

**ĐẠI HỌC ĐÀ NẴNG**  
**TRƯỜNG ĐẠI HỌC BÁCH KHOA**  
**KHOA CÔNG NGHỆ THÔNG TIN**

---



**BÁO CÁO**  
**PBL3: ĐỒ ÁN LẬP TRÌNH ỨNG DỤNG**  
**HỆ THỐNG HỖ TRỢ TÌM KIẾM VÀ CHIA SẺ**  
**CÁC ĐỊA ĐIỂM DU LỊCH**

**Giảng viên hướng dẫn: TS.GV. Đặng Thiên Bình.**

**Sinh viên thực hiện: Lê Nguyễn Châu Anh,**  
**Hồ Nguyễn Thảo Nguyên,**  
**Hà Thị Kim Oanh,**  
**Nguyễn Như Quỳnh.**

**Lớp: 23T\_NHAT2.**

**Nhóm: 23.16A.**

**Đà Nẵng, 06 – 2025**



## LỜI MỞ ĐẦU

Trong bối cảnh công nghệ thông tin ngày càng phát triển và lan rộng vào mọi lĩnh vực của đời sống, nhu cầu tra cứu và chia sẻ thông tin du lịch của người dân cũng ngày càng tăng cao, đặc biệt là với giới trẻ – những người năng động, yêu thích khám phá và thường xuyên di chuyển. Tuy nhiên, hiện nay việc tìm kiếm các địa điểm du lịch phù hợp vẫn còn gặp nhiều khó khăn do thông tin phân tán, thiếu sự đánh giá thực tế từ cộng đồng, cũng như chưa có nền tảng hỗ trợ kết nối trải nghiệm người dùng một cách hiệu quả.

Hiện nay, có nhiều vùng đất giàu tiềm năng du lịch với nhiều danh lam thắng cảnh, di tích lịch sử, và nền văn hóa đặc sắc. Thế nhưng, nhiều địa điểm thú vị vẫn chưa được khai thác đúng mức hoặc chưa được biết đến rộng rãi. Nhận thấy được thực trạng này, nhóm chúng em đã lựa chọn đề tài “Hệ thống hỗ trợ tìm kiếm và chia sẻ các địa điểm du lịch” làm dự án cho học phần PBL3.

Mục tiêu của dự án là xây dựng một hệ thống thân thiện, dễ sử dụng, cho phép người dùng tìm kiếm, đánh giá và chia sẻ thông tin về các địa điểm du lịch. Dự án không chỉ nhằm giải quyết nhu cầu thực tế của người dùng mà còn là cơ hội để nhóm áp dụng kiến thức đã học vào thực tiễn, nâng cao kỹ năng lập trình hệ thống, làm việc nhóm và giải quyết vấn đề.

Chúng em xin chân thành cảm ơn sự quan tâm, hướng dẫn tận tình của thầy và sự hỗ trợ của các bạn đã giúp nhóm hoàn thành dự án này.

Cấu trúc của đồ án môn học gồm:

Chương 1: Tổng quan đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Phân tích và thiết kế hệ thống

Chương 4: Chương trình và kết quả

Chương 5: Kết luận và hướng phát triển

## MỤC LỤC

LỜI MỞ ĐẦU .....	3
MỤC LỤC.....	4
DANH MỤC HÌNH ẢNH .....	5
I. TỔNG QUAN ĐỀ TÀI .....	7
1. Mục tiêu đề tài .....	7
2. Phạm vi và đối tượng nghiên cứu .....	7
3. Đối tượng sử dụng .....	7
II. CƠ SỞ LÝ THUYẾT.....	8
1. ASP.NET Core .....	8
2. Hướng đối tượng (OOP) .....	8
3. Công nghệ phía Frontend (HTML, CSS, JavaScript).....	9
4. Entity Framework Core (EF Core) .....	10
5. Hệ quản trị Cơ sở dữ liệu: SQL Server.....	11
6. Công cụ sử dụng .....	12
7. Quản lý mã nguồn: GitHub.....	12
III. PHÂN TÍCH VÀ THIẾT KẾ ỨNG DỤNG .....	14
1. Use – case Diagram .....	14
2. Thiết kế cơ sở dữ liệu.....	19
3. Cấu trúc thư mục dự án.....	27
IV. TRIỂN KHAI ỨNG DỤNG .....	32
1. Người quản lý (Admin).....	34
2. Người dùng (User) .....	44
V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	56
1. Kết luận .....	56
2. Hướng phát triển .....	56
TÀI LIỆU THAM KHẢO.....	58

## DANH MỤC HÌNH ẢNH

Hình 1: Visual Studio Code.....	12
Hình 2: SQL Server Management Studio 2022 .....	12
Hình 3: Use-case diagram .....	14
Hình 4: Data Diagram .....	19
Hình 5: Sơ đồ cấu trúc thư mục .....	27
Hình 6: Giao diện đăng nhập .....	32
Hình 7: Giao diện đăng nhập bằng Facebook .....	33
Hình 8: Giao diện đăng nhập bằng Google .....	34
Hình 9: Giao diện Trang tổng quan của Admin .....	35
Hình 10: Biểu đồ tỷ lệ phân bố bài viết .....	35
Hình 11: Bảng hoạt động gần đây .....	36
Hình 12: Thêm địa điểm du lịch mới .....	37
Hình 13: Quản lý bài viết .....	38
Hình 14: Trang Tương tác .....	39
Hình 15: Biểu đồ tương tác theo thời gian .....	39
Hình 16: Biểu đồ phân bố tương tác .....	40
Hình 17: Bài viết có nhiều tương tác nhất .....	40
Hình 18: Địa điểm có nhiều tương tác nhất .....	41
Hình 19: Người dùng tương tác nhiều nhất .....	41
Hình 20: Quản lý Báo cáo vi phạm .....	42
Hình 21: Quản lý người dùng .....	42
Hình 22: Quản lý Đánh giá .....	43
Hình 23: Quản lý Bình luận .....	43
Hình 24: Giao diện Trang chủ .....	44
Hình 25: Chức năng Tìm kiếm .....	45
Hình 26: Bộ lọc địa điểm .....	45
Hình 27: Chi tiết địa điểm .....	46
Hình 28: Đánh giá địa điểm .....	47
Hình 29: Yêu thích/Chia sẻ Địa điểm .....	47
Hình 30: Viết bài chia sẻ .....	48
Hình 31: Bài viết của tôi kèm trạng thái .....	48
Hình 32: Tương tác yêu thích .....	49
Hình 33: Bình luận bài viết .....	49
Hình 34: Chia sẻ bài viết .....	50
Hình 35: Báo cáo bài viết .....	50
Hình 36: Thông tin cá nhân .....	51
Hình 37: Đổi mật khẩu .....	51
Hình 38: Bộ sưu tập ảnh .....	52

Hình 39: Bài viết của tôi .....	52
Hình 40: Đánh giá của tôi .....	53
Hình 41: Bình luận của tôi .....	53
Hình 42: Địa điểm và Bài viết yêu thích.....	54
Hình 43: Chỉnh sửa bài viết .....	55
Hình 44: Xoá bài viết .....	55
Hình 45: Liên hệ.....	56

## I. TỔNG QUAN ĐỀ TÀI

### 1. Mục tiêu đề tài

- Mục đích thực hiện đề tài: Học tập cách phát triển ứng dụng hệ thống hoàn chỉnh, từ việc thiết kế cơ sở dữ liệu, xây dựng giao diện người dùng thân thiện đến triển khai các chức năng nghiệp vụ phức tạp. Đồng thời, nâng cao khả năng tư duy phân tích hệ thống, thiết kế kiến trúc phần mềm hợp lý và tối ưu hóa hiệu suất ứng dụng, tạo ra sản phẩm có tính thực tiễn cao phục vụ cộng đồng du lịch.

- Mục tiêu đề tài: Tìm hiểu và ứng dụng công nghệ .NET Framework cùng với hệ quản trị cơ sở dữ liệu SQL Server để xây dựng hệ thống hỗ trợ tìm kiếm và chia sẻ thông tin các địa điểm du lịch. Qua đó, học được cách tổ chức mã nguồn theo mô hình MVC, thiết kế cơ sở dữ liệu quan hệ và phát triển giao diện hệ thống responsive.

### 2. Phạm vi và đối tượng nghiên cứu

- Phạm vi và đối tượng nghiên cứu: Nghiên cứu về công nghệ ASP.NET, Entity Framework, Trắc thư viện JavaScript/jQuery, Bootstrap CSS framework, và SQL Server qua việc xây dựng hệ thống tìm kiếm và chia sẻ thông tin các địa điểm du lịch.

- Phương pháp nghiên cứu: Sử dụng kiến thức học được trên lớp ở các học phần "Lập trình .NET", "Cơ sở dữ liệu" kết hợp với tham khảo tài liệu chính thức từ Microsoft Documentation, Stack Overflow và đặc biệt là theo sát sự hướng dẫn của giảng viên. Áp dụng phương pháp phát triển phần mềm theo mô hình Waterfall với các giai đoạn: Phân tích yêu cầu → Thiết kế → Cài đặt → Kiểm thử → Triển khai.

### 3. Đối tượng sử dụng

- Người dùng (User): Là những người đăng ký tài khoản trên hệ thống để sử dụng đầy đủ các chức năng như đăng bài chia sẻ địa điểm du lịch, đăng hình ảnh, viết đánh giá, bình luận và tương tác với cộng đồng. Nhóm này chủ yếu bao gồm các bạn trẻ yêu du lịch, nhóm bạn, gia đình, và cả những người dân địa phương có mong muốn giới thiệu các điểm đến hấp dẫn tại quê hương mình.

- Khách truy cập (Guest): Là những người không đăng nhập nhưng vẫn có thể truy cập hệ thống để tìm kiếm, xem thông tin về các địa điểm du lịch, đọc đánh giá, tham khảo hình ảnh và trải nghiệm. Dù không có quyền tương tác như User, nhóm này là một phần quan trọng giúp lan tỏa thông tin và thu hút người dùng tiềm năng.

- Người quản trị (Admin): Có quyền truy cập các chức năng quản lý hệ thống như duyệt bài đăng, kiểm duyệt nội dung, xóa báo cáo vi phạm, quản lý người dùng và duy trì hoạt động ổn định của hệ thống. Vai trò này giúp đảm bảo chất lượng thông tin và sự an toàn trong môi trường cộng đồng.

## II. CƠ SỞ LÝ THUYẾT

### 1. ASP.NET Core

- ASP.NET Core là một framework mã nguồn mở, đa nền tảng được phát triển bởi Microsoft, dùng để xây dựng các ứng dụng hệ thống hiện đại, dịch vụ API, và các ứng dụng dựa trên đám mây. Đây là phiên bản kế thừa và cải tiến mạnh mẽ từ ASP.NET Framework truyền thống, được thiết kế để tối ưu hóa hiệu suất, tăng tính linh hoạt và hỗ trợ phát triển trên nhiều hệ điều hành như Windows, macOS, và Linux.

- Kiến trúc MVC (Model-View-Controller) trong ASP.NET Core:

+ Model: Đại diện cho dữ liệu và logic nghiệp vụ của ứng dụng. Trong dự án này, các lớp như User, TouristSpot, Post đóng vai trò là Model, chứa các thuộc tính dữ liệu và có thể bao gồm các phương thức xử lý liên quan đến dữ liệu đó. Chúng tương tác với cơ sở dữ liệu thông qua Entity Framework Core.

+ View: Chịu trách nhiệm hiển thị dữ liệu cho người dùng. Trong ASP.NET Core MVC, Views thường là các tệp Razor (.cshtml) kết hợp HTML, CSS và mã C# (sử dụng cú pháp Razor) để tạo ra giao diện người dùng động và tương tác.

+ Controller: Đóng vai trò trung gian, tiếp nhận các HTTP request từ người dùng, xử lý logic nghiệp vụ (bằng cách tương tác với Model để truy vấn hoặc cập nhật dữ liệu), sau đó lựa chọn và truyền dữ liệu cho View thích hợp để tạo ra HTTP response trả về cho trình duyệt.

### 2. Hướng đối tượng (OOP)

- Lập trình hướng đối tượng (OOP) là một phương pháp lập trình dựa trên khái niệm về lớp và đối tượng. OOP tập trung vào các đối tượng thao tác hơn là logic để thao tác chúng, giúp code dễ quản lý, tái sử dụng được và dễ bảo trì.

- Để hiểu rõ hơn, hãy xem qua một số khái niệm cơ bản trong OOP:

+ Đối tượng (Object): Đối tượng trong OOP bao gồm hai thành phần chính:

- Thuộc tính (Attribute): Là những thông tin, đặc điểm của đối tượng. Ví dụ, một đối tượng “User” có thuộc tính như FullName, DateOfBirth, PhoneNumber...
- Phương thức (Method): Là những hành vi mà đối tượng có thể thực hiện. Ví dụ, đối tượng “User” có thể đánh giá, bình luận, đăng bài...

+ Lớp (Class): Lớp là sự trừu tượng hóa của đối tượng. Những đối tượng có những đặc tính tương tự nhau sẽ được tập hợp thành một lớp. Lớp cũng bao gồm thuộc tính và phương thức. Một đối tượng sẽ được xem là một thực thể của lớp.

- Ứng dụng tính chất của OOP trong dự án:



+ Đóng gói: Các lớp như User sử dụng thuộc tính private (UserId, Password) và cung cấp getter/setter để kiểm soát truy cập, đảm bảo bảo mật dữ liệu nhạy cảm như mật khẩu hoặc token đặt lại mật khẩu (PasswordResetToken).

+ Kế thừa: Trong dự án, tính kế thừa được áp dụng ở phần xử lý người dùng. Các loại người dùng như khách truy cập, người dùng đăng ký, và quản trị viên đều chia sẻ chung các thuộc tính cơ bản như Username, Password, và được phân quyền dựa trên thuộc tính Role. Mặc dù không định nghĩa các lớp con riêng biệt, hệ thống vẫn sử dụng ý tưởng kế thừa logic để tổ chức quyền truy cập và chức năng tương ứng cho từng loại người dùng.

+ Đa hình: Tính chất đa hình được thể hiện trong cách hệ thống xử lý các loại bài viết khác nhau trong cùng một bảng Post. Mỗi bài viết có một thuộc tính TypeOfPost để phân loại nội dung. Tuy tất cả các loại bài viết đều lưu trong cùng bảng Post, nhưng khi hiển thị hoặc xử lý logic trong giao diện, hệ thống áp dụng tư tưởng đa hình: các loại bài viết được xử lý theo cách khác nhau tùy vào TypeOfPost. Mặc dù trong mã nguồn không tách ra các lớp GuidePost, ExperiencePost, v.v., nhưng tư tưởng đa hình vẫn được áp dụng thông qua xử lý logic động dựa trên kiểu dữ liệu, thay vì viết nhiều khối if rời rạc. Điều này giúp mã dễ mở rộng nếu sau này thêm các loại bài viết mới.

+ Trừu tượng: Các lớp như Category và TouristSpot trừu tượng hóa các khái niệm thực tế (danh mục và địa điểm du lịch), cho phép mô hình hóa dữ liệu phức tạp một cách đơn giản.

+ Ưu điểm: OOP giúp mô hình hóa những thứ phức tạp dưới dạng cấu trúc đơn giản, tái sử dụng mã, dễ bảo trì (sửa lỗi trong một lớp không ảnh hưởng đến lớp khác), và hỗ trợ mở rộng (thêm loại bài viết mới trong Post).

### **3. Công nghệ phía Frontend (HTML, CSS, JavaScript)**

- HTML (HyperText Markup Language):

HTML là ngôn ngữ đánh dấu cơ bản được sử dụng để tạo nên cấu trúc và nội dung của các trang của hệ thống. Nó sử dụng các "thẻ" (tags) để định nghĩa các thành phần khác nhau của một trang, ví dụ như tiêu đề, đoạn văn, hình ảnh, liên kết, và các biểu mẫu nhập liệu. Trong dự án này, HTML được dùng để xây dựng khung sườn cho các trang hiển thị danh sách địa điểm, chi tiết địa điểm, trang đăng bài, trang thông tin người dùng, v.v.

- CSS (Cascading Style Sheets):

CSS là ngôn ngữ được sử dụng để mô tả cách trình bày và định dạng giao diện của một tài liệu HTML. Nó kiểm soát các yếu tố như màu sắc, phông chữ, bố cục,

khoảng cách, và các hiệu ứng trực quan khác. CSS giúp hệ thống du lịch trở nên hấp dẫn, dễ đọc, và thân thiện với người dùng.

+ Bootstrap: Dự án có sử dụng Bootstrap, một CSS framework phổ biến, để tăng tốc độ phát triển giao diện, cung cấp các thành phần UI đã được thiết kế sẵn (như hệ thống lưới responsive, nút bấm, biểu mẫu, thanh điều hướng) và đảm bảo hệ thống hiển thị tốt trên nhiều kích thước màn hình khác nhau (responsive design).

- JavaScript:

+ JavaScript là một ngôn ngữ lập trình kịch bản, chủ yếu chạy ở phía trình duyệt của người dùng (client-side), cho phép tạo ra hệ thống động và tương tác cao.

+ Trong dự án, JavaScript được sử dụng để:

- Xử lý các sự kiện từ người dùng như click chuột, nhập liệu vào form.
- Kiểm tra tính hợp lệ của dữ liệu nhập vào form (form validation) phía client.
- Thực hiện các yêu cầu AJAX (Asynchronous JavaScript and XML/JSON) để tải dữ liệu từ server mà không cần tải lại toàn bộ trang (ví dụ: hiển thị kết quả tìm kiếm gợi ý, cập nhật trạng thái yêu thích).
- Thao tác với DOM (Document Object Model) để thay đổi nội dung và giao diện của hệ thống một cách linh hoạt.

+ Thư viện jQuery có thể được sử dụng để đơn giản hóa việc viết code JavaScript, đặc biệt là trong việc thao tác DOM, xử lý sự kiện và thực hiện các request AJAX.

#### **4. Entity Framework Core (EF Core)**

- Entity Framework Core (EF Core) là một Object-Relational Mapper (ORM) mã nguồn mở, nhẹ, có thể mở rộng và đa nền tảng cho .NET. Nó cho phép các nhà phát triển làm việc với cơ sở dữ liệu bằng cách sử dụng các đối tượng .NET (POCO - Plain Old CLR Objects) quen thuộc, thay vì phải viết trực tiếp các câu lệnh SQL.

