải nằm trong các dấu ngoặc móc.
* Thuộc tính (Properties): Thuộc tính là các cách thức mà bạn có thể tạo kiểu cho một phần tử HTML. Vì vậy, với CSS, bạn chỉ cần lựa chọn thuộc tính mà bạn muốn tác động nhất trong bộ quy tắc bạn đã tạo ra.
* Giá trị thuộc tính: Được nằm ở bên phải của thuộc tính. Việc lựa chọn một thuộc tính trong số đó phụ thuộc vào số lần xuất hiện của thuộc tính.

#### Cách nhúng CSS vào website

❖ Có ba cách nhúng CSS vào website:

* *Nhúng CSS trực tiếp CSS vào tài liệu HTML (Inline CSS)*

Với cách này, chúng ta đặt mã CSS vào thẳng thuộc tính style của phần tử. Và với cách nhúng trực tiếp, mã CSS sẽ chỉ tác động lên chính phần tử đó.

Để chỉ định nhiều quy tắc CSS, chúng ta có thể sử dụng dấu chấm phẩy để ngăn cách giữa các rules. Các rules này sẽ được đặt bằng cách sử dụng thuộc tính “style” kèm theo tên thuộc tính, giá trị thuộc tính.

* *Nội tuyến (Internal CSS)*

Với cách nhúng nội tuyến, bạn cần dùng thẻ &lt;style&gt; để tạo ra khu vực viết CSS. Bạn có thể để CSS ở bất kỳ đâu trong HTML, nhưng nên đặt &lt;style&gt; trong thẻ &lt;head&gt;.

* *Ngoại tuyến (External CSS)*

Trong External CSS, chúng ta sẽ sử dụng phần tử “link” để thêm các

style sheet ở bên ngoài vào trong tài liệu HTML.

Trước hết chúng ta cần tạo các rules (quy tắc) trong một file riêng có phần đuôi mở rộng là .css. Tiếp theo, bạn cần thêm file CSS này vào phần tử head trong tài liệu HTML.

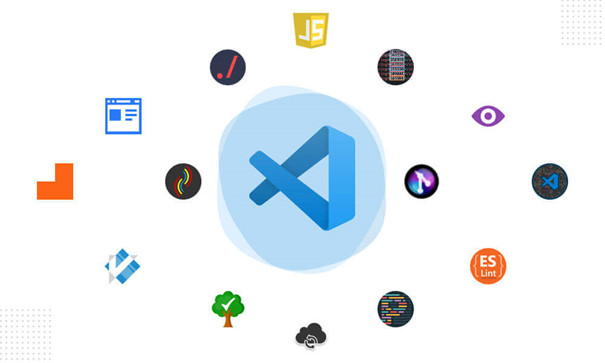
Đây là phương pháp phổ biến nhất để nhúng CSS vào tài liệu HTML. Với cách chèn này, các lập trình viên có thể viết CSS cho nhiều trang web khác nhau và bổ sung cùng một file CSS cho tất cả các trang tương tự.

### 3. Phần mềm Visual Studio Code

- Visual studio là một môi trường phát triển tích hợp (IDE) từ Microsoft, là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Năm 1997, phần mềm lập trình nay có tên mã Project Boston. Nhưng sau đó, Microsoft đã kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

-Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển

ứng dụng, bao gồm  trình chỉnh sửa mã, trình thiết kế, gỡ lỗi.



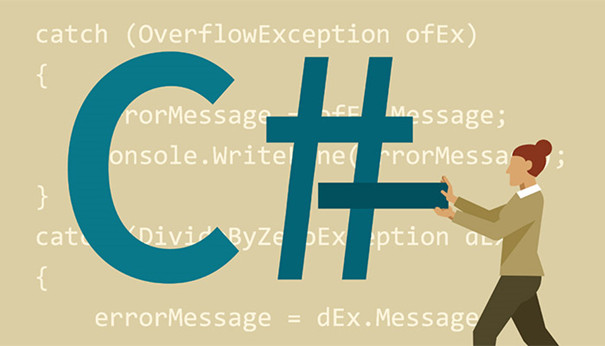
Hình : Phần mềm Visual Studio

Phần mềm Visual studio được chia thành 2 phiên bản Visual Studio Enterprise và Visual Studio Professional, các phiên bản cao cấp có tính phí này được sử dụng nhiều bởi các [công ty chuyên về lập trình](https://mona.software/). Bên cạnh đó, Microsoft cũng cho ra mắt phiên bản Community (phiên bản miễn phí) của gói phần mềm, cung cấp cho người dùng những tính năng cơ bản nhất, phù hợp với các đối tượng lập trình không chuyên, mới tiếp cận tìm hiểu về công nghệ.

* **Giới thiệu các tính năng phần mềm Visual Studio**
* Đa nền tảng

Phần mềm lập trình Visual Studio của Microsoft hỗ trợ sử dụng trên nhiều nền tảng khác nhau. Không giống như các trình viết code khác, Visual Studio sử dụng được trên cả Windows, Linux và Mac Systems. Điều này cực kỳ tiện lợi cho lập trình viên trong quá trình ứng dụng.

* Đa ngôn ngữ lập trình



Hình : Ngôn ngữ lập trình C#

Không chỉ hỗ trợ đa nền tảng, Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic#:~:text=Visual%20Basic%20(vi%E1%BA%BFt%20t%E1%BA%AFt%20VB,h%E1%BB%A3p%20(IDE)%20k%E1%BA%BFt%20b%C3%B3.), JavaScript,… Bởi vậy, Visual Studio có thể dễ dàng phát hiện và thông báo cho bạn khi các chương trình có lỗi.

* Hỗ trợ website

Visual Studio code cũng hỗ trợ website, đặc biệt trong công việc soạn thảo và [thiết kế web](https://mona.solutions/thiet-ke-website).

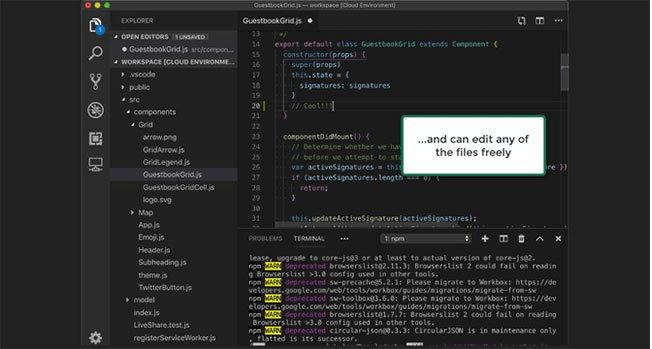
* Kho tiện ích mở rộng phong phú

Mặc dù Visual Studio có hệ thống các ngôn ngữ hỗ trợ lập trình khá đa dạng. Nhưng nếu lập trình viên muốn sử dụng một ngôn ngữ khác, bạn có thể dễ dàng tải xuống các tiện ích mở rộng. Tính năng hấp dẫn này được hoạt động như một phần chương trình độc lập nên không lo làm giảm hiệu năng của phần mềm.

* Lưu trữ phân cấp

Phần lớn các tệp dữ liệu đoạn mã của  Visual Studio đều được đặt trong các thư mục tương tự nhau. Đồng thời, Visual Studio cũng cung cấp một số thư một cho các tệp đặc biệt để bạn lưu trữ an toàn, dễ tìm, dễ sử dụng hơn.

* Kho lưu trữ an toàn

****

Hình : Kho lưu trữ an toàn

Với Visual Studio, bạn có thể hoàn toàn yên tâm về tính lưu trữ, bởi phần mềm đã được kết nối GIT và một số kho lưu trữ an toàn được sử dụng phổ biến hiện nay.

* Màn hình đa nhiệm

Visual Studio sở hữu tính năng màn hình đa nhiệm, cho phép người dùng mở cùng lúc nhiều tập tin, thư mục dù chúng  có thể không liên quan tới nhau.

* Hỗ trợ viết code

Khi sử dụng code vào trong lập trình, với Visual Studio, công cụ này có thể đề xuất tới các lập trình viên một số tùy chọn thay thế nhằm điều chỉnh đôi chút để đoạn code áp dụng thuận tiện hơn cho người dùng.

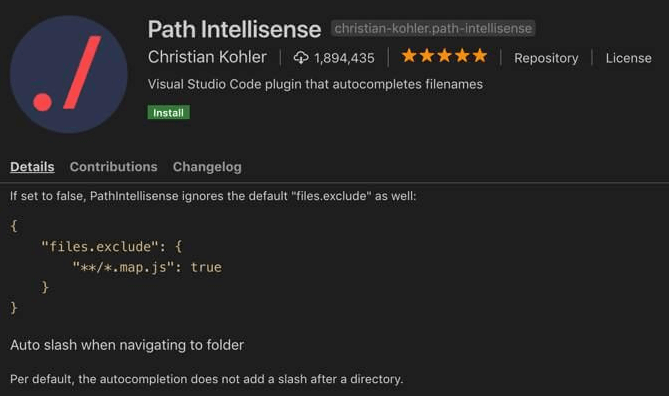
* Hỗ trợ thiết bị đầu cuối

Phần mềm Visual Studio cũng tích hợp các loại thiết bị đầu cuối, giúp người dùng không cần chuyển đổi giữa hai màn hình hay trở về thư mục gốc khi thực hiện một thao tác cần thiết nào đó.

