**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN ĐIỆN TỬ - VIỄN THÔNG**

~~~~~ 🙟🕮🙝 ~~~~~



**BÁO CÁO ĐỒ ÁN I**

**ĐỀ TÀI:**

**THIẾT KẾ WEBSITE QUẢN LÍ KHÁCH SẠN**

|  |  |  |
| --- | --- | --- |
| ***Giảng viên hướng dẫn*** | ***:*** | ***PGS.TS Nguyễn Thúy Anh*** |
| ***Sinh viên thực hiện*** | ***:*** | ***Nguyễn Anh Tú*** |
| ***Mã số sinh viên*** | ***:*** | ***20164468*** |
| ***Lớp*** | ***:*** | ***Điện tử 07 – K61*** |

**Hà Nội, tháng 6 năm 2019**

**Đánh giá quyển báo cáo đồ án**

**(Dùng cho giảng viên hướng dẫn)**

Giảng viên đánh giá : ………………………………………………………………...

Sinh viên thực hiện : ...………………………………………………………………

Tên đề tài : ………………………………………………………………...

………………………………………………………………………………………...

***Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:***

***Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)***

***Nhận xét thêm của Thầy/Cô*** *(giảng viên hướng dẫn nhận xét về thái độ và tinh thần làm việc của sinh viên)*

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

Ngày: / /2019

Người nhận xét

(Ký và ghi rõ họ tên)

# LỜI NÓI ĐẦU

Ngày nay với sự phát triển mạnh mẽ của khoa học kỹ thuật từ lý thuyết đến ứng dụng, người ta đang cố gắng đưa công nghệ thông tin vào tất cả mọi mặt của đời sống, trong đó lĩnh vực quản lí đặt phòng. **Trong nội dung đồ án này,** **em thực hiện thiết kế một website quản lí khách sạn**. Trang web này có ý nghĩa quản lí hay giới thiệu khách sạn cụ thể là dịch vụ , phòng ăn , nhà nghỉ, danh lam thắng cảnh,…

Trong thời gian làm đồ án, em đã gặp không ít khó khăn nhưng được sự giúp đỡ hướng dẫn tận tình của các thầy cô giáo, các bạn sinh viên Viện Điện Tử - Viễn Thông cho nên em đã hoàn thành được đề tài này.

Em xin chân thành cảm ơn giảng viên Nguyễn Thúy Anh – người trực tiếp hướng dẫn, đã tận tình hướng dẫn và giúp đỡ em hoàn thành đề tài cho đồ án này.

Em xin chân thành cảm ơn!

# BẢNG PHÂN CÔNG CÔNG VIỆC

**Bảng A. Phân công công việc qua các tuần.**

|  |  |
| --- | --- |
| **Tuần** | **Công việc thực hiện** |
| Tuần 1 | Tìm hiểu và chọn đề tài |
| Tuần 2 | Đưa ra yêu cầu chức năng và phi chức năng của hệ thống |
| Tuần 3 | Tìm kiếm tài liệu, học cách vẽ sơ đồ và cơ sở dữ liệu, … |
| Tuần 4 | Tìm hiểu về các ngôn ngữ cần thiết để xây dựng trang web |
| Tuần 5 | Xây dựng mô hình khái quát về trang web |
| Tuần 6-7 | Tìm hiểu và viết giao diện cơ bản dựa vào kiến thức đã có và trau dồi thêm từ 3 ngôn ngữ HTML, CSS, Javascript |
| Tuần 8-13 | Xây dựng cơ sở dữ liệu trên MySQL |
| Tuần 14-17 | Xây dựng các chức năng của trang web và nhúng giao diện |
| Tuần 18 | Hoàn thành phần mềm và viết báo cáo sơ lược |
| Tuần 19 | Hoàn thành báo cáo |

# MỤC LỤC

[LỜI NÓI ĐẦU 1](#_Toc533751058)

[BẢNG PHÂN CÔNG CÔNG VIỆC 2](#_Toc533751059)

[MỤC LỤC 3](#_Toc533751060)

[DANH SÁCH HÌNH VẼ 5](#_Toc533751061)

[DANH SÁCH BẢNG BIỂU 6](#_Toc533751062)

[MỞ ĐẦU 7](#_Toc533751063)

[1. Đặt vấn đề 7](#_Toc533751064)

[2. Mục đích nghiên cứu 7](#_Toc533751065)

[3. Đối tượng nghiên cứu và phạm vi của hệ thống 7](#_Toc533751066)

[4. Phương pháp nghiên cứu 8](#_Toc533751067)

[5. Ý nghĩa thực tiễn của ứng dụng 8](#_Toc533751068)

[PHẦN NỘI DUNG 9](#_Toc533751069)

[Chương 1: Cơ sở lý thuyết 9](#_Toc533751070)

[1.1 Mô hình MVC 9](#_Toc533751071)

[1.2 Giới thiệu về PHP 12](#_Toc533751072)

[1.3 HTML và CSS 14](#_Toc533751074)

[1.4 Javascript 15](#_Toc533751075)

[1.5 Bootstrap 15](#_Toc533751075)