- Vai trò và các khái niệm chính trong dự án:

+ DbContext: Đại diện cho một session làm việc với cơ sở dữ liệu. Nó chứa các DbSet đại diện cho các bảng và quản lý việc kết nối, theo dõi thay đổi của các đối tượng, và lưu các thay đổi đó vào cơ sở dữ liệu.

+ DbSet<TEntity>: Mỗi DbSet trong DbContext tương ứng với một bảng trong cơ sở dữ liệu (ví dụ: DbSet<User> tương ứng với bảng Users). Nó cho phép thực hiện các thao tác truy vấn (sử dụng LINQ), thêm, sửa, xóa dữ liệu trên bảng đó.

+ LINQ to Entities: Cho phép viết các truy vấn dữ liệu bằng ngôn ngữ LINQ (Language Integrated Query) trực tiếp trong mã C#. EF Core sẽ dịch các truy vấn LINQ này thành các câu lệnh SQL phù hợp để thực thi trên cơ sở dữ liệu.

+ Migrations: Là một tính năng cho phép quản lý và cập nhật schema của cơ sở dữ liệu một cách tự động khi có sự thay đổi trong các lớp Model (theo phương pháp Code-First).

- Lợi ích của EF Core: Giúp tăng tốc độ phát triển, giảm thiểu code truy cập dữ liệu lặp đi lặp lại, cung cấp khả năng truy vấn type-safe, và hỗ trợ làm việc với nhiều hệ quản trị cơ sở dữ liệu khác nhau.

## 5. Hệ quản trị Cơ sở dữ liệu: SQL Server

- Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mạnh mẽ, toàn diện và được sử dụng rộng rãi, phát triển bởi tập đoàn Microsoft. Nó cung cấp một nền tảng vững chắc để lưu trữ, quản lý và truy xuất dữ liệu có cấu trúc một cách hiệu quả và an toàn cho nhiều loại ứng dụng khác nhau, từ các ứng dụng doanh nghiệp lớn đến hệ thống và dịch vụ trực tuyến.

- Đặc điểm chính của SQL Server trong dự án:

+ Tính toàn vẹn dữ liệu cao: SQL Server hỗ trợ mạnh mẽ các cơ chế để đảm bảo tính toàn vẹn và nhất quán của dữ liệu thông qua việc sử dụng khóa chính (Primary Keys), khóa ngoại (Foreign Keys) để định nghĩa mối quan hệ giữa các bảng, cùng với các ràng buộc (Constraints) như NOT NULL, UNIQUE, và CHECK. Điều này rất quan trọng để duy trì sự chính xác của thông tin người dùng, địa điểm, bài viết và các tương tác khác trên hệ thống.

+ Ngôn ngữ truy vấn T-SQL (Transact-SQL): SQL Server sử dụng T-SQL, một phần mở rộng của ngôn ngữ SQL chuẩn, cung cấp nhiều hàm, thủ tục lưu trữ (Stored Procedures), và các cấu trúc lập trình mạnh mẽ cho phép thực hiện các thao tác truy vấn và xử lý dữ liệu phức tạp.

+ Bảo mật nâng cao: Cung cấp các tính năng bảo mật đa lớp, từ xác thực người dùng, phân quyền chi tiết trên các đối tượng cơ sở dữ liệu, đến mã hóa dữ liệu, giúp bảo vệ thông tin nhạy cảm của người dùng và dữ liệu của ứng dụng.

+ Công cụ quản lý và phát triển mạnh mẽ: SQL Server Management Studio (SSMS) là một công cụ đồ họa tích hợp, cung cấp giao diện trực quan để thiết kế, quản trị, giám sát và phát triển cơ sở dữ liệu một cách hiệu quả.

+ Tích hợp chặt chẽ với hệ sinh thái .NET: SQL Server hoạt động rất tốt và có sự tương thích cao với các công nghệ .NET như ASP.NET Core và Entity Framework Core, giúp quá trình phát triển ứng dụng trở nên mượt mà và hiệu quả hơn. Entity

Framework Core có thể dễ dàng ánh xạ các đối tượng C# sang các bảng trong SQL Server và ngược lại.

- Ứng dụng của SQL Server trong dự án:

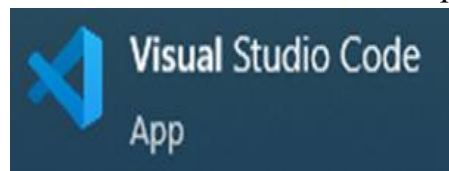
+ SQL Server được sử dụng làm nơi lưu trữ trung tâm cho toàn bộ dữ liệu của ứng dụng, bao gồm:

- Thông tin tài khoản người dùng (tên đăng nhập, mật khẩu, email, thông tin cá nhân, vai trò).
- Dữ liệu chi tiết về các địa điểm du lịch (tên, địa chỉ, mô tả, hình ảnh, danh mục,...).
- Nội dung các bài viết về địa điểm, cảm nang, trải nghiệm của người dùng.
- Các đánh giá, bình luận của người dùng về địa điểm và bài viết.
- Thông tin về các tương tác như lượt yêu thích, lượt chia sẻ.
- Dữ liệu về các báo cáo vi phạm.

+ Khả năng xử lý các truy vấn phức tạp của SQL Server cho phép hệ thống tìm kiếm và lọc địa điểm, bài viết một cách hiệu quả, cũng như hiển thị các thông tin liên quan một cách nhanh chóng cho người dùng.

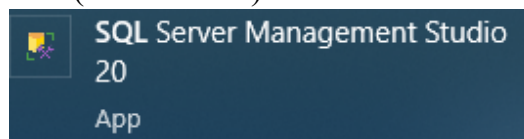
## 6. Công cụ sử dụng

- Môi trường lập trình C#: Visual Studio Code kết hợp với .NET SDK.



*Hình 1: Visual Studio Code*

- Quản lý cơ sở dữ liệu: Microsoft SQL Server, quản lý thông qua SQL Server Management Studio (SSMS 2022).



*Hình 2: SQL Server Management Studio 2022*

## 7. Quản lý mã nguồn: GitHub

- Trong quá trình phát triển hệ thống tìm kiếm và chia sẻ các địa điểm du lịch, nhóm sử dụng **GitHub** làm nền tảng chính để lưu trữ và quản lý mã nguồn. GitHub được xây dựng dựa trên hệ thống kiểm soát phiên bản phân tán **Git**, cho phép các thành viên trong nhóm cộng tác hiệu quả, theo dõi lịch sử thay đổi và triển khai các tính năng một cách có hệ thống.

Dự án được quản lý tại kho lưu trữ:

- **GitHub**

- Repository:** [https://github.com/nhuquynhdayyy/project\\_based\\_learning\\_3](https://github.com/nhuquynhdayyy/project_based_learning_3)

- **Tên repository:** project\_based\_learning\_3

Tại đây, nhóm thực hiện các thao tác như tạo nhánh phát triển (MAINNNN/, WAIN/, anh/, nenny/...), tạo pull request, thảo luận qua các issue, cũng như cập nhật thường xuyên để đồng bộ tiến độ giữa các thành viên.

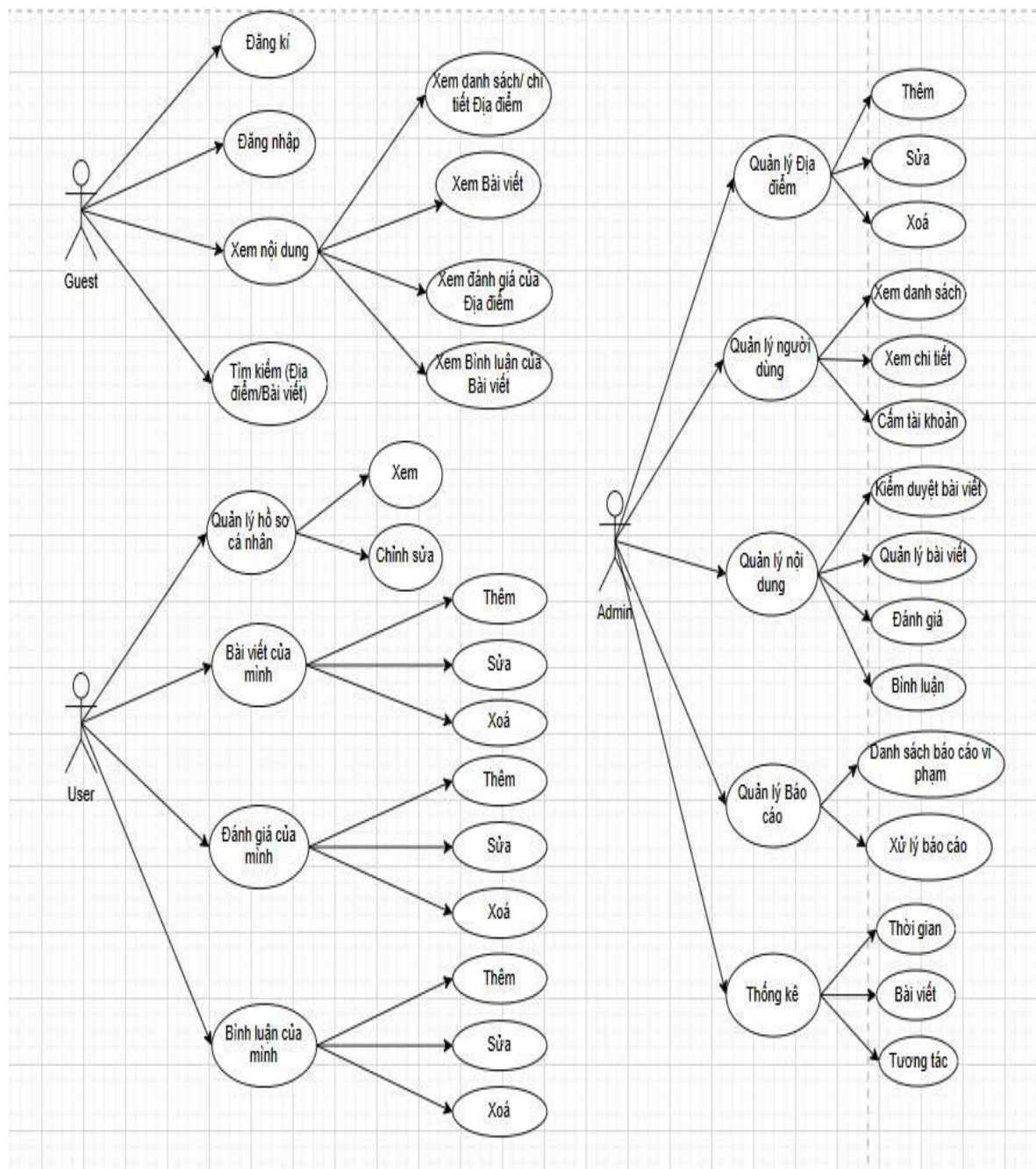
**Lợi ích GitHub mang lại cho dự án:**

- **Quản lý phiên bản:** Theo dõi toàn bộ lịch sử phát triển dự án, có thể phục hồi mã nguồn về bất kỳ thời điểm nào.
- **Hỗ trợ làm việc nhóm:** Mỗi thành viên làm việc trên nhánh riêng, sau đó hợp nhất thông qua pull request, giúp tránh xung đột mã.
- **Tích hợp CI/CD:** Hỗ trợ kết nối với các dịch vụ tự động kiểm thử, build hoặc triển khai Hệ thống (có thể mở rộng về sau).
- **Lưu trữ linh hoạt:** Dự án có thể được đặt ở chế độ công khai để chia sẻ hoặc riêng tư nếu cần bảo mật.

Việc sử dụng GitHub góp phần giúp dự án được phát triển có tổ chức, chuyên nghiệp, đồng thời hỗ trợ việc trình bày và minh chứng năng lực trong các buổi báo cáo hoặc đánh giá.

### III. PHÂN TÍCH VÀ THIẾT KẾ ỨNG DỤNG

#### 1. Use – case Diagram



Hình 3: Use-case diagram

**- Đặc tả UserCase**

**+ Đăng nhập**

Tên UseCase	Đăng nhập
Tác nhân	User
Mô tả	User thực hiện đăng nhập tài khoản để sử dụng các chức năng hệ thống
Điều kiện tiên quyết	Nếu sử dụng đăng nhập truyền thống bằng Username/Email và Password thì yêu cầu hệ thống đã có dữ liệu tài khoản
Dòng dự kiện chính	1. Người dùng truy cập trang đăng nhập 2. Nhập Username/Email và Password 3. Nhấn nút “Đăng nhập”
Kết quả	Nếu thông tin đúng, chuyển đến trang chủ. Nếu sai, hiển thị thông báo lỗi.

**+ Đăng ký**

Tên UseCase	Đăng ký
Tác nhân	Guest
Mô tả	Người dùng chưa có tài khoản sẽ đăng ký để sử dụng hệ thống
Điều kiện tiên quyết	Hệ thống đang hoạt động
Dòng dự kiện chính	1. Người dùng truy cập trang đăng ký 2. Nhập Username, Email và Password 4. Nhấn nút “Đăng ký”
Kết quả	Nếu hợp lệ, tài khoản được tạo và chuyển đến trang đăng nhập

+ Thêm địa điểm du lịch

Tên UseCase	Thêm địa điểm du lịch
Tác nhân	Admin
Mô tả	Người quản trị thêm địa điểm du lịch mới lên hệ thống
Điều kiện tiên quyết	Đã đăng nhập với vai trò Admin
Dòng dự kiện chính	1. Truy cập chức năng "Quản lý địa điểm" 2. Nhập thông tin địa điểm (tên, mô tả, hình ảnh, vị trí,...) 3. Nhấn nút “Thêm địa điểm”
Kết quả	Lưu lại thông tin và hiển thị trên hệ thống

+ Đăng bài viết

Tên UseCase	Đăng bài viết
Tác nhân	User
Mô tả	Người dùng đăng bài chia sẻ
Điều kiện tiên quyết	Đã đăng nhập
Dòng dự kiện chính	1. Truy cập mục “Đăng bài” 2. Nhập tiêu đề, nội dung, hình ảnh minh họa 3. Nhấn “Đăng bài”
Kết quả	Gửi yêu cầu kiểm duyệt bài viết lên hệ thống quản trị. Nếu bài viết được duyệt, bài viết sẽ được hiển thị lên hệ thống

+ Đánh giá địa điểm

Tên UseCase	Đánh giá địa điểm
Tác nhân	User
Mô tả	Người dùng viết nhận xét, đánh giá một địa điểm



Điều kiện tiên quyết	Đã đăng nhập
Dòng dự kiện chính	<ol style="list-style-type: none"> <li>1. Truy cập trang chi tiết địa điểm</li> <li>2. Nhập nội dung đánh giá và số sao (có thể đính kèm hình ảnh)</li> <li>3. Nhấn nút “Gửi đánh giá”</li> </ol>
Kết quả	Đánh giá được hiển thị công khai tại địa điểm

+ Bình luận bài viết

Tên UseCase	Bình luận bài viết
Tác nhân	User
Mô tả	Người dùng bình luận dưới bài viết chia sẻ của người khác
Điều kiện tiên quyết	Đã đăng nhập
Dòng dự kiện chính	<ol style="list-style-type: none"> <li>1. Truy cập bài viết</li> <li>2. Nhập nội dung bình luận</li> <li>3. Nhấn “Gửi”</li> </ol>
Kết quả	Bình luận hiển thị dưới bài viết

+ Báo cáo Bài viết/Bình luận/Đánh giá

Tên UseCase	Báo cáo nội dung (Bài viết/Bình luận/Đánh giá)
Tác nhân	User
Mô tả	Người dùng báo cáo bài viết, bình luận hoặc đánh giá vi phạm
Điều kiện tiên quyết	Đã đăng nhập
Dòng dự kiện chính	<ol style="list-style-type: none"> <li>1. Truy cập vào trang chi tiết Bài viết/Địa điểm</li> <li>2. Chọn mục muốn báo cáo (Bài viết/Bình luận/Đánh giá)</li> <li>3. Nhấn vào biểu tượng “Báo cáo”</li> </ol>