* Hỗ trợ Git

Do kết nối với [GitHub](https://github.com/) nên Visual Studio cho phép hỗ trợ sao chép, kéo thả trực tiếp. Các mã code này sau đó cũng có thể thay đổi và lưu lại trên phần mềm.

* Intellisense

****

Hình : Intellisense

Tính năng nhắc Intellisense được sử dụng hầu hết trong các phần mềm lập trình, bao gồm cả Visual Studio. Tuy nhiên, so với các trình viết mã, Visual Studio vẫn được đánh giá cao  về tính chuyên nghiệp. Đặc biệt, tính năng này còn có thể phát hiện tất cả các đoạn mã không đầy đủ, nhắc lập trình viên, gợi ý sửa đổi, khai báo biến tự động trong trường hợp lập trình viên quên, giúp bổ sung cú pháp  còn thiếu,…

* Tính năng comment

Một tính năng cũng khá hay ho, hỗ trợ cho người lập trình trong trường hợp “nhớ nhớ quên quên” đó là tính năng bình luận. Tính năng này cho phép lập trình viên để lại nhận xét, giúp dễ dàng ghi nhớ công việc cần hoàn thành, không bỏ sót công đoạn nào.



Hình : Visual Studio 2022

* Những **ưu điểm** vượt trội hơn hẳn bất kỳ IDE nào khác như:
* Ít dung lượng
* Tính năng phong phú, đa dạng và mạnh mẽ
* Hỗ trợ nhiều nền tảng khác nhau (Linux, Windows, Mac)
* Hỗ trợ đa ngôn ngữ (HTML, CSS, C#, F#, C/C++, JSON, JavaScript,…)
* Tính năng Intellisense chuyên nghiệp
* Giao diện thân thiện dễ sử dụng
* Cập nhật đa dạng phiên bản cho nhiều đối tượng sử dụng khác nhau,….

## II. Giới thiệu về cơ sở dữ liệu Mysql

 MySQL là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [tự do nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị 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, FreeBSD, Unix, NetBSD, Novell MetWare, 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](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng NodeJs, PHP hay Perl,...

Một số đặc điểm của Mysql :

* Khả năng mở rộng và linh hoạt : Máy chủ cơ sở dữ liệu MySQL đáp ứng nhiều tính năng linh hoạt, nó có sức chứa để xử lý các ứng dụng được nhúng sâu với 1MB dung lượng để chạy kho dữ liệu khổng lồ lên đến hàng terabytes thông tin. Đặc tính đáng chú ý của MySQL là sự linh hoạt về platform với tất cả các phiên bản của Windows, Unix và Linux đang được hỗ trợ. Và đương nhiên, tính chất mã nguồn mở của MySQL cho phép tùy biến theo ý muốn để thêm các yêu cầu phù hợp cho database server.
* Hiệu năng cao: Với kiến trúc storage-engine cho phép các chuyên gia cơ sở dữ liệu cấu hình máy chủ cơ sở dữ liệu MySQL đặc trưng cho các ứng dụng chuyên biệt. Dù ứng dụng là website dung lượng lớn phục vụ hàng triệu người/ngày hay hệ thống xử lý giao dịch tốc độ cao thì MySQL đều đáp ứng được khả năng xử lý khắt khe của mọi hệ thống. Với những tiện ích tải tốc độ cao, cơ chế xử lý nâng cao khác và đặc biệt bộ nhớ cache, MySQL đưa ra tất cả nhưng tính năng cần có cho hệ thống doanh nghiệp khó tính hiện nay.
* Tính sẵn sàng cao: MySQL đảm bảo sự tin cậy và có thể sử dụng ngay. MySQL đưa ra nhiều tùy chọn có thể “mì ăn liền” ngay từ cấu hình tái tạo chủ/tớ tốc độ cao, để các nhà phân phối thứ 3 có thể đưa ra những điều hướng có thể dùng ngay duy nhất cho server cơ sở dữ liệu MySQL.
* Hỗ trợ giao dịch mạnh mẽ: MySQL đưa ra một trong số những engine giao dịch cơ sở dữ liệu tốt nhất trên thị trường. Các đặc trưng bao gồm, khóa mức dòng không hạn chế, hỗ trợ [giao dịch ACID](https://vi.wikipedia.org/wiki/ACID) hoàn thiện, khả năng giao dịch được phân loại và hỗ trợ giao dịch đa dạng (multi-version) mà người đọc không bao giờ cản trở cho người viết và ngược lại. Dữ liệu được đảm bảo trong suốt quá trình server có hiệu lực, các mức giao dịch độc lập được chuyên môn hóa, khi phát hiện có lỗi khóa chết ngay tức thì.
* Bảo vệ dữ liệu mạnh mẽ: Việc quan trọng của các doanh nghiệp là bảo mật dữ liệu, MySQL tích hợp các tính năng bảo mật an toàn tuyệt đối. Với việc xác nhận truy cập cơ sở dữ liệu, MySQL trang bị các kỹ thuật mạnh, chỉ có người sử dụng đã được xác nhận mới truy cập được vào cơ sở dữ liệu. SSH và SSL cũng được hỗ trợ để đảm bảo kết nối an toàn và bảo mật. Tiện ích backup và recovery cung cấp bởi MySQL và các hãng phần mềm thứ 3 cho phép backup logic và vật lý cũng như recovery toàn bộ hoặc tại một thời điểm.
* Phát triển ứng dụng toàn diện: MySQL trở thành cơ sở dữ liệu mã nguồn mở phổ biến nhất hiện nay một phần là do cung cấp hỗ trợ hỗn hợp cho bất cứ sự phát triển ứng dụng nào cần. Trong [cơ sở dữ liệu](http://sqladvice.com/co-du-lieu-la-gi/), hỗ trợ có thể được tìm thấy trong các trigger, stored procedure, cursor, view, ANSI-standard SQL,… MySQL cũng cung cấp các bộ kết nối như: JDBC, [ODBC](https://vi.wikipedia.org/wiki/ODBC),… để tất cả các form của ứng dụng sử dụng MySQL như một server quản lý dữ liệu được đề xuất hàng đầu.
* Quản lý dễ dàng: Cài đặt MySQL khá nhanh và trung bình từ khi download phần mềm tới khi cài đặt thành công chỉ mất chưa đầy 15 phút. Cho dù platform là Linux, Microsoft Windows, Macintosh hoặc Unix thì quá trình cũng diễn ra nhanh chóng. Khi đã cài đặt, tính năng quản lý như tự khởi động lại, tự động mở rộng không gian và cấu hình động sẵn sàng cho người quản trị cơ sở dữ liệu bắt đầu làm việc. MySQL cung cấp toàn bộ công cụ quản lý đồ họa cho phép một DBA quản lý, sửa chữa và điều khiển hoạt động của nhiều server MySQL từ một máy trạm đơn.
* Mã nguồn mở và tự do hỗ trợ 24/7: Nhiều doanh nghiệp còn băn khoăn trong việc giao toàn bộ cho phần mềm mã nguồn mở bởi khó có thể tìm được hỗ trợ hay bảo mật an toàn phục vụ chuyên nghiệp. Với MySQL mọi sự cam kết đều rõ ràng, MySQL cam kết bồi thường khi gặp sự cố.
* Chi phí sở hữu thấp: Sử dụng MySQL cho các dự án, doanh nghiệp đều nhận thấy sự tiết kiệm chi phí đáng kể. Người dùng MySQL cũng không mất nhiều thời gian để sửa chữa hoặc vấn đề thời gian chết.

# CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Khảo sát hệ thống

### Khảo sát sơ bộ

* Tổ chức:
* Admin:
  + Quản lý hoạt động của trang web, quản lý 1 số thông tin cơ bản của trang web.
  + Quản lý thông tin địa điểm,thông tin giá cả đi lại, khách sạn mới tương ứng có thể thêm sửa xóa các thông tin này trên trang web.
  + Quản lý thông tin, tài khoản khách hàng tương ứng có thể tìm kiếm, thêm, sửa, xóa tài khoản khách hàng.
  + Quản lý thông tin về địa điểm du lịch, cập nhật giá cả, thông tin mô tả về địa điểm du lịch.
* Khách hàng:
  + Có thể truy cập vào trang web, xem thông tin trang web, tìm kiếm địa điểm du lịch, tạo tài khoản, đăng nhập để lựa chọn đặt vé du lịch, liên hệ hỗ trợ, cũng như đánh giá và nhận xét.