[1.6 Cơ sở dữ liệu MySQL 19](#_Toc533751076)

[Chương 2: Phân tích thiết kế hệ thống 20](#_Toc533751077)

[2.1 Mô tả về hệ thống 20](#_Toc533751078)

[2.1.1 Yêu cầu chức năng 20](#_Toc533751079)

[2.1.2 Yêu cầu phi chức năng 20](#_Toc533751080)

[2.1.3 Yêu cầu hệ thống 21](#_Toc533751081)

[2.2 Phân tích thiết kế 21](#_Toc533751082)

[2.2.1 Mô hình hóa yêu cầu 21](#_Toc533751083)

[Chương 3: Kết quả đạt được và định hướng phát triển 26](#_Toc533751086)

[3.1 Kết quả đạt được của hệ thống 26](#_Toc533751087)

[3.2 Định hướng phát triển 29](#_Toc533751088)

[PHẦN KẾT LUẬN 30](#_Toc533751089)

[TÀI LIỆU THAM KHẢO 31](#_Toc533751090)

# DANH SÁCH HÌNH VẼ

[**Hình 1.1 Mô hình hoạt động MVC.** 10](file:///C:\Users\ADMIN\Downloads\BaoCaoDoAnI.docx#_Toc535024171)

[**Hình 1.2 Bố cục HTML cơ bản cho 1 trang web.** 15](file:///C:\Users\ADMIN\Downloads\BaoCaoDoAnI.docx#_Toc535024172)

[**Hình 2.1 Kiến trúc của hệ thống.** 20](#_Toc535024166)

[**Hình 3.1 Giao diện đăng nhập khi truy cập vào URL của trang Web.** 24](#_Toc535024155)

# DANH SÁCH BẢNG BIỂU

[**Bảng A. Phân công công việc qua các tuần.** 2](#_Toc534501076)

# MỞ ĐẦU

## 1. Đặt vấn đề

Ngày này vấn đề đang được nhà nước và đẩy mạnh phát triển là ngày dịch vụ về du lịch .Bời vì nó đem lại cho chúng ta nguồn thu nhập lớn về kinh tế bên cạnh đó cũng làm tăng thêm cái nét đẹp của con người Việt Nam đến các bạn bè quốc tế cũng như năm châu.

Hiện nay ở Việt Nam , vấn đề quản bá thì còn chưa phát triển dẫn đến thị trường du lịch còn hạn chế. Qua tìm hiểu thì em thấy để mọi người biết đến chúng ta nhiều hơn thì chỉ có phương tiện duy nhất là Internet .Từ đó em muốn xây dựng một trang web nhằm giới thiệu một khu du lịch cụ thể là Sapa ,nhằm quản bá cũng như giới thiệu cho mọi người thế giời biết thêm về nó . Cụ thể website của em sẽ có nội dung giới thiệu về Sapa và các loại dịch vụ ở đó .

## 2. Mục đích nghiên cứu

Để giới thiệu cũng như nâng cao phát triển ngành du lịch của nước ta để mọi người trên thế giới sẽ biết đến đất nước hình chữ S , bên cạnh đó còn biết thêm về phong tục tập quán về các nét đẹp truyền thống của đất nước ta.

Để quan trọng nhất là không chỉ người Việt Nam hay người nước ngoài khi muốn đi nghỉ ở đâu đó không biết nên đặt phòng hay các vấn đề sinh hoạt hay dịch vụ ở nơi nghỉ dưỡng đó như thế nào

Từ đó để khắc phục cũng như phát triển em muốn xây dựng 1 trang website quản lí khách sạn để mọi người có thể biết các cảnh đẹp cũng như có thể đặt phòng như thế nào và giá cả ra sao cho hợp lí nhất

## 3. Đối tượng nghiên cứu và phạm vi của hệ thống

**Đối tượng:**

* Mô hình MVC
* Ngôn ngữ lập trình PHP
* Ngôn ngữ đánh dấu siêu văn bản HTML và ngôn ngữ CSS
* Ngôn ngữ lập trình Javascript
* Cơ sở dữ liệu MySQL

**Phạm vi của hệ thống:** Hệ thống được ứng dụng trong các hệ thống đặt phòng quy mô nhỏ

## 4. Phương pháp nghiên cứu

Trong đề tài này, em đã sử dụng các phương pháp nghiên cứu:

* Phương pháp tham khảo tài liệu: bằng cách thu thập thông tin từ sách, tạp chí về điện tử, viễn thông, truy cập từ mạng internet.
* Phương pháp thực nghiệm: Thử nghiệm liên tục để rút ra kinh nghiệm cũng như cố gắng hoàn thiện những yêu cầu mà nhóm đề ra.

## 5. Ý nghĩa thực tiễn của ứng dụng

Websites được áp dụng cho việc đặt mượn phòng giúp cho sinh viên có thể mượn phòng một cách thuận tiện nhất. Đối với ban quản lí sẽ có thể theo giõi việc cho mượn phòng một cách khoa học và hệ thống nhất.