	4. Chọn lý do và gửi báo cáo
Kết quả	Báo cáo được gửi đến người quản trị

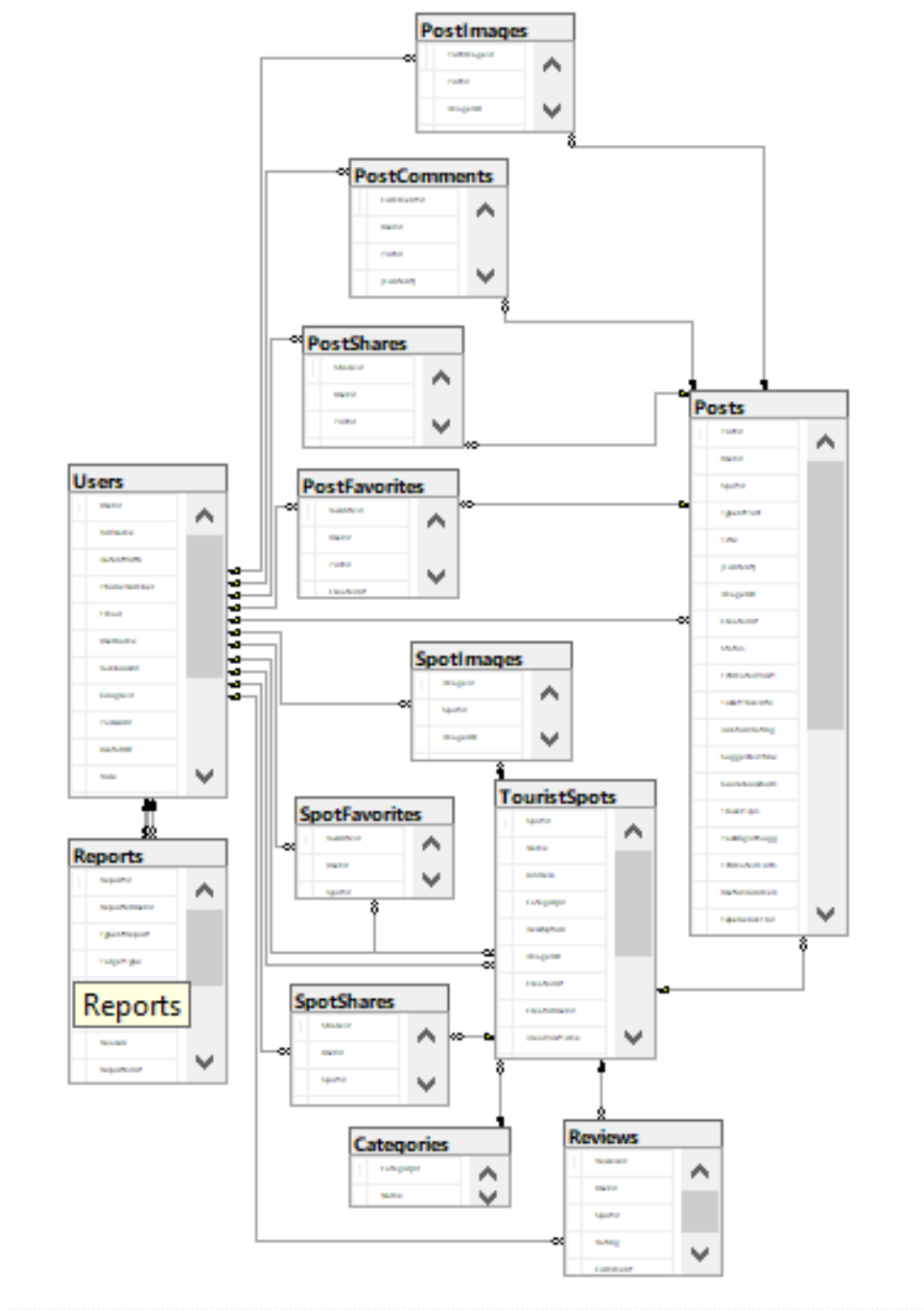
+ Yêu thích Bài viết/Địa điểm

Tên UseCase	Thêm Bài viết/Địa điểm vào danh sách yêu thích
Tác nhân	User
Mô tả	Người dùng lưu địa điểm/bài viết yêu thích để xem lại sau
Điều kiện tiên quyết	Đã đăng nhập
Dòng dữ kiện chính	1. Truy cập vào trang chi tiết Bài viết/Địa điểm 2. Nhấn biểu tượng “Yêu thích”
Kết quả	Địa điểm/bài viết được lưu vào danh sách yêu thích cá nhân

+ Chia sẻ Bài viết/Địa điểm

Tên UseCase	Chia sẻ
Tác nhân	User
Mô tả	Người dùng chia sẻ bài viết hoặc địa điểm lên mạng xã hội
Điều kiện tiên quyết	Đã đăng nhập
Dòng dữ kiện chính	1. Truy cập vào trang chi tiết Bài viết/Địa điểm 2. Nhấn vào nút “Chia sẻ” 3. Chọn nền tảng mạng xã hội
Kết quả	Đường link bài viết/địa điểm được chia sẻ

## 2. Thiết kế cơ sở dữ liệu



Hình 4: Data Diagram

### - BẢNG USERS (Người dùng)

+ Khóa chính: UserId (tăng tự động)

+ Các trường còn lại:

- FullName: Họ và tên đầy đủ của người dùng (Bắt buộc, tối đa 100 ký tự)
- DateOfBirth: Ngày sinh của người dùng (Tùy chọn)

- PhoneNumber: Số điện thoại (Tùy chọn, định dạng Phone)
- Email: Địa chỉ email (Bắt buộc, định dạng Email)
- Username: Tên đăng nhập (Bắt buộc, 3-50 ký tự, duy nhất)
- FacebookId: ID tài khoản Facebook (Tùy chọn, dành cho đăng nhập bằng Facebook)
- GoogleId: ID tài khoản Google (Tùy chọn, dành cho đăng nhập bằng Google)
- Password: Mật khẩu (Bắt buộc, tối thiểu 6 ký tự)
- AvatarUrl: Đường dẫn ảnh đại diện (Mặc định: "/images/default-avatar.png")
- Role: Vai trò người dùng (Mặc định: "User")
- CreatedAt: Thời gian tạo tài khoản (Mặc định: DateTime.Now)
- LastLoginAt: Thời gian đăng nhập lần cuối (Tùy chọn)
- Bio: Tiểu sử/Mô tả bản thân (Tùy chọn, tối đa 500 ký tự)
- UserStatus: Trạng thái người dùng (Mặc định: "Hoạt động")
- PasswordResetToken: Token đặt lại mật khẩu (Tùy chọn, tối đa 256 ký tự)
- PasswordResetTokenExpiry: Thời gian hết hạn token đặt lại mật khẩu (Tùy chọn)

#### - BẢNG CATEGORIES (Danh mục địa điểm)

+ Khóa chính: CategoryId (tăng tự động)

+ Các trường còn lại:

- Name: Tên danh mục (Bắt buộc, tối đa 50 ký tự, ví dụ: "Tỉnh/Thành phố du lịch", "Biển/Đảo",...)

#### - BẢNG TOURISTSPOTS (Địa điểm du lịch)

+ Khóa chính: SpotId (tăng tự động)

+ Khóa ngoại:

- CategoryId tham chiếu tới CategoryId trong bảng Categories: Danh mục của địa điểm
- CreatorUserId tham chiếu tới UserId trong bảng Users: Người quản trị tạo địa điểm

+ Các trường còn lại:

- Name: Tên địa điểm (Bắt buộc, tối đa 100 ký tự)
- Address: Địa chỉ địa điểm (Bắt buộc)
- Description: Mô tả chi tiết về địa điểm (Tùy chọn, dạng văn bản dài)
- ImageUrl: Ảnh đại diện của địa điểm (Mặc định: "/images/default-spotImage.png")
- CreatedAt: Thời gian tạo địa điểm (Mặc định: DateTime.Now)
- IdealVisitTime: Thời gian tham quan lý tưởng (Ví dụ: "Tháng 3-5 và 9-11")
- AvailableServices: Các dịch vụ có sẵn (Ví dụ: "Khách sạn, Nhà hàng, Tour")
- TravelTips: Mẹo du lịch (Dạng văn bản dài)
- MapEmbedUrl: URL nhúng bản đồ Google Maps
- VideoEmbedUrl: URL nhúng video YouTube giới thiệu

+ Trường tính toán:

- IsLikedByCurrentUser: Trạng thái yêu thích của người dùng hiện tại (Không lưu DB, tính toán runtime)

#### - BẢNG REVIEWS (Đánh giá địa điểm)

+ Khóa chính: ReviewId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người đánh giá
- SpotId tham chiếu tới SpotId trong bảng TouristSpots: Địa điểm được đánh giá

+ Các trường còn lại:

- Rating: Điểm đánh giá (Bắt buộc, từ 1-5 sao)
- Comment: Nội dung đánh giá (Bắt buộc)
- ImageUrl: Ảnh kèm theo đánh giá (Mặc định: "/images/default-review.png")
- CreatedAt: Thời gian tạo đánh giá (Mặc định: DateTime.Now)

#### - BẢNG POSTS (Bài viết về Địa điểm/Cảm nang/Trải nghiệm)

+ Khóa chính: PostId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Tác giả bài viết
- SpotId tham chiếu tới SpotId trong bảng TouristSpots: Địa điểm liên quan

+ Các trường còn lại:

- TypeOfPost: Loại bài viết (Bắt buộc: "Địa điểm", "Cẩm nang", "Trải nghiệm")
- Title: Tiêu đề bài viết (Bắt buộc, tối đa 100 ký tự)
- Content: Nội dung chính (Bắt buộc, dạng văn bản dài)
- ImageUrl: Ảnh đại diện bài viết (Mặc định: "/images/default-postImage.png")
- CreatedAt: Thời gian tạo bài viết (Mặc định: DateTime.Now)
- Status: Trạng thái duyệt bài (Enum: Pending/Approved/Rejected, Mặc định: Pending)

+ Các trường cho loại "Địa điểm":

- EstimatedVisitTime: Thời gian tham quan ước tính (Ví dụ: "2-3 giờ")
- TicketPriceInfo: Thông tin giá vé (Ví dụ: "Miễn phí", "100.000 VND")
- LocationRating: Điểm đánh giá địa điểm (0-5)
- SuggestedItinerary: Lịch trình gợi ý

+ Các trường cho loại "Cẩm nang":

- GuidebookSummary: Tóm tắt cẩm nang
- TravelTips: Mẹo du lịch
- PackingListSuggestions: Gợi ý đồ mang theo
- EstimatedCosts: Chi phí tham khảo
- UsefulDocumentsHtml: Tài liệu hữu ích (dạng HTML)

+ Các trường cho loại "Trải nghiệm":

- ExperienceEndDate: Ngày kết thúc trải nghiệm
- Companions: Người đồng hành (Ví dụ: "Gia đình", "Bạn bè")
- ApproximateCost: Chi phí ước tính

- OverallExperienceRating: Đánh giá tổng quan (0-10)
- RatingLandscape: Điểm cảnh quan (0-5)
- RatingFood: Điểm ẩm thực (0-5)
- RatingService: Điểm dịch vụ (0-5)
- RatingPrice: Điểm giá cả (0-5)
- ExperienceHighlights: Điểm nổi bật
- ExperienceItinerarySummary: Tóm tắt hành trình
- Advice: Lời khuyên

**- BẢNG POSTCOMMENTS (Bình luận bài viết)**

+ Khóa chính: CommentId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người bình luận
- PostId tham chiếu tới PostId trong bảng Posts: Bài viết được bình luận

+ Các trường còn lại:

- Content: Nội dung bình luận (Bắt buộc)
- ImageUrl: Ảnh kèm theo bình luận (Mặc định: "/images/default-comment.png")
- CreatedAt: Thời gian tạo bình luận (Mặc định: DateTime.Now)

**- BẢNG SPOTIMAGES (Ảnh địa điểm)**

+ Khóa chính: ImageId (tăng tự động)

+ Khóa ngoại:

- SpotId tham chiếu tới SpotId trong bảng TouristSpots: Địa điểm chứa ảnh
- UploadedBy tham chiếu tới UserId trong bảng Users: Người tải ảnh lên

+ Các trường còn lại:

- ImageUrl: Đường dẫn ảnh (Bắt buộc, Mặc định: "/images/default-spotImage.png")
- UploadedAt: Thời gian tải ảnh (Mặc định: DateTime.Now)

**- BẢNG POSTIMAGES (Ảnh bài viết)**

+ Khóa chính: PostImageId (tăng tự động)

+ Khóa ngoại:

- PostId tham chiếu tới PostId trong bảng Posts: Bài viết chứa ảnh
- UploadedBy tham chiếu tới UserId trong bảng Users: Người tải ảnh lên

+ Các trường còn lại:

- ImageUrl: Đường dẫn ảnh (Bắt buộc, Mặc định: "/images/default-postImage.png")
- UploadedAt: Thời gian tải ảnh (Mặc định: DateTime.Now)

#### **- BẢNG SPOTFAVORITES (Yêu thích địa điểm)**

+ Khóa chính: FavoriteId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người yêu thích
- SpotId tham chiếu tới SpotId trong bảng TouristSpots: Địa điểm được yêu thích

+ Các trường còn lại:

- CreatedAt: Thời gian thêm vào yêu thích (Mặc định: DateTime.Now)

#### **- BẢNG POSTFAVORITES (Yêu thích bài viết)**

+ Khóa chính: FavoriteId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người yêu thích
- PostId tham chiếu tới PostId trong bảng Posts: Bài viết được yêu thích

+ Các trường còn lại:

- CreatedAt: Thời gian thêm vào yêu thích (Mặc định: DateTime.Now)

#### **- BẢNG SPOTSHARES (Chia sẻ địa điểm)**

+ Khóa chính: ShareId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người chia sẻ
- SpotId tham chiếu tới SpotId trong bảng TouristSpots: Địa điểm được chia sẻ



+ Các trường còn lại:

- SharedOn: Nền tảng chia sẻ (Bắt buộc: "Facebook", "Twitter", "Instagram", "Email")
- SharedAt: Thời gian chia sẻ (Mặc định: DateTime.Now)

**- BẢNG POSTSHARES (Chia sẻ bài viết)**

+ Khóa chính: ShareId (tăng tự động)

+ Khóa ngoại:

- UserId tham chiếu tới UserId trong bảng Users: Người chia sẻ
- PostId tham chiếu tới PostId trong bảng Posts: Bài viết được chia sẻ

+ Các trường còn lại:

- SharedOn: Nền tảng chia sẻ (Bắt buộc: "Facebook", "Twitter", "Instagram", "Email")
- SharedAt: Thời gian chia sẻ (Mặc định: DateTime.Now)

**- BẢNG REPORTS (Báo cáo vi phạm)**

+ Khóa chính: ReportId (tăng tự động)

+ Khóa ngoại:

- ReporterUserId tham chiếu tới UserId trong bảng Users: Người báo cáo
- ReportedUserId tham chiếu tới UserId trong bảng Users: Người bị báo cáo
- AdminUserId tham chiếu tới UserId trong bảng Users: Người quản trị xử lý

+ Các trường còn lại:

- TypeOfReport: Loại báo cáo (Enum: Spam, InappropriateContent, IncorrectInformation, CopyrightInfringement, Harassment, HateSpeech, Other)
- TargetType: Loại đối tượng bị báo cáo (Enum: Post, Comment, Review, User, Spot)
- TargetId: ID của đối tượng bị báo cáo
- Reason: Lý do báo cáo (Bắt buộc, tối đa 1000 ký tự)
- ReportedAt: Thời gian báo cáo (Mặc định: DateTime.Now)

- Status: Trạng thái xử lý (Enum: Pending, Resolved, Dismissed, Mặc định: Pending)
- ResolvedAt: Thời gian xử lý
- AdminNotes: Ghi chú của admin (Tùy chọn, tối đa 500 ký tự)

## **- MỐI QUAN HỆ CHÍNH GIỮA CÁC BẢNG**

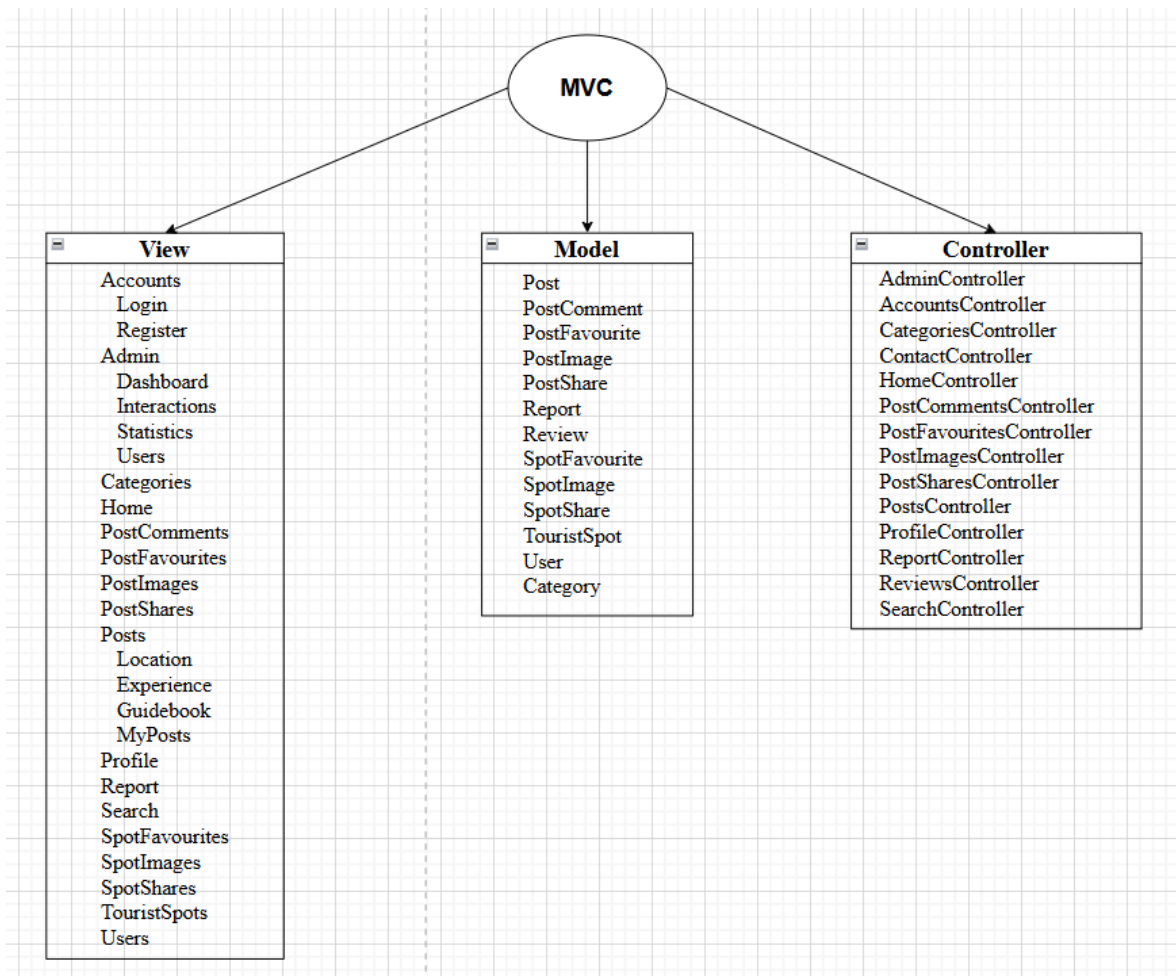
### **+ Quan hệ 1-N (One-to-Many):**

- Users → Reviews: Một người dùng có thể tạo nhiều đánh giá
- Users → Posts: Một người dùng có thể tạo nhiều bài viết
- Users → PostComments: Một người dùng có thể tạo nhiều bình luận
- Users → TouristSpots: Một người quản trị có thể tạo nhiều địa điểm
- Categories → TouristSpots: Một danh mục có thể chứa nhiều địa điểm
- TouristSpots → Reviews: Một địa điểm có thể có nhiều đánh giá
- TouristSpots → Posts: Một địa điểm có thể có nhiều bài viết liên quan
- Posts → PostComments: Một bài viết có thể có nhiều bình luận

### **+ Quan hệ N-N (Many-to-Many) thông qua bảng trung gian:**

- Users ↔ TouristSpots (thông qua SpotFavorites): Người dùng yêu thích địa điểm
- Users ↔ Posts (thông qua PostFavorites): Người dùng yêu thích bài viết
- Users ↔ TouristSpots (thông qua SpotShares): Người dùng chia sẻ địa điểm
- Users ↔ Posts (thông qua PostShares): Người dùng chia sẻ bài viết

### 3. Cấu trúc thư mục dự án



Hình 5: Sơ đồ cấu trúc thư mục

Dự án được xây dựng theo mô hình kiến trúc **MVC (Model - View - Controller)** của ASP.NET Core. Toàn bộ mã nguồn được tổ chức thành các thư mục chính với chức năng rõ ràng, hỗ trợ cho việc phát triển, bảo trì và mở rộng hệ thống. Cụ thể:

- **Controllers (Bộ điều khiển):** Chứa các lớp Controller chịu trách nhiệm điều hướng và xử lý các yêu cầu HTTP từ người dùng. Mỗi controller tương ứng với một nhóm chức năng cụ thể:

+ Nhóm quản lý bài viết:

- PostsController.cs: CRUD bài viết
- PostCommentsController.cs: Bình luận trên bài viết
- PostImagesController.cs: Hình ảnh đính kèm bài viết
- PostFavoritesController.cs: Tính năng yêu thích bài viết
- PostSharesController.cs: Chia sẻ bài viết

+ Nhóm quản lý địa điểm du lịch:

- TouristSpotsController.cs: CRUD địa điểm du lịch
- CategoriesController.cs: Danh mục của địa điểm
- ReviewsController.cs: Đánh giá cho địa điểm du lịch
- SpotImagesController.cs: Hình ảnh đính kèm địa điểm
- SpotFavoritesController.cs: Tính năng yêu thích địa điểm
- SpotSharesController.cs: Chia sẻ địa điểm du lịch

+ Nhóm quản lý người dùng:

- UsersController.cs: Thao tác với thông tin người dùng
- AccountsController.cs: Xử lý đăng ký, đăng nhập
- ProfileController.cs: Quản lý hồ sơ cá nhân

+ Các controller chức năng khác:

- HomeController.cs: Trang chủ và điều hướng chính
- AdminController.cs: Quản trị hệ thống
- SearchController.cs: Tìm kiếm thông tin
- ReportController.cs: Báo cáo và thống kê
- ContactController.cs: Hiện thị trang liên hệ

Tổng cộng có **19 controller**, thể hiện quy mô lớn và tính phân chia rõ ràng theo module chức năng.

