

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Vĩnh Long, ngày tháng năm

Giảng viên hướng dẫn

(Ký tên và ghi rõ họ tên)

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Vĩnh Long, ngày tháng năm

Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến thầy Võ Thành C, người đã trực tiếp hướng dẫn và tận tình chỉ bảo em trong suốt quá trình thực hiện đề tài "Xây dựng Website tra cứu các di tích lịch sử tỉnh Vĩnh Long".

Nhờ có những định hướng, sự góp ý quý báu và những kiến thức chuyên môn mà thầy truyền đạt, em đã có thể hoàn thiện sản phẩm một cách tốt nhất. Việc thực hiện đề tài này không chỉ giúp em nâng cao kỹ năng lập trình, thiết kế mà còn giúp em hiểu thêm về những giá trị lịch sử cao đẹp của quê hương Vĩnh Long.

Dù đã cố gắng hoàn thành bài báo cáo với tất cả tâm huyết, nhưng do kiến thức và kinh nghiệm còn hạn chế, sản phẩm của em còn nhiều thiếu sót. Em rất mong nhận được những ý kiến đóng góp của thầy để website ngày càng hoàn thiện hơn.

Em xin kính chúc thầy Võ Thành C thật nhiều sức khỏe, hạnh phúc và gặt hái được nhiều thành công trong sự nghiệp trồng người.

Em xin trân trọng cảm ơn!

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	16
1.1. Đặt vấn đề	16
1.1.1. Bối cảnh lịch sử và văn hóa	16
1.2. Mục tiêu đồ án	16
1.2.1. Mục tiêu tổng quát	16
1.2.2. Mục tiêu cụ thể	16
1.2.3. Đối tượng hướng đến	17
1.3. Tổng quát về cấu trúc đồ án	17
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT	18
2.1. Khái niệm về Front-end và Back-end	18
2.1.1. Khái niệm Front-end	18
2.1.2. Ngôn ngữ cốt lõi:	18
2.1.3. Khái niệm Back-end	19
2.1.4. Cách thức hoạt động và ngôn ngữ của Back-end	19
2.1.5. Ngôn ngữ Back-end	20
2.2. So sánh cơ bản giữa Front-end và Back-end	21
2.3. Môi quan hệ hỗ trợ	22
2.4. Vai trò cốt lõi	22
2.5. Front-end hỗ trợ Back-end	22
2.6. Back-end hỗ trợ Front-end	22
2.7. My SQL là gì	23
2.7.1. khái niệm cơ bản	23

2.7.2. Các phần của SQL	23
2.7.3. Cách thức hoạt động	23
2.7.4. Ưu và nhược điểm MySQL	23
2.7.5. Một số thuật ngữ thường gặp:	24
2.8. PHP là gì	25
2.8.1. Khái niệm PHP	25
2.8.2. Ứng dụng PHP trong lập trình	25
2.8.3. Ưu và nhược điểm của PHP	26
2.9. UX/UI là gì	27
2.9.1. khái niệm UX/UI	27
2.9.2. Đặc điểm và vai trò	27
2.9.3. Tầm quan trọng và ảnh hưởng đến người dùng	27
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	29
3.1. Quy trình nghiên cứu và phát triển	29
3.1.1. Khảo sát và phân tích yêu cầu	29
3.1.2. Nghiên cứu công nghệ	29
3.1.3. Thiết kế hệ thống	29
3.1.4. Phát triển và kiểm thử	30
3.2. Phân tích đặc điểm và yêu cầu hệ thống	30
3.2.1. Yêu cầu chức năng	30
3.2.2. Yêu cầu phi chức năng	31
3.3. Thiết kế hệ thống kiến trúc	32
3.4. Mô hình MVC (Model-View-Controller)	33

3.5. Luồng xử lý dữ liệu	33
3.6. Mô hình ERD (Entity Relationship Diagram)	34
3.6.1. Mối liên hệ	35
3.7. Cài đặt và triển khai hệ thống	36
3.7.1. Yêu cầu hệ thống	36
3.7.2. Cấu trúc thư mục dự án	36
3.8. Kiểm tra hệ thống	37
3.9. Tổng kết chương	38
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	40
4.1. Tổng quan về dự án	40
4.1.1. Tổng quan	40
4.1.2. Chức năng	40
4.2. Phần trang chủ của trang web	41
4.2.1. Tổng quan về trang chủ	41
4.2.2. Phần Header	42
4.2.3. Phần Footer	42
4.2.4. Banner chính	43
4.2.5. Phần thẻ card	44
4.3. Tổng quan về phần đăng ký	45
4.3.1. Giao diện chính đăng ký	45
4.3.2. Hướng dẫn đăng ký tài khoản	46
4.3.3. Cách thức hoạt động	47
4.4. Phần đăng nhập	48

4.4.1. Giao diện chính đăng nhập	48
4.4.2. Hướng dẫn đăng nhập	49
4.4.3. Cách thức hoạt động	50
4.5. Tổng quan về phân tra cứu	51
4.5.1. Giao diện chính tra cứu	51
4.5.2. Hướng dẫn tra cứu	53
4.5.3. Cách thức hoạt động	55
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	58
5.1. Kết quả đạt được	58
5.2. Giao diện và trải nghiệm người dùng	58
5.2.1. Giao diện	58
5.2.2. Tính năng kỹ thuật	58
5.3. Đóng góp mới	59
5.3.1. Về mặt công nghệ	59
5.3.2. Về mặt ứng dụng	59
5.3.3. Về mặt phương pháp	59
5.4. Hạn chế khuyết điểm	59
5.4.1. Về chức năng	59
5.4.2. Về kỹ thuật	59
5.4.3. Về nội dung	60
5.5. ĐỀ XUẤT PHÁT TRIỂN	60
5.5.1. Ngắn hạn (3-6 tháng)	60
5.5.2. Trung hạn (6-12 tháng)	60

5.5.3. Dài hạn (1-2 năm)	60
5.5.4. Đề xuất về tổ chức và vận hành	60
5.6. Ý nghĩa thực tiễn	61
5.6.1. Đối với cộng đồng	61
5.6.2. Đối với ngành công nghệ	61
5.6.3. Đối với nghiên cứu khoa học	61
DANH MỤC TÀI LIỆU THAM KHẢO	62
PHỤ LỤC	63

DANH MỤC HÌNH ẢNH

Hình 4.2.1 Hình ảnh trang chủ	41
Hình 4.2.2 Hình ảnh Header	42
Hình 4.2.3 Hình ảnh Footer	42
Hình 4.2.4 Hình ảnh Banner Thứ 1	43
Hình 4.2.5 Hình ảnh Banner Thứ 2	43
Hình 4.2.6 Hình ảnh Banner Thứ 2	43
Hình 4.2.6 Hình ảnh phần thẻ Card	44
Hình 4.3.1 Hình ảnh trang Đăng ký	45
Hình 4.3.1. Hình ảnh hướng dẫn đăng ký	47
Hình 4.3.2 Hình ảnh cơ sở dữ liệu được lưu lại	48
Hình 4.4.1 Hình ảnh trang Đăng nhập	48
Hình 4.4.2 Hình ảnh hướng dẫn đăng nhập	50
Hình 4.4.3 Hình ảnh đăng nhập thành công	50
Hình 4.5.1 Hình ảnh trang đăng ký	51
Hình 4.5.2 Hình ảnh bộ lọc theo lọc theo địa điểm	51
Hình 4.5.3 Hình ảnh bộ lọc theo loại hình	52
Hình 4.5.4 Hình ảnh nút đặt lại bộ lọc	52
Hình 4.5.5 Hình ảnh thanh tìm kiếm	52
Hình 4.5.6 Hình ảnh phần tìm kiếm nhanh	53
Hình 4.5.7 Hình ảnh trạng thái phản hồi người dùng	53
Hình 4.5.8 Hình ảnh cách thức hoạt động	55
Hình 4.5.8 Hình ảnh sau khi tra cứu	56

Hình 4.5.9 Hình ảnh bản đồ.....	56
---------------------------------	----

DANH MỤC BẢNG BIỂU

Bảng 2.1.1: Các ngôn ngữ chính trong Back-end	20
Bảng 2.2.1: So sánh cơ bản giữa Front-end và Back-end	21
Bảng 3.3.1 Sơ đồ thiết kế hệ thống kiến trúc	32
Bảng 3.4.1 Sơ đồ mô hình MVC	33
Bảng 3.6.1 Bảng Users	34
Bảng 3.6.2 Bảng di tích	34
Bảng 3.6.3: sơ đồ mối liên hệ giữa các bảng	35
Bảng 3.7.1 Bảng Cấu trúc thư mục dự án	36

TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

1. Vấn đề nghiên cứu

Trong bối cảnh chuyển đổi số, nhu cầu tiếp cận thông tin lịch sử một cách nhanh chóng và chính xác trở nên vô cùng thiết yếu. Đề tài tập trung giải quyết bài toán xây dựng một hệ thống thông tin trực tuyến chuyên biệt nhằm giới thiệu các di tích lịch sử trên địa bàn Tỉnh Vĩnh Long. Vấn đề trọng tâm mà nghiên cứu hướng đến là giúp người dùng khắc phục những khó khăn trong việc tìm kiếm thông tin thủ công, từ đó có thể tra cứu và tìm hiểu về các địa danh lịch sử một cách thuận tiện, nhanh chóng và trực quan nhất.

2. Các hướng tiếp cận

Để giải quyết bài toán nêu trên, đề tài đã xác định các hướng tiếp cận khoa học và thực tế như sau:

Tiếp cận về nghiệp vụ và dữ liệu: Thực hiện khảo sát hiện trạng thực tế và xác định các yêu cầu cụ thể của đề tài. Quá trình này bao gồm việc thu thập thông tin, hình ảnh và các tài liệu liên quan trực tiếp đến các di tích lịch sử tại Vĩnh Long để làm giàu kho dữ liệu cho hệ thống.

Tiếp cận về công nghệ: Tập trung nghiên cứu và ứng dụng bộ ba ngôn ngữ lập trình nền tảng của công nghệ Web bao gồm HTML, CSS và JavaScript. Đây là hướng tiếp cận phù hợp cho một đồ án cơ sở ngành, giúp sinh viên nắm vững kiến trúc giao diện và tư duy lập trình hệ thống.

Tiếp cận về kiến trúc: Hệ thống được thiết kế theo mô hình phân tách rõ rệt giữa thành phần dữ liệu, thành phần xử lý và thành phần giao diện để đảm bảo tính module hóa và dễ dàng bảo trì.

3. Cách thức giải quyết vấn đề

Vấn đề được giải quyết thông qua một lộ trình thực hiện chặt chẽ kéo dài 8 tuần (từ tháng 11 đến tháng 12 năm 2025) với các bước cụ thể:

Giai đoạn Phân tích và Thiết kế: Sau khi hoàn thành đề cương, sinh viên tiến hành phân tích thiết kế dữ liệu và xử lý. Đặc biệt, việc thiết kế cơ sở dữ liệu và xây dựng kiến trúc hệ thống (bao gồm cả Front-end và Back-end) được thực hiện song song để tạo ra khung xương vững chắc cho website.

Giai đoạn Phát triển (Lập trình): Tập trung xây dựng các nhóm chức năng cho hai đối tượng người dùng riêng biệt:

Quản trị viên: Xây dựng các module quản lý website, các chức năng nghiệp vụ quan trọng như Thêm, Xóa, Sửa thông tin di tích, quản lý người dùng và tích hợp bản đồ chỉ dẫn để tối ưu hóa khả năng giám sát.

Người dùng: Phát triển các tính năng tương tác cơ bản như đăng ký, đăng nhập và công cụ tra cứu thông tin di tích.

Giai đoạn Kiểm thử và Tối ưu: Đây là bước quan trọng nhất để đảm bảo chất lượng sản phẩm. Hệ thống sẽ trải qua quy trình kiểm thử toàn diện về chức năng (Functionality), giao diện (UI) và hiệu suất (Performance) để phát hiện và sửa lỗi trước khi bàn giao.

4. Kết quả đạt được

Kết thúc quá trình nghiên cứu và thực hiện, đề tài dự kiến đạt được các sản phẩm sau:

Sản phẩm phần mềm: Một Website giới thiệu di tích lịch sử tỉnh Vĩnh Long hoàn chỉnh, vận hành ổn định và đáp ứng đầy đủ các yêu cầu về tra cứu thông tin đã đặt ra.