### Tài liệu đặc tả người dùng

#### Khảo sát chi tiết

Các hoạt động của hệ thống:

* *Quản lý thông tin địa điểm:*

+ Admin cần đăng nhập vào hệ thống

+ Tìm kiếm địa điểm theo tên. Với những địa điểm cần thay đổi thông tin, admin click vào biểu tượng sửa bên cạnh mỗi địa điểm để thay đổi một số thông tin địa điểm như tên, miêu tả,…

* *Quản lý thông tin người dùng:*

+ Admin cần đăng nhập vào hệ thống

+ Admin có thể tìm kiếm theo tên, theo ngày tạo và cũng có thể lọc dữ liệu theo quyền của người dùng. Từ đó admin có thể sửa thông tin tài khoản khách hàng hay xóa tài khoản.

* *Đăng ký tài khoản:*

+ Người dùng cũng có thể tự tạo tài khoản cho mình để truy cập trang web nhưng chỉ với quyền là user.

+ Click vào nút “Register” để hệ thống hiển thị màn hình tạo tài khoản rồi điền thông tin và click nút “Sign up” để xác nhận tạo tài khoản.

* *Đăng nhập:*

+ Để đăng nhập trước tiên người dùng cần có một tài khoản của trang web. Khi truy cập vào trang web, người dùng click vào nút “Login” hay khi người dùng truy cập vào 1 trang không thuộc quyền, hệ thống sẽ hiển thị trang đăng nhập yêu cầu người dùng đăng nhập.

+Nhập thông tin và click “Login”. Hệ thống sẽ hiển thị màn hình trang chủ theo quyền truy cập của tài khoản.

* *Tìm kiếm địa điểm:*

+ Khách hàng có thể tìm kiếm thông tin địa điểm bằng cách truy cập vào trang web. Ở trang chủ hiển thị 1 vài địa điểm giới thiệu, hay 1 số địa điểm được xem nhiều nhất.

+ Khách hàng cũng có thể truy cập trực tiếp vào “About us” để xem nhiều sản phẩm hơn, tìm kiếm theo tên.

* *Đặt chỗ trực tuyền:*

+ Khách hàng cần đăng nhập vào hệ thống.

+ Chọn vào mục “Offers” để xem thông tin các địa điểm và các khách sạn.

+ Sau khi đã chọn được địa điểm và khách sạn ưng ý sẽ click vào “Book” để đặt chỗ.

* *Đánh giá và nhận xét:*

+ Khách hàng cần đăng nhập vào hệ thống.

+ Kéo xuống ở trang chủ xem đánh giá các địa điểm du lịch hoặc chọn vào mục “About us” tìm các địa điểm click “See reviews” để xem và đánh giá địa điểm đó.

* *Liên hệ hỗ trợ:*

+ Khách hàng cần đăng nhập vào hệ thống.

+ Chọn các biểu tượng icon hoặc chọn vào mục “Contact” sau đó điền các thông tin và nội dung cần gửi đến admin.

* *Đọc tin tức:*

+ Khách hàng mở trang web và chọn mục “News” để xem và đọc các tin tức nổi bật được website cập nhật thường xuyên.

#### Yêu cầu chức năng:

* Khách hàng có thể vào xem thông tin website, tìm kiếm nhanh các địa điểm du lịch theo từ khóa, địa điểm, giá cả,..cho phép người dùng đăng nhập và đăng ký tài khoản, đặt chỗ trực tuyến, đánh giá và nhân xét về các địa điểm du lịch.
* Admin có thể đăng nhập , quản lý thông tin tài khoản của khách hàng, quản lý một thông tin của trang web, hỗ trợ người dùng giải quyết các vấn đề kỹ thuật hoặc tài khoản của họ.

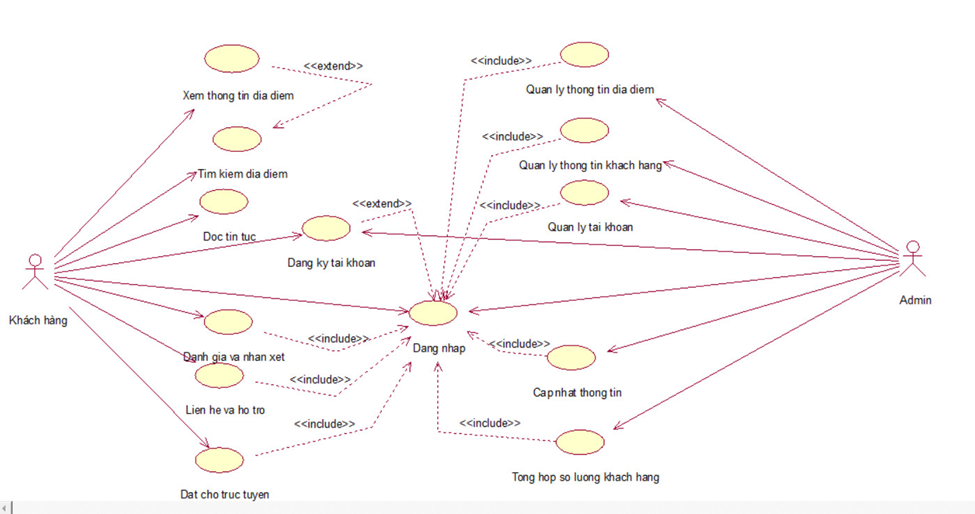
#### Yêu cầu phi chức năng:

* Trang web hỗ trợ cứu hộ hay trang quản trị hoạt động 24/24h, giao diện(UI) dễ nhìn có thiện cảm với người dùng ngoài ra các vấn đề thao tác của người dùng cần đơn giản dễ thao tác tránh cảm giác gây ức chế khi người dùng thao tác.
* Ổn định, xử lý nhanh, Tốc độ tải trang hay thực hiện công việc không được quá lâu, hiện các thông báo khi kết thúc một thao tác công việc nào đó.
* Trang web có thể tương tích với các trình duyệt khác nhau các thiết bị truy cập khác nhau.
* An toàn, bảo mật: Đảm bảo an toàn thông tin người dùng. Các thao tác nghiệp vụ của người quản trị chỉ thực hiện được khi có quyền.
* Độ tin cậy: Yêu cầu website cần cung cấp thông tin chính xác và đáng tin cậy về các địa điểm du lịch, giúp người dùng đưa ra quyết định đúng đắn khi lựa chọn địa điểm.
* Đảm bảo hệ thống website hoạt động bình thường khi có nhiều người cùng truy cập và thao tác trên đó.
* Sử dụng được nhiều ngôn ngữ.
* Ngăn chặn những trang quảng cáo không tốt xuất hiện trên trang web.

## Phân tích hệ thống

### Biểu đồ use case

* Actor, mô tả actor:
* Khách hàng: là người có thể vào website, xem thông tin địa điểm, đăng nhập, đăng ký tạo tài khoản, tìm kiếm địa điểm, đánh giá và nhận xét, liên hệ và hỗ trợ, đặt chỗ trực tuyến, đọc tin tức.
* Admin: là người có thể vào website, quản lý thông tin địa điểm, quản lý thông tin khách hàng, cập nhật thông tin, quản lý tải khoản, tổng hợp số lượng khách hàng đặt chỗ..
* CSDL
* Các use case:
* Khách hàng: xem thông tin địa điểm, đăng nhập, đăng ký tạo tài khoản, tìm kiếm địa điểm, đánh giá và nhận xét, liên hệ và hỗ trợ, đặt chỗ trực tuyến, đọc tin tức.
* Admin: quản lý thông tin địa điểm, quản lý thông tin khách hàng, cập nhật thông tin, quản lý tải khoản, tổng hợp số lượng khách hàng.



Hình : Sơ đồ usecase

### Mô tả chi tiết các use case

#### Mô tả use case Đăng ký tài khoản

* **Tên Use Case:**

Đăng ký tài khoản

* **Mô tả vắn tắt:**

Use case này cho phép khách hàng đăng ký tài khoản hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi khách hàng nhấn vào nút đăng ký trên thanh menu. Hệ thống hiển thị màn hình đăng ký và hiển thị yêu cầu nhập thông tin cá nhân.
2. Khách hàng nhập thông tin đăng ký bao gồm họ, tên, email, mật khẩu sau đó kích vào nút “đăng ký” hệ thống sẽ cập nhật thông tin tài khoản trong bảng TAIKHOAN. Hệ thống hiển thị thông tin tài khoản, đăng ký thành công và use case kết thúc.