- **Models (Mô hình dữ liệu):** Chứa các lớp đại diện cho dữ liệu và logic nghiệp vụ của hệ thống:

+ Các entity chính:

- User.cs: Thông tin người dùng
- Post.cs: Bài viết
- TouristSpot.cs: Địa điểm du lịch
- Category.cs: Danh mục phân loại
- Report.cs: Báo cáo vi phạm
- Review.cs, PostComment.cs: Đánh giá và bình luận

- PostImage.cs, SpotImage.cs: Quản lý hình ảnh
- PostFavorite.cs, SpotFavorite.cs: Mục yêu thích
- PostShares.cs, SpotShares.cs: Chia sẻ

+ Cấu hình hệ thống:

- ApplicationDbContext.cs: Lớp kết nối CSDL sử dụng Entity Framework Core
- ErrorViewModel.cs: Xử lý hiển thị lỗi

+ Thư mục con:

- ViewModels/: Chứa các lớp trung gian truyền dữ liệu từ Controller đến View
- Weather/: Xử lý dữ liệu thời tiết (tích hợp API thời tiết)

- **Views (Giao diện người dùng):** Chứa các **template giao diện** viết bằng Razor (.cshtml), được tổ chức theo cấu trúc tương ứng với Controllers:

+ Nhóm Views bài viết:

- Posts/, PostComments/, PostImages/, PostFavorites/, PostShares/: Giao diện quản lý bài viết và các tính năng liên quan

+ Nhóm Views địa điểm:

- TouristSpots/, Reviews/, SpotImages/, SpotFavorites/, SpotShares/: Giao diện quản lý địa điểm du lịch

+ Nhóm Views người dùng:

- Users/, Profile/, Accounts/: Giao diện tài khoản và hồ sơ người dùng

+ Views chức năng chính:

- Home/: Trang chủ và landing page
- Admin/: Trang quản trị hệ thống
- Search/: Giao diện tìm kiếm
- Categories/: Phân loại danh mục
- Report/: Báo cáo vi phạm

+ Files cấu hình View:

- Shared/: Chứa layout và partial views dùng chung
- \_ViewImports.cshtml: Import các namespace và tag helpers

- `_ViewStart.cshtml`: Thiết lập layout mặc định
- **wwwroot (Tài nguyên tĩnh)**: Thư mục public chứa các tài nguyên tĩnh được trình duyệt truy cập trực tiếp:
  - `css/`, `cssadmin/`: Stylesheet tùy chỉnh
  - `js/`: JavaScript files
  - `images/`: Hình ảnh, logo
  - `lib/`: Thư viện frontend (Bootstrap, jQuery, etc.)
  - `uploads/posts`: File upload ảnh đính kèm bài viết từ người dùng
- **Services (Tầng dịch vụ)**: Chứa các lớp xử lý business logic và truy cập dữ liệu (data access), giúp tách biệt hoàn toàn khỏi Controller nhằm tăng tính mô-đun và dễ bảo trì:

Trong dự án, có sử dụng Email service:

  - `IEmailSender.cs`: Interface định nghĩa các phương thức gửi email.
  - `SmtplibEmailSender.cs`: Lớp hiện thực việc gửi email thông qua giao thức SMTP (hữu ích trong chức năng liên hệ, khôi phục mật khẩu...).
- **Utilities (Tiện ích)**: Tập hợp các helper classes, tiện ích mở rộng, và hàm dùng chung được sử dụng nhiều nơi trong toàn dự án:
  - `FileHelper.cs`: chứa các phương thức xử lý tệp tin (upload, xóa, tạo đường dẫn, kiểm tra định dạng...).
- **Helpers (Hỗ trợ)**: Chứa các lớp hỗ trợ custom helper và tiện ích bổ sung cho giao diện, mô hình dữ liệu hoặc binding:
  - `EnumExtensions.cs`: Class mở rộng để hiển thị Enum dưới dạng chuỗi có ý nghĩa (dùng trong dropdown hoặc view).
- **Migrations (Quản lý Database)**: Lưu trữ các **Entity Framework migration files** để:
  - Theo dõi thay đổi database schema
  - Đồng bộ cấu trúc CSDL giữa các môi trường
  - Hỗ trợ version control cho database
- **Cấu hình dự án**
  - + Files khởi động:

- Program.cs: Entry point và dependency injection configuration
- TourismWeb.csproj: Project file định nghĩa dependencies
- TourismWeb.sln: Solution file cho Visual Studio

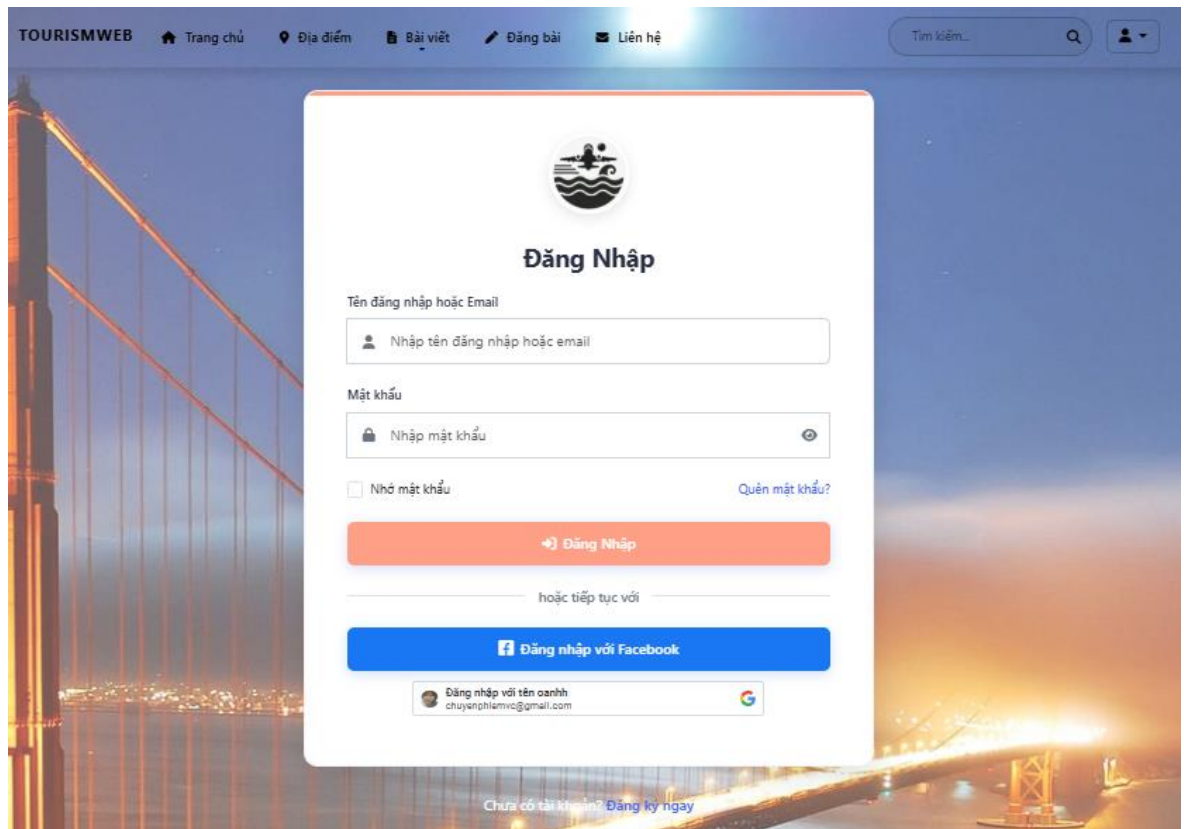
+ Files cấu hình:

- appsettings.json: Cấu hình production (connection strings, API keys)
- appsettings.Development.json: Cấu hình development environment
- .gitignore: Loại trừ files không cần version control
- .hintrc: Cấu hình code analysis và suggestions

#### IV. TRIỂN KHAI ỨNG DỤNG

Đăng nhập: Thông tin đăng nhập bao gồm username/email và password, dựa vào thông tin đăng nhập hệ thống sẽ kiểm tra vai trò người dùng và điều hướng đến giao diện người dùng hoặc quản lý tương ứng.

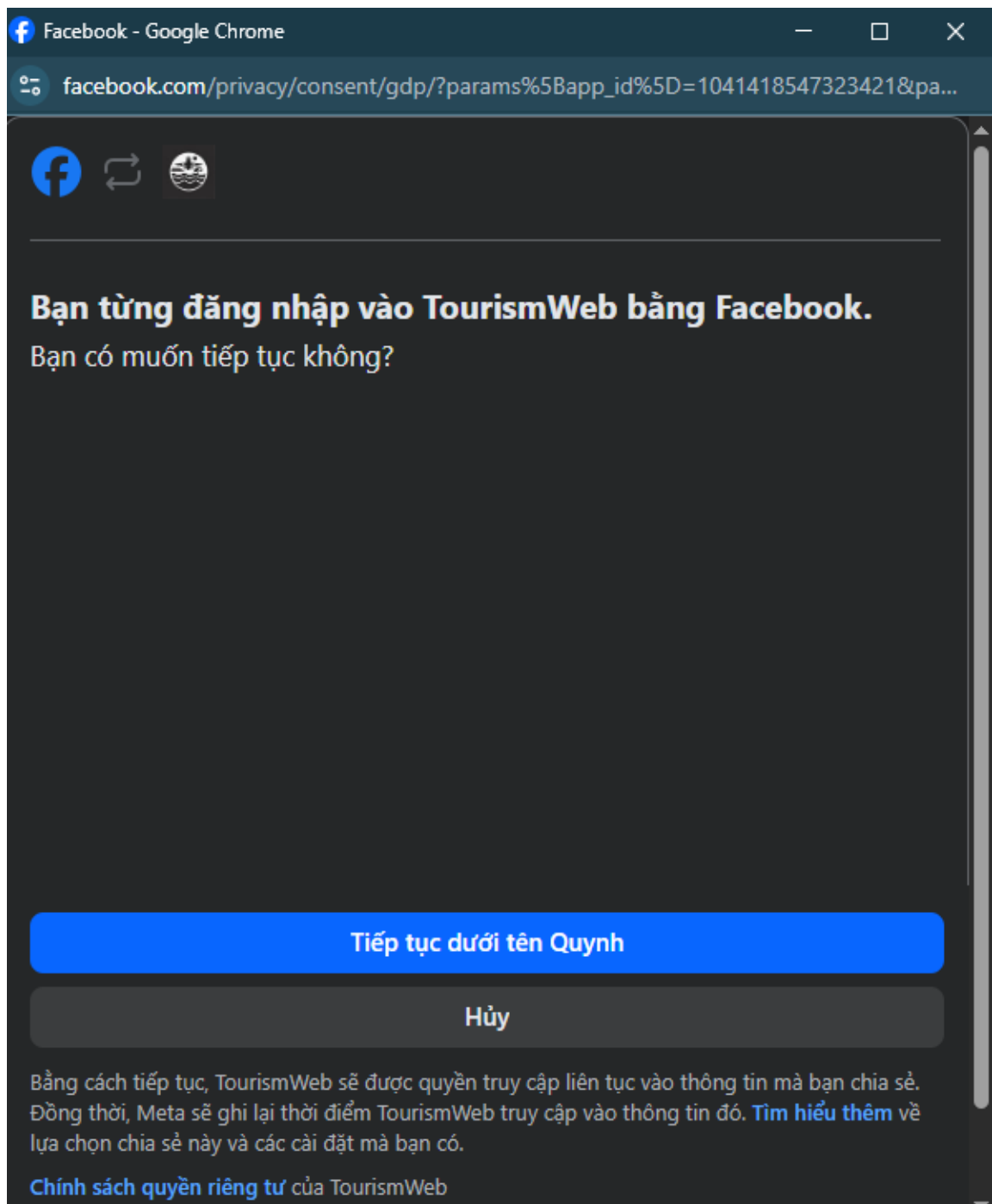
- Đăng nhập truyền thống (Username/Email & Password)



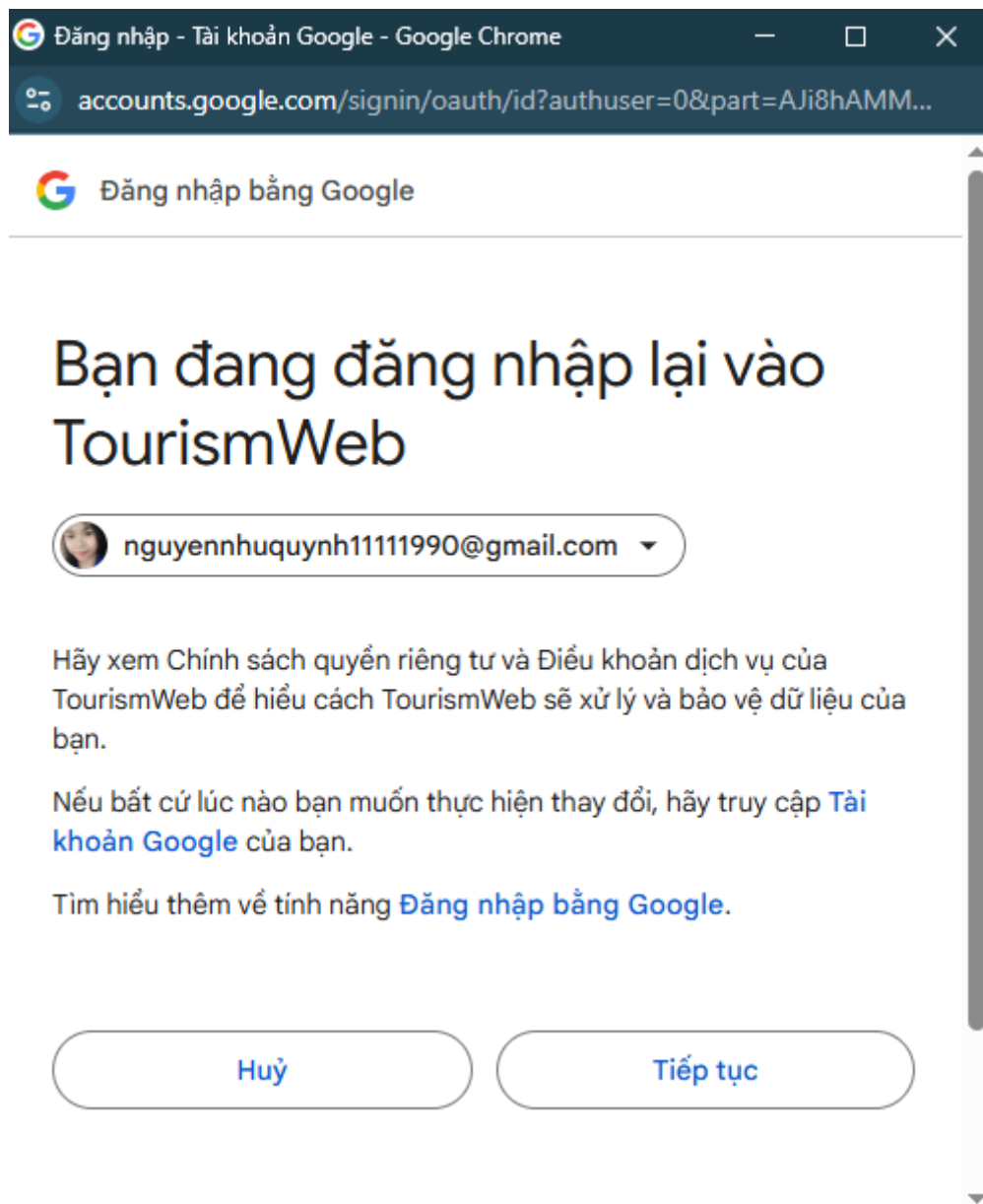
Hình 6: Giao diện đăng nhập



- Đăng nhập qua mạng xã hội (Facebook, Google)