# PHẦN NỘI DUNG

## Chương 1: Cơ sở lý thuyết

### Mô hình MVC

MVC là từ viết tắt của **'Model View Controller'**. Nó đại diện cho các nhà phát triển kiến ​​trúc áp dụng khi xây dựng các ứng dụng. Với kiến ​​trúc MVC, chúng ta xem xét cấu trúc ứng dụng liên quan đến cách luồng dữ liệu của ứng dụng của chúng ta hoạt động như thế nào

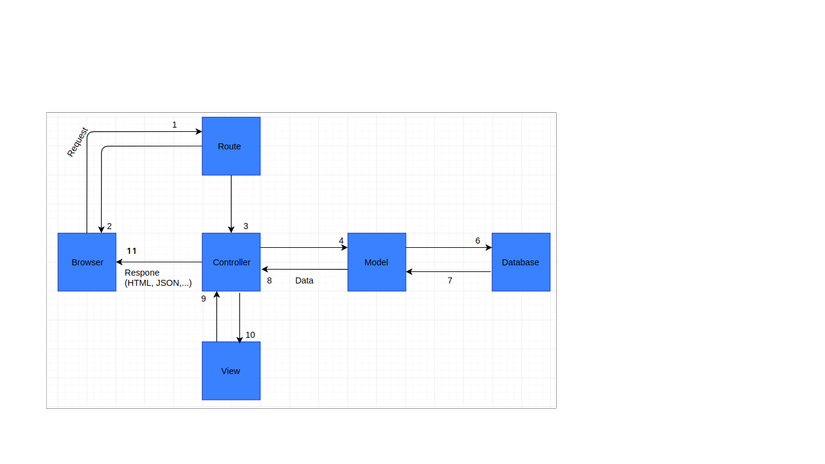
Mô hình MVC được chia làm 3 lớp xử lý gồm Model – View – Controller :

* **Model :** là nơi chứa những nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (mysql, mssql… ); nó sẽ bao gồm các class/function xử lý nhiều nghiệp vụ như kết nối database, truy vấn dữ liệu, thêm – xóa – sửa dữ liệu…
* **View :** là nới chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.
* **Controller :** là nới tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View.

***Sự tương tác giữa các thành phần***

* **Controller** tương tác với qua lại với **View**
* **Controller** tương tác qua lại với **Model**
* **Model** và **View** không có sự tương tác với nhau mà nó tương tác với nhau thông qua **Controller**.

Kịch bản mô hình hoạt động theo mô hình MVC



**Hình 1.1 Mô hình hoạt động MVC.**

**Kịch bản 1:** Người dùng chỉ gửi yêu cầu chuyển từ trang hiện tại sang một trang khác của web không có yêu cầu về dữ liệu.

* Nếu như người lập trình thực hiện việc **redirect** ở **Controller** thì lúc này luồng hoạt động là: **1** → **3** → **11**. **Request** được gửi từ **Browser** (trình duyệt) đến **Route** (nơi định tuyến hay phân định các request sẽ được xử lý ở đâu là trực tiếp tại **Route** hay là tại **Controller** nào đó).
* Nếu người lập trình để việc **redirect** trang ở trực tiếp tại **Route** thì luồng hoạt động của nó là: **1** → **2** yêu cầu chuyển trang sẽ được xử lý ngay tại **Route** mà không cần gọi đến **Controller**.

**Kịch bản 2:** Người dùng gửi 1 **request** **redirect** về 1 trang khác của web có trả về dữ liệu (ví dụ như xem danh sách các Tour đã đặt).

* Luồng hoạt động của nó là: **1** → **3** → **4** → **6** → **5** → **8** → **10** → **9** → **11**. Sau khi **request** gửi về **Route** được **Route** chuyển về xử lý tại **Controller**, lúc này các yêu cầu liên quan đến dữ liệu sẽ được xử lý và **Controller** sẽ có sự tương tác với **Model** để lấy dữ liệu, **Controller** sẽ sử dụng các **lớp/hàm** trong **Model** cần thiết để lấy ra những dữ liệu chính xác. **Model** tương tác với **Database** để lấy dữ liệu, dữ liệu trả về được gửi về **Model** từ **Model** gửi lại về **Controller**, **Controller** gọi đến **View** phù hợp với **request** kèm theo dữ liệu cho **View**, **View** sẽ lắp dữ liệu tương ứng vào **HTML** và gửi lại một **HTML** cho **Controller** sau khi thực hiện xong nhiệm vụ của mình. Hoàn tất các công đoạn trên **Controller** sẽ trả kết quả về **Browser**

**Kịch bản 3:** Người dùng chỉ yêu cầu dữ liệu nhưng không chuyển trang

* Luồng hoạt động của nó là: **1** → **3** → **4** → **6** → **5** → **8** → **11**. Thứ tự xử lý tương tự ở kịch bản 2 nhưng đến khi **Controller** nhận được dữ liệu trả về thì không gọi đến **View** mà trả **respone**lại cho trình duyệt thông qua **API**, dữ liệu trả về thường sẽ là dạng **JSON**.

=>**Trên đây là 1 số kịch bản thường xảy ra với request ở client gửi lên Server và được xử lý theo mô hình MVC**

***Ưu điểm MVC***

* Trình tự xử lý rất rõ ràng
* Mô hình MVC quy hoạch các class/function vào các thành phần riêng biêt **Controller - Model - View**, việc đó làm cho quá trình phát triển - quản lý - vận hành - bảo trì web diễn ra thuận lợi hơn, tạo ra được các chức năng chuyên biệt hoá đồng thời kiểm soát được luồng xử lý.
* Tạo mô hình chuẩn cho dự án, khi người có chuyên môn ngoài dự án tiếp cận với dự án dễ dàng hơn.
* Mô hình đơn giản, dễ hiểu, xử lý những nghiệp vụ đơn giản, và dễ dàng triển khai với các dự án nhỏ.

***Nhược điểm mô hình MVC***

* Đối với các dự án có tính phức tạp cao thì mô hình MVC trở nên không khả dụng.