* **Luồng rẽ nhánh:**

1. Sai Tên/Mật Khẩu : Tại bước 2 trong luồng cơ bản nếu khách hàng nhập một tên hay mật khẩu, hệ thống sẽ hiển thị một thông báo lỗi tên người dùng đã tồn tại hoặc mật khẩu không hợp lệ. Khách hàng có thể chọn quay về luồng cơ bản để nhập lại, hoặc bỏ qua thao tác khi đó use case kết thúc.
2. Bỏ qua : Tại bước 2 trong luồng cơ bản nếu khách hàng kich vào nút “bỏ qua” thì use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Không có.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

#### Mô tả use case Đăng nhập

* Tên Use Case:

Đăng nhập

* Mô tả vắn tắt:

Use case này cho phép người dùng (khách hàng & admin) đăng nhập vào hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng muốn đăng nhập vào hệ thống. Hệ thống sẽ hiển thị màn hình yêu cầu nhập tên đăng nhập và mật khẩu.
2. Người dùng nhập tên và mật khẩu sau đó kích vào nút “Đăng nhập”**.** Hệ thống kiểm tra tên đăng nhập, mật khẩu và quyền truy nhập trong bảng TAIKHOANvà hiển thị màn hình hệ thống với các menu tương ứng với quyền. Use case kết thúc.
   * **Luồng rẽ nhánh:**
3. Tại bước 2 trong luồng cơ bản nếu người dùng nhập sai tên hoặc/và mật khẩu thì hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại. Người dùng nhập lại để tiếp tục, hoặc kích vào nút “Thoát” và use case kết thúc.
4. Tại bước 2 trong luồng cơ bản nếu hệ thống không thể kết nối được với cơ sở dữ liệu thì sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không được nhập sai quá 5 lần.

* **Tiền điều kiện:**

Không có.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

#### Mô tả use case Tìm kiếm địa điểm

* **Tên Use case:**

Tìm kiếm địa điểm

* **Mô tả vắn tắt:**

Use case này cho phép khách hàng tìm kiếm địa điểm mong muốn.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng nhập tên địa điểm muốn tra cứu vào ô tìm kiếm. Hệ thống lấy thông tin từ bảng DIADIEM và hiển thị: ảnh minh họa, tiêu đề, giới thiệu,..
2. Khách hàng kích vào ảnh minh họa một địa điểm trong danh sách. Hệ thống lấy thông tin chi tiết của địa điểm được chọn gồm: ảnh minh họa, tiếu đề, tên địa điểm, thông tin địa điểm,.. từ bảng DIADIEM và hiển thị ra màn hình. Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại thời điểm bất kì trong quá trình thực hiện use case nếu không kết nối được với hệ thống cơ sở dữ liệu thì hệ thống sê hiển thị một thông báo lỗi và use case kết thúc.
2. Tại bước 1 trong luồng cơ bản khi hệ thống không tìm thấy tên địa điểm nào thuộc địa điểm khách hàng muốn tìm kiếm, thì sẽ hiển thị một thông báo. Use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có

* **Tiền điều kiện:**

Không có

* **Hậu điều kiện:**

Không có

* **Điểm mở rộng:**

Không có

#### Mô tả use case Đặt chỗ trực tuyến

* **Tên Use Case:**

Đặt chỗ trực tuyến

* **Mô tả vắn tắt:**

Use case này cho phép khách hàng đặt phòng, chỗ ở có trong phần offers

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi khách hàng bấm vào offers. Hệ thống lấy thông tin danh sách của các phòng ở bao gồm: gồm mã phòng, tên phòng, địa chỉ, số lượng,... trong bảng PHONG và hiển thị lên màn hình.
2. Khách hàng kích vào nút “Book” của một phòng. Hệ thống sẽ lấy thông tin gồm: ảnh , tên phòng, số lượng, địa chỉ, đánh giá,... và hiển thị ra màn hình.
3. Khách hàng kích vào nút “Book”. Hệ thống sẽ xác nhận và đặt chỗ. Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 1 trong luồng cơ bản: Nếu cơ sở dữ liệu chưa được nhập thì hệ thống sẽ hiển thị thông báo . Use case kết thúc.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có

* **Tiền điều kiện:**

Người dùng phải đăng nhập vào hệ thống trước khi đặt chỗ trực tuyến.

* **Hậu điều kiện:**

**Không có**

* **Điểm mở rộng:**

Không có.

#### Mô tả use case Quản lý tài khoản

* **Tên Use Case:**

Quản lý tài khoản.

* **Mô tả vắn tắt:**

Use Case này cho phép người quản trị xem, sửa trạng thái, xóa các thông tin trong bảng TAIKHOAN.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người quản trị kích vào “Tài khoản” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các tài khoản gồm: mã người dùng, tên tài khoản đăng nhập, mật khẩu, quyền quản trị trong cơ sở dữ liệu và hiển thị các lên màn hình.
2. Sửa thông tin tài khoản:
3. Người quản trị kích vào nút “Sửa” trên một dòng chứa thông tin tài khoản. Hệ thống sẽ lấy thông tin cũ của tài khoản được chọn gồm: mã người dùng, tên tài khoản đăng nhập, mật khẩu, quyền quản trị từ bảng TAIKHOAN và hiển thị lên màn hình.
4. Người quản trị nhập thông tin mới cho mã người dùng, tên tài khoản đăng nhập, mật khẩu, quyền quản trị và kích vào nút “Cập nhật”. Hệ thống sẽ sửa thông tin của tài khoản được chọn trong bảng TAIKHOAN và hiển thị danh sách tài khoản sau khi cập nhật.
5. Xóa tài khoản:
6. Người quản trị kích vào nút “Xóa” trên một dòng chứa tài khoản. Hệ thống sẽ đưa ra thông báo “Bạn có chắc chắn muốn xóa tài khoản này?” lên màn hình.
7. Người quản trị kích vào nút “Đồng ý” trên thông báo hệ thống đã đưa ra. Hệ thống sẽ xóa tài khoản được chọn khỏi bảng TAIKHOAN và hiển thị danh sách các tài khoản sau khi cập nhật.

Use case kết thúc.

* **Luồng rẽ nhánh:**
* Tại bước 2b hoặc 3b trong luồng cơ bản nếu người quản trị kích vào nút “Hủy bỏ” hệ thống sẽ bỏ qua thao tác tương ứng và hiển thị danh sách các tài khoản trong bảng TAIKHOAN.
* Tại bước 2b nếu người quản trị nhập các thông tin không hợp lệ **hoặc sai định dạng, hệ thống sẽ đưa ra 1 thông báo lỗi yêu cầu nhập** lại. Người quản trị có thể nhập lại để tiếp tục hoặc nhấn “Hủy bỏ” để kết thúc.
* Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.
* **Các yêu cầu đặc biệt:**

Use case này chỉ cho phép một số vai trò như người quản trị thực hiện.

* **Tiền điều kiện:**

Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi thực hiện use case này.

* **Hậu điều kiện:**

Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.

* **Điểm mở rộng:**

Không có.

#### Mô tả use case Tổng hợp số lượng khách hàng

* **Tên use case:**

Tổng hợp số lượng khách hàng.

* **Mô tả vắn tắt:**

Use case này cho phép người quản trị thống kê kết quả số lượng khách hàng, hằng ngày, tháng, năm.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Khi người quản trị nhấn chọn “Tổng hợp” hệ thống sẽ lấy dữ liệu từ hệ cơ sở dữ liệu hiện ra các báo cáo tổng hợp số lượng khách hàng,..
2. Người quản trị chọn tổng hợp, báo cáo cần thiết rồi nhấn xuất file, hệ thống sẽ xuất các báo cáo ra file Excel.

* **Luồng rẽ nhánh:**

1. Tại bước 2 luồng cơ bản nếu người quản trị nhấn “Huỷ” thì yêu cầu tổng hợp bị huỷ bỏ và use case kết thúc.
2. Tại bất kì thời điểm nào trong luồng cơ bản, nếu không kết nối được với CSDL thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng phải đăng nhập tài khoản với vai trò người quản trị.