Hình 7: Giao diện đăng nhập bằng Facebook



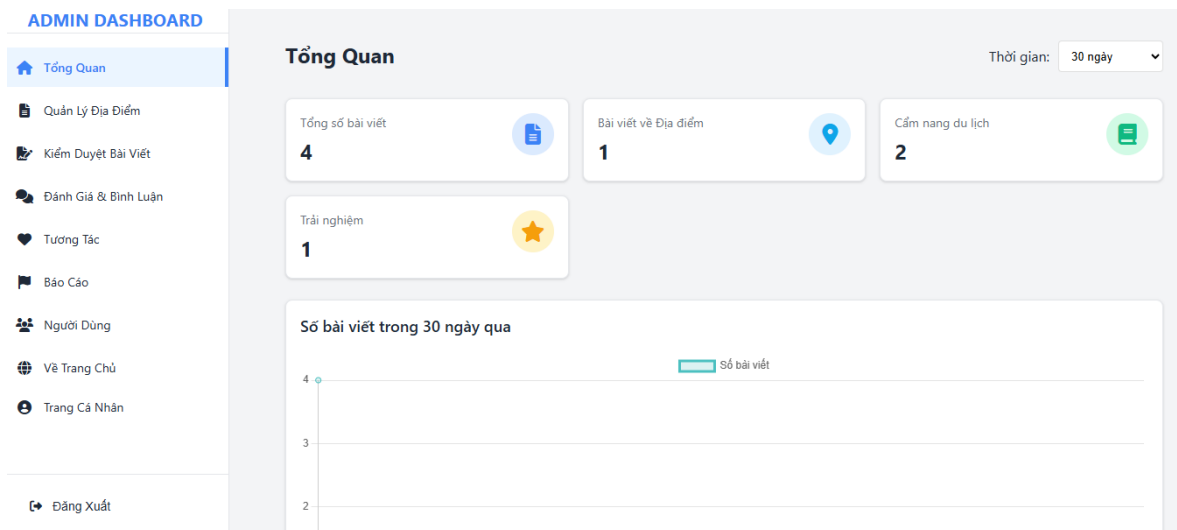
Hình 8: Giao diện đăng nhập bằng Google

### 1. Người quản lý (Admin)

Đăng nhập với vai trò “Quản lý” bao gồm các chức năng:

#### a. Trang tổng quan

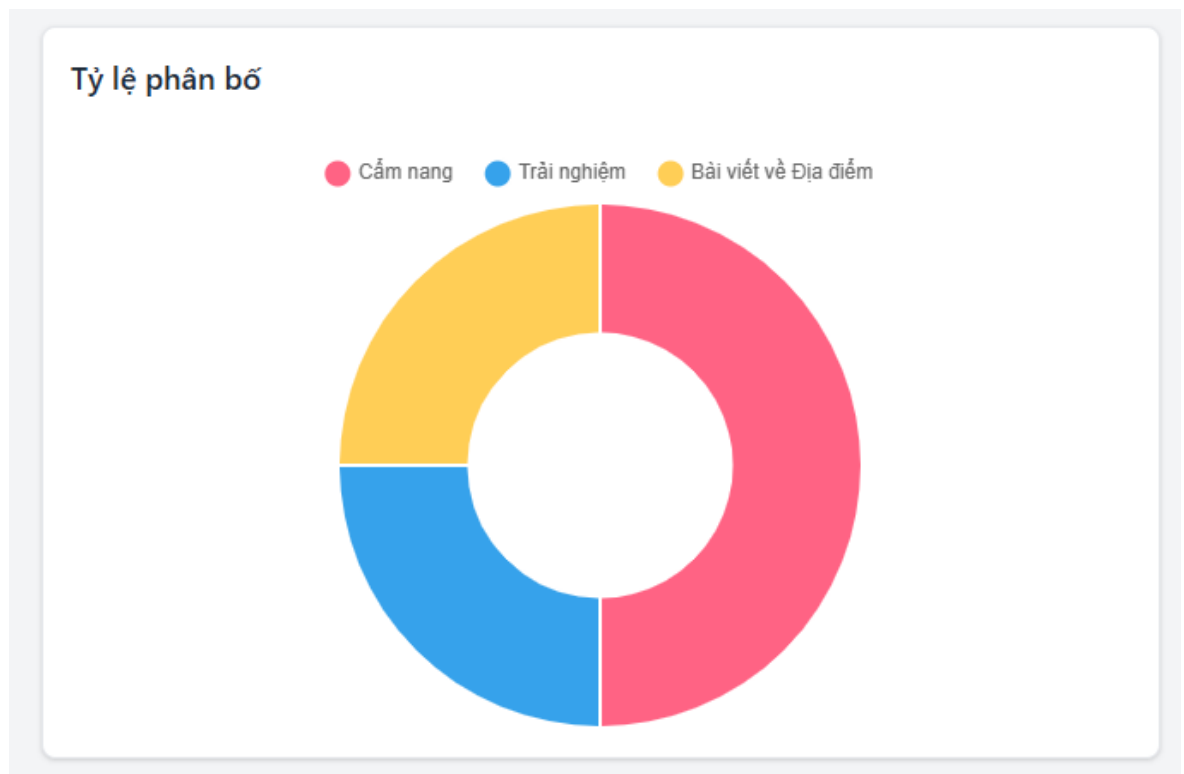
- **Thẻ thống kê:** Hiện thị số liệu tổng hợp



Hình 9: Giao diện Trang tổng quan của Admin





### - Biểu đồ trực quan:

- Xu hướng bài viết theo thời gian (30 ngày, 7 ngày, tất cả, tùy chỉnh)
- Biểu đồ tròn phân bố loại nội dung



Hình 10: Biểu đồ tỷ lệ phân bố bài viết

- **Bảng hoạt động:** Liệt kê các cập nhật mới nhất với các thông tin đi kèm

Hoạt động gần đây					<a href="#">Xem tất cả</a>
TIÊU ĐỀ	LOẠI	TÁC GIẢ	NGÀY ĐĂNG	HÀNH ĐỘNG	
Phú Yên – Nét đẹp nguyên sơ níu chân lữ khách	 Địa điểm	oanhha	17/05/2025 00:50	<a href="#">Xem</a>	
Biển xanh, đảo đẹp, hải sản rẻ, dịch vụ tốt – Nha Trang đáng để đi ít nhất một lần trong đời!	 Trải nghiệm	oanhha	17/05/2025 00:47	<a href="#">Xem</a>	
Phố cổ Hội An: Bước chân vào ký ức thương cảng sầm uất một thời	 Cẩm nang	chouanh	17/05/2025 00:39	<a href="#">Xem</a>	
Hướng dẫn du lịch Huế từ A-Z	 Cẩm nang	nhuquynh	17/05/2025 00:37	<a href="#">Xem</a>	

Hình 11: Bảng hoạt động gần đây

## b. Quản lý Địa điểm Du lịch

- **Chức năng thêm mới:**

- Thông tin cơ bản: Tên, địa chỉ, danh mục
- Nội dung: Mô tả chi tiết, hình ảnh đại diện
- Thông tin bổ sung: Thời gian tham quan, dịch vụ, mẹo du lịch
- Tích hợp: Google Maps, Video YouTube

- **Giao diện:** Form nhập liệu đầy đủ với các nút "Tạo địa điểm" và "Quay lại"

Thêm địa điểm du lịch mới

Thông tin cơ bản

Tên địa điểm

\*

Nhập tên địa điểm du lịch

Địa chỉ

\*

Nhập địa chỉ chi tiết

Danh mục

\*

-- Chọn danh mục --

Hình ảnh

Xem trước ảnh

Chọn tệp

Không có tệp nào được chọn

Hoặc URL:

/images/default-spotimage.png

Mô tả

\*

Nhập mô tả chi tiết về địa điểm du lịch...

Thông tin bổ sung

Thời gian lý tưởng

VD: Tháng 3 - Tháng 5

Dịch vụ có sẵn

VD: Khách sạn, Nhà hàng (cách nhau bởi dấu phẩy)

Mẹo du lịch

Nhập các mẹo du lịch, mỗi mẹo một dòng...

Bản đồ (Google Maps)

Dán URL src từ iframe Google Maps

Video (YouTube)

Dán URL src từ iframe YouTube

Tạo địa điểm

Hình 12: Thêm địa điểm du lịch mới

### c. Kiểm duyệt Bài viết

#### - Bộ lọc trạng thái:













- Tất cả bài viết
- Chờ duyệt
- Đã duyệt
- Từ chối

- **Bảng quản lý:** Hiện thị STT, tiêu đề, loại, tác giả, ngày đăng, trạng thái

- **Hành động:** Xem, sửa, xóa từng bài viết

**Quản Lý Bài Viết**

Tất cả Chờ duyệt Đã duyệt Từ chối

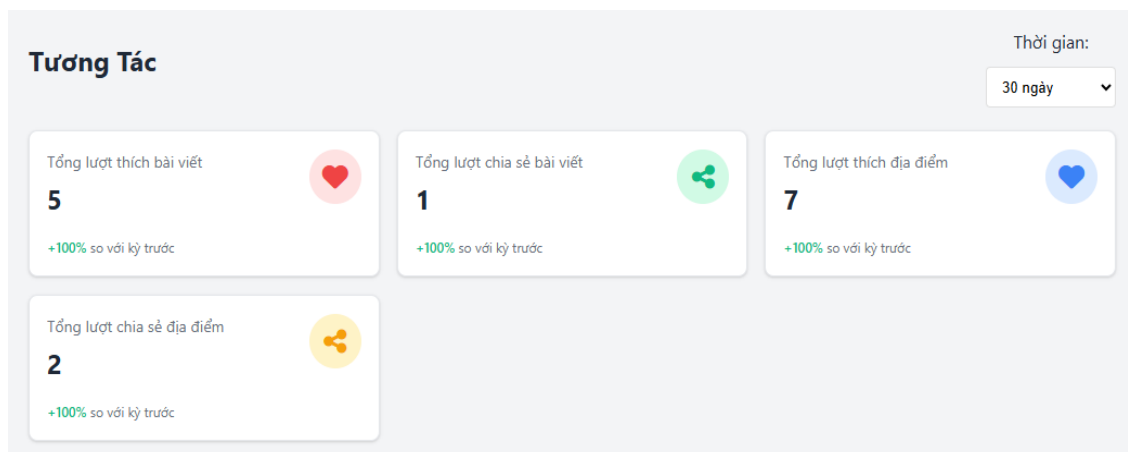
STT	TIÊU ĐỀ	LOẠI	TÁC GIẢ	NGÀY ĐĂNG	TRẠNG THÁI	HÀNH ĐỘNG
1	Phú Yên – Nét đẹp nguyên sơ niu chân lữ khách Phú Yên không phải là nơi ồn ào, hào nhoáng như...	Địa điểm	oanhha	17/05/2025	Đã duyệt	  
2	Biển xanh, đảo đẹp, hải sản rẻ, dịch vụ tốt – Nha Trang đáng để đi ít nhất một lần trong đời! Ngày 1: Chạm mặt thành phố biển – thư giãn v...	Trải nghiệm	oanhha	17/05/2025	Đã duyệt	  
3	Phố cổ Hội An: Bước chân vào ký ức thương cảng sầm uất một thời Lịch trình tham quan gợi ý Dưới đây là lịch trình g...	Cắm nang	chouanh	17/05/2025	Đã duyệt	  
4	Hướng dẫn du lịch Huế từ A-Z Thời điểm lý tưởng nhất để du lịch Huế: Tháng 1 - ...	Cắm nang	nhuquynh	17/05/2025	Đã duyệt	  

Hình 13: Quản lý bài viết

### d. Thống kê Tương tác

#### - Số liệu tổng hợp:

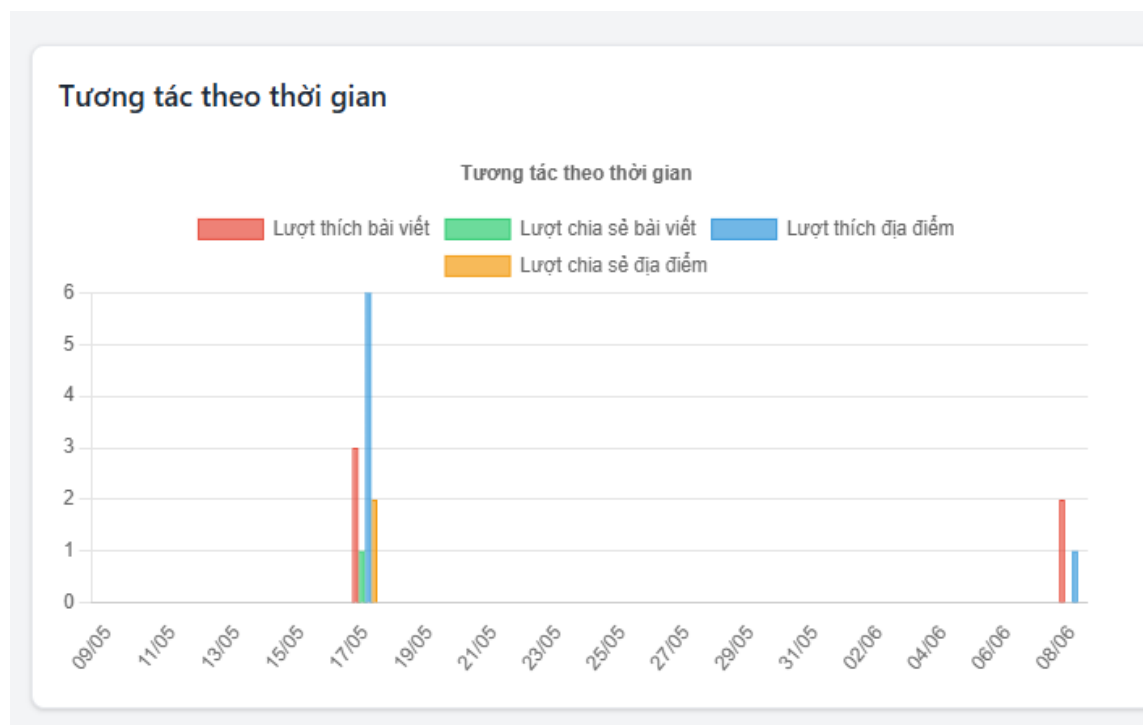
- Lượt thích/chia sẻ cho bài viết và địa điểm
- Bộ lọc thời gian linh hoạt



Hình 14: Trang Tương tác

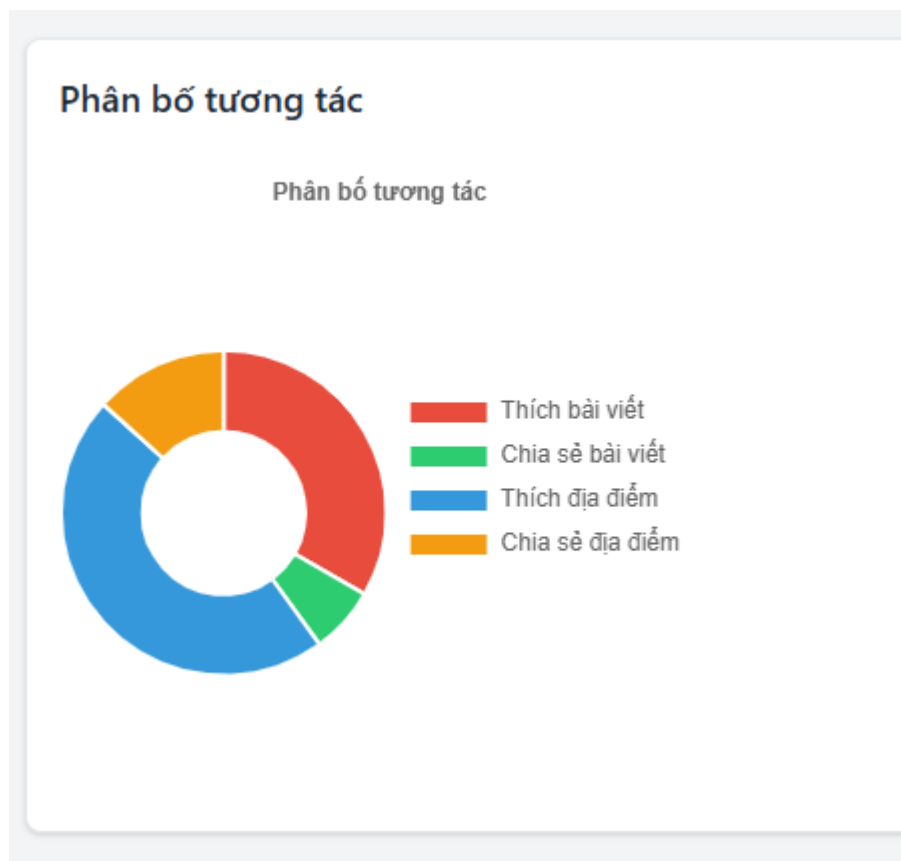
**- Biểu đồ phân tích:**

- Xu hướng tương tác theo thời gian



Hình 15: Biểu đồ tương tác theo thời gian

- Tỷ trọng các loại tương tác



Hình 16: Biểu đồ phân bố tương tác

- **Bảng xếp hạng:** Top bài viết, địa điểm, người dùng có nhiều tương tác

Bài viết có nhiều tương tác nhất					Xem tất cả
TIÊU ĐỀ	LOẠI	LƯỢT THÍCH	LƯỢT CHIA SẺ	TỔNG TƯƠNG TÁC	
<a href="#">Phố cổ Hội An: Bước chân vào ký ức thương cảng sầm uất một thời</a>	Cẩm nang	3	0	3	
<a href="#">Phú Yên – Nét đẹp nguyên sơ níu chân lữ khách</a>	Địa điểm	1	1	2	
<a href="#">Biển xanh, đảo đẹp, hải sản rẻ, dịch vụ tốt – Nha Trang đáng để đi ít nhất một lần trong đời!</a>	Trải nghiệm	1	0	1	
<a href="#">Hướng dẫn du lịch Huế từ A-Z</a>	Cẩm nang	0	0	0	