Sản phẩm học thuật: Quyền báo cáo đồ án chi tiết, trình bày đầy đủ quy trình từ khảo sát, phân tích thiết kế đến triển khai và kiểm thử hệ thống theo đúng tiêu chuẩn đào tạo của Khoa Công nghệ thông tin.

Về mặt thời gian: Đảm bảo hoàn thành đúng tiến độ cam kết dưới sự hướng dẫn và xác nhận của giảng viên chuyên môn.

MỞ ĐẦU

1. Lý do chọn đề tài

Trong thời đại chuyên đổi số, việc ứng dụng công nghệ thông tin để bảo tồn và phát huy các giá trị văn hóa, lịch sử là một xu thế tất yếu. Tỉnh Vĩnh Long là vùng đất có bề dày lịch sử với nhiều di tích quan trọng, tuy nhiên việc tiếp cận thông tin của người dân và du khách đôi khi còn gặp nhiều khó khăn do tài liệu phân tán,.

Xuất phát từ thực tế đó, đề tài "Xây dựng Website giới thiệu các di tích lịch sử trên địa bàn Tỉnh Vĩnh Long" được lựa chọn nhằm tạo ra một kênh thông tin chính thống, trực quan. Việc xây dựng website không chỉ giúp người dùng tra cứu thông tin một cách thuận tiện, nhanh chóng mà còn là cơ hội để sinh viên áp dụng các kiến thức cốt lõi về lập trình web như HTML, CSS và JavaScript vào giải quyết một bài toán cụ thể trong thực tiễn,.

2. Mục đích nghiên cứu

Mục tiêu chính của đề án này là:

Xây dựng hoàn chỉnh một hệ thống Website có khả năng cung cấp thông tin chi tiết, hình ảnh và vị trí địa lý của các di tích lịch sử tại Vĩnh Long,.

Cung cấp công cụ quản trị giúp việc cập nhật, chỉnh sửa thông tin di tích và quản lý người dùng trở nên khoa học và dễ dàng.

Nâng cao trải nghiệm người dùng thông qua các tính năng tra cứu thông minh và tích hợp bản đồ chỉ dẫn.

Củng cố kỹ năng phân tích thiết kế hệ thống, quy trình phát triển phần mềm thực tế.

3. Đối tượng và phạm vi nghiên cứu

3.1. Đối tượng nghiên cứu:

Đối tượng thực tế: Các thông tin, hình ảnh, tài liệu liên quan đến các di tích lịch sử hiện hữu trên địa bàn Tỉnh Vĩnh Long (sau khi sáp nhập 2025).

Đối tượng kỹ thuật: Quy trình xây dựng website sử dụng bộ ba ngôn ngữ lập trình HTML, CSS và JavaScript; các phương pháp phân tích thiết kế dữ liệu và kiến trúc hệ thống Front-end/Back-end.

3.1. Phạm vi nghiên cứu:

- Phạm vi nội dung: Tập trung vào các chức năng cốt lõi bao gồm: Quản lý thông tin di tích (thêm, xóa, sửa), quản lý tài khoản người dùng, chức năng đăng ký/đăng nhập và hệ thống tra cứu thông tin kèm bản đồ.

- Phạm vi không gian: Các di tích lịch sử trên địa bàn Tỉnh Vĩnh Long.

- Phạm vi thời gian: Đề tài được triển khai và hoàn thiện trong vòng 8 tuần, từ ngày 03/11/2025 đến ngày 28/12/2025,.

CHƯƠNG 1: TỔNG QUAN

1.1. Đặt vấn đề

1.1.1. Bối cảnh lịch sử và văn hóa

Việt Nam là một quốc gia có bề dày lịch sử hàng nghìn năm, với hệ thống di tích lịch sử, văn hóa phong phú và đa dạng trải dài khắp 63 tỉnh thành. Thống kê của Bộ Văn hóa, Thể thao và Du lịch, cả nước hiện có hơn 40.000 di tích lịch sử văn hóa được xếp hạng, trong đó có nhiều di tích mang tầm quan trọng quốc gia và quốc tế.

Vùng đồng bằng sông Cửu Long, với vị trí địa lý đặc biệt tại cửa ngõ phía Nam của đất nước, không chỉ là vùng đất màu mỡ về nông nghiệp mà còn là nơi lưu giữ những giá trị văn hóa lịch sử độc đáo. Trong đó, tỉnh Vĩnh Long (sau khi sáp nhập tỉnh 2025 gồm Trà Vinh, Bến Tre và Vĩnh Long) địa phương tiêu biểu với hệ thống di tích đa dạng, phản ánh quá trình hình thành và phát triển của vùng đất Nam Bộ.[1]

1.2. Mục tiêu đề án

1.2.1. Mục tiêu tổng quát

Xây dựng một hệ thống website tra cứu di tích lịch sử cho hai tỉnh Vĩnh Long và Trà Vinh, nhằm số hóa thông tin di tích, tạo ra một kênh thông tin chính thức, đầy đủ và dễ tiếp cận cho người dân, du khách và các nhà nghiên cứu.

1.2.2. Mục tiêu cụ thể

Về mặt kỹ thuật:

Thiết kế và phát triển website responsive, tương thích với mọi thiết bị (desktop, tablet, mobile), xây dựng cơ sở dữ liệu hoàn chỉnh chứa thông tin chi tiết về 48 di tích tiêu biểu, tích hợp hệ thống tìm kiếm thông minh với nhiều tiêu chí lọc, phát triển giao diện người dùng thân thiện, trực quan và dễ sử dụng.

Về mặt chức năng:

Cung cấp thông tin đầy đủ về từng di tích: tên gọi, địa chỉ, lịch sử, đặc điểm kiến trúc

giá trị văn hóa, hiển thị hình ảnh chất lượng cao của các di tích, tích hợp thông tin vị trí GPS và liên kết với các ứng dụng bản đồ phổ biến, xây dựng hệ thống quản lý người dùng với chức năng đăng ký, đăng nhập.

Về mặt nội dung:

- Tổng hợp và biên soạn thông tin chính xác, đầy đủ về các di tích
- Phân loại di tích theo nhiều tiêu chí: loại hình, địa phương, thời gian
- Cung cấp thông tin lịch sử, văn hóa phong phú và hấp dẫn
- Hướng dẫn du lịch và thông tin thực tế cho du khách

1.2.3. Đối tượng hướng đến

Người dân địa phương

Du khách trong và ngoài nước

Nhà nghiên cứu, sinh viên

Cơ quan quản lý

1.3. Tổng quát về cấu trúc đồ án

Đồ án được tổ chức thành 10 chương chính, từ việc phân tích yêu cầu, thiết kế hệ thống, triển khai thực hiện đến kiểm thử và đánh giá kết quả. Mỗi chương đều có mục tiêu rõ ràng và nội dung cụ thể, tạo thành một quy trình phát triển phần mềm hoàn chỉnh và khoa học.

Với việc kết hợp giữa lý thuyết và thực hành, đồ án không chỉ tạo ra một sản phẩm công nghệ hữu ích mà còn thể hiện quá trình học tập, nghiên cứu và ứng dụng kiến thức một cách có hệ thống và sáng tạo.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1. Khái niệm về Front-end và Back-end

2.1.1. Khái niệm Front-end

Front end là một thuật ngữ phổ biến được sử dụng trong phát triển web. Thuật ngữ này được dùng để chỉ phần giao diện mà người dùng trực tiếp tương tác khi sử dụng một ứng dụng web hoặc website. Chẳng hạn màu sắc và kiểu văn bản, hình ảnh, đồ thị và bảng, nút, menu điều hướng [2].

Front end được thiết lập để thực hiện hai nhiệm vụ chính là tăng khả năng đáp ứng và hiệu suất hoạt động của website hoặc ứng dụng. Chính vì vậy, các lập trình viên giao diện người dùng (Front end Developer) phải đảm bảo rằng trang web hoặc ứng dụng được tối ưu hóa cho các thiết bị và kích thước màn hình khác nhau, đồng thời họ phải có khả năng khắc phục mọi sự cố phát sinh trong quá trình phát triển [3].

2.1.2. Ngôn ngữ cốt lõi:

Ngôn ngữ cốt lõi của Front-end bao gồm bộ 3 nền tảng: **HTML**, **CSS** và **Java Script**.

- **HTML là gì :**

HTML có tên đầy đủ là (Hypertext Markup Language) nghĩa là ngôn ngữ đánh dấu siêu văn bản. HTML thường được sử dụng để tạo và cấu trúc các phần trong trang web và ứng dụng, phân chia các đoạn văn, heading, link, blockquotes,...

HTML không phải là một ngôn ngữ lập trình mà chỉ là một ngôn ngữ đánh dấu. Điều này đồng nghĩa với việc HTML không thể thực hiện các chức năng “động”. Nói cách khác, HTML tương tự như phần mềm Microsoft Word, chỉ có tác dụng định dạng các thành phần có trong website.[4]

- CSS là gì :

CSS là viết tắt của (Cascading Style Sheets) nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (HTML). Nói ngắn gọn hơn là ngôn ngữ tạo phong cách cho trang web. Bạn có thể hiểu đơn giản rằng, nếu HTML đóng vai trò định dạng các phần tử trên website như việc tạo ra các đoạn văn bản, các tiêu đề, bảng,...thì CSS sẽ giúp chúng ta có thể thêm style vào các phần tử HTML đó như đổi bố cục, màu sắc trang, đổi màu chữ, font chữ, thay đổi cấu trúc.

- Java Script là gì :

JavaScript còn được viết tắt là (JS) là một ngôn ngữ lập trình, chạy dưới dạng ngôn ngữ kịch bản phía máy khách trong môi trường hosting. Môi trường hosting phổ biến nhất là trình duyệt web, dùng ngôn ngữ này để tạo tính tương tác cho trang web.

2.1.3. Khái niệm Back-end

Backend (còn được viết là Back-end hay Back end) là phần mà người dùng không thể nhìn thấy của một ứng dụng hoặc trang web nhưng đóng vai trò quan trọng trong việc đảm bảo hoạt động của ứng dụng diễn ra mượt mà.

Nói một cách đơn giản, backend là phía “máy chủ” của ứng dụng, chịu trách nhiệm xử lý dữ liệu, quản lý cơ sở dữ liệu, và đảm bảo mọi thứ hoạt động đúng cách để người dùng có thể tương tác với giao diện người dùng (frontend) một cách dễ dàng.

2.1.4. Cách thức hoạt động và ngôn ngữ của Back-end

Nhận Yêu Cầu: Backend nhận yêu cầu từ người dùng thông qua HTTP.

Xác Thực và Xử Lý: Kiểm tra thông tin đăng nhập và quyền truy cập. Sau đó, truy vấn cơ sở dữ liệu để lấy hoặc cập nhật dữ liệu.

Xử Lý Logic: Thực hiện các phép toán và quy trình theo yêu cầu của ứng dụng.

Tạo Phản Hồi: Tạo phản hồi dựa trên kết quả xử lý và gửi lại cho giao diện người dùng.

Quản Lý Tải: Sử dụng kỹ thuật như phân bổ tải và bộ nhớ đệm để xử lý nhiều yêu cầu đồng thời và tối ưu hóa hiệu suất.