* **Hậu điều kiện:**

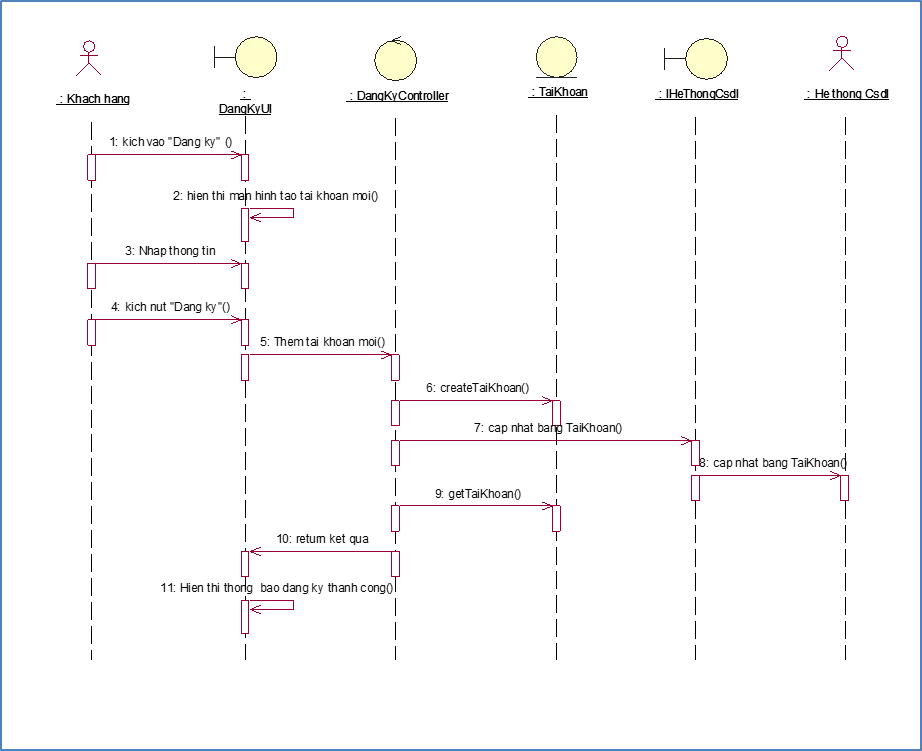
Không có.

* **Điểm mở rộng:**

Không có.