Hình 17: Bài viết có nhiều tương tác nhất



Địa điểm du lịch có nhiều tương tác nhất					<a href="#">Xem tất cả</a>
TÊN ĐỊA ĐIỂM	DANH MỤC	LƯỢT THÍCH	LƯỢT CHIA SẺ	TỔNG TƯƠNG TÁC	
<a href="#">Huế</a>	Tỉnh/Thành phố du lịch	3	1	4	
<a href="#">Đà Nẵng</a>	Tỉnh/Thành phố du lịch	1	1	2	
<a href="#">Phố cổ Hội An</a>	Tỉnh/Thành phố du lịch	2	0	2	
<a href="#">Nha Trang</a>	Biển - Đảo	1	0	1	
<a href="#">Phú Yên</a>	Biển - Đảo	0	0	0	

Hình 18: Địa điểm có nhiều tương tác nhất

Người dùng tương tác nhiều nhất						<a href="#">Xem tất cả</a>
NGƯỜI DÙNG	THÍCH BÀI VIẾT	CHIA SẺ BÀI VIẾT	THÍCH ĐỊA ĐIỂM	CHIA SẺ ĐỊA ĐIỂM	TỔNG TƯƠNG TÁC	
<b>nhuquynh</b> nhuquynh@gmail.com	3	1	2	1	7	
<b>chouanh</b> chouanh@gmail.com	1	0	3	1	5	
<b>oanhha</b> oanh@gmail.com	1	0	2	0	3	
<b>abc</b> abc@gmail.com	0	0	0	0	0	
<b>nguyen</b> nguyen@gmail.com	0	0	0	0	0	

Hình 19: Người dùng tương tác nhiều nhất













### e. Quản lý Báo cáo vi phạm

#### - Công cụ tìm kiếm:

- Thanh tìm kiếm báo cáo
- Lọc theo loại báo cáo, đối tượng, trạng thái

- **Bảng chi tiết:** ID, người báo cáo, loại vi phạm, đối tượng bị báo, lý do, ngày báo cáo

- **Trạng thái xử lý:** Chờ xử lý, đã xử lý, đã bỏ qua

ADMIN DASHBOARD									
Tổng Quan	Quản Lý Báo Cáo								
Quản Lý Địa Điểm	Trạng thái: <span>Tất cả</span>								
Kiểm Duyệt Bài Viết	<input type="text" value="Tìm kiếm báo cáo..."/> <span>Tất cả loại báo cáo</span> <span>Tất cả đối tượng</span> <span>Lọc</span>								
Đánh Giá & Bình Luận	ID	NGƯỜI BÁO CÁO	LOẠI BÁO CÁO	ĐỐI TƯỢNG	NGƯỜI BỊ BÁO CÁO	LÝ DO (TÓM TẮT)	NGÀY BÁO CÁO	TRẠNG THÁI	HÀNH ĐỘNG
Tương Tác	12	 chouanh chouanh@gmail.com	Ví phạm bản quyền	Bài viết (0: 6)	 nhuquynh nhuquynh@gmail.com	của tôi	06/06/2025 17:33	Đã xử lý	
Báo Cáo	11	 chouanh chouanh@gmail.com	Thông tin không chính xác	Đánh giá (0: 2)	 nhuquynh nhuquynh@gmail.com	sai	06/06/2025 17:32	Đã xử lý	
Người Dùng	10	 nhuquynh nhuquynh@gmail.com	Nghôn từ thô ghet	Bình luận (0: 4)	 oanhha oanh@gmail.com	ghet	06/06/2025 17:28	Đã xử lý	
Về Trang Chủ	9	 oanhha oanh@gmail.com	Khác	Người dùng (0: 5)	 nhuquynh nhuquynh@gmail.com	khác	16/05/2025 19:39	Đã xử lý	
Đăng Xuất									

Hình 20: Quản lý Báo cáo vi phạm











## f. Quản lý Người dùng

### - Công cụ quản lý:

- Tìm kiếm người dùng
- Lọc theo vai trò và trạng thái

- **Thông tin hiển thị:** Tên, email, vai trò, ngày tham gia, trạng thái hoạt động













- **Hành động:** Xem chi tiết, chỉnh sửa, xóa tài khoản

ADMIN DASHBOARD									
Tổng Quan	Quản Lý Người Dùng								
Quản Lý Địa Điểm	<input type="text" value="Tìm kiếm người dùng..."/> <span>Tất cả vai trò</span> <span>Tất cả trạng thái</span> <span>Lọc</span>								
Kiểm Duyệt Bài Viết	<input type="checkbox"/>	NGƯỜI DÙNG	EMAIL	VAI TRÒ	BÀI VIẾT	NGÀY THAM GIA	TRẠNG THÁI	HÀNH ĐỘNG	
Đánh Giá & Bình Luận	<input type="checkbox"/>	 Như Quỳnh nguyennhuquynh11111990@gmail.com	nguyennhuquynh11111990@gmail.com	User	0	17/05/2025	Hoạt động		
Tương Tác	<input type="checkbox"/>	 nhuquynh @nhuquynh	nhuquynh@gmail.com	User	1	16/05/2025	Hoạt động		
Báo Cáo	<input type="checkbox"/>	 chouanh @chouanh	chouanh@gmail.com	User	1	16/05/2025	Hoạt động		
Người Dùng	<input type="checkbox"/>	 oanhha @oanhha	oanh@gmail.com	User	2	16/05/2025	Hoạt động		
Về Trang Chủ	<input type="checkbox"/>	 nguyen @nguyen	nguyen@gmail.com	Admin	0	16/05/2025	Hoạt động		
Đăng Xuất	<div>Hiện thị</div>								













Hình 21: Quản lý người dùng

**g. Đánh giá và Bình luận**

- **Quản lý nội dung:** Đảm bảo nội dung phù hợp và có giá trị

Đánh Giá & Bình Luận						
<div>★ Đánh Giá (Địa điểm)    💬 Bình Luận (Bài viết)</div> <div><input type="text" value="Tìm nội dung, người dùng..."/> <input type="text" value="Tất cả đánh giá (sao)"/> <input type="button" value="Lọc"/></div>						
NGƯỜI DÙNG	NỘI DUNG	ẢNH ĐÁNH GIÁ	ĐỊA ĐIỂM	ĐÁNH GIÁ (SAO)	NGÀY ĐĂNG	HÀNH ĐỘNG
 <b>nhuquynh</b> nhuquynh@gmail.com	Biển Quy Nhơn siêu...		Biển Quy Nhơn	★★★★★ 5.0	08/06/2025 20:08	 
 <b>oanhha</b> oanh@gmail.com	Rất đẹp		Phố cổ Hội An	★★★★★ 5.0	17/05/2025 01:00	 
 <b>chouanh</b> chouanh@gmail.com	Cũng được		Nha Trang	★★★★☆ 4.0	17/05/2025 00:59	 

Hình 22: Quản lý Đánh giá

Đánh Giá & Bình Luận						
<div>★ Đánh Giá (Địa điểm)    💬 Bình Luận (Bài viết)</div> <div><input type="text" value="Tìm nội dung, người dùng..."/> <input type="text" value="Tất cả loại bài viết"/> <input type="button" value="Lọc"/></div>						
NGƯỜI DÙNG	NỘI DUNG	ẢNH BÌNH LUẬN	BÀI VIẾT	LOẠI BÀI VIẾT	NGÀY ĐĂNG	HÀNH ĐỘNG
 <b>nhuquynh</b> nhuquynh@gmail.com	Bài viết rất hữu...		Phố cổ Hội An: Bước chân vào kỷ ức thương cảng sầm uất một thời	Cảm nang	08/06/2025 19:54	 
 <b>chouanh</b> chouanh@gmail.com	Hay		Biển xanh, đảo đẹp, hải sản rẻ, dịch vụ tốt – Nha Trang đáng để đi it nhất một lần trong đời!	Trải nghiệm	17/05/2025 00:59	 
 <b>nhuquynh</b> nhuquynh@gmail.com	Bài viết hữu ích		Phú Yên – Nét đẹp nguyên sơ níu chân lữ khách	Địa điểm	17/05/2025 00:57	 

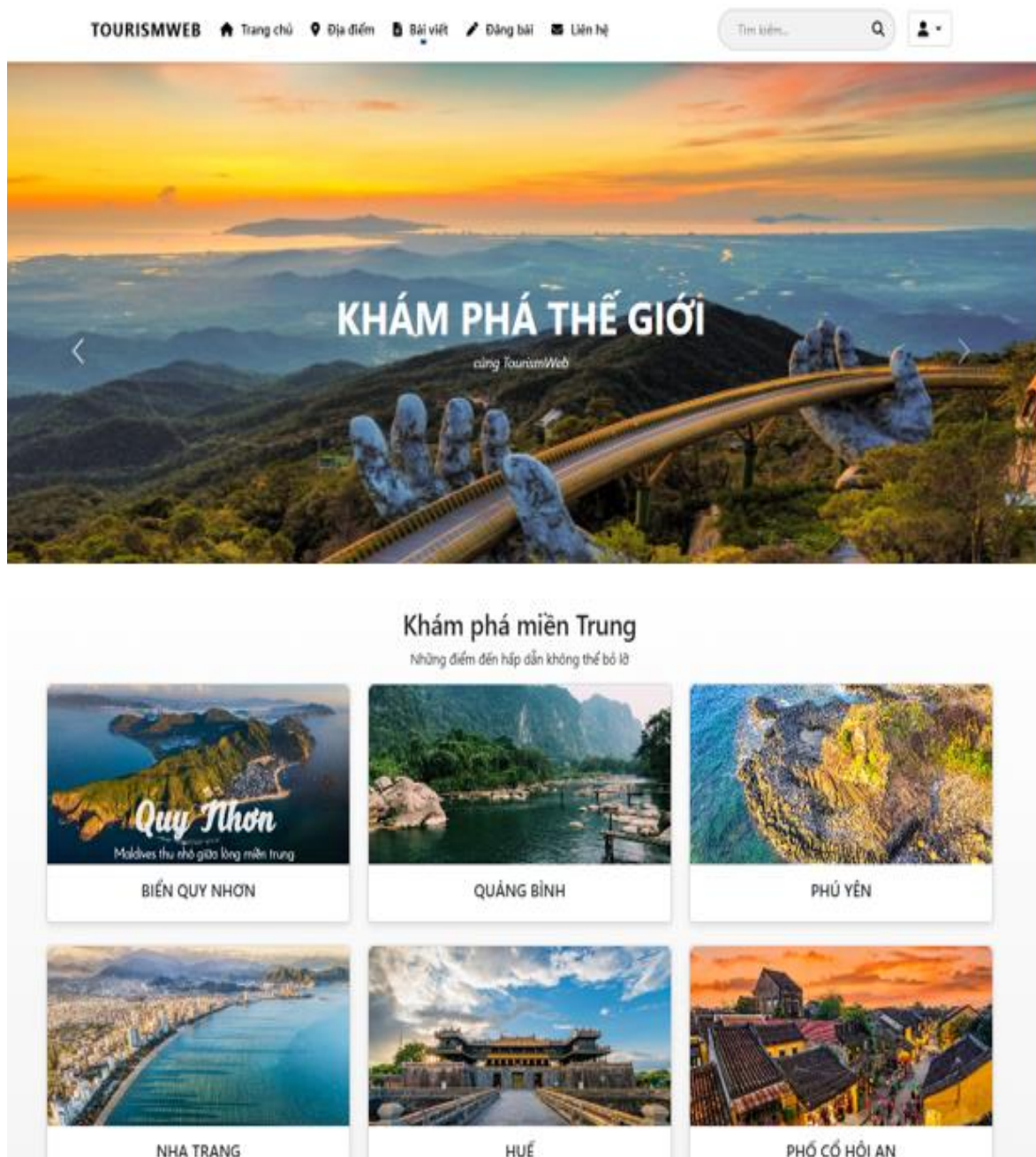
Hình 23: Quản lý Bình luận

## 2. Người dùng (User)

Giao diện người dùng được thiết kế hiện đại, đơn giản và dễ sử dụng, hỗ trợ đầy đủ các thao tác liên quan đến việc tìm kiếm địa điểm, tương tác cộng đồng và chia sẻ trải nghiệm du lịch. Giao diện tương thích trên cả máy tính và thiết bị di động.

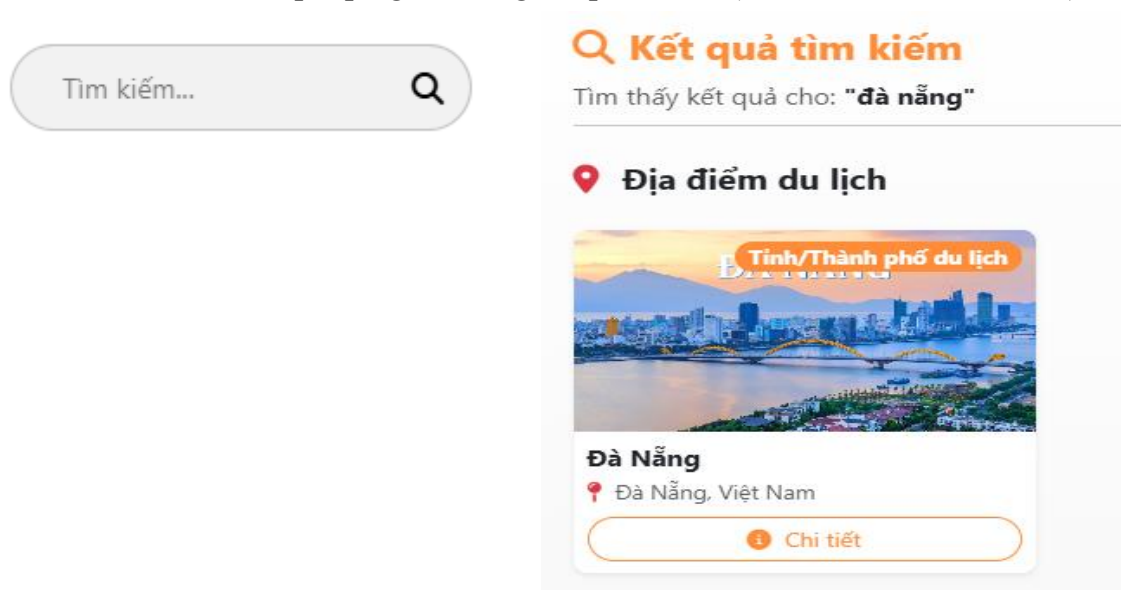
### a. Trang chủ và tìm kiếm địa điểm

- Hiện thị danh sách địa điểm du lịch nổi bật.



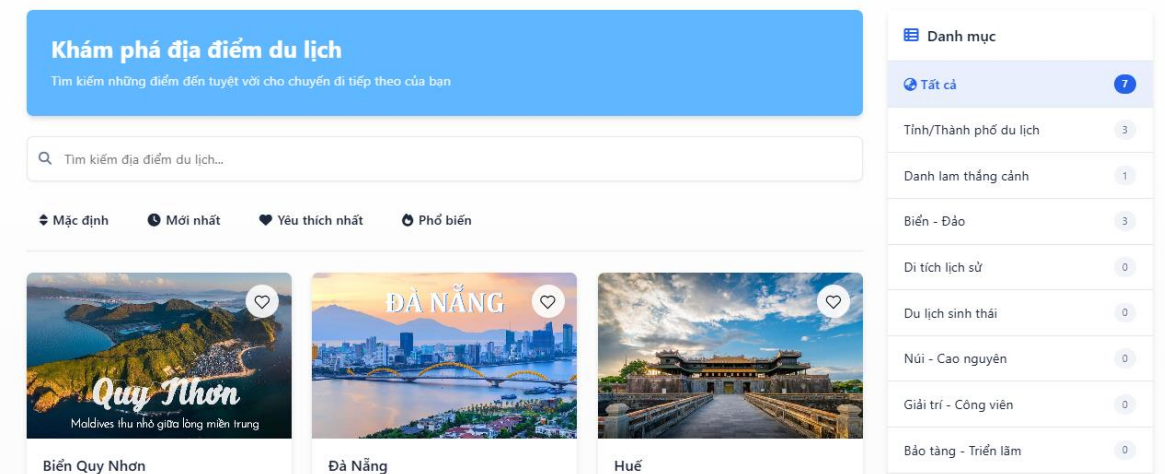
Hình 24: Giao diện Trang chủ

- Thanh tìm kiếm cho phép người dùng nhập từ khóa (tên địa điểm, tỉnh thành).



Hình 25: Chức năng Tìm kiếm

- Bộ lọc thông minh theo danh mục, theo thời gian thêm mới nhất, được yêu thích hoặc phổ biến.