2.1.5. Ngôn ngữ Back-end

Dưới đây là các ngôn ngữ chính được dùng trong [2]Back-end

Ngôn ngữ	Điểm mạnh	ứng dụng phổ biến
JavaScript (Node.js)	Full-stack, nhanh, nhiều thư viện	API, microservices
Python	Dễ học, nhiều framework	Web app, AI tích hợp
Java	Ổn định, hiệu năng cao	Doanh nghiệp, ngân hàng
C#	Tích hợp Microsoft	Ứng dụng doanh nghiệp
PHP	Dễ triển khai	CMS, web app nhỏ
Go	Hiệu năng, song song	Microservices, cloud
Ruby	Phát triển nhanh	Startup, MVP

Bảng 2.1.1: Các ngôn ngữ chính trong Back-end

2.2. So sánh cơ bản giữa Front-end và Back-end

	Front-end	Back-end
Mục tiêu chính	Xử lý giao diện người dùng (UI), những gì người dùng nhìn thấy và tương tác trực tiếp, bao gồm: Các nút, menu, hình động, biểu mẫu và thiết kế hình ảnh tổng thể.	Xử lý logic phía máy chủ, bộ máy ẩn giúp mọi thứ hoạt động. Nó liên quan đến cơ sở dữ liệu, xử lý dữ liệu, bảo mật và liên lạc với các hệ thống bên ngoài.
Trách nhiệm	Thiết kế và xây dựng giao diện người dùng bằng HTML, CSS và JavaScript (đôi khi là các framework như React, Angular, Vue.js).	Viết mã phía máy chủ bằng các ngôn ngữ như Python, Java, Node.js, PHP, v.v.
	Đảm bảo giao diện người dùng phản hồi nhanh và hoạt động trên các thiết bị khác nhau [3].	Xây dựng và duy trì cơ sở dữ liệu để lưu trữ và quản lý dữ liệu.
	Tối ưu hóa hiệu suất cho thời gian tải nhanh.	Phát triển API để liên lạc giữa giao diện người dùng và các hệ thống khác.
Kỹ năng & Công cụ	HTML, CSS, JavaScript, nguyên tắc thiết kế, kiến thức UI/UX, framework front-end, công cụ gỡ lỗi.	Ngôn ngữ lập trình, quản lý cơ sở dữ liệu, quản trị máy chủ, các biện pháp bảo mật tốt nhất, phát triển API, sử dụng hệ thống kiểm soát phiên bản Git.

Bảng 2.2.1: So sánh cơ bản giữa Front-end và Back-end

2.3. Mối quan hệ hỗ trợ

Mối quan hệ hỗ trợ giữa Front-end và Back-end là mối quan hệ phụ thuộc hai chiều, cùng phối hợp để tạo nên một hệ thống web/app hoàn chỉnh, ổn.

2.4. Vai trò cốt lõi

Front-end: Hiển thị giao diện người dùng (UI/UX), xử lý tương tác như click, nhập liệu, validation cơ bản; đồng thời gửi và nhận dữ liệu với Back-end qua API. Công nghệ phổ biến: HTML, CSS, JavaScript, React, Vue, Angular...

Back-end: Đảm nhiệm xử lý logic nghiệp vụ, quản lý cơ sở dữ liệu, xác thực và phân quyền người dùng, cung cấp API cho Front-end. Công nghệ phổ biến: PHP, Java, Node.js, Python, .NET, MySQL, PostgreSQL...

2.5. Front-end hỗ trợ Back-end

Front-end giúp giảm tải cho Back-end bằng cách thực hiện kiểm tra dữ liệu đầu vào (độ dài, định dạng), chuẩn hóa dữ liệu gửi lên theo đúng cấu trúc API, phản hồi lỗi rõ ràng để Back-end dễ debug, đồng thời tối ưu trải nghiệm người dùng qua loading và cache phía client. Nhờ có Front-end tốt, Back-end ít lỗi hơn và hiệu năng hệ thống cao hơn.

2.6. Back-end hỗ trợ Front-end

Back-end hỗ trợ Front-end bằng việc cung cấp API rõ ràng, ổn định, trả dữ liệu đúng – đủ – nhất quán. Ngoài ra, Back-end đảm bảo bảo mật (JWT, session, OAuth...) và xử lý các logic phức tạp, giúp Front-end nhẹ hơn. Nhờ Back-end tốt, Front-end dễ phát triển, dễ mở rộng và mang lại trải nghiệm mượt mà cho người dùng.

2.7. My SQL là gì

2.7.1. khái niệm cơ bản

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) mã nguồn mở, được sử dụng rộng rãi để lưu trữ, truy xuất và quản lý dữ liệu trong các website, ứng dụng và hệ thống phần mềm. MySQL hoạt động theo mô hình client–server, trong đó máy khách gửi yêu cầu và máy chủ MySQL xử lý, trả về kết quả tương ứng.

2.7.2. Các phần của SQL

SQL chứa ba phần:

Định nghĩa dữ liệu bao gồm các câu lệnh giúp bạn xác định cơ sở dữ liệu và các đối tượng của nó, ví dụ: table, view, trigger, stored procedures, v.v.

Thao tác dữ liệu chứa các câu lệnh cho phép bạn cập nhật và truy vấn dữ liệu.

Điều khiển dữ liệu cho phép bạn cấp quyền cho người dùng truy cập dữ liệu cụ thể trong cơ sở dữ liệu.

2.7.3. Cách thức hoạt động

MySQL tạo cơ sở dữ liệu để lưu trữ và thao tác dữ liệu, xác định mối quan hệ của từng bảng. Client có thể đưa ra yêu cầu bằng cách nhập các câu lệnh SQL cụ thể trên MySQL. Server application sẽ phản hồi với thông tin được yêu cầu và nó sẽ xuất hiện ở phía máy khách.

2.7.4. Ưu và nhược điểm MySQL

Ưu điểm MySQL:

An toàn: Vì MySQL sở hữu nhiều tính năng bảo mật cấp cao, mã hóa thông tin đăng nhập và chứng thực từ host đều khả dụng.

Dễ sử dụng: MySQL ổn định và dễ sử dụng trên nhiều hệ điều hành và cung cấp một hệ thống các hàm tiện ích lớn.

Khả năng mở rộng: Với MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa có thể mở rộng khi cần thiết.

Hiệu năng cao: Hỗ trợ nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).

Nhược điểm MySQL:

Giới hạn: Nó vẫn bị hạn chế về một số chức năng cần thiết.

Dung lượng hạn chế: Nếu số bản ghi càng lớn thì việc truy xuất dữ liệu sẽ phức tạp và khó khăn hơn. Bạn cần phải áp dụng nhiều thủ thuật để nâng cấp tốc độ truy xuất dữ liệu lên.

2.7.5. Một số thuật ngữ thường gặp:

Database (còn gọi là cơ sở dữ liệu): Là một tập hợp có tổ chức của dữ liệu có cấu trúc, thường được lưu trữ điện tử trong hệ thống máy tính. Mỗi cơ sở dữ liệu được quản lý bởi một hệ quản trị cơ sở dữ liệu (DBMS). Trong MySQL, database chứa nhiều bảng quan hệ dùng để lưu trữ thông tin như người dùng, sản phẩm, giao dịch,...

Open Source: Thuật ngữ chỉ phần mềm có mã nguồn công khai, cho phép bất kỳ ai xem, sửa đổi và phân phối lại. MySQL là một phần mềm mã nguồn mở, nghĩa là người dùng có thể sử dụng miễn phí và tùy biến theo nhu cầu.

Mô hình Client – Server: Là kiến trúc phổ biến trong mạng máy tính, trong đó Client gửi yêu cầu và Server phản hồi dữ liệu. MySQL hoạt động theo mô hình này: ứng dụng (client) gửi truy vấn đến máy chủ MySQL (server), server xử lý và trả kết quả về cho client. Ví dụ điển hình: truy cập cơ sở dữ liệu của website, ứng dụng email, trình duyệt web.

2.8. PHP là gì

2.8.1. Khái niệm PHP

PHP là viết tắt của từ Hypertext Preprocessor. Đây là một ngôn ngữ lập trình kịch bản (scripting language) mã nguồn mở và là tập hợp con của các ngôn ngữ script như JavaScript và Python. Ngôn ngữ này chủ yếu được dùng để phát triển các ứng dụng từ phía server, bên cạnh đó còn được sử dụng cho frontend và backend.

PHP rất thích hợp với web và có thể nhúng vào trong trang HTML bằng cách sử dụng cặp thẻ `<?php ?>`. Các tệp lệnh PHP chỉ có thể được diễn giải trên một máy chủ đã cài PHP [5].

2.8.2. Ứng dụng PHP trong lập trình

Các ứng dụng phổ biến của PHP trong lĩnh vực công nghệ thông tin bao gồm:

Xây dựng và cấu hình hệ thống máy chủ: Đây là một trong những ứng dụng chính của PHP. Các lập trình viên PHP thường phải thực hiện nhiệm vụ như phân tích mã nguồn PHP, triển khai máy chủ web và trình duyệt web.

Phát triển các dòng lệnh: PHP Developers sẽ tạo ra các tệp lệnh để thực thi mã PHP mà không cần máy chủ. Phương pháp này thường được sử dụng trên nền tảng hệ điều hành phổ biến như Linux hoặc Windows.

Xây dựng ứng dụng: PHP có thể được áp dụng để phát triển các ứng dụng phần mềm bằng cách tận dụng các tính năng mạnh mẽ sẵn có. PHP-GTK thường được lựa chọn làm nền tảng phát triển phần mềm vì đó là một phần mở rộng của PHP không được tích hợp sẵn trong các bản phân phối chính thức hiện nay.

Hỗ trợ cho các loại cơ sở dữ liệu khác nhau: Một website tích hợp cơ sở dữ liệu tốt sẽ giúp quản lý dữ liệu, sao lưu và đặc biệt là tạo bản sao lưu dữ liệu để đối phó với các vấn đề an ninh mạng có thể xảy ra.

2.8.3. Ưu và nhược điểm của PHP

Ưu điểm

PHP là một ngôn ngữ lập trình mã nguồn mở, miễn phí và dễ học, giúp tiết kiệm chi phí cho cá nhân lẫn doanh nghiệp. Nó linh hoạt, chạy tốt trên nhiều hệ điều hành và dễ dàng tích hợp với các ngôn ngữ khác.

PHP sở hữu cộng đồng lớn cùng hệ sinh thái thư viện phong phú, hỗ trợ lập trình viên nhanh chóng tìm giải pháp khi gặp vấn đề. Bên cạnh đó, PHP kết nối hiệu quả với nhiều hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL hay SQLite, đồng thời được tối ưu cho phát triển web backend với khả năng xử lý request nhanh và thao tác tiện lợi với form, session, cookie.

PHP còn là nền tảng của nhiều CMS phổ biến như WordPress, Joomla, Drupal, khiến nó trở thành kỹ năng bắt buộc đối với những ai làm việc trong lĩnh vực xây dựng website.

Nhược điểm

PHP tuy có nhiều ưu điểm nhưng cũng tồn tại một số hạn chế. Do là ngôn ngữ mã nguồn mở, các lỗ hổng bảo mật thường bị công khai sớm, dễ bị kẻ xấu khai thác nếu hệ thống chưa kịp cập nhật.

Về giao diện và hiệu suất, PHP chủ yếu xử lý logic phía server nên cần kết hợp với HTML, CSS, JavaScript để hoàn thiện phần hiển thị, đồng thời có thể chạy chậm hơn các ngôn ngữ hiện đại nếu không được tối ưu tốt.

PHP dễ học nhưng khó chuẩn hóa, khiến nhiều dự án trở nên rối rắm và khó bảo trì khi thiếu quy tắc lập trình rõ ràng. Đặc biệt, PHP chưa phải lựa chọn tối ưu cho các ứng dụng thời gian thực như chat trực tuyến hay game online, nơi Node.js thường mang lại hiệu quả cao hơn.

.

2.9. UX/UI là gì

2.9.1. khái niệm UX/UI

UI (User Interface) là giao diện người dùng, tập hợp các yếu tố đồ họa và tương tác giữa người dùng với sản phẩm số (trang web, ứng dụng,...). Mục đích của giao diện là thu hút sự chú ý của người xem và làm cho giao diện trực quan bằng cách thiết kế màu sắc, kiểu chữ, bố cục,... đồng thời xây dựng hình ảnh thương hiệu.

UX (User Experience) là trải nghiệm người dùng, bao gồm toàn bộ quá trình mà người dùng trải qua khi sử dụng sản phẩm số, từ những tương tác qua các chức năng, hiệu ứng, điều hướng, màu sắc,... nhằm đảm bảo sản phẩm số thân thiện, dễ dàng điều hướng, đáp ứng kỳ vọng của người dùng [3].

2.9.2. Đặc điểm và vai trò

Đặc điểm

UX (User Experience - Trải nghiệm): Là "Cảm giác". Tập trung vào sự tiện lợi, logic, tốc độ và cách bố trí tính năng sao cho dễ dùng nhất.