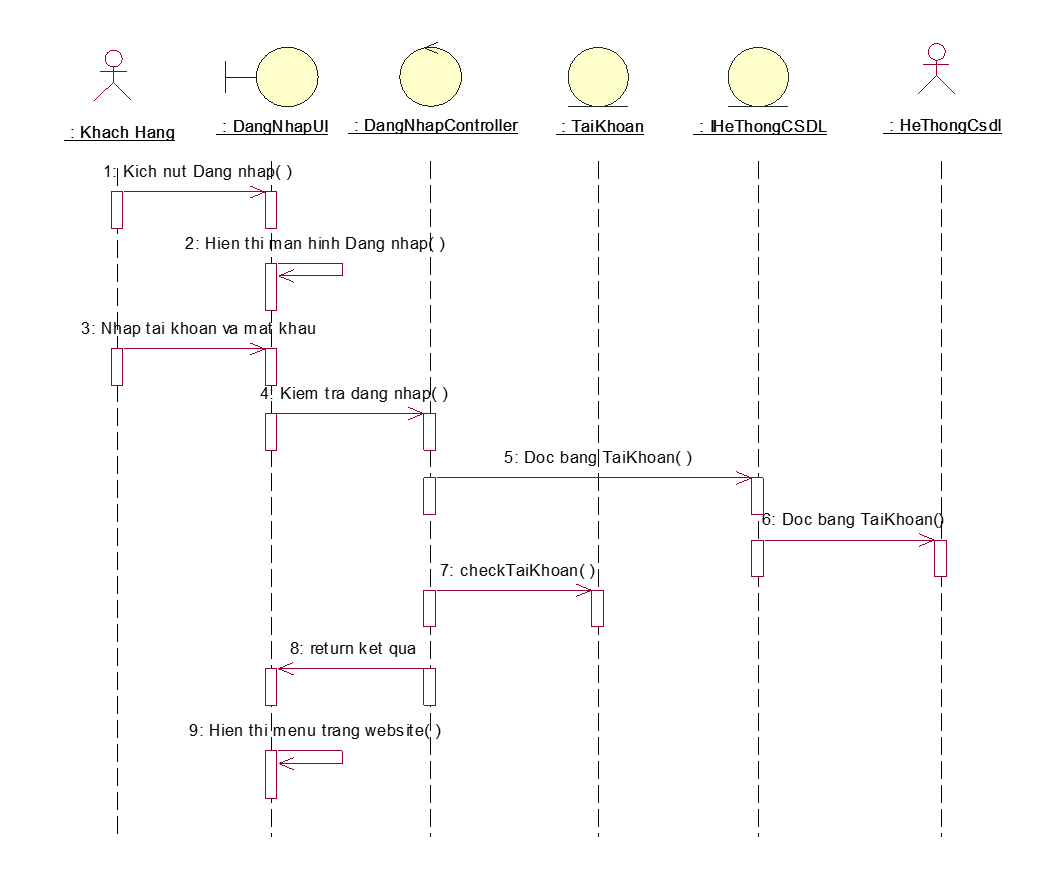
### Biểu đồ trình tự

#### Use case Đăng ký tài khoản



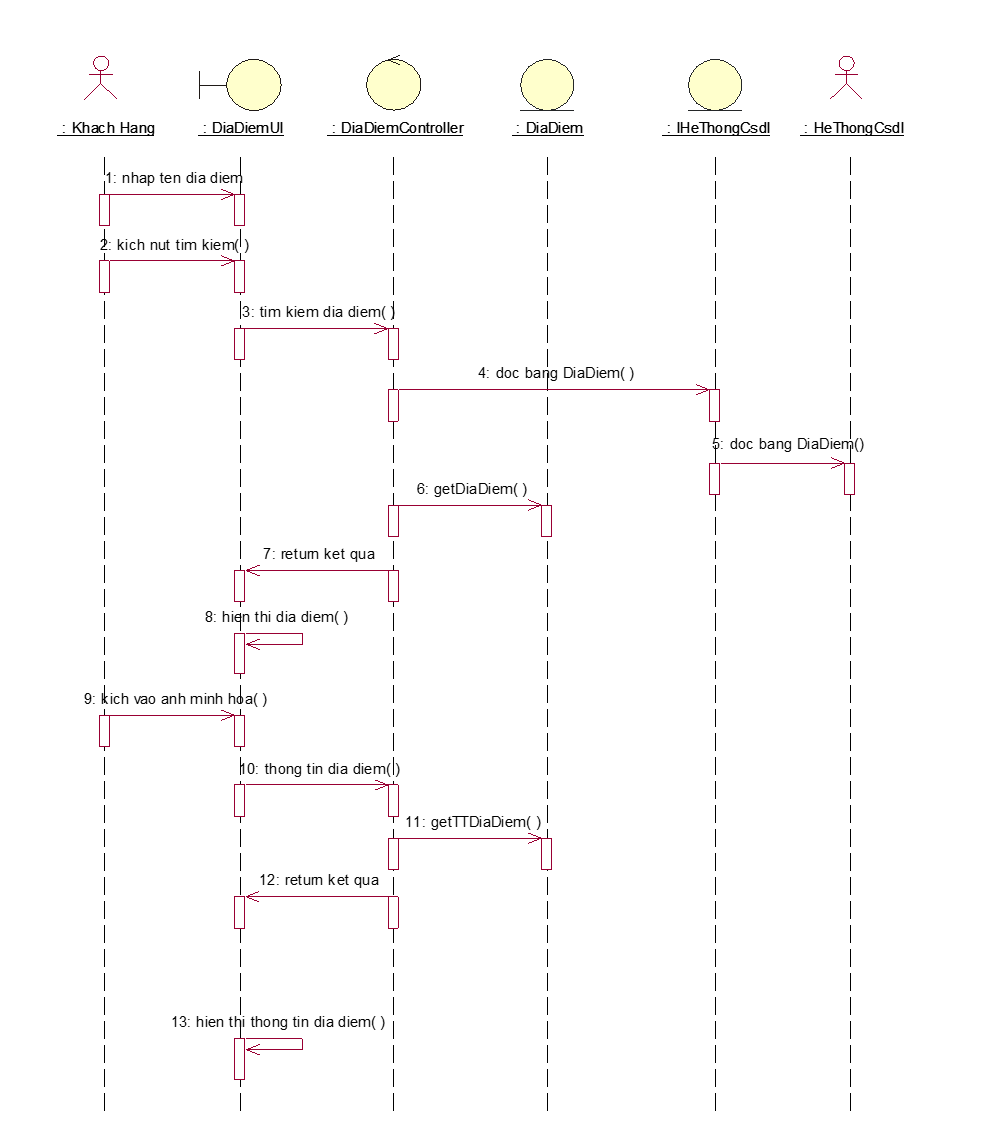
Hình : Use case Đăng kí tài khoản

#### Use case Đăng nhập



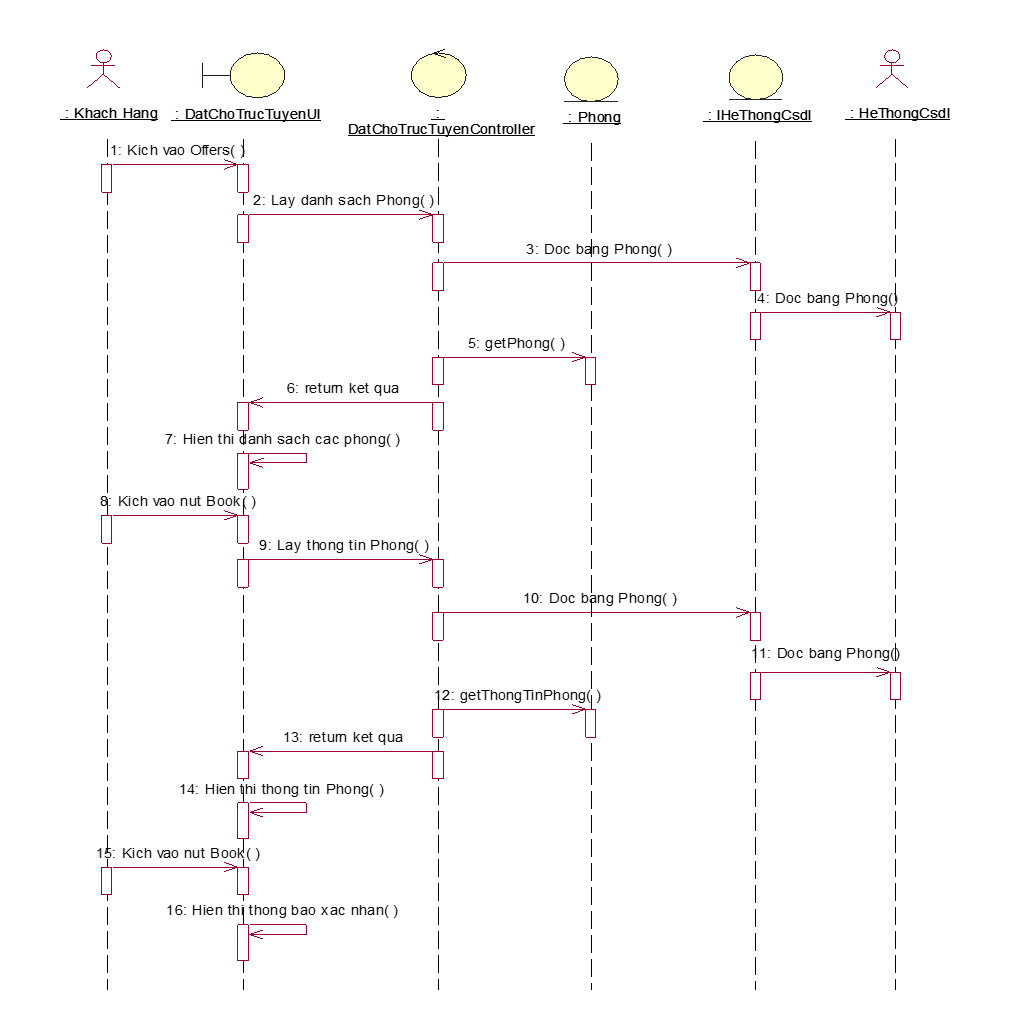
Hình : Use case Đăng nhập

#### Use case Tìm kiếm địa điểm



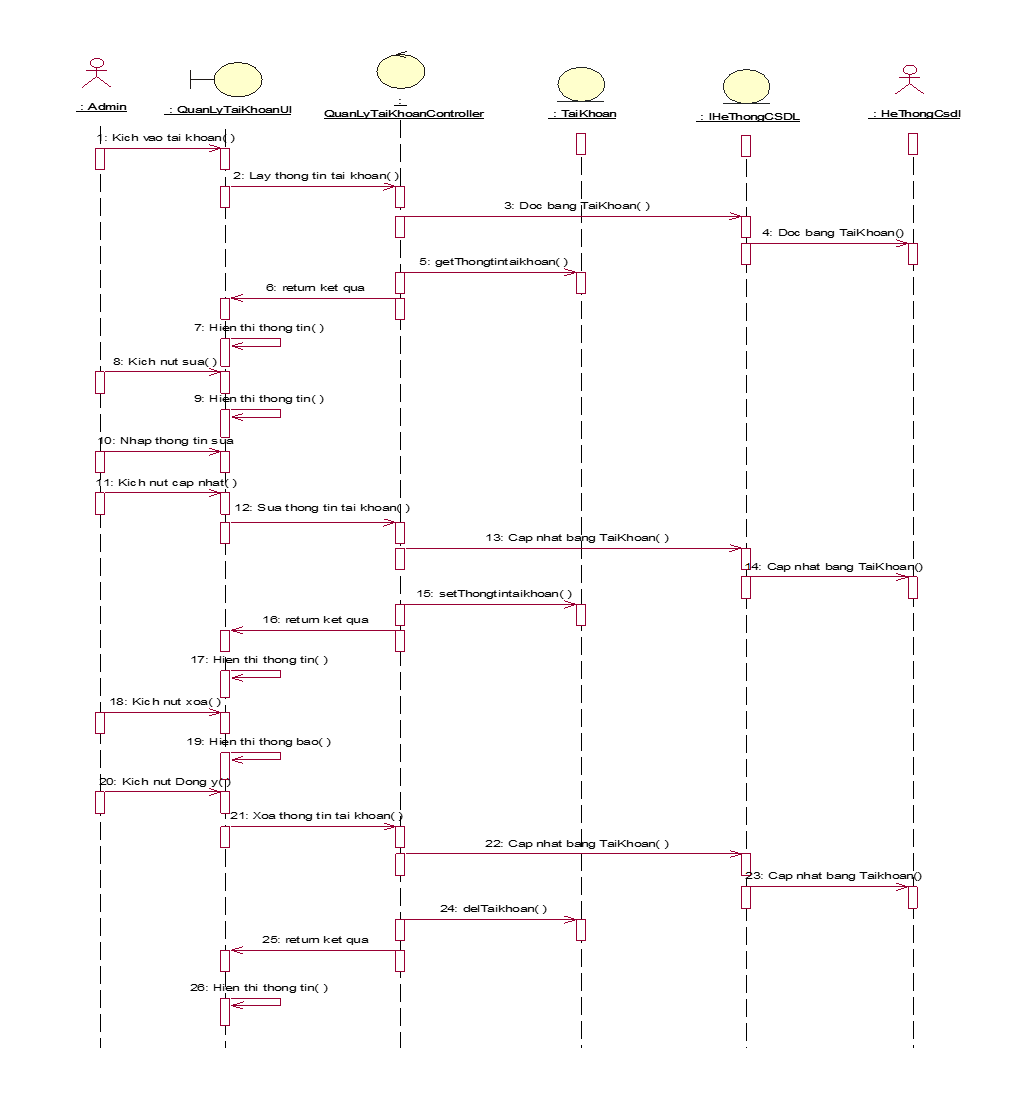
Hình : Use case Tìm kiếm địa điểm

#### Use case Đặt chỗ trực tuyến



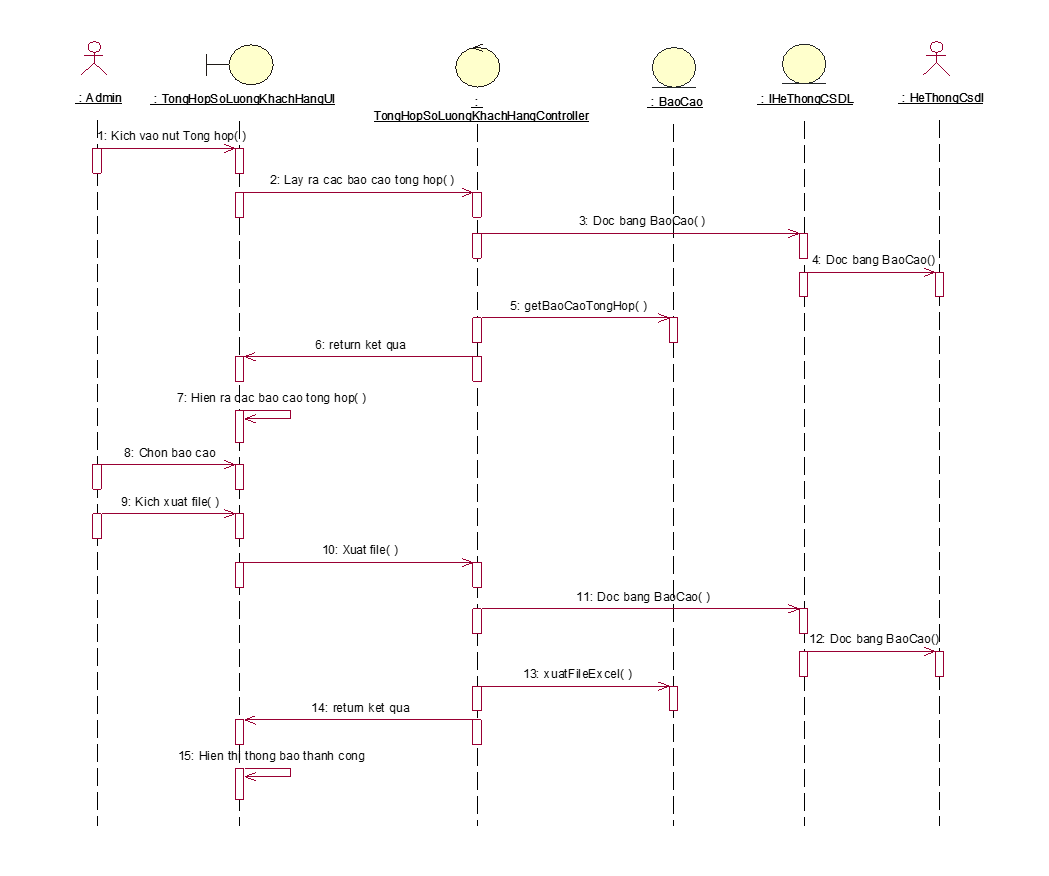
Hình : Use case Đặt chỗ trực tuyến

#### Use case Quản lý tài khoản



Hình : Use case Quản lý tài khoản

#### Use case Tổng hợp số lượng khách hàng



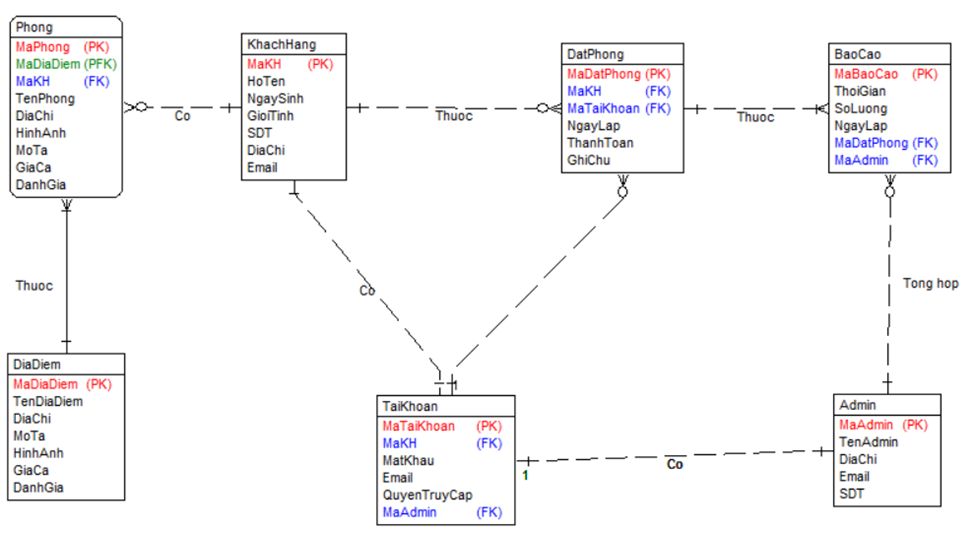
Hình : Use case Tổng hợp số lượng khách hàng

### Mô hình hoá dữ liệu của hệ thống

#### Các yêu cầu về dữ liệu

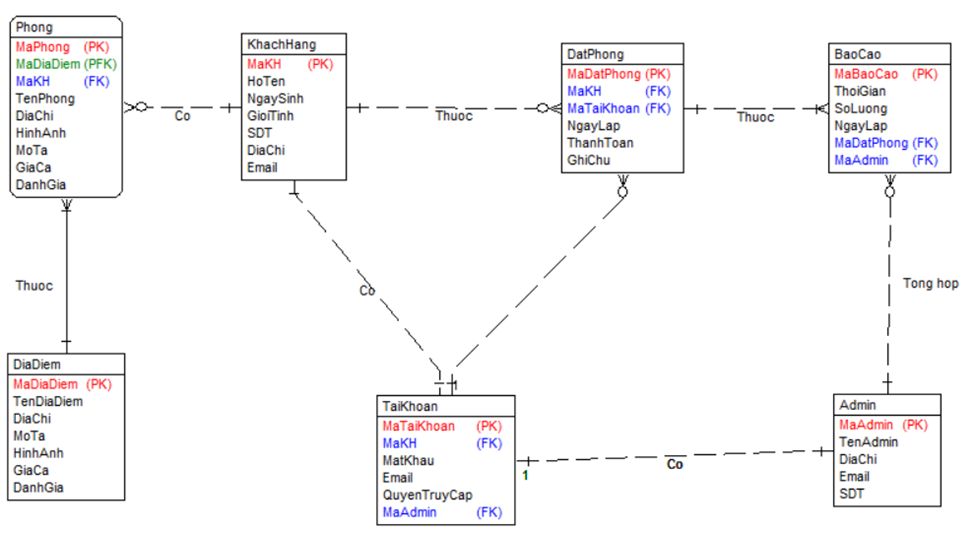
* Từ nhu cầu thực tế của người dùng để chuyển hóa thành dữ liệu
* **Tài khoản:** gồm mã tài khoản, mã khách hàng, mã admin, mật khẩu, email và quyền truy cập. Mỗi khách hàng có một tài khoản. Mỗi tài khoản phải thuộc về 1 khách hàng. Ngoài ra, mỗi admin có một tài khoản.