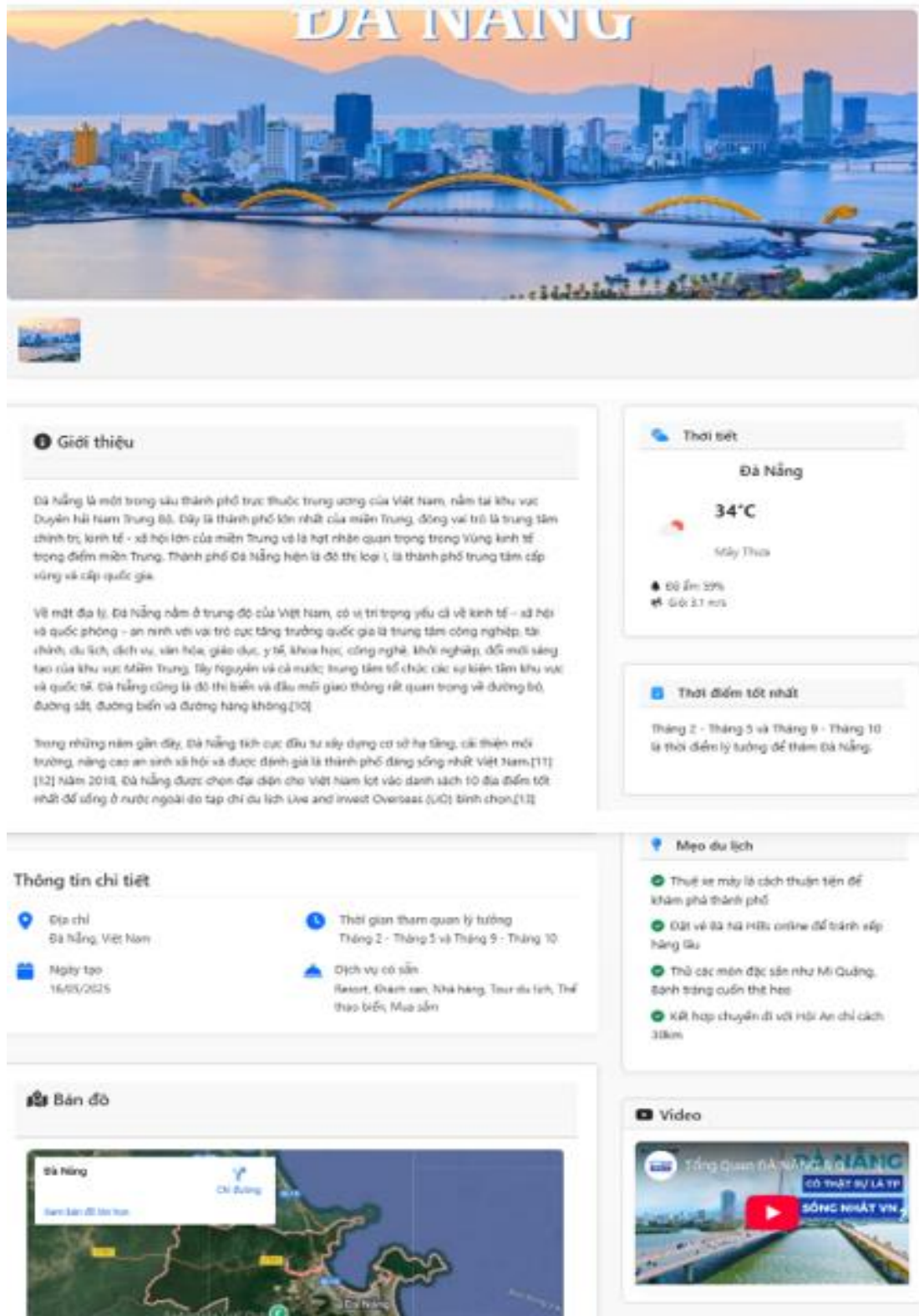


Hình 26: Bộ lọc địa điểm

## b. Xem chi tiết địa điểm

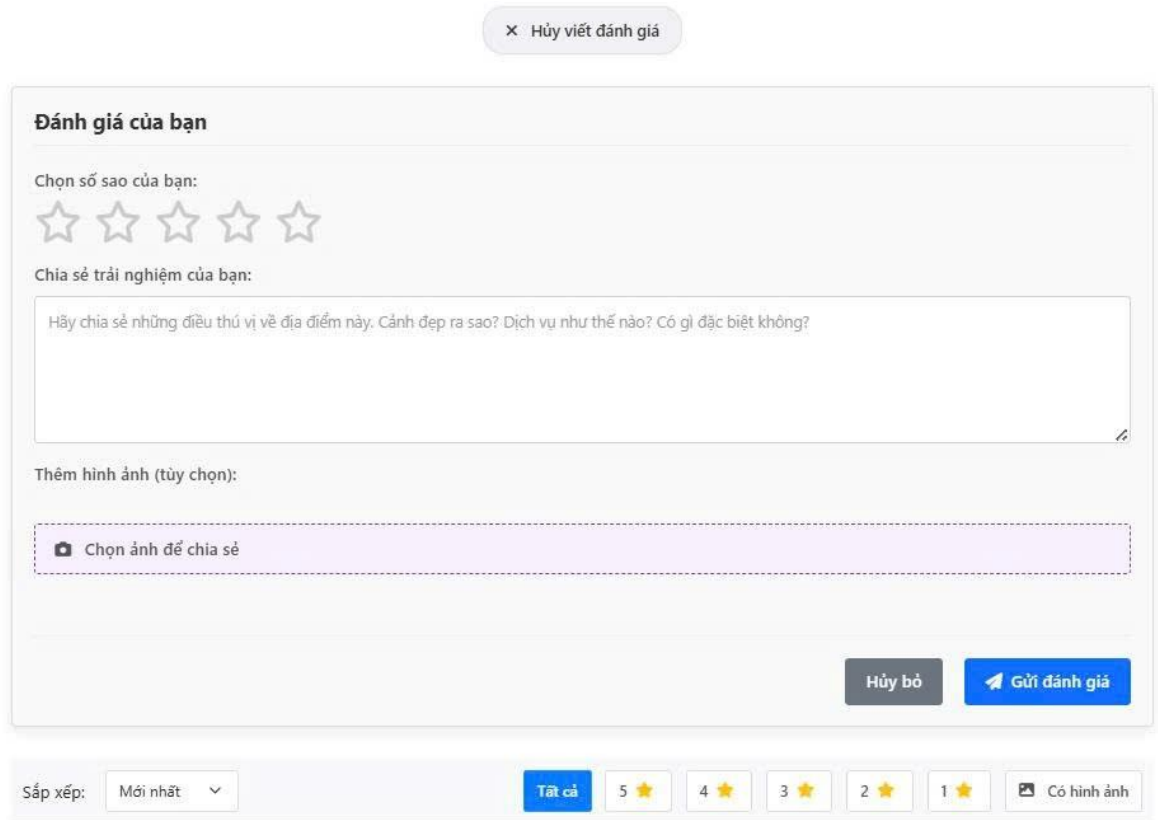
- Hiện thị đầy đủ thông tin như tên địa điểm, mô tả, hình ảnh, bản đồ vị trí (Google Maps), thời tiết, mẹo du lịch, video.





Hình 27: Chi tiết địa điểm

- **Chức năng đánh giá địa điểm:** Người dùng có thể chọn mức đánh giá từ 1 đến 5 sao, viết một đoạn nhận xét và đính kèm hình ảnh.



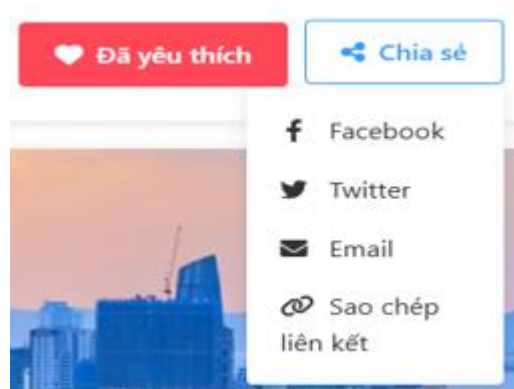
Form đánh giá địa điểm với các trường:

- Đánh giá của bạn
- Chọn số sao của bạn: 5 ngôi sao trống
- Chia sẻ trải nghiệm của bạn:   
Hãy chia sẻ những điều thú vị về địa điểm này. Cảnh đẹp ra sao? Dịch vụ như thế nào? Có gì đặc biệt không?
- Thêm hình ảnh (tùy chọn):   
Chọn ảnh để chia sẻ
- Nút: Hủy bỏ, Gửi đánh giá
- Sắp xếp: Mới nhất
- Nút: Tất cả
- Chọn số sao: 5, 4, 3, 2, 1
- Nút: Có hình ảnh

Hình 28: Đánh giá địa điểm

- **Chức năng yêu thích địa điểm:** Người dùng có thể lưu địa điểm vào danh sách “Địa điểm yêu thích”.

- **Chức năng chia sẻ địa điểm:** Chia sẻ nhanh qua Facebook, Twitter, Gmail hoặc sao chép liên kết để gửi cho bạn bè.



Hình 29: Yêu thích/Chia sẻ Địa điểm

### c. Viết bài chia sẻ

- Giao diện viết bài gồm:

- Nhập tiêu đề
- Chọn địa điểm du lịch liên quan
- Chọn loại bài viết (Cẩm nang, Trải nghiệm, Địa điểm)
- Nhập nội dung chi tiết
- Tải ảnh bìa và ảnh phụ minh họa
- Người dùng phải đồng ý điều khoản sử dụng trước khi đăng.

The screenshot shows a form for creating a new article. It has several sections: 'Tiêu đề bài viết' (Article Title) with a text input; 'Địa điểm du lịch' (Travel Location) and 'Loại bài viết' (Article Type) with dropdown menus; 'Nội dung chi tiết' (Detailed Content) with a large text area; 'Ảnh bìa bài viết' (Article Cover Image) with a 'Chọn ảnh đẹp' (Choose beautiful image) button; and 'Ảnh phụ / Hình ảnh bổ sung' (Additional Images / Supplementary Images) with a 'Chọn tệp' (Choose file) button and a note about image quality. At the bottom, there is a checkbox for 'Tôi đồng ý với điều khoản sử dụng' (I agree with the terms of use) and buttons for 'Hủy bỏ' (Cancel) and 'Đăng bài viết' (Publish article).

Hình 30: Viết bài chia sẻ

- Sau khi đăng bài, hệ thống sẽ chuyển bài viết sang trạng thái "chờ duyệt".
- Admin có quyền phê duyệt, chỉnh sửa hoặc từ chối hiển thị nếu nội dung vi phạm.

### Bài viết của tôi

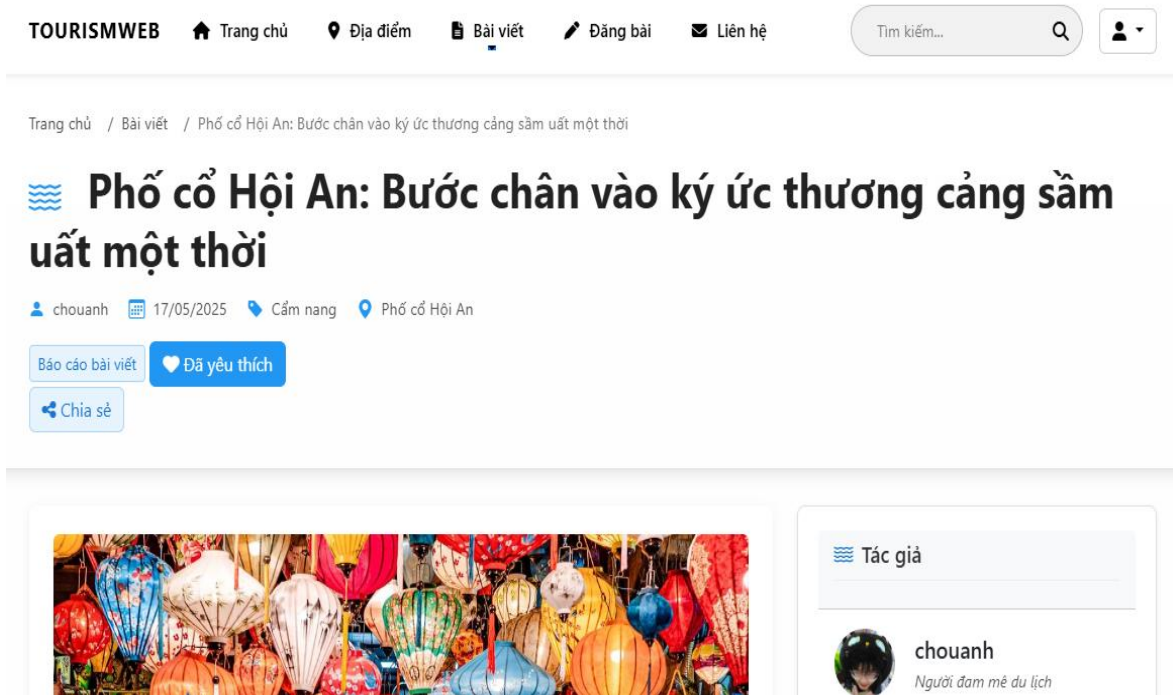
The screenshot shows the 'Bài viết của tôi' (My Articles) section. It features a blue button 'Tạo bài viết mới' (Create new article). Below is a card for an article titled 'Hướng dẫn du lịch Huế từ A-Z' (A-Z Guide to Hue Travel). The card displays a thumbnail image, the location 'Huế', type 'Cẩm nang' (Guide), and creation date '17/05/2025 00:37'. A yellow 'Chờ duyệt' (Waiting for review) badge is in the top right corner. At the bottom of the card are buttons for 'Chi tiết' (Details), 'Sửa' (Edit), and 'Xóa' (Delete).

Hình 31: Bài viết của tôi kèm trạng thái



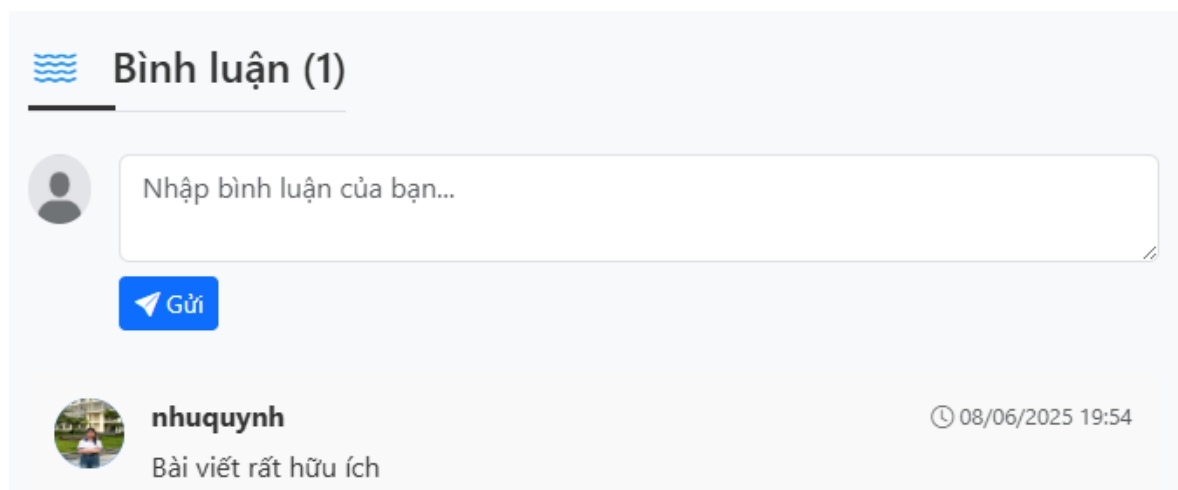
#### d. Tương tác cộng đồng

- Trên mỗi bài viết được hiển thị công khai, người dùng có thể:
  - **Yêu thích (thả tim)** bài viết → Bài viết được lưu vào mục “Bài viết yêu thích”.



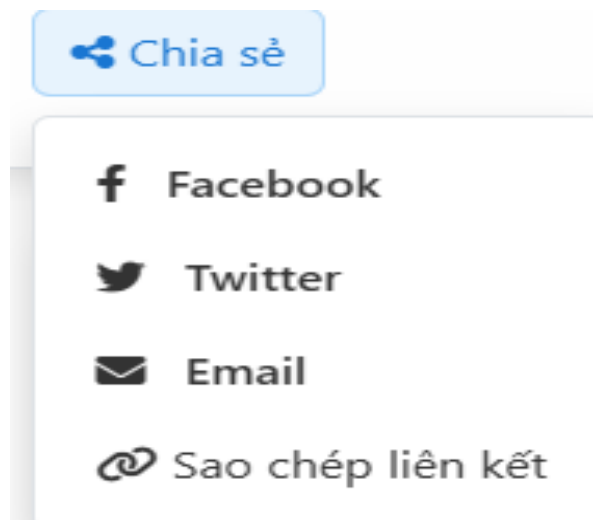
Hình 32: Tương tác yêu thích

- **Bình luận** bài viết để chia sẻ ý kiến hoặc đặt câu hỏi.



Hình 33: Bình luận bài viết

- **Chia sẻ bài viết** qua Facebook, Twitter, Gmail hoặc sao chép liên kết.



Hình 34: Chia sẻ bài viết

- **Báo cáo bài viết** nếu phát hiện nội dung không phù hợp.

A screenshot of a "Báo cáo nội dung" (Report Content) form. The title "Báo cáo nội dung" is in bold blue text. Below it is the subtitle "Giúp chúng tôi duy trì cộng đồng an toàn và tích cực". The form has two main sections: "Loại báo cáo" (Report Type) with a dropdown menu showing "-- Chọn loại báo cáo --", and "Lý do chi tiết" (Detailed Reason) with a text area containing the placeholder "Vui lòng mô tả chi tiết lý do báo cáo để chúng tôi có thể xử lý một cách phù hợp nhất...". There are yellow warning triangle icons next to both sections. At the bottom is a red button with a bell icon and the text "Gửi báo cáo".

Hình 35: Báo cáo bài viết

### e. Trang cá nhân (User Profile)

- Người dùng có một khu vực riêng để quản lý hoạt động:
  - **Thông tin tài khoản:** Họ tên, ngày sinh, email, số điện thoại – có thể chỉnh sửa.

The screenshot displays a user profile interface. On the left, there is a circular profile picture of a person, the username 'nhuquynh', a status 'Chưa có tiểu sử' with a plus icon, and a post count '1 Bài viết'. The main area is titled 'Thông tin tài khoản' and 'Mật khẩu & Bảo mật'. Under 'Thông tin tài khoản', the 'Dữ liệu cá nhân' section includes a 'Tên đầy đủ' field with 'nhuquynh', a 'Ngày sinh' section with dropdowns for 'Ngày', 'Tháng', and 'Năm', and 'Hãy thay đổi' and 'Lưu thay đổi' buttons. The 'Email' section shows 'nhuquynh@gmail.com' with a 'Nhận nhận thông báo' button. The 'Số di động' section shows a field with '0000000000' and 'Hãy thay đổi' and 'Lưu thay đổi' buttons.