UI (User Interface - Giao diện): Là "Cái nhìn". Tập trung vào màu sắc, hình ảnh, phông chữ và sự bắt mắt để tạo ấn tượng thị giác.

Vai trò

Với người dùng: Giúp họ dùng ứng dụng nhanh hơn, dễ hơn và cảm thấy hài lòng, không bị ức chế.

Với doanh nghiệp: Giúp bán được hàng, giữ chân khách hàng và giảm chi phí sửa lỗi hoặc hỗ trợ kỹ thuật.

2.9.3. Tầm quan trọng và ảnh hưởng đến người dùng

Tầm quan trọng

Trong thực tế, UX/UI đóng vai trò là xương sống quyết định sự thành bại của một sản phẩm số, bởi đây là điểm chạm duy nhất giữa công nghệ và con người. Tầm quan trọng của nó nằm ở việc chuyển hóa các mã nguồn phức tạp thành một

hành trình trải nghiệm đơn giản, giúp người dùng đạt được mục đích một cách nhanh chóng nhất

Ảnh hưởng đến người dùng

UX/UI tác động trực tiếp đến cả tâm lý lẫn hành vi thường nhật. Khi tương tác với một giao diện mượt mà và logic, người dùng nảy sinh cảm xúc tích cực, cảm thấy hài lòng và gắn bó lâu dài với sản phẩm. Ngược lại, một thiết kế tồi sẽ gây ra sự ức chế, mệt mỏi và khiến họ ngay lập tức rời bỏ ứng dụng để tìm kiếm lựa chọn thay thế. Xa hơn, UX/UI còn góp phần định hình thói quen sử dụng công nghệ của xã hội, biến những thao tác phức tạp trở thành bản năng, giúp cuộc sống hiện đại trở nên tiện lợi và tiết kiệm thời gian hơn.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1. Quy trình nghiên cứu và phát triển

3.1.1. Khảo sát và phân tích yêu cầu

Trong giai đoạn đầu, nghiên cứu tiến hành khảo sát thực địa tại các tỉnh Vĩnh Long, Trà Vinh và Bến Tre nhằm tìm hiểu về các di tích lịch sử tiêu biểu. Song song với đó, việc phỏng vấn người dùng được thực hiện để nắm bắt nhu cầu tra cứu thông tin di tích của du khách và người dân địa phương. Ngoài ra, cũng phân tích các hệ thống hiện có bằng cách nghiên cứu các website du lịch và các công cụ tra cứu di tích đang tồn tại. Từ kết quả khảo sát và phân tích, nhóm xác định các yêu cầu chức năng cần thiết cho hệ thống, làm cơ sở cho các bước nghiên cứu tiếp theo.

3.1.2. Nghiên cứu công nghệ

Lựa chọn và nghiên cứu các công nghệ phù hợp để xây dựng hệ thống. Về Front-end, sử dụng các công nghệ hiện đại như: HTML5, CSS3 và JavaScript ES6+. Về Back-end, hệ thống được phát triển trên nền tảng PHP 7.4+ kết hợp với cơ sở dữ liệu MySQL 8.0. Công cụ phát triển chính bao gồm: Visual Studio Code, XAMPP và phpMyAdmin, đảm bảo môi trường lập trình thuận tiện và hiệu quả.

3.1.3. Thiết kế hệ thống

Thiết kế hệ thống bao gồm ba phần chính:

Thứ nhất là thiết kế cơ sở dữ liệu với mô hình ERD và schema database để quản lý dữ liệu một cách khoa học.

Thứ hai là thiết kế giao diện thông qua các bản wireframe, mockup và prototype nhằm hình dung rõ ràng trải nghiệm người dùng.

Cuối cùng là thiết kế kiến trúc hệ thống theo mô hình MVC, đảm bảo luồng xử lý dữ liệu hợp lý và dễ dàng mở rộng trong tương lai.

3.1.4. Phát triển và kiểm thử

Quá trình phát triển được tiến hành theo từng module, bắt đầu từ Front-end, sau đó đến Back-end và cuối cùng là tích hợp toàn hệ thống. Mỗi chức năng đều được kiểm thử kỹ lưỡng thông qua unit test và integration test để đảm bảo tính ổn định. Sau khi hoàn thiện, hệ thống được tối ưu hóa cả về hiệu suất cơ sở dữ liệu lẫn giao diện frontend, nhằm mang lại trải nghiệm tốt nhất cho người dùng.

3.2. Phân tích đặc điểm và yêu cầu hệ thống

3.2.1. Yêu cầu chức năng

Quản lý thông tin di tích

Hệ thống phải cho phép quản lý và hiển thị thông tin về các di tích. Người dùng có thể xem danh sách di tích được phân loại theo từng tỉnh hoặc thành phố. Khi chọn một di tích cụ thể, hệ thống sẽ hiển thị đầy đủ thông tin chi tiết bao gồm tên, mô tả, hình ảnh, địa chỉ và loại hình. Ngoài ra, hệ thống hỗ trợ chức năng tìm kiếm di tích theo từ khóa để người dùng dễ dàng tra cứu. Bên cạnh đó, người dùng có thể lọc danh sách di tích theo loại như tôn giáo, lịch sử, kiến trúc hoặc văn hóa nhằm tối ưu trải nghiệm tìm kiếm.

Quản lý người dùng

Hệ thống cần cung cấp chức năng quản lý người dùng. Người dùng mới có thể đăng ký tài khoản bằng cách nhập thông tin cơ bản như họ tên, email và mật khẩu. Sau khi đăng ký, họ có thể đăng nhập để sử dụng các tính năng của hệ thống và đăng xuất khi không còn nhu cầu. Đồng thời, hệ thống phải quản lý phiên làm việc (session) để đảm bảo tính bảo mật và duy trì trải nghiệm liên tục cho người dùng trong suốt quá trình sử dụng.

Giao diện và trải nghiệm người dùng

Hệ thống phải được thiết kế với giao diện thân thiện và tối ưu cho mọi thiết bị, đảm bảo tính responsive trên máy tính, máy tính bảng và điện thoại. Khi hiển thị hình ảnh di tích, hệ thống cần tích hợp chức năng lightbox để người dùng có thể xem ảnh chi tiết một cách trực quan. Ngoài ra, hệ thống phải hỗ trợ tích hợp bản đồ như Google Maps hoặc Bing Maps để hiển thị vị trí chính xác của các di tích. Cuối cùng, giao diện cần có navigation menu và breadcrumb nhằm giúp người dùng dễ dàng điều hướng và theo dõi vị trí hiện tại trong hệ thống.

3.2.2. Yêu cầu phi chức năng

Hiệu suất (Performance)

Hệ thống phải đảm bảo hiệu suất cao trong quá trình vận hành. Thời gian tải trang cần được tối ưu để luôn dưới 3 giây nhằm mang lại trải nghiệm mượt mà cho người dùng. Ngoài ra, hệ thống phải hỗ trợ đồng thời trên 100 người dùng truy cập mà vẫn duy trì tính ổn định. Để đạt được điều này, việc tối ưu hóa hình ảnh và các tài nguyên tĩnh là bắt buộc, giúp giảm dung lượng tải và tăng tốc độ phản hồi.

Bảo mật (Security)

Bảo mật là yếu tố quan trọng hàng đầu trong hệ thống. Mật khẩu người dùng phải được mã hóa bằng thuật toán **bcrypt** để đảm bảo an toàn dữ liệu. Hệ thống cần có cơ chế bảo vệ chống lại các cuộc tấn công **SQL Injection**, đồng thời thực hiện kiểm tra và xác thực dữ liệu đầu vào nhằm ngăn chặn các hành vi khai thác lỗ hổng bảo mật.

Khả năng sử dụng (Usability)

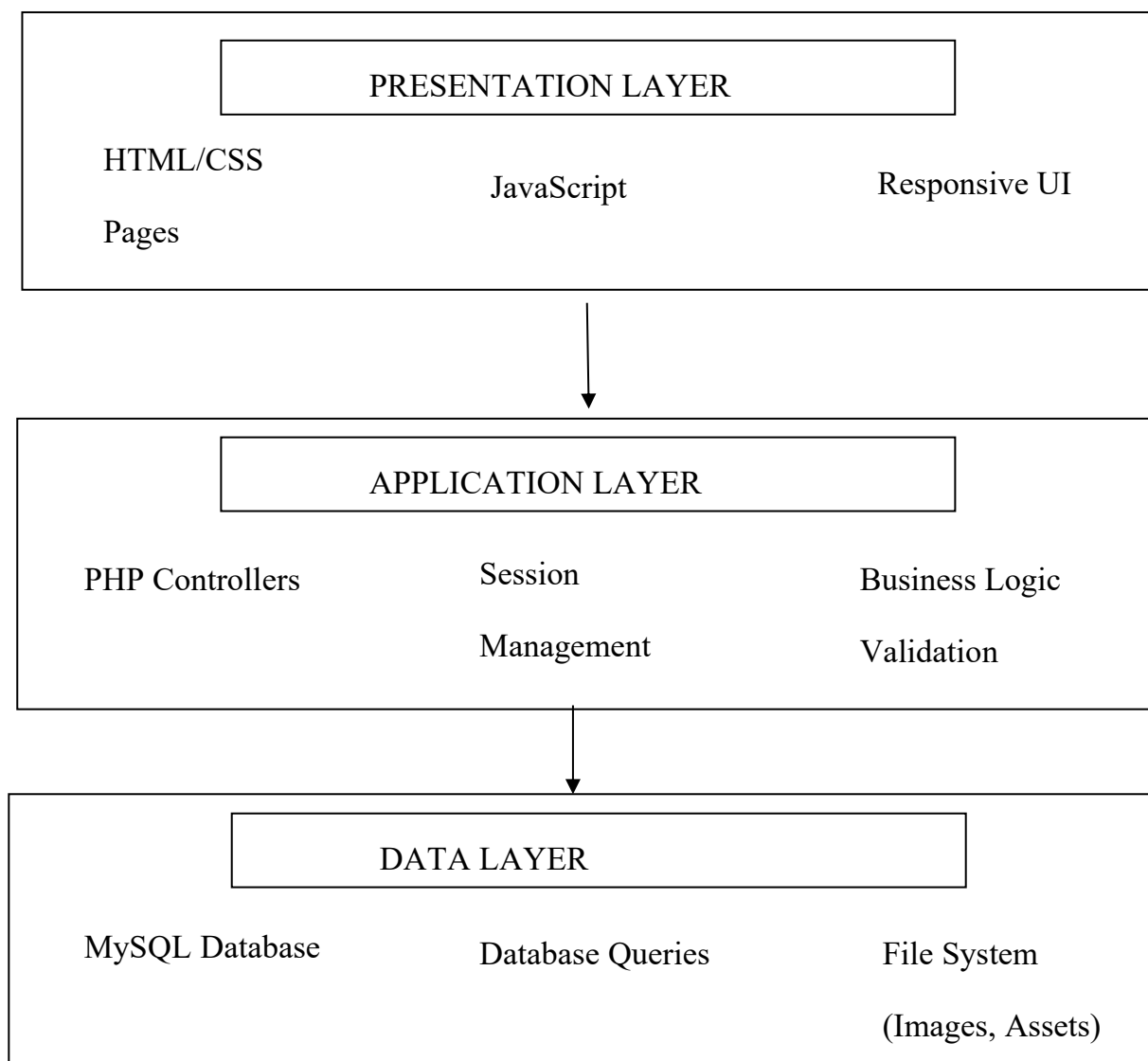
Hệ thống phải được thiết kế với giao diện trực quan, dễ sử dụng, giúp người dùng thao tác thuận tiện ngay cả khi mới tiếp cận. Toàn bộ giao diện và nội dung cần hỗ trợ tiếng Việt hoàn chỉnh để phù hợp với đối tượng người dùng trong nước.

Ngoài ra, hệ thống phải tương thích với các trình duyệt phổ biến như Chrome, Firefox, Edge và Safari, đảm bảo tính nhất quán trong trải nghiệm.