* **Khách hàng:** gồm mã khách hành, họ tên, ngày sinh, giới tính, số điện thoại, địa chỉ, email. . Mỗi khách hàng có thể có 0,1 hoặc nhiều đơn đặt phòng. Mỗi đơn đặt phòng thuộc về một và chỉ một khách hàng. Mỗi khách hàng có 0,1 hoặc nhiều phòng và 1 phòng chỉ thuộc 1 khách hàng.
* **Phòng:** gồm mã phòng, mã địa điểm, mã khách hàng, tên phòng, ngày lập, địa chỉ, mô tả, hình ảnh, giá cả, đánh giá. Mỗi phòng sẽ thuộc về 1 địa điểm và mỗi địa điểm có 1 hoặc nhiều phòng.
* **Địa điểm:** các thông tin địa điểm được xác định gồm mã địa điểm, tên địa điểm, địa chỉ, mô tả, hình ảnh, giá cả, đánh giá.
* **Đặt phòng:** gồm mã đặt phòng, mã khách hang, mã tài khoản, ngày lập, thanh toán, ghi chú. Mỗi đơn đặt phòng sẽ thuộc về 1 hoặc nhiều báo cáo. Mỗi báo cáo có 1 và chỉ 1 đơn đặt phòng.
* **Admin:** bao gồm mã admin, tên admin, địa chỉ, email, số điện thoại. Mỗi admin sẽ tổng hợp 0,1 hoặc nhiều báo cáo. Mỗi báo cáo chỉ thuộc về một admin.
* **Báo cáo:** gồm mã báo cáo, mã đặt phòng, mã admin, thời gian, số lượng.

#### Biểu đồ thực thể liên kết mức logic



Hình : Biểu đồ thực thể liên kết mức logic

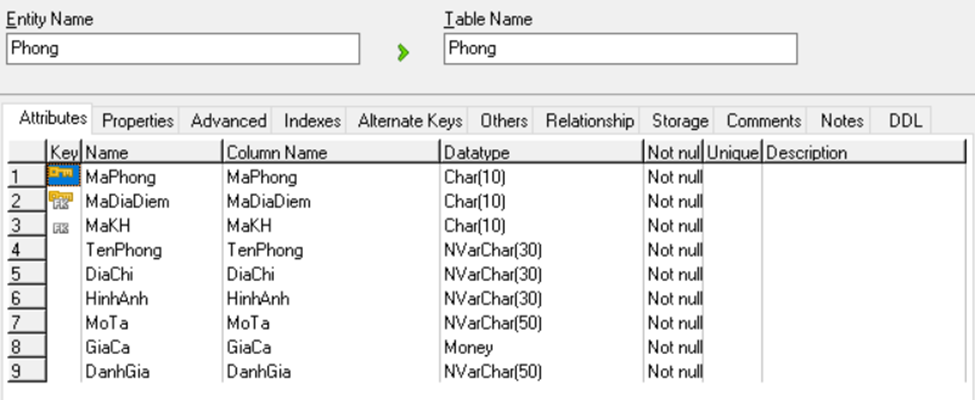
#### Biểu đồ thực thể liên kết mức vật lý