### 1.2 Giới thiệu về PHP

**PHP** (viết tắt hồi quy "PHP: Hypertext Preprocessor") là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) kịch bản hay một loại [mã lệnh](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_l%E1%BB%87nh&action=edit&redlink=1) chủ yếu được dùng để phát triển các [ứng dụng](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) viết cho máy chủ, [mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F), dùng cho mục đích tổng quát. Nó rất thích hợp với [web](https://vi.wikipedia.org/wiki/Internet) và có thể dễ dàng nhúng vào trang [HTML](https://vi.wikipedia.org/wiki/HTML). Do được tối ưu hóa cho các [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web), tốc độ nhanh, nhỏ gọn, cú pháp giống [C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) web phổ biến nhất thế giới.

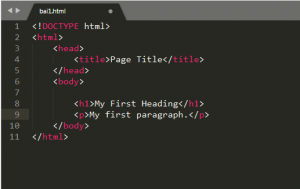
Các phiên bản PHP:

* **PHP/FI do**[**Rasmus Lerdorf**](https://vi.wikipedia.org/w/index.php?title=Rasmus_Lerdorf&action=edit&redlink=1)**tạo ra năm**[**1994**](https://vi.wikipedia.org/wiki/1995)
* **PHP 3 được**[**Andi Gutmans**](https://vi.wikipedia.org/w/index.php?title=Andi_Gutmans&action=edit&redlink=1)**và**[**Zeev Suraski**](https://vi.wikipedia.org/w/index.php?title=Zeev_Suraski&action=edit&redlink=1)**tạo ra năm 1997**
* **PHP 4 được giới thiệu vào giữa năm**[**1999**](https://vi.wikipedia.org/wiki/1999)
* **PHP 5 Ngày**[**29 tháng 6**](https://vi.wikipedia.org/wiki/29_th%C3%A1ng_6)**năm**[**2003**](https://vi.wikipedia.org/wiki/2003)**, PHP 5 Beta 1 đã chính thức được công bố để cộng đồng kiểm nghiệm**
* **PHP 6**

### 1.3 HTML và CSS

HTML là chữ viết tắt của cụm từ **HyperText Markup Language**, có nghĩa là **“ngôn ngữ đánh dấu siêu văn bản”**. Cha đẻ của HTML là Tim Berners-Lee, cũng là người khai sinh ra World Wide Web và là chủ tịch của World Wide Web Consortium (W3C – tổ chức thiết lập ra các chuẩn trên môi trường Internet). HTML dùng để mô tả cấu trúc, các thuộc tính liên quan đến cách hiển thị của một đoạn văn bản nào đấy ra trình duyệt. Các phần tử HTML là các khối xây dựng nên các trang web. Các phần tử HTML được biểu diễn bằng thẻ. Trình duyệt không hiển thị các thẻ HTML, nhưng sử dụng chúng để hiển thị nội dung của trang.

Bố cục cơ bản của HTML trên một trang web thường được chia thành 2 phần  chính nằm trong cặp thẻ **<html> </html>:** là head và body



**Hình 1.2 Bố cục HTML cơ bản cho 1 trang web.**

**Trong đó:**

* **<!DOCTYPE html>**: Dùng để xác định tài liệu này thuộc dạng html cho trình duyệt biết.
* **<html> </html>**: Là phần tử gốc, nằm ngoài cùng mỗi trang web, có nhiệm vụ bao toàn bộ nội dung của trang web
* **<head> </head>**: Phần mở đầu. Có nhiệm vụ khai báo tất cả thông tin của trang web
* **<title> </title>**: Thẻ tiêu đề: Dùng để khai báo tiêu đề cho mỗi trang web. Tiêu đề này sẽ hiển thị ở thanh tab trên cùng của trình duyệt.
* **<body> </body>**: Phần nội dung. Chứa tất cả nội dung sẽ hiển thị ra trên trình duyệt
* **<h1> </h1>** và**<p> </p>** : Các thẻ định dạng văn bản.

**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.

### 1.4 Javascript

JavaScript là một ngôn ngữ kịch bản (scripting language) được dùng để tạo các script ở máy client (client-side script) và máy server (server-side script). Các script ở máy client được thực thi tại trình duyệt, các script ở máy server được thực hiện trên server. Chương này sẽ giới thiệu cho chúng ta về ngôn ngữ Javascript, và cách chèn một script vào trong tài liệu HTML.

HTML lúc đầu được phát triển như là một định dạng của tài liệu có thể chuyển dữ liệu trênInternet. Tuy nhiên, không lâu sau đó, trọng tâm của HTML nặng tính hàn lâm và khoa học dần chuyển hướng sang người dùng thường nhật vì ngày nay người dùng xem Internet như là một nguồn thông tin và giải trí. Các trang Web ngày càng mang tính sáng tạo và đẹp mắt hơn nhằm thu hút nhiều người dùng hơn. Nhưng thực chất kiểu dáng và nội dung bên trong vẫn không thay đổi. Và người dùng hầu như không thể điều khiển trên trang Web mỗi khi nó được hiển thị.