Khả năng mở rộng (Scalability)

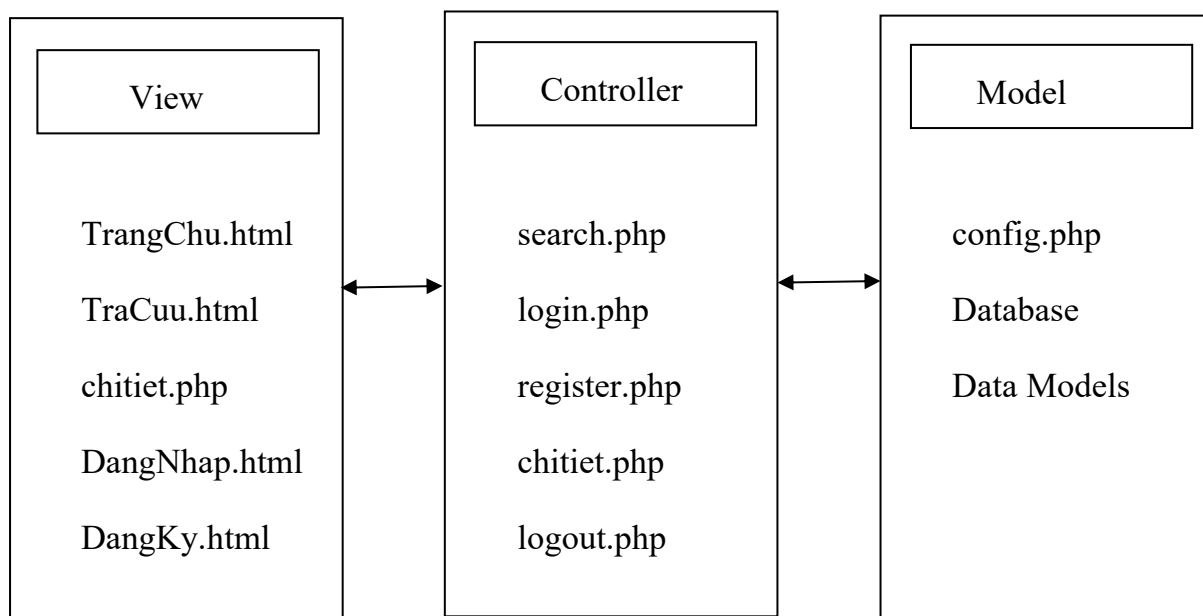
Hệ thống cần có khả năng mở rộng để đáp ứng nhu cầu phát triển trong tương lai. Cơ sở dữ liệu phải được thiết kế theo hướng mở rộng, cho phép lưu trữ và xử lý khối lượng dữ liệu ngày càng lớn. Bên cạnh đó, cấu trúc mã nguồn phải rõ ràng, dễ bảo trì và phát triển, giúp đội ngũ kỹ thuật có thể nâng cấp hoặc bổ sung tính năng mới một cách thuận lợi.

3.3. Thiết kế hệ thống kiến trúc



Bảng 3.3.1 Sơ đồ thiết kế hệ thống kiến trúc

3.4. Mô hình MVC (Model-View-Controller)



Bảng 3.4.1 Sơ đồ mô hình MVC

3.5. Luồng xử lý dữ liệu

Luồng tìm kiếm di tích:

User Input → TraCuu.html → JavaScript → search.php → MySQL → JSON
Response → Display Results

Luồng đăng nhập:

Login Form → DangNhap.html → login.php → Validate → Session → Redirect

Luồng xem chi tiết:

Click Detail → chitiet.php?id=X → Query Database → Render HTML → Display

3.6. Mô hình ERD (Entity Relationship Diagram)

BẢNG USERS

USERS
id (PK, INT, AUTO_INCREMENT)
username (VARCHAR(50), UNIQUE)
email (VARCHAR(100), UNIQUE)
password (VARCHAR(255))
created_at (TIMESTAMP)

Bảng 3.6.1 Bảng Users

BẢNG DI TÍCH

DITICH
id (PK, INT, AUTO_INCREMENT)
ten_ditich (VARCHAR(255))
dia_chi (TEXT)
mo_ta (TEXT)
loai_ditich (VARCHAR(100))
nam_xay_dung (VARCHAR(50))
tinh (VARCHAR(50))
hinh_anh (VARCHAR(255))
latitude (DECIMAL(10,8))
longitude (DECIMAL(11,8))
created_at (TIMESTAMP)

Bảng 3.6.2 Bảng di tích

3.6.1. Mỗi liên hệ

Sơ đồ mỗi liên hệ



Bảng 3.6.3: sơ đồ mỗi liên hệ giữa các bảng

Ý nghĩa của việc thiết lập:

Quản lý dữ liệu: Giúp bạn biết chính xác di tích nào được tạo bởi ai và vào thời gian nào (thông qua cột `created_at` ở cả hai bảng).

Truy xuất thông tin: Bạn có thể dễ dàng viết câu lệnh để lấy ra danh sách tất cả di tích của một người dùng cụ thể hoặc hiển thị tên người đăng khi xem chi tiết một di tích.

Ràng buộc toàn vẹn: Đảm bảo không có di tích nào tồn tại mà "không có chủ" (nếu bạn thiết lập thuộc tính NOT NULL cho khóa ngoại).

3.7. Cài đặt và triển khai hệ thống

3.7.1. Yêu cầu hệ thống

Server Requirements:

Web Server: Apache 2.4+ hoặc Nginx 1.18+

PHP: Version 7.4+ (khuyến nghị 8.0+)

Database: MySQL 8.0+ hoặc MariaDB 10.4+

Storage: Tối thiểu 500MB (cho code + images)

RAM: Tối thiểu 512MB (khuyến nghị 1GB+)

Client Requirements:

Browser: Chrome 90+, Firefox 88+, Safari 14+, Edge 90+

JavaScript: Enabled

Screen Resolution: 320px+ width

3.7.2. Cấu trúc thư mục dự án



Bảng 3.7.1 Bảng Cấu trúc thư mục dự án

Tương tác Người dùng (Frontend): Các file .html, style.css và script.js phối hợp để hiển thị giao diện chuyên nghiệp (như bộ lọc tìm kiếm màu gradient hay form đăng nhập có icon).

Xử lý Dữ liệu (Backend): Các file .php đóng vai trò trung gian. Ví dụ: Khi nhấn nút **Tìm Kiếm** ở TraCuu.html, dữ liệu sẽ được gửi đến search.php để truy vấn cơ sở dữ liệu.

Lưu trữ (Database): File database.sql chứa cấu trúc **Schema** (Bảng người dùng, bảng di tích).

Quản lý Hình ảnh:

images/ (hình ảnh di tích - nội dung)

img/banner/ (hình ảnh giao diện - trang trí).

3.8. Kiểm tra hệ thống

Unit Testing:

Test các function PHP riêng lẻ

Test validation form

Test database queries

Integration Testing:

Test luồng đăng ký → đăng nhập

Test luồng tìm kiếm → xem chi tiết

Test tích hợp với bản đồ

System Testing:

Test toàn bộ hệ thống end-to-end

Test trên nhiều trình duyệt khác nhau

Test responsive trên mobile/tablet

User Acceptance Testing:

Test với người dùng thực tế

Thu thập feedback và cải thiện

Test Cases chính

Tìm kiếm di tích

Input: Keyword "Bác Hồ"

Expected: Hiện thị "Khu Tưởng Niệm Bác Hồ"

Actual: Pass

Đăng ký tài khoản

Input: Username "testuser", Email "test@email.com", Password "123456"

Expected: Tạo tài khoản thành công, redirect đến trang chủ

Actual: Pass

Xem chi tiết di tích

Input: Click vào di tích ID=1

Expected: Hiện thị thông tin chi tiết, hình ảnh, bản đồ

Actual: Pass

3.9. Tổng kết chương

Chương này đã trình bày chi tiết quá trình thiết kế và cài đặt hệ thống website tra cứu di tích lịch sử, bao gồm:

- Quy trình nghiên cứu có hệ thống từ khảo sát yêu cầu đến triển khai
- Thiết kế kiến trúc rõ ràng theo mô hình MVC và 3-layer architecture
- Database schema tối ưu với đầy đủ index và ràng buộc
- Giao diện responsive thân thiện với người dùng

-
- Quy trình cài đặt chi tiết và dễ thực hiện
 - Kiểm thử toàn diện đảm bảo chất lượng hệ thống
 - Chiến lược tối ưu hóa và bảo trì dài hạn

Hệ thống đã được phát triển thành công với tất cả các chức năng hoạt động ổn định, đáp ứng đầy đủ yêu cầu đề ra và sẵn sàng cho việc triển khai thực tế.

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

4.1. Tổng quan về dự án

4.1.1. Tổng quan

Đồ án được tổ chức thành 10 chương chính, từ việc phân tích yêu cầu, thiết kế hệ thống, triển khai thực hiện đến kiểm thử và đánh giá kết quả. Mỗi chương đều có mục tiêu rõ ràng và nội dung cụ thể, tạo thành một quy trình phát triển phần mềm hoàn chỉnh và khoa học.

Với việc kết hợp giữa lý thuyết và thực hành, đồ án không chỉ tạo ra một sản phẩm công nghệ hữu ích mà còn thể hiện quá trình học tập, nghiên cứu và ứng dụng kiến thức một cách có hệ thống và sáng tạo.

4.1.2. Chức năng

Đồ án được tổ chức thành 10 chương chính, từ việc phân tích yêu cầu, thiết kế hệ thống, triển khai thực hiện đến kiểm thử và đánh giá kết quả. Mỗi chương đều có mục tiêu rõ ràng và nội dung cụ thể, tạo thành một quy trình phát triển phần mềm hoàn chỉnh và khoa học.

Với việc kết hợp giữa lý thuyết và thực hành, đồ án không chỉ tạo ra một sản phẩm công nghệ hữu ích mà còn thể hiện quá trình học tập, nghiên cứu và ứng dụng kiến thức một cách có hệ thống và sáng tạo.

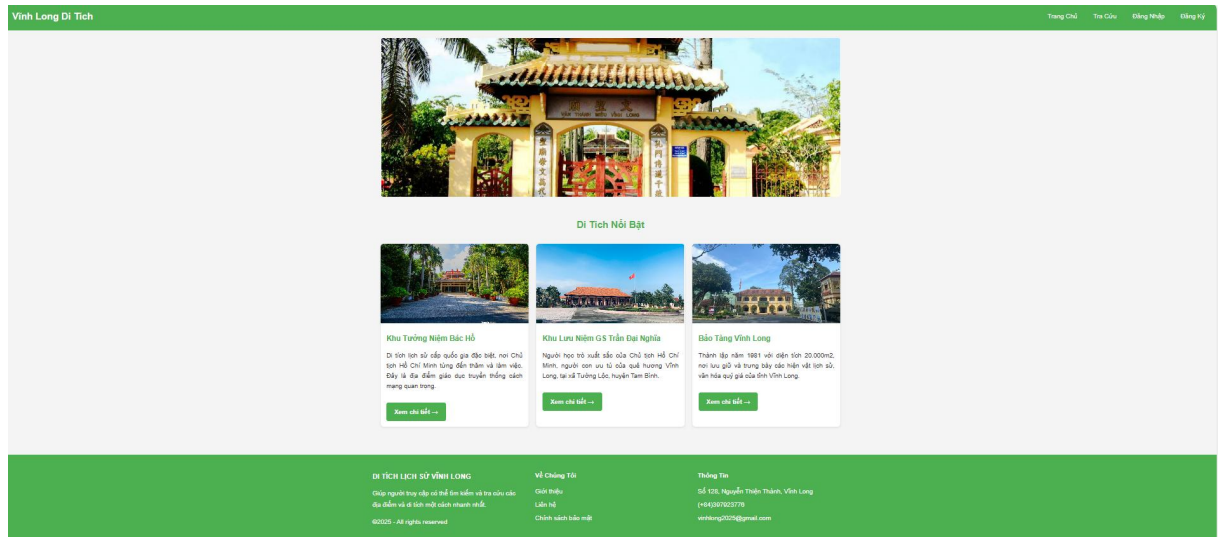
Hệ thống “website giúp người dùng tra cứu thông tin về các di tích lịch sử trên địa bàn Tỉnh Vĩnh Long” đã được tích hợp các chức năng: trang chủ đăng nhập, đăng ký, tra cứu.

Lưu thông tin đăng nhập người dùng trên cơ sở dữ liệu thông qua truy vấn.