Hình : Biểu đồ thực thể mức vật lý

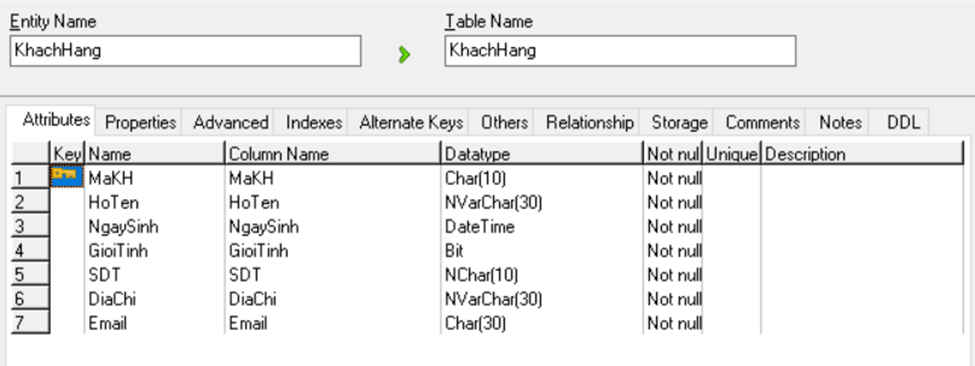
### Thiết kế cơ sở dữ liệu

#### Bảng phòng



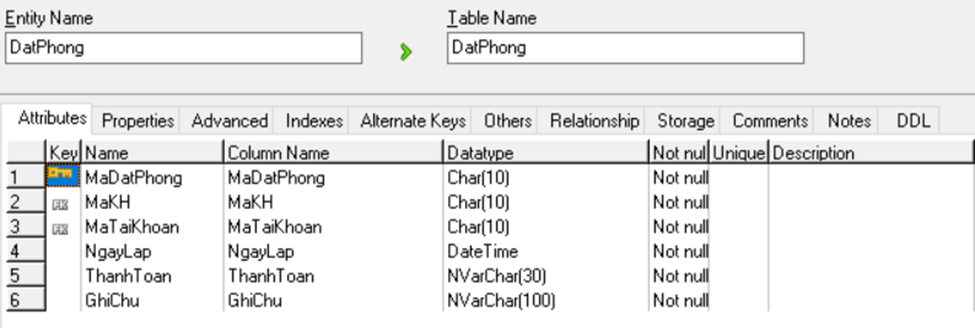
Hình : Bảng phòng

#### Bảng Khách hàng



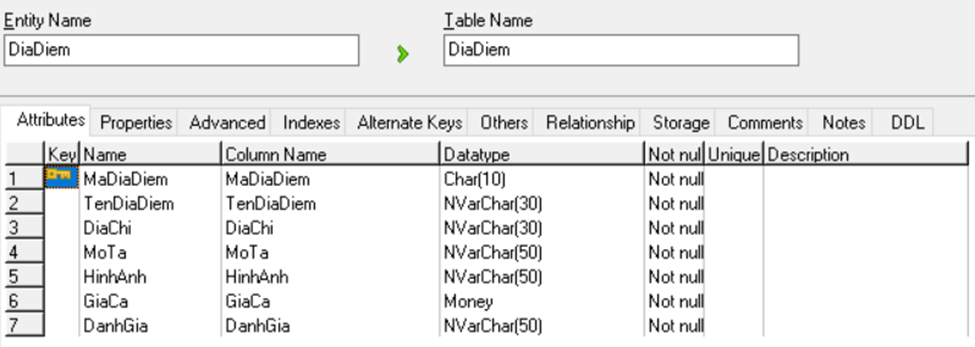
Hình : Bảng khách hàng

#### Bảng Đặt phòng



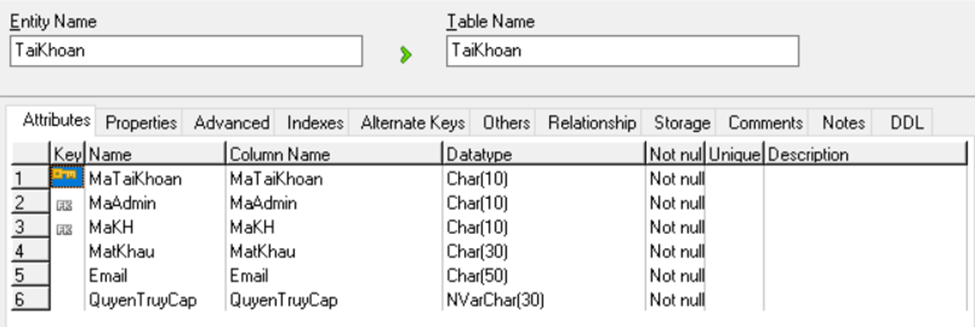
Hình : Bảng Đặt phòng

#### Bảng Địa điểm



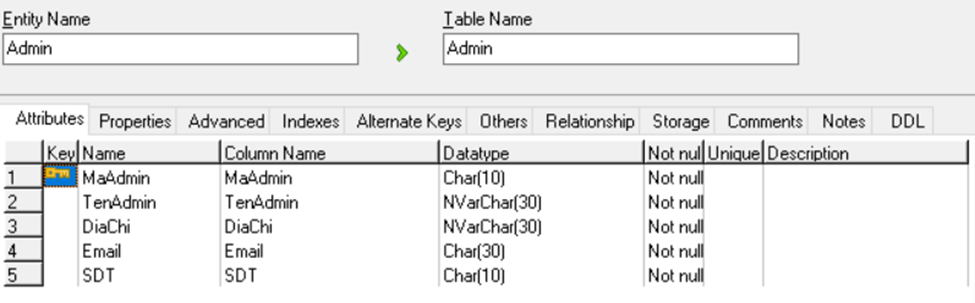
Hình : Bảng Địa điểm

#### Bảng Tài khoản



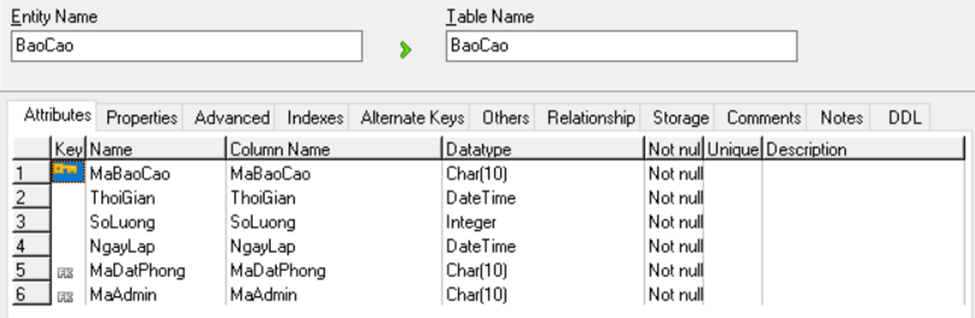
Hình : Bảng Tài khoản

#### Bảng Admin



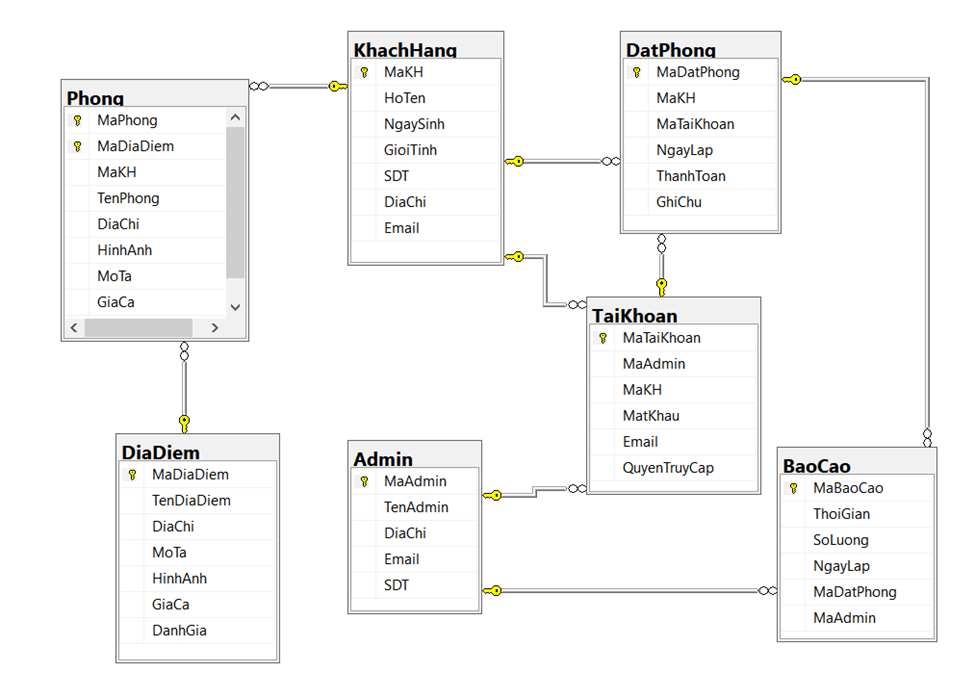
Hình : Bảng Admin

#### Bảng báo cáo



Hình : Bảng Báo cáo

#### Mô hình cơ sở dữ liệu mức vật lý



Hình : Mô hình cơ sở dữ liệu mức vật lý