Javascript được phát triển như là một giải pháp cho vấn đề nêu trên. Javascript là một ngôn ngữ kịch bản được Sun Microsystems và Netscape phát triển. Nó được dùng để tạo các trang Web động và tương tác trên Internet. Đối với những người phát triển HTML, Javascript rất hữu ích trong việc xây dựng các hệ thống HTML có thể tương tác với người dùng

**Hiệu ứng và quy tắc JavaScript**

JavaScript là một ngôn ngữ lập trình được nhúng được trong các trang HTML. JavaScript nâng cao tính động và khả năng tương tác cho web-site bằng cách sử dụng các hiệu ứng của nó như thực hiện các phép tính, kiểm tra form, viết các trò chơi, bổ sung các hiệu ứng đặc biệt, tuỳ biến các chọn lựa đồ hoạ, tạo ra các mật khẩu bảo mật và hơn thế nữa.

**Chúng ta có thể sử dụng JavaScript để:**

**Tương tác với người dùng**. Chúng ta có thể viết mã để đáp lạI các sự kiện. Các sự này sẽ có thể phát sinh bởi người dùng - - nhấp chuột hay được phát sinh từ hệ thống - - định lại kích thước của trang và v.v.

**Thay đổi nội dung động**. Mã JavaScript có thể dùng để thay đổi nội dung và vị trí các phần tử một cách động trên một trang nhằm đáp lại sự tương tác với người dùng.

**Kiểm tra tính hợp lệ dữ liệu**. Chúng ta có thể viết mã nhằm kiểm tra tính hợp lệ của dữ liệu do người dùng nhập vào trước khi nó được gửi lên Web server để xử lý.

Giống như các ngôn ngữ khác, JavaScript cũng tuân thủ một số quy tắc ngữ pháp căn bản. Việc nắm vững các quy tắc ngữ pháp này có thể giúp ta đọc được script và tự viết các script không bị lỗi.

Một số trong các luật này bao gồm:

**Dùng Caps.**JavaScript phân biệt chữ hoa chữ thường

**Dùng Pairs.**Trong JavaScript, luôn luôn có cặp ký hiệu mở và đóng. Lỗi sẽ xuất hiện khi bỏ sót hoặc đặt sai một trong hai ký hiệu này.

**Dùng Spaces (các ký tự trắng).** Như HTML, JavaScript thường bỏ qua ký tự trắng. Trong JavaScript, ta có thể thêm vào các ký tự trắng hoặc các tab giúp cho ta dễ dàng đọc hay sửa các file script.

**Dùng Chú thích (Comments).** Các chú thích giúp ta ghi chú về chức năng của đoạn script, thờI gian và ngườI tạo ra đoạn script.

Mặc dù cả client-side JavaScript và server-side JavaScript đều dựa trên một ngôn ngữ nền tảng như nhau, nhưng mỗI loạI còn có thêm những tính năng chuyên biệt phù hợp với môi trường mà nó chạy. Nghĩa là, client-side JavaScript bao gồm các đối tượng được định nghĩa sẵn chỉ có thể sử dụng trên trình duyêt, Server-side JavaScript bao gồm các đối tượng và các hàm được định nghĩa sẵn chỉ có thể sử dùng trong các ứng dụng phía server (server-side applications)

**Các công cụ JavaScript và IDE, và môi trường thực thi**

Các công cụ sinh mã JavaScript và IDE giúp tạo ra mã JavaScript rất hữu hiệu. . Các công cụ này còn giúp ta nhanh chóng phát triển website của mình.

Một vài công cụ JavaScript và IDE được đề cập dưới đây:

**Dialog Box.**Công cụ này tự động tạo mã để sinh ra các hộp thoại tuỳ biến trên các trình duyệt khác nhau (*alert, confirm, prompt, v.v.*) mang lại kiểu dáng chuyên nghiệp cho website.

**Pop-up Menu builder.**Chỉ cần đưa vào các lựa chọn, công cụ này sẽ tự động tạo ra các pop-up menu trên các trình duyệt khác nhau.

**Remotes.**Tự động sinh mã để mở ra một cửa sổ popup.

Như chúng ta đã biết, JavaScript có thể được chạy trên máy khách (client) và máy chủ (server). Bên phía máy khách, trình duyệt sẽ thực thi mã lệnh javascript trong trang web khi mở nó. Bên phía máy chủ, mã javascript sẽ được thực thi tại máy chủ và do máy chủ thực hiện.

**Client-side Java Script**

Khi máy client yêu cầu một trang HTML, server sẽ kiểm tra xem trang đó có chứa script hay không. Nếu nó chứa các client-side script, server sẽ chuyển toàn bộ tài liệu bao gồm mã lệnh JavaScript và nội dung HTML của nó cho trình duyệt. Khi trình duyệt nhận được tài liệu đó, nó thực thi các mã lệnh HTML và JavaScript mà không cần bất kì sự tương tác nào với server.

**JavaScript trên Web Server**

Chúng ta có thể nhúng các lệnh JavaScript chạy trên server (server-side script) vào trong tài liệu HTML. Quá trình tạo ra các ứng dụng server-side là một quá trình gồm hai giai đọan.

Các trang HTML có chứa các câu lệnh JavaScript của cả client-side và server-side đều được tạo ra cùng với các file JavaScript. Tất cả các file này sẽ được biên dịch thành dạng mã thực thi được là bytecode.

Khi trình duyệt yêu cầu trang HTML, run-time engine sẽ thực thi mã lệnh server-side JavaScript rồi trả trang HTML về cho trình duyệt.