Tích hợp gg map vào website giúp người dùng có thể tra cứu nhanh chóng và đúng nhất các địa điểm, di tích lịch sử.

4.2. Phần trang chủ của trang web

4.2.1. Tổng quan về trang chủ



Hình 4.2.1 Hình ảnh trang chủ

Bố cục và Cấu trúc (Layout):

Header (Thanh điều hướng): Có logo/tên trang web "Vĩnh Long Di Tích" và các menu chức năng: Trang Chủ, Tra Cứu, Đăng Nhập, Đăng Ký. Bố cục này chuẩn và dễ điều hướng.

Banner/Hero Image: Sử dụng hình ảnh Văn Thánh Miếu Vĩnh Long rất đặc trưng, khổ rộng, tạo ấn tượng thị giác tốt ngay khi vào trang.

Phần nội dung chính (Di Tích Nổi Bật): Sử dụng dạng thẻ (Cards) để hiển thị các địa danh. Đây là cách trình bày hiện đại, sạch sẽ và dễ nhìn trên cả máy tính lẫn điện thoại.

Footer (Chân trang): Đầy đủ thông tin về bản quyền, giới thiệu ngắn, các liên kết phụ và thông tin liên hệ (địa chỉ, số điện thoại, email).

Nội dung (Content):

Đã liệt kê được các địa điểm tiêu biểu của Vĩnh Long như: Khu Tưởng Niệm Bác Hồ, Khu Lưu Niệm GS Trần Đại Nghĩa, và Bảo Tàng Vĩnh Long.

Mỗi thẻ di tích đều có: Hình ảnh minh họa, tiêu đề, mô tả ngắn gọn và nút "Xem chi tiết" để dẫn dắt người dùng.

4.2.2. Phần Header



Hình 4.2.2 Hình ảnh Header

Tên thương hiệu/Logo: Chữ "Vĩnh Long Di Tích" được đặt ở phía bên trái, sử dụng phông chữ không chân (sans-serif), màu trắng và in đậm để làm nổi bật tên trang web.

Menu điều hướng: Nằm ở phía bên phải, giúp người dùng chuyển đổi giữa các trang khác nhau, bao gồm:

Trang Chủ: Liên kết về trang chính.

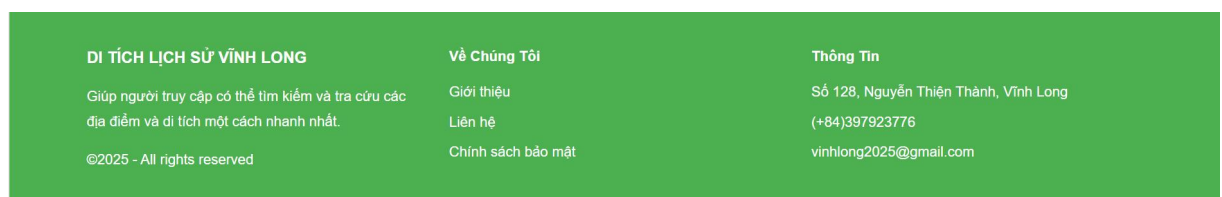
Tra Cứu: Chức năng tìm kiếm thông tin về các di tích.

Đăng Nhập: Dành cho người dùng đã có tài khoản.

Đăng Ký: Dành cho người dùng mới tạo tài khoản.

Màu sắc chủ đạo: Toàn bộ thanh Header sử dụng nền màu xanh lá cây, tạo cảm giác thân thiện và phù hợp với chủ đề về văn hóa, lịch sử.

4.2.3. Phần Footer



Hình 4.2.3 Hình ảnh Footer

4.2.4. Banner chính



Hình 4.2.4 Hình ảnh Banner Thứ 1



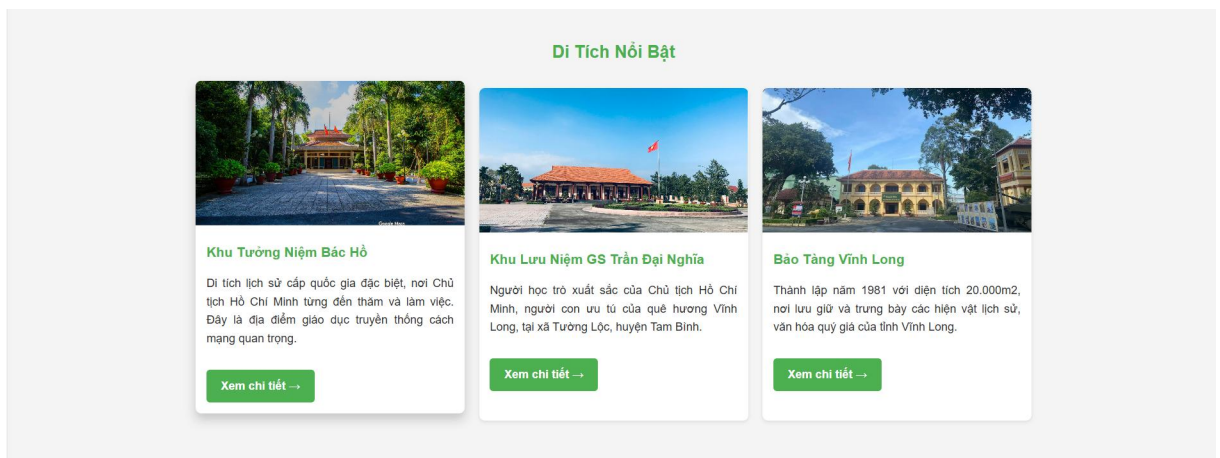
Hình 4.2.5 Hình ảnh Banner Thứ 2



Hình 4.2.6 Hình ảnh Banner Thứ 2

Vị trí: Banner được đặt trang trọng ngay phía trên phần nội dung chính và bên dưới thanh header, đóng vai trò là điểm nhấn thu hút thị giác đầu tiên khi người dùng truy cập vào trang. Kết hợp cùng với hiệu ứng chuyển động qua lại giữa các ảnh giúp người dùng khi vừa truy cập vào website hướng thẳng vào mắt họ là hình ảnh sống động và đỡ nhàm chán.

4.2.5. Phần thẻ card



Hình 4.2.6 Hình ảnh phần thẻ Card

Cấu trúc của mỗi Card

Mỗi thẻ di tích được thiết kế đồng nhất với các thành phần:

Hình ảnh minh họa: Nằm ở trên cùng, hiển thị phong cảnh đặc trưng của từng di tích.

Tiêu đề di tích: Sử dụng font chữ đậm, màu xanh lá cây nổi bật.

Đoạn mô tả ngắn: Tóm tắt các thông tin quan trọng nhất về di tích (như ý nghĩa lịch sử, diện tích, hoặc vị trí địa lý).

Nút hành động (Call to Action): Nút "Xem chi tiết →" màu xanh lá cây với chữ trắng, giúp người dùng chuyển hướng sâu hơn vào thông tin cụ thể.

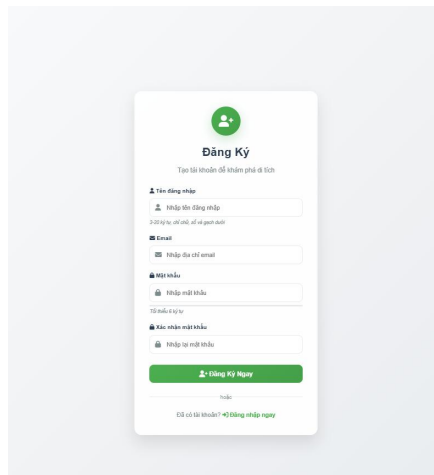
Đặc điểm thiết kế

Hiệu ứng thị giác: Các thẻ có đổ bóng nhẹ (drop shadow) và bo góc, tạo cảm giác hiện đại và tách biệt rõ ràng với nền trang web.

Bố cục: Được sắp xếp theo dạng lưới (grid) nằm ngang, giúp người dùng dễ dàng so sánh và lựa chọn thông tin khi lướt qua.

4.3. Tổng quan về phần đăng ký

4.3.1. Giao diện chính đăng ký



Hình 4.3.1 Hình ảnh trang Đăng ký

Về thiết kế

Giao diện đồng bộ: Bạn duy trì cùng một phong cách thiết kế (form trắng, nền xám nhạt, đổ bóng nhẹ) và bảng màu xanh lá cây, giúp người dùng cảm thấy sự liên mạch giữa hai tính năng.

Biểu tượng (Icon): Biểu tượng ở đầu form đã được thay đổi thành icon "Thêm người dùng", phù hợp với mục đích tạo tài khoản mới.

Các chức năng

Mở rộng các trường thông tin: Ngoài Tên đăng nhập và Mật khẩu, bạn đã thêm trường Email (có icon thư) để hỗ trợ xác thực và liên lạc.

Xác nhận mật khẩu: Thêm ô "Xác nhận mật khẩu" giúp giảm thiểu sai sót khi người dùng thiết lập mật khẩu lần đầu.

Hướng dẫn người dùng (Validation Hints): thêm các dòng chú thích nhỏ bên dưới các ô nhập liệu:

Chú thích cho Tên đăng nhập: "3-20 ký tự, chỉ chữ, số và gạch dưới".

Chú thích cho Mật khẩu: "Tối thiểu 6 ký tự".

Nút hành động chính: Nút "Đăng Ký Ngay" nổi bật, rõ ràng.

Điều hướng ngược: Dòng "Đã có tài khoản? Đăng nhập ngay" giúp người dùng dễ dàng quay lại nếu họ đi nhầm trang.

4.3.2. Hướng dẫn đăng ký tài khoản

Bước 1: Truy cập trang đăng ký

Mở website và click vào nút "Đăng Ký" trên thanh menu

Bước 2: Điền thông tin đăng ký

Nhập đầy đủ các thông tin sau:

-Tên đăng nhập

Từ 3-20 ký tự

Chỉ được sử dụng chữ cái, số và dấu gạch dưới (_)

Ví dụ: `nguyen_van_a`, `user123`

- Email

Địa chỉ email hợp lệ

Ví dụ: `example@gmail.com`

- Mật khẩu

Tối thiểu 6 ký tự

Nên kết hợp chữ hoa, chữ thường, số và ký tự đặc biệt

Hệ thống sẽ hiển thị độ mạnh mật khẩu

Xác nhận mật khẩu

Nhập lại mật khẩu để xác nhận

Bước 3: Hoàn tất đăng ký

Click nút "Đăng Ký Ngay"

Hệ thống sẽ kiểm tra thông tin và thông báo kết quả

Nếu thành công, bạn sẽ được chuyển về trang chủ

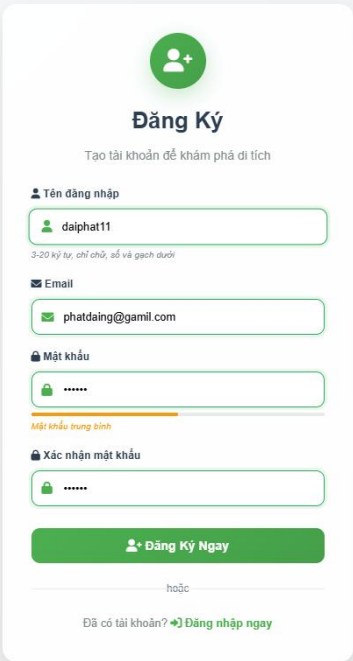
Lưu ý khi đăng ký:

Tên đăng nhập phải duy nhất (chưa có ai sử dụng)

Email phải hợp lệ và chưa được đăng ký

Mật khẩu và xác nhận mật khẩu phải giống nhau

4.3.3. Cách thức hoạt động

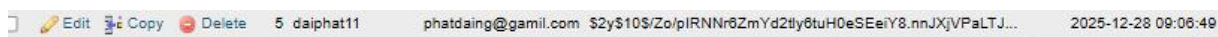


The screenshot shows a registration form titled "Đăng Ký" (Register) with the subtitle "Tạo tài khoản để khám phá di tích" (Create an account to explore the relic). The form contains the following fields and elements:

- Tên đăng nhập** (Username): A text input field containing "daiphat11". Below the field is a small note: "3-20 ký tự, chỉ chữ, số và gạch dưới" (3-20 characters, only letters, numbers, and underscores).
- Email**: A text input field containing "phatdaing@gmail.com".
- Mật khẩu** (Password): A password input field with a strength indicator bar below it. The text "Mật khẩu trung bình" (Average password) is displayed.
- Xác nhận mật khẩu** (Confirm password): A second password input field.
- Đăng Ký Ngay** (Register Now): A prominent green button.
- hoặc** (or): A separator word.
- Đăng nhập ngay** (Login now): A link for existing users, preceded by the text "Đã có tài khoản?" (Already have an account?).