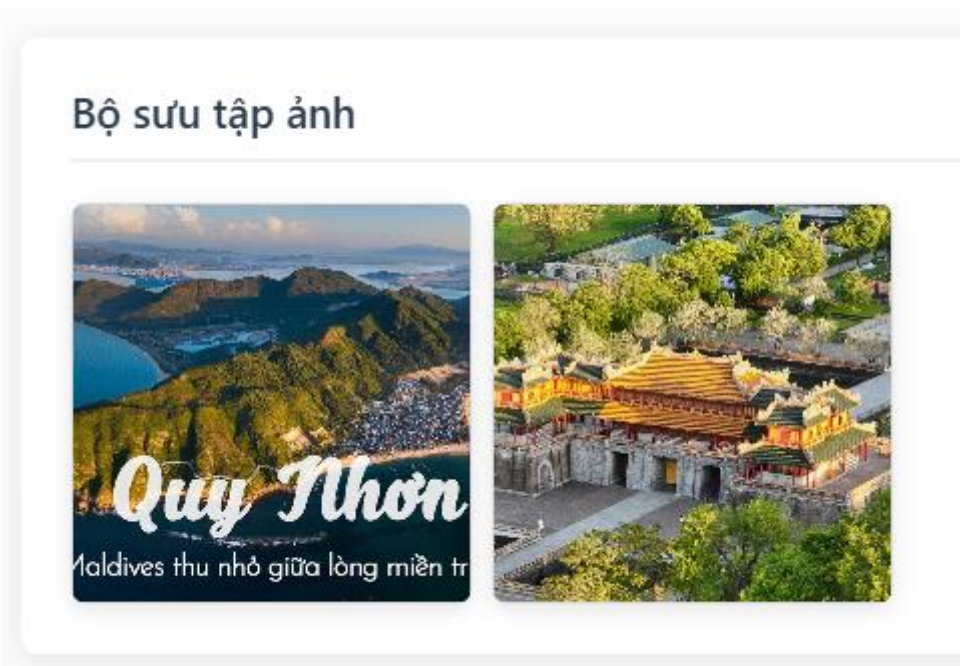
Hình 36: Thông tin cá nhân

- **Đổi mật khẩu:** Yêu cầu xác thực mật khẩu cũ trước khi cập nhật.

The screenshot shows the 'Mật khẩu & Bảo mật' section of the user profile. It includes a sub-header 'Mật khẩu & Bảo mật' and a description 'Quản lý mật khẩu và các tùy chọn bảo mật cho tài khoản của bạn'. Below this, there are three input fields: 'Mật khẩu hiện tại', 'Mật khẩu mới', and 'Xác nhận mật khẩu mới'. A blue button labeled 'Cập nhật mật khẩu' is located at the bottom right.

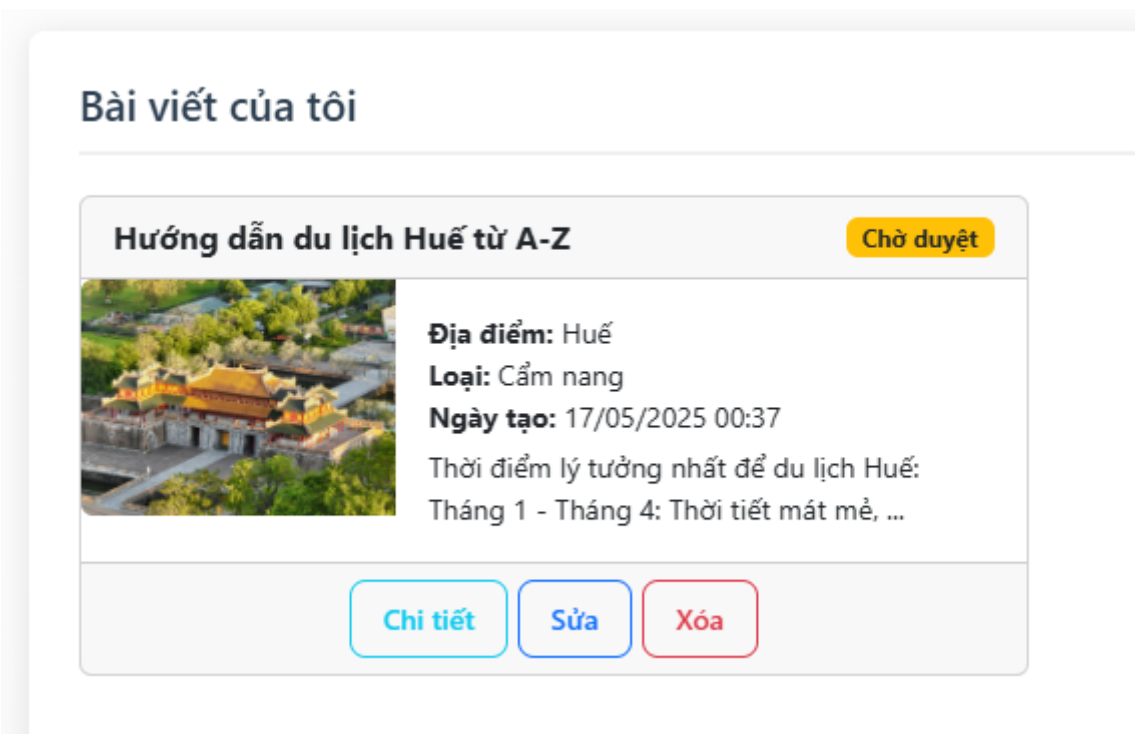
Hình 37: Đổi mật khẩu

- **Bộ sưu tập ảnh:** Tập hợp ảnh người dùng đã tải lên trong các bài viết hoặc đánh giá.



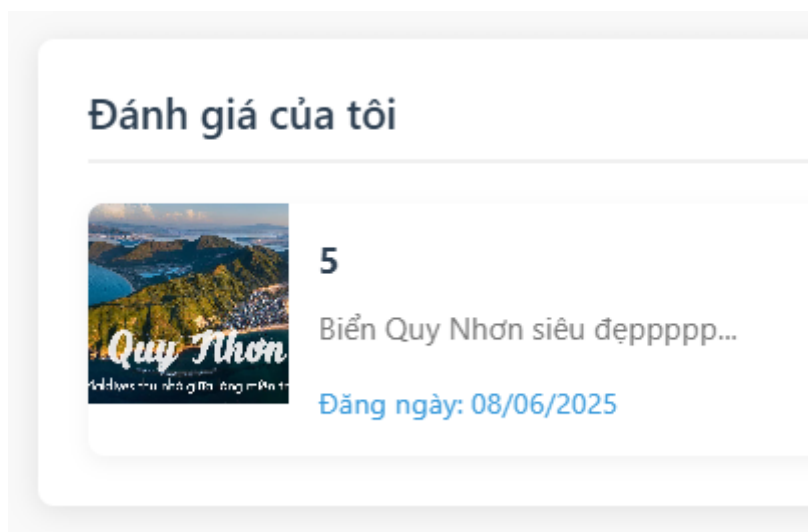
Hình 38: Bộ sưu tập ảnh

- **Bài viết của tôi:** Liệt kê tất cả bài viết của người dùng kèm trạng thái (đã đăng/chờ duyệt/bị từ chối), có thể sửa hoặc xóa.



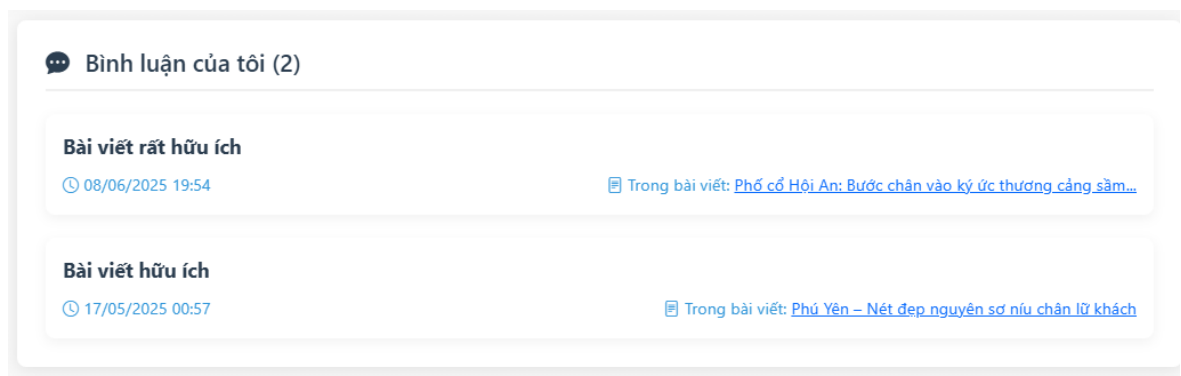
Hình 39: Bài viết của tôi

- **Đánh giá của tôi:** Các địa điểm người dùng đã đánh giá.



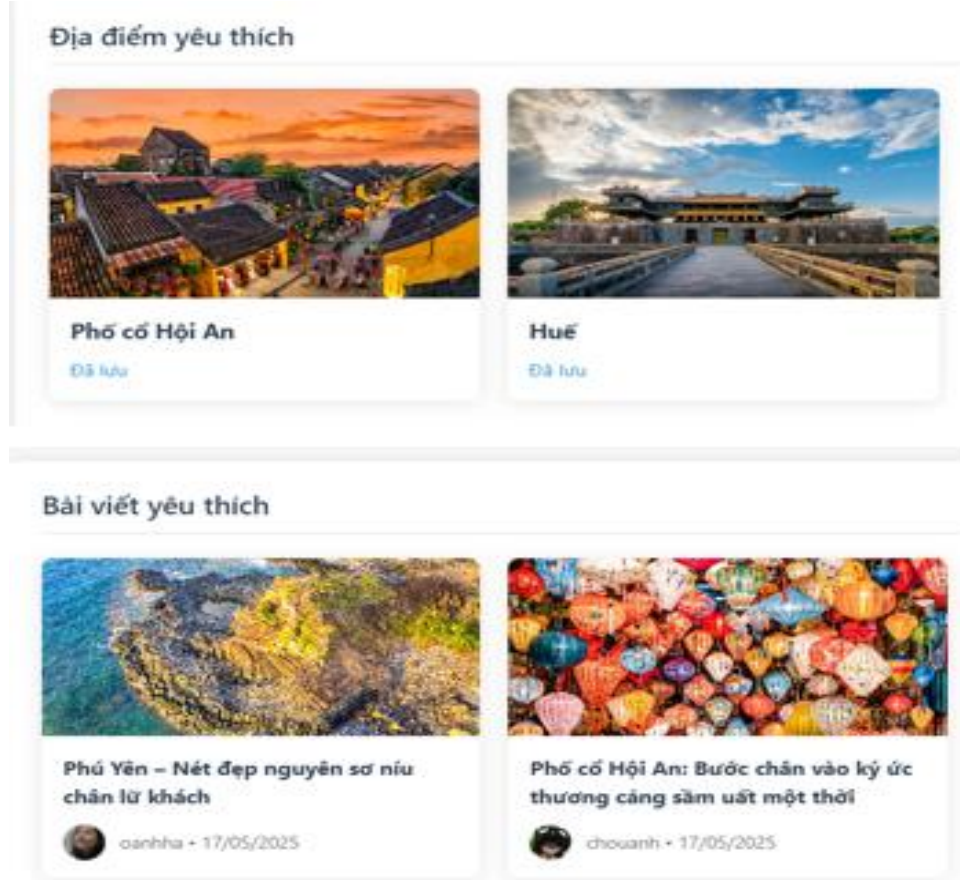
Hình 40: Đánh giá của tôi

- **Bình luận của tôi:** Toàn bộ bình luận đã gửi.



Hình 41: Bình luận của tôi

- **Địa điểm yêu thích và Bài viết yêu thích:** Giúp xem lại nhanh các nội dung yêu thích.



Hình 42: Địa điểm và Bài viết yêu thích

#### f. Chỉnh sửa và xoá bài viết

- Người dùng có thể **chỉnh sửa nội dung** bất kỳ bài viết nào của mình:

Chỉnh sửa bài viết

Địa điểm du lịch

Huế

Loại bài viết

Cẩm nang

Tiêu đề

Hướng dẫn du lịch Huế từ A-Z

Ảnh bìa chính

Chọn tệp

Không có tệp nào được chọn

Chọn ảnh mới nếu bạn muốn thay thế ảnh hiện tại.

Ảnh phụ

Nội dung chi tiết

Thời điểm lý tưởng nhất để du lịch Huế:

Tháng 1 - Tháng 4: Thời tiết mát mẻ, ít mưa, nhiệt độ dao động từ 20-28°C, rất thích hợp để tham quan các di tích.  
Tháng 9 - Tháng 11: Thời tiết dễ chịu, mặc dù có thể có mưa nhẹ, nhưng cảnh sắc thiên nhiên rất đẹp.  
Thời điểm nên tránh:

Tháng 5 - Tháng 8: Thời tiết nóng bức, nhiệt độ có thể lên đến 38-40°C, không thuận lợi cho việc tham quan ngoài trời.  
Tháng 10 - Tháng 12: Mùa mưa cao điểm, có thể gặp mưa lớn và ngập lụt.  
Các sự kiện đặc biệt:

❗ Sau khi chỉnh sửa, bài viết của bạn sẽ được gửi lại để kiểm duyệt.


← Quay lại


Lưu thay đổi


Hình 43: Chỉnh sửa bài viết

Sau khi chỉnh sửa, bài viết sẽ **trở lại trạng thái "chờ duyệt"**.

- Người dùng có thể **xoá bài viết** vĩnh viễn nếu không còn muốn giữ nội dung đó trên hệ thống. Xoá bài cần xác nhận để tránh thao tác nhầm.

  
**Xác nhận xóa**  
Bạn có chắc chắn muốn xóa bài viết này không?

 **Cảnh báo:** Hành động này không thể hoàn tác. Tất cả dữ liệu liên quan đến bài viết sẽ bị xóa vĩnh viễn.

 **Thông tin bài viết**

User	nhuquynh
Spot	Huế, Việt Nam
TypeOfPost	Cẩm nang
Title	Hướng dẫn du lịch Huế từ A-Z
Content	Thời điểm lý tưởng nhất để du lịch Huế: Tháng 1 - Tháng 4: Thời tiết mát mẻ, ít mưa, nhiệt độ dao động từ 20-28°C, rất thích hợp để tham quan các di tích. Tháng 9 - Tháng 11: Th...
ImageUrl	/uploads/posts/b2f12404-ffcd-4006-92e0-6b34f44c4d1c.jpg
CreatedAt	5/17/2025 12:37:35 AM

Xóa bài viết

Quay lại danh sách

Hình 44: Xóa bài viết



**g. Liên hệ với Ban quản trị:**

- Giao diện trang liên hệ được thiết kế trực quan, giúp người dùng dễ dàng tìm thấy và lựa chọn hình thức liên hệ phù hợp.



*Hình 45: Liên hệ*

## V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 1. Kết luận

- Hệ thống hỗ trợ tìm kiếm và chia sẻ các địa điểm du lịch đã đáp ứng được các tính năng cơ bản của một nền tảng du lịch trực tuyến. Ứng dụng có thể hỗ trợ cho du khách dễ dàng tìm kiếm những địa điểm du lịch hấp dẫn theo nhu cầu, cùng với đó cũng giúp cho các nhà cung cấp dịch vụ du lịch và cộng đồng người dùng chia sẻ thông tin một cách thuận tiện hơn.

- Song bên cạnh đó vẫn còn nhiều thiếu sót trong ứng dụng như vẫn chưa chú trọng trong việc bảo mật tài khoản người dùng và xác thực thông tin địa điểm du lịch được chia sẻ, cũng như tối ưu hóa hiệu suất tìm kiếm khi dữ liệu lớn.

- Nhìn chung đồ án này đã giúp chúng em có cái nhìn tổng quát hơn về phát triển ứng dụng, áp dụng lập trình vào các bài toán thực tế trong lĩnh vực du lịch, đặc biệt là ứng dụng về công nghệ .NET và quản lý cơ sở dữ liệu SQL Server, kết hợp với những kỹ thuật phát triển Hệ thống hiện đại. Đây là bước đệm cần thiết để nâng cao tư duy phân tích hệ thống cũng như khả năng phát triển phần mềm nhằm tạo nên nền tảng để tiếp cận với các dự án lớn hơn phía trước.

### 2. Hướng phát triển

- Hiện tại, Hệ thống đã tích hợp khá nhiều chức năng hỗ trợ việc tìm kiếm và chia sẻ thông tin các địa điểm du lịch, tạo nền tảng cơ bản cho một cộng đồng du lịch trực tuyến. Tuy nhiên, để có thể triển khai ứng dụng này vào thực tế một cách hiệu



quả và bền vững, vẫn còn nhiều khía cạnh quan trọng cần được cải thiện và phát triển thêm.

+ Về mặt bảo mật và xác thực, đây là thách thức lớn nhất mà ứng dụng cần giải quyết. Mặc dù các tính năng đăng nhập, đăng ký và chia sẻ thông tin đã được triển khai, nhưng chúng vẫn thiếu những biện pháp bảo mật mạnh mẽ như mã hóa mật khẩu hay xác thực hai yếu tố (2FA), tạo ra lỗ hổng bảo mật tiềm ẩn. Thêm vào đó, việc thiếu cơ chế kiểm duyệt thông tin địa điểm du lịch do người dùng chia sẻ có thể dẫn đến tình trạng thông tin sai lệch, ảnh hưởng đến trải nghiệm của cộng đồng người dùng.

+ Về khía cạnh mở rộng chức năng, hệ thống hiện tại chủ yếu tập trung vào tìm kiếm và chia sẻ các thông tin cơ bản. Để trở thành nền tảng du lịch hoàn chỉnh, cần bổ sung các tính năng như hệ thống đánh giá địa điểm, công cụ lập kế hoạch du lịch, và dịch vụ đặt tour/khách sạn trực tuyến. Việc phát triển ứng dụng di động cũng là yếu tố then chốt để người dùng dễ dàng truy cập khi di chuyển. Việc tích hợp bản đồ tương tác, dự báo thời tiết, thông tin giao thông, và hệ thống gợi ý thông minh dựa trên sở thích cá nhân sẽ tạo ra trải nghiệm du lịch toàn diện và được cá nhân hóa.

## TÀI LIỆU THAM KHẢO

- [1] M. Halvorson, *Microsoft Visual C# Step by Step*, 9th ed. Redmond, WA: Microsoft Press, 2021.
- [2] J. Price, *C# 9.0 and .NET 5 – Modern Cross-Platform Development*, 6th ed. Birmingham: Packt Publishing, 2020.
- [3] Microsoft Docs, “SQL Server documentation,” 2023. [Online].  
Available: <https://learn.microsoft.com/en-us/sql/sql-server/>
- [4] Microsoft Docs, “ADO.NET overview,” 2023. [Online].  
Available: <https://learn.microsoft.com/en-us/dotnet/framework/data/adonet/ado-net-overview>
- [5] Microsoft Docs, “Entity Framework Core documentation,” 2023. [Online].  
Available: <https://learn.microsoft.com/en-us/ef/core/>