Một số công dụng của script server-side bao gồm:

* Kết nối vào các cơ sở dữ liệu
* Chia sẻ thông tin cho những người dùng của một ứng dụng
* Truy cập vào hệ thống file trên server

**Nhúng JavaScript vào trong trang Web**

Chúng ta có thẻ chèn các lệnh JavaScritp vào trong một tài liệu HTML theo những cách sau đây:

Nhúng các câu lệnh trực tiếp vào trong tài liệu bằng cách sử dụng thẻ <SCRIPT> Liên kết file nguồn JavaScript với tài liệu HTML. Đặt các biểu thức JavaScript làm giá trị cho thuộc tính của thẻ HTML. Dùng như trình xử lý sự kiện trong các thẻ HTML

### 1.5 Boostrap

Bootstrap là 1 framework HTML, CSS, và JavaScript cho phép người dùng dễ dàng thiết kế website theo 1 chuẩn nhất định, tạo các website thân thiện với các thiết bị cầm tay như mobile, ipad, tablet,...

Bootstrap là một trong những framework được sử dụng nhiều nhất trên thế giới để xây dựng nên một website. Bootstrap đã xây dựng nên 1 chuẩn riêng và rất được người dùng ưa chuộng. Chính vì thế, chúng ta hay nghe tới một cụm từ rất thông dụng "Thiết kế theo chuẩn Bootstrap".

Từ cái "chuẩn mực" này, chúng ta có thể thấy rõ được những điểm thuận lợi khi sử dụng Bootstrap.

Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cẩn có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.

Responsive: Bootstrap xây dựng sẵn reponsive css trên các thiết bị Iphones, tablets, và desktops. Tính năng này khiến cho người dùng tiết kiệm được rất nhiều thời gian trong việc tạo ra một website thân thiện với các thiết bị điện tử, thiết bị cầm tay.

Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera). Tuy nhiên, với IE browser, Bootstrap chỉ hỗ trợ từ IE9 trở lên. Điều này vô cùng dễ hiểu vì IE8 không support HTML5 và CSS3.

Có 2 cách để bạn có thể sử dụng Bootstrap trên web của bạn.

1. Download Bootstrap packet từ [getbootstrap.com](http://getbootstrap.com/)
2. Thêm Bootstrap từ CDN

### 1.6 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 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. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau.

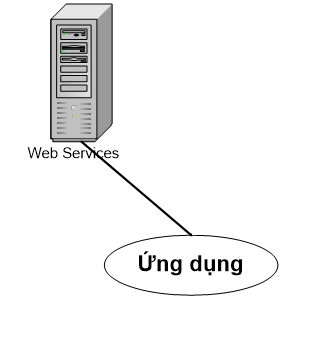
MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL)

MySQL được sử dụng cho việc bổ trợ [PHP](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,…

## Chương 2: Phân tích thiết kế hệ thống

### 2.1 Mô tả về hệ thống

Hệ thống được xây dựng theo mô hình trao đổi dữ liệu client-server. Trong đó ứng dụng phía thiết bị phía client là máy tính truy cập web tạo các truy vấn lên máy chủ là máy chủ web của ứng dụng.



**Hình 2.1 Kiến trúc của hệ thống.**

#### 2.1.1 Yêu cầu chức năng

***Các chức năng của hệ thống bao gồm:***

* Đặt phòng (khách hàng)
* Quản lý phòng ̣( nhân viên hay quản lí )

#### 2.1.2 Yêu cầu phi chức năng

* Chương trình có thể đảm bảo làm việc 24/7
* Chương trình có thể vận hành tốt khi mà cơ sở dữ liệu có thể tăng đột biến trong tương lai.
* Chương trình có thể thực hiện chính xác kết quả với sai số nhỏ với cơ sở dữ liệu lớn.
* Chương trình phải đẹp và đơn giản để những người có ít kinh nghiệm máy tính có thể sử dụng.

#### 2.1.3 Yêu cầu hệ thống

* Có kết nối Internet.

### 2.2 Phân tích thiết kế

#### 2.2.1 Mô hình hóa yêu cầu

***Xác định các tác nhân:***

* **Khách hàng**: Là người trực tiếp sử dụng phần mềm để đặt lịch thuận tiện cho đặt phòng.
* **Quản lý hay nhân viên** : Là người trực tiếp sử dụng phần mềm để cập nhật các danh mục, quản lý tài nguyên , xác nhận lịch đặt phòng.