Hình 4.3.1. Hình ảnh hướng dẫn đăng ký

Khi bạn đã điền đầy đủ thông tin vào phần đăng ký, sau khi đăng ký thành công sẽ chuyển sang phần Trang Chủ và dữ liệu thông tin của bạn sẽ được lưu lại.

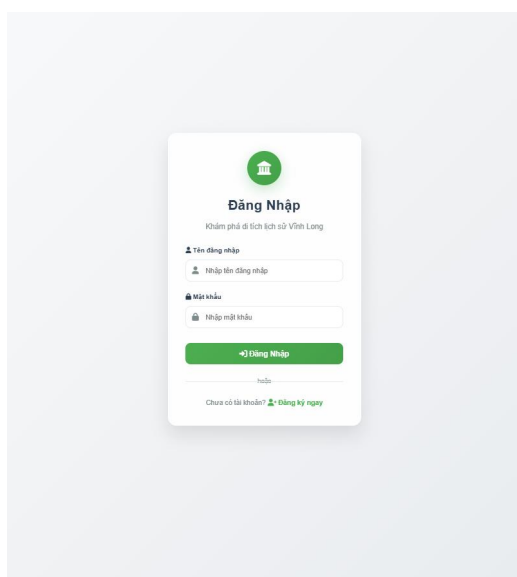


Hình 4.3.2 Hình ảnh cơ sở dữ liệu được lưu lại

Đảm bảo thông tin của bạn được bảo mật một cách tốt nhất có thể

4.4. Phần đăng nhập

4.4.1. Giao diện chính đăng nhập



Hình 4.4.1 Hình ảnh trang Đăng nhập

Giao diện người dùng

Thiết kế đã tạo được một form đăng nhập tối giản (minimalist) với tông màu xanh lá cây chủ đạo, mang lại cảm giác thân thiện và phù hợp với chủ đề lịch sử/văn hóa.

Bố cục cân đối: Các thành phần (logo, tiêu đề, ô nhập liệu, nút bấm) được sắp xếp cân giữa, có khoảng trắng (white space) hợp lý, giúp người dùng dễ tập trung.

Hiệu ứng hình ảnh: Có sử dụng đổ bóng (box-shadow) cho khung đăng nhập, tạo chiều sâu cho giao diện.

Các thành phần chức năng

Logo/Icon: Một biểu tượng hình ngôi đèn/di tích nằm trong vòng tròn xanh, giúp nhận diện thương hiệu ngay lập tức.

Tiêu đề & Slogan: "Đăng Nhập" và dòng phụ "Khám phá di tích lịch sử Vĩnh Long" giúp xác định mục đích của trang.

- Tên đăng nhập: placeholder "Nhập tên đăng nhập".

- Mật khẩu: placeholder "Nhập mật khẩu".

Nút hành động (CTA): Nút "Đăng Nhập" lớn, màu xanh nổi bật với icon mũi tên đi kèm.

Điều hướng phụ: Dòng chữ "Chưa có tài khoản? Đăng ký ngay" để chuyển hướng người dùng mới.

4.4.2. Hướng dẫn đăng nhập

Bước 1: Truy cập trang đăng nhập

Click vào nút "Đăng Nhập" trên thanh menu

Hoặc truy cập trực tiếp: `DangNhap.html`

Bước 2: Nhập thông tin đăng nhập

Tên đăng nhập: Nhập tên đăng nhập đã đăng ký

Mật khẩu: Nhập mật khẩu tương ứng

Bước 3: Đăng nhập

Click nút "Đăng Nhập"

Nếu thông tin chính xác, bạn sẽ được chuyển về trang chủ

Nếu sai thông tin, hệ thống sẽ thông báo

4.4.3. Cách thức hoạt động



Hình 4.4.2 Hình ảnh hướng dẫn đăng nhập

Khi bạn đăng nhập tài khoản của bạn đã tạo trên phần đăng ký, sau khi đăng nhập thành công sẽ hiện lên dòng chữ “Đăng nhập thành công! Chào mừng bạn trở lại!”.



Hình 4.4.3 Hình ảnh đăng nhập thành công

Sau khi bạn nhấn “OK” sẽ chuyển sang phần Trang Chủ bạn có thể thực hiện chức năng Tra Cứu của website.

4.5. Tổng quan về phần tra cứu

4.5.1. Giao diện chính tra cứu

The screenshot shows a web interface titled "Tra Cứu Di Tích Lịch Sử" (Search Historical Sites). It features a purple search bar with the text "Bộ lọc tìm kiếm" (Search filters). Inside the bar, there are two dropdown menus: "Tỉnh/Thành phố:" (Province/City) and "Loại di tích:" (Type of site), both currently set to "Tất cả" (All). Below these is a button labeled "Đặt lại bộ lọc" (Reset filters). To the right of the search bar is a green button labeled "Tìm Kiếm" (Search). Below the search bar is a text input field with the placeholder "Nhập tên di tích, địa chỉ, hoặc từ khóa..." (Enter the name of the site, address, or keyword...). Below this is a section titled "Tìm kiếm nhanh:" (Quick search) with five buttons: "Bác Hồ", "Chùa", "Đình", "Nhà thờ", and "Bảo tàng". At the bottom, there is a message: "Hãy nhập từ khóa hoặc chọn bộ lọc để bắt đầu tìm kiếm." (Please enter a keyword or select a filter to start searching.).

Hình 4.5.1 Hình ảnh trang đăng ký

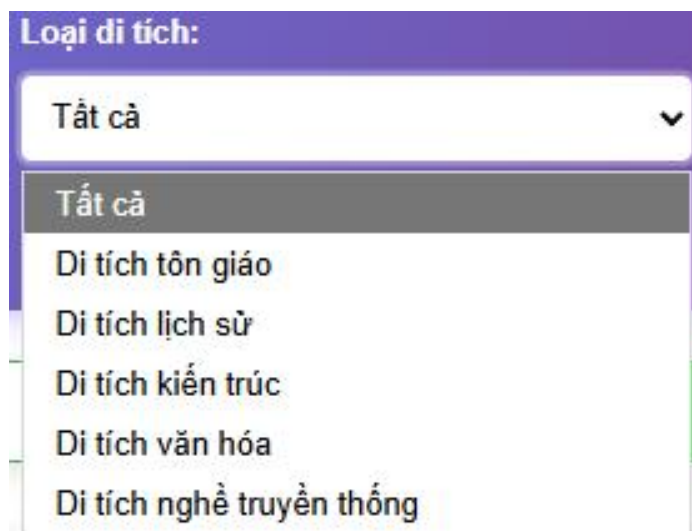
Hệ thống bộ lọc

Lọc theo địa điểm: Cho phép chọn theo Tỉnh/Thành phố (giúp thu hẹp phạm vi tìm kiếm).

The screenshot shows a close-up of the "Tỉnh/Thành phố:" dropdown menu. The menu is open, showing a list of options: "Tất cả" (All), "Vĩnh Long", "Trà Vinh", and "Bến Tre". The "Tất cả" option is currently selected.

Hình 4.5.2 Hình ảnh bộ lọc theo lọc theo địa điểm

Lọc theo loại hình: Phân loại di tích (ví dụ: tâm linh, lịch sử, kiến trúc...) giúp người dùng tìm đúng sở thích.



Hình 4.5.3 Hình ảnh bộ lọc theo loại hình

Nút "Đặt lại bộ lọc": Một chi tiết nhỏ nhưng rất quan trọng về mặt trải nghiệm, giúp người dùng xóa nhanh các lựa chọn để tìm lại từ đầu.



Hình 4.5.4 Hình ảnh nút đặt lại bộ lọc

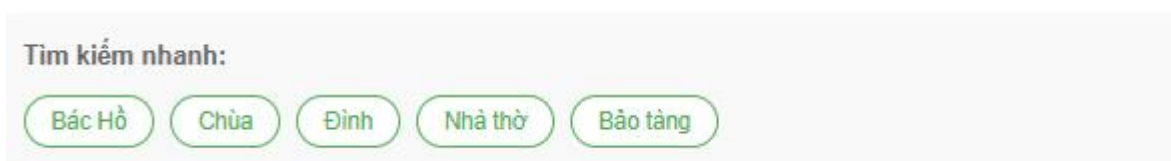
Công cụ tìm kiếm linh hoạt

Thanh tìm kiếm (Search Bar): Có placeholder hướng dẫn cụ thể ("Nhập tên di tích, địa chỉ, hoặc từ khóa..."). Việc cho phép tìm theo từ khóa giúp hệ thống thông minh hơn.



Hình 4.5.5 Hình ảnh thanh tìm kiếm

Tìm kiếm nhanh (Quick Search/Tags): Đây là điểm sáng nhất trong thiết kế này. Các tag như Bác Hồ, Chùa, Đình, Nhà thờ, Bảo tàng giúp người dùng truy cập ngay lập tức vào các chủ đề phổ biến mà không cần gõ phím.



Hình 4.5.6 Hình ảnh phần tìm kiếm nhanh

Trạng thái phản hồi người dùng

Dòng chữ "Hãy nhập từ khóa hoặc chọn bộ lọc để bắt đầu tìm kiếm" ở phía dưới là một cách xử lý "Empty State" (trạng thái trống) rất tốt. Nó hướng dẫn người dùng biết phải làm gì tiếp theo thay vì để một khoảng trắng vô nghĩa.

Hãy nhập từ khóa hoặc chọn bộ lọc để bắt đầu tìm kiếm.

Hình 4.5.7 Hình ảnh trạng thái phản hồi người dùng

4.5.2. Hướng dẫn tra cứu

Cách 1: Tra cứu cơ bản (Không cần đăng nhập)

Truy cập trang tra cứu

Click "Tra Cứu" trên menu

Tìm kiếm

Nhập từ khóa vào ô tìm kiếm

Ví dụ: "chùa", "đình", "Vĩnh Long"

Click nút "Tìm kiếm"

Cách 2: Tra cứu nâng cao (Cần đăng nhập)

Đăng nhập vào hệ thống (theo hướng dẫn ở trên)

Sử dụng bộ lọc nâng cao

Lọc theo loại di tích (chùa, đình, nhà thờ...)

Lọc theo địa điểm

Lọc theo thời gian xây dựng

Xem thông tin chi tiết

Click vào tên di tích để xem chi tiết

Xem hình ảnh, mô tả, lịch sử

Xem vị trí GPS trên bản đồ

Thông tin hiển thị khi tra cứu:

Tên di tích: Tên chính thức của di tích

Địa chỉ: Vị trí cụ thể của di tích

Mô tả: Thông tin chi tiết về lịch sử, kiến trúc

Hình ảnh: Ảnh thực tế của di tích

Tọa độ GPS: Vị trí chính xác trên bản đồ

Năm xây dựng: Thời gian xây dựng (nếu có)

4.5.3. Cách thức hoạt động

Tra Cứu Di Tích Lịch Sử

Bộ lọc tìm kiếm

Tỉnh/Thành phố: Tất cả ▼ Loại di tích: Tất cả ▼

[Đặt lại bộ lọc](#)

[Tìm Kiếm](#)

Tìm kiếm nhanh:

[Bác Hồ](#) [Chùa](#) [Đình](#) [Nhà thờ](#) [Bảo tàng](#)

✓ **Tìm thấy 2 kết quả**

Đền thờ Bác Hồ Trà Vinh

📍 Trà Vinh

Địa chỉ: Ấp Vĩnh Hội, Xã Long Đức, TP. Trà Vinh

Loại: Di tích lịch sử

Năm xây dựng: 1970

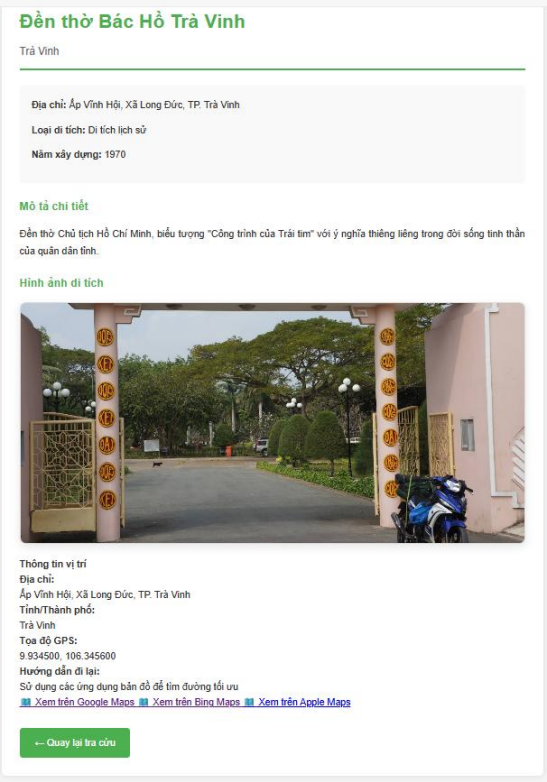
Đền thờ Chủ tịch Hồ Chí Minh, biểu tượng "Công trình của Trái tim" với ý nghĩa thiêng liêng trong đời sống tinh thần của quân dân tỉnh....