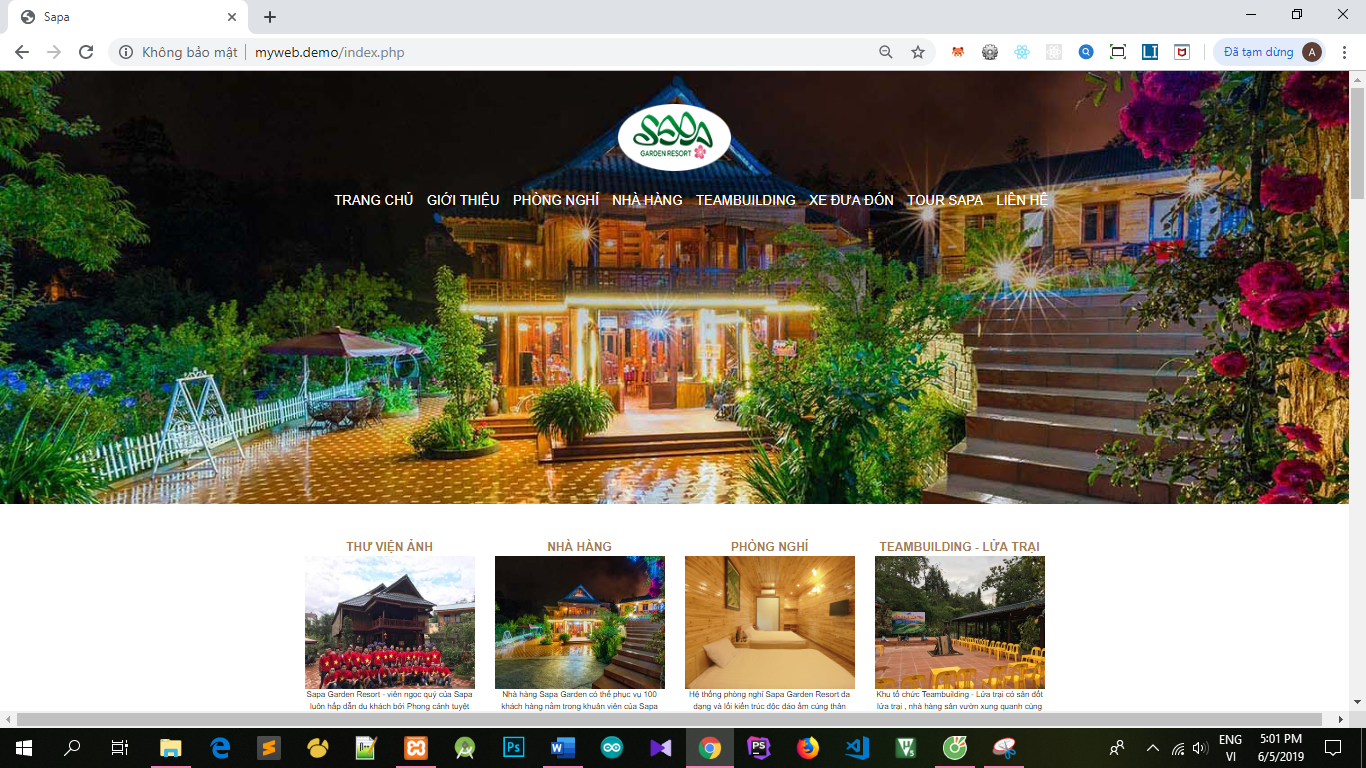
***Khái quát về website:***

* Các dự liệu quan trọng sẽ được lưu trữ trên sever cụ thể là trên MySQL và được quản lý hay nhân viên theo dõi cũng như cập nhật thêm thông tin như sửa , xóa ,....
* Về phía khách hàng thì sẽ có thể đặt phòng các dự liệu đặt trên trang web sẽ được đẩy lên MySQL phía bên quản lí sẽ nhận được các thông tin đó

## Chương 3: Kết quả đạt được và định hướng phát triển

### 3.1 Kết quả đạt được của hệ thống

Hệ thống bước đầu được hoàn thiện và sẽ càng hoàn thiện hơn nếu được đưa vào sử dụng thực tiễn và nhận được sự góp ý của phía người dùng. Dưới đây, là một số kết quả bước đầu đạt được của hệ thống mà em đã xây dựng.



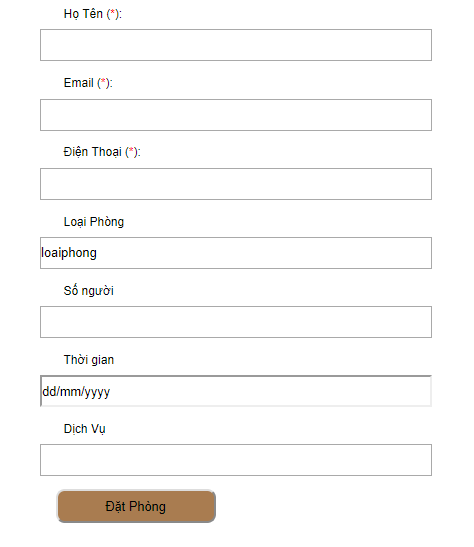
**Hình 3.1 Giao diện trang chủ khi truy cập vào URL của trang Web.**



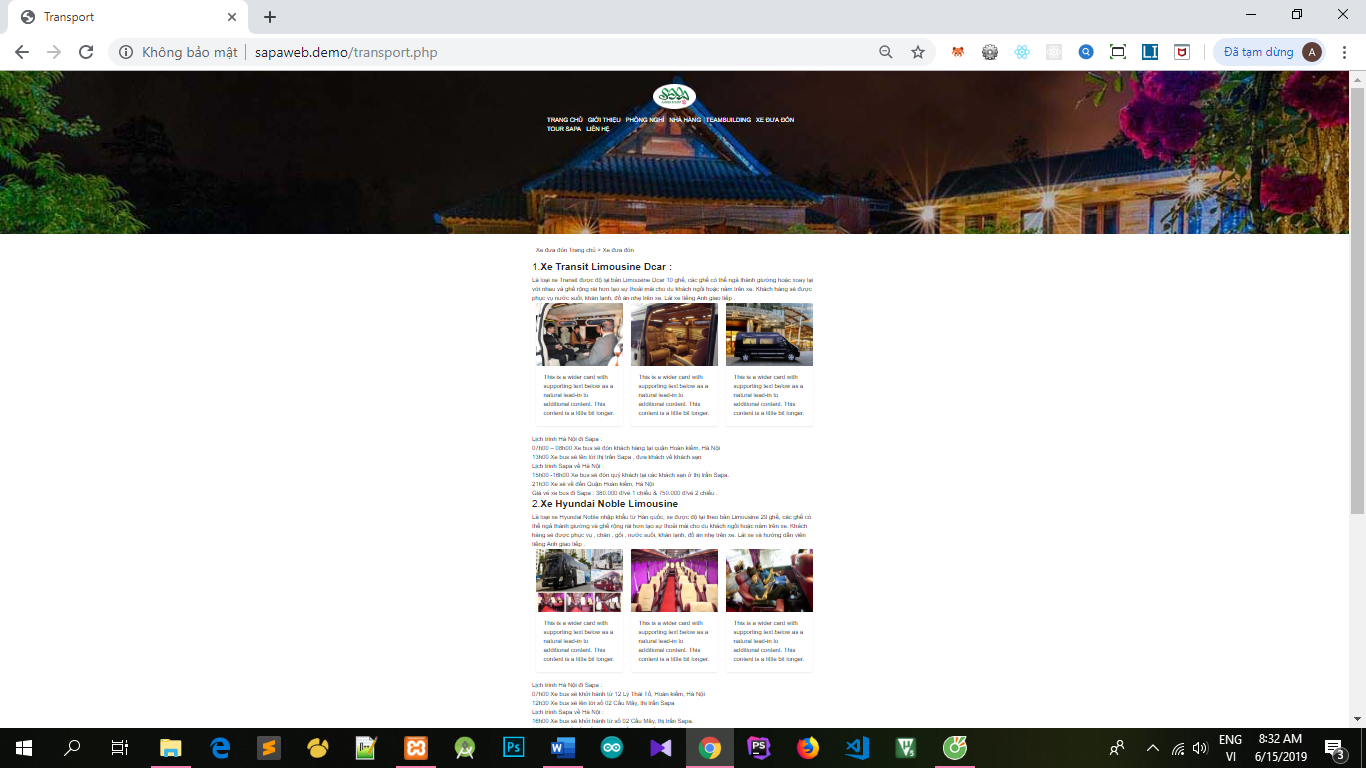
**Hình 3.2 Giao diện giới thiệu khách sạn**



**Hình 3.3. Giới thiệu phòng**



**Hình 3.4 Form đặt phòng.**



**Hình 3.5 Trang Xe đưa đón.**

### 3.2 Định hướng phát triển

Em đã thực hiện về căn bản là đầy đủ chức năng giới thiệu phòng cũng như đặt phòng bên cạnh đó thì vẫn còn một số phần chưa hoàn chỉnh. Do vậy, trong thời gian tiếp theo em sẽ:

* Nghiên cứu thêm các công nghệ mới để áp dụng vào trang Web, giúp trang Web xử lý các request nhanh hơn, bảo mật của hệ thống được nâng lên tốt hơn.
* Ứng dụng modules NodeJs vào hệ thống để Real Time quá trình đặt lịch và xác nhận lịch hẹn, để từ đó sinh viên có thể biết thêm được thông tin cuộc hẹn lân cận (Ví dụ: Sinh viên biết được trong khung giờ mà mình hẹn lịch, có bao nhiêu cuộc hẹn cùng như thế, khả năng được chấp nhận lịch hẹn của mình là bao nhiêu %).
* Đưa hệ thống chạy trên nền tảng di động, nền tảng mà đang thành xu thế phát triển rất mạnh trong thời gian gần đây.
* Kết nối với cơ sở dữ liệu về sinh viên của nhà trường để thuận tiện cho việc quản lí và giám sát.

# PHẦN KẾT LUẬN

Do sự hạn hẹp về thời gian và trình độ nên website mà em xây dựng chỉ mới đáp ứng được những nhu cầu cơ bản, nếu đưa vào thực tế sử dụng thì phải giải quyết tốt bài toán về các tình huống có thể xảy ra.

Website về cơ bản đã hoàn thành nhưng không thể tránh khỏi những lỗi về tình huống, mà điều này cần thu nhận trong quá trình sử dụng Website một thời gian. Tuy vậy, em đã rất nổ lực và website đã hoàn thành được những nội dung chính sau: giúp sinh viên có thể đặt phòng gián tiếp giúp giảm thiểu chi phí cũng như giúp ban quản lí kí túc xá quản lí, tiếp nhận, sắp xếp và quản lí thông tin giúp quá trình đặt diễn ra nhanh chóng, chính xác. Ngoài ra, em đã vận dụng tốt ngôn ngữ HTML, CSS, JS, PHP,mô hình MVC, MySQL vào việc xây dựng website. Phần giao diện người dùng: giao diện thân thiện, cho phép bạn đọc dễ dàng xem tin, đặt lịch mà website cung cấp.

Trong quá trình thực hiện đề tài sẽ không thể mắc phải những sai lầm, thiết xót do vốn kiến thức còn hạn hẹp. Em mong sẽ nhận được những sự góp ý, phê bình, chia sẻ của cô để đề tài này và những đề tài tiếp theo được hoàn thiện hơn.

Em xin chân thành cảm ơn cô!

# TÀI LIỆU THAM KHẢO

[1] <https://laravel.com/docs/5.7>

[2] <https://giaphiep.com/docs/5.3/installation>

[3] <https://api.jquery.com/>

[4] <https://thachpham.com/web-development/html-css/html-va-css-can-ban-danh-cho-cho-moi-nguoi.html>