[Xem chi tiết →](#)

Hình 4.5.8 Hình ảnh cách thức hoạt động

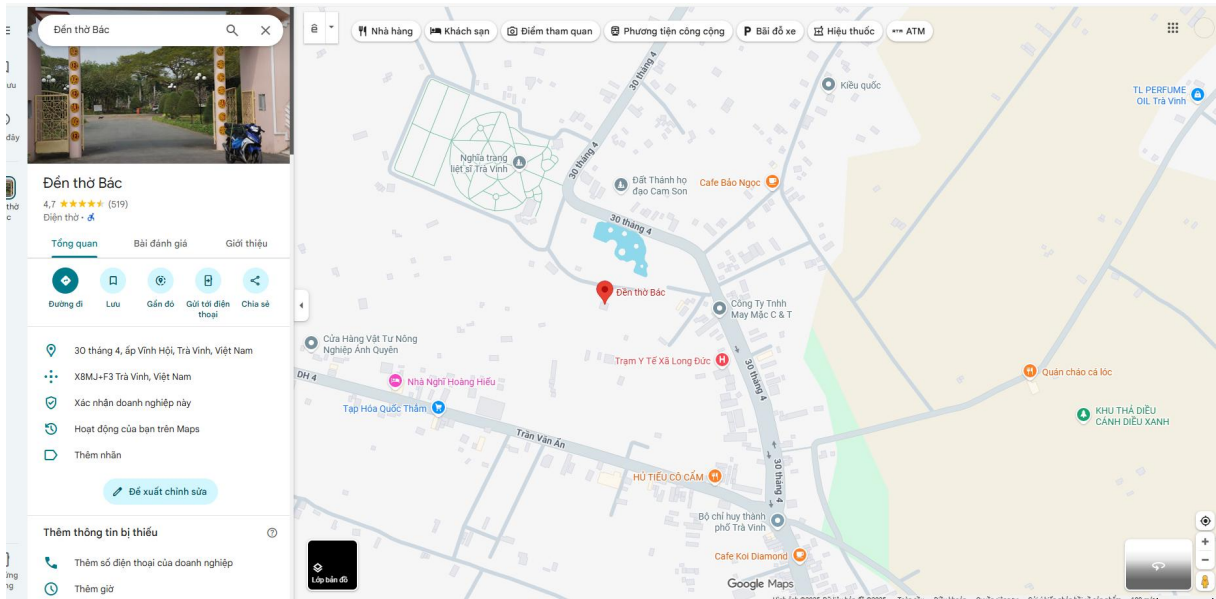
Khi bạn vào phần tìm kiếm để tra cứu thông tin, khi nhập tên di tích hoặc sử dụng thanh tìm kiếm nhanh sẽ hiển thị ra tên các di tích mà bạn tìm kiếm.

Website giúp người dùng tra cứu thông tin về các di tích lịch sử trên địa bàn Tỉnh Vĩnh Long



Hình 4.5.8 Hình ảnh sau khi tra cứu

Khi nhấn vào sản phẩm sẽ hiển thị ra chi tiết đầy đủ thông tin, vị trí, tọa độ và năm xây dựng của di tích. Website còn có chức năng hiển thị vị trí chính xác trên gg map.



Hình 4.5.9 Hình ảnh bản đồ

Tổng kết chương

Chương này đã trình bày chi tiết giao diện cách thức hoạt động website tra cứu di tích lịch sử, bao gồm:

- Tổng quan về phần giao diện của trang website giúp người dùng hiểu rõ hơn.
- Cách thức hoạt động và hướng dẫn sử dụng giúp cho việc đăng ký, đăng nhập và tra cứu trở nên đơn giản với người dùng mới lần đầu vào website.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết quả đạt được

Hoàn thành hệ thống website

Xây dựng thành công website tra cứu di tích lịch sử Vĩnh Long với đầy đủ chức năng cơ bản

Triển khai hệ thống quản lý người dùng với tính năng đăng ký, đăng nhập và phân quyền

Phát triển module tra cứu thông tin di tích với khả năng tìm kiếm nâng cao

Tích hợp hệ thống hiển thị chi tiết di tích bao gồm hình ảnh, mô tả và vị trí GPS

Cơ sở dữ liệu và thông tin

Xây dựng cơ sở dữ liệu chứa thông tin 47 di tích lịch sử tại Vĩnh Long

Thiết kế cấu trúc dữ liệu tối ưu với các bảng users, monuments và categories

Tổ chức thông tin di tích theo tiêu chuẩn khoa học với đầy đủ metadata

5.2. Giao diện và trải nghiệm người dùng

5.2.1. Giao diện

Thiết kế giao diện responsive tương thích đa thiết bị

Áp dụng nguyên tắc UX/UI hiện đại với màu sắc và typography phù hợp

Tối ưu hóa tốc độ tải trang và hiệu suất hệ thống

5.2.2. Tính năng kỹ thuật

Tích hợp Google Maps API để hiển thị vị trí di tích

Xây dựng hệ thống tìm kiếm với thuật toán LIKE và FULLTEXT

Triển khai bảo mật cơ bản với mã hóa mật khẩu và validation dữ liệu

Phát triển API RESTful cho các chức năng tra cứu

5.3. Đóng góp mới

5.3.1. Về mặt công nghệ

Ứng dụng công nghệ web hiện đại (HTML5, CSS3, JavaScript ES6, PHP 7+) vào lĩnh vực bảo tồn di sản

Tích hợp thành công các API bên thứ ba (Google Maps) với hệ thống nội bộ

Phát triển kiến trúc MVC đơn giản phù hợp với quy mô dự án nhỏ và vừa

5.3.2. Về mặt ứng dụng

Số hóa thông tin di tích lịch sử Vĩnh Long dưới dạng website tương tác

Tạo ra công cụ tra cứu trực tuyến miễn phí phục vụ cộng đồng và du khách

Xây dựng nền tảng mở rộng cho các tỉnh thành khác trong khu vực Đồng bằng sông Cửu Long

5.3.3. Về mặt phương pháp

Áp dụng quy trình phát triển phần mềm từ phân tích yêu cầu đến triển khai

Sử dụng phương pháp thiết kế database-first cho hệ thống quản lý nội dung

Triển khai testing và debugging có hệ thống đảm bảo chất lượng sản phẩm

5.4. Hạn chế khuyết điểm

5.4.1. Về chức năng

Chưa có tính năng đánh giá và bình luận từ người dùng

Thiếu module quản trị nâng cao cho việc cập nhật thông tin di tích

5.4.2. Về kỹ thuật

Chưa áp dụng framework hiện đại để tăng tính bảo mật

Thiếu hệ thống cache và tối ưu hóa cho lượng truy cập lớn

Chưa có backup tự động và disaster recovery plan

5.4.3. Về nội dung

Thông tin một số di tích chưa đầy đủ do hạn chế trong thu thập dữ liệu

Chưa có phiên bản đa ngôn ngữ để phục vụ du khách quốc tế

Thiếu tính năng tour ảo và multimedia tương tác

5.5. ĐỀ XUẤT PHÁT TRIỂN

5.5.1. Ngắn hạn (3-6 tháng)

Bổ sung tính năng quản trị nâng cao cho admin

Tích hợp hệ thống đánh giá và phản hồi từ người dùng

Tối ưu hóa SEO và tăng tốc độ tải trang

Phát triển ứng dụng mobile native (Android/iOS)

5.5.2. Trung hạn (6-12 tháng)

Mở rộng cơ sở dữ liệu bao gồm các tỉnh lân cận

Tích hợp AI/ML cho hệ thống gợi ý di tích phù hợp

Phát triển tính năng tour ảo 360 độ

Xây dựng API public cho các ứng dụng bên thứ ba

5.5.3. Dài hạn (1-2 năm)

Phát triển thành nền tảng quốc gia về di sản văn hóa

Tích hợp blockchain để xác thực thông tin di tích

Ứng dụng AR/VR cho trải nghiệm tương tác

Xây dựng hệ sinh thái du lịch thông minh

5.5.4. Đề xuất về tổ chức và vận hành

Thành lập ban quản lý chuyên trách

Xây dựng quy trình cập nhật thông tin định kỳ

Hợp tác với Sở Văn hóa và Du lịch các tỉnh thành

Tìm kiếm nguồn tài trợ bền vững cho việc duy trì và phát triển

5.6. Ý nghĩa thực tiễn

5.6.1. Đối với cộng đồng

Nâng cao nhận thức về giá trị di sản văn hóa địa phương, tạo công cụ học tập và nghiên cứu cho học sinh, sinh viên, hỗ trợ phát triển du lịch văn hóa bền vững

5.6.2. Đối với ngành công nghệ

Minh chứng khả năng ứng dụng công nghệ thông tin vào bảo tồn di sản

Tạo tiền đề cho các dự án số hóa di sản quy mô lớn hơn

Đóng góp vào xu hướng chuyển đổi số trong lĩnh vực văn hóa

5.6.3. Đối với nghiên cứu khoa học

Cung cấp dữ liệu số cho các nghiên cứu về lịch sử và văn hóa

Tạo nền tảng cho việc phân tích và so sánh di tích liên vùng

Hỗ trợ công tác lập hồ sơ di sản văn hóa quốc gia và thế giới

Dự án đã hoàn thành các mục tiêu đề ra và tạo nền tảng vững chắc cho việc phát triển hệ thống quản lý di sản văn hóa số trong tương lai.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] Hồ Minh, “Di tích lịch sử văn hóa Vĩnh Long.” Accessed: Nov. 09, 2025. [Online]. Available: <https://portal.vinhlong.gov.vn/portal/wpthuvientinh/thuvientinh/page/xemtin.cpx?item=637c3316e324da0e1e37eb69>
- [2] Linh Trao, “Front End vs Back End: Khác nhau và Hỗ trợ nhau như thế nào?” Accessed: Nov. 10, 2025. [Online]. Available: <https://itviec.com/blog/front-end-vs-back-end/>
- [3] Hà My, “Thiết kế UI UX là gì? Hướng dẫn từ A – Z cho người mới bắt đầu.” Accessed: Nov. 22, 2025. [Online]. Available: <https://itviec.com/blog/thiet-ke-ui-ux-cho-nguoi-moi-bat-dau/>
- [4] Daisy, “HTML, CSS, JavaScript là gì?” Accessed: Nov. 10, 2025. [Online]. Available: <https://itviec.com/blog/thiet-ke-ui-ux-cho-nguoi-moi-bat-dau/>
- [5] TopDev, “Ngôn ngữ lập trình PHP là gì? Tất tần tật những điều bạn cần biết về PHP.” Accessed: Nov. 20, 2025. [Online]. Available: <https://topdev.vn/blog/ngon-ngu-lap-trinh-php-la-gi-tat-tan-tat-nhung-dieu-ban-can-biet-ve-php/>

PHỤ LỤC