TRƯỜNG ĐẠI HỌC TRÀ VINH

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------🕮-----------**

****

**BÁO CÁO KẾT THÚC HỌC PHẦN**

**PHÁT TRIỂN ỨNG DỤNG WEB VỚI MÃ NGUỒN MỞ**

**XÂY DỰNG HỆ THỐNG QUẢN LÝ THẾ GIỚI ẢO TRONG TRUYỆN ONLINE**

|  |  |
| --- | --- |
| *Giáo viên hướng dẫn*  ThS. Phạm Thị Trúc Mai | *Sinh viên thực hiện:*  Họ tên: Ngô Huỳnh Quốc Khang  Mã số sinh viên: 110122092  Lớp: DA22TTD  Họ tên: Nguyễn Đinh Tuấn Khoa  Mã số sinh viên: 110122094  Lớp: DA22TTD  Họ tên: Phạm Duy Tân  Mã số sinh viên: 110122243  Lớp: DA22TTD |

***Vĩnh Long, Tháng 01 Năm 2026***

TRƯỜNG ĐẠI HỌC TRÀ VINH

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------🕮-----------**

****

**BÁO CÁO KẾT THÚC HỌC PHẦN**

**PHÁT TRIỂN ỨNG DỤNG WEB VỚI MÃ NGUỒN MỞ**

**XÂY DỰNG HỆ THỐNG QUẢN LÝ THẾ GIỚI ẢO TRONG TRUYỆN ONLINE**

|  |  |
| --- | --- |
| *Giáo viên hướng dẫn*  ThS. Phạm Thị Trúc Mai | *Sinh viên thực hiện:*  Họ tên: Ngô Huỳnh Quốc Khang  Mã số sinh viên: 110122092  Lớp: DA22TTD  Họ tên: Nguyễn Đinh Tuấn Khoa  Mã số sinh viên: 110122094  Lớp: DA22TTD  Họ tên: Phạm Duy Tân  Mã số sinh viên: 110122243  Lớp: DA22TTD |

***Vĩnh Long, Tháng 01 Năm 2026***

**LỜI CẢM ƠN**

Đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến các thầy cô, những người đã hết mực chỉ bảo, hướng dẫn và sẵn sàng giải đáp thắc mắc, cung cấp những kiến thức hữu ích giúp em hoàn thành công việc trong suốt quá trình thực hiện báo cáo này.

Mặc dù đã cố gắng để thực hiện đề tài một cách nghiêm túc và tập trung trong suốt quá trình thu thập, tìm hiểu, phân tích và tổng hợp nhiều nguồn tài liệu khác nhau, song do sự hiểu biết còn chưa cao, trình độ chuyên môn còn hạn chế ở nhiều mặt và chưa có nhiều kỹ năng trong việc thực hiện các công việc trong chuyên ngành nên trong quá trình biên soạn đồ án không thể tránh khỏi các thiết sót. Do đó nếu như có những điểm trong báo cáo còn chưa hợp lí hoặc sai sót em rất mong được các thầy cô đóng góp thêm các ý kiến để có thể học hỏi và tiếp thu thêm các kiến thức còn thiếu sót.

Em xin chân thành cảm ơn!

*Sinh viên ký và ghi rõ họ tên*

|  |  |  |
| --- | --- | --- |
| **Sinh viên 1** | **Sinh viên 2** | **Sinh viên 3** |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Vĩnh Long, ngày … tháng … năm 20…*

**Giáo viên hướng dẫn**

*(Ký và ghi rõ họ tên)*

**NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN**

*Vĩnh Long, ngày … tháng … năm 20…*

**Giáo viên hướng dẫn**

*(Ký và ghi rõ họ tên)*

**MỤC LỤC**

[LÝ DO CHỌN ĐỀ TÀI xii](#_Toc218817993)

[CHƯƠNG 1. NGHIÊN CỨU LÝ THUYẾT 1](#_Toc218817994)

[1.1. Tổng quan về thiết kế giao diện web 1](#_Toc218817995)

[1.1.1 Tổng quan HTML 1](#_Toc218817996)

[1.1.2 Giới thiệu ngôn ngữ HTML 1](#_Toc218817997)

[*1.1.2.1* Cấu trúc của các thành phần HTML 1](#_Toc218817998)

[*1.1.2.2* Cấu trúc trang HTML 1](#_Toc218817999)

[1.1.3 Tổng quan về CSS 2](#_Toc218818000)

[*1.1.3.1* Giới thiệu ngôn ngữ CSS 2](#_Toc218818001)

[*1.1.3.2* Chức năng của CSS 2](#_Toc218818002)

[*1.1.3.3* Chèn định dạng CSS vào tài liệu HTML 3](#_Toc218818003)

[1.1.4 Tổng quan JavaScript 3](#_Toc218818004)

[*1.1.4.1* Giới thiệu ngôn ngữ JavaScript 3](#_Toc218818005)

[*1.1.4.2* Các tính năng của JavaScript 3](#_Toc218818006)

[*1.1.4.3* Sử dụng JavaScript trong tài liệu HTML 4](#_Toc218818007)

[1.1.5 Tổng quan Bootstrap 4](#_Toc218818008)

[*1.1.5.1* Giới thiệu về framework Bootstrap 4](#_Toc218818009)

[*1.1.5.2* Bootstrap 4](#_Toc218818010)

[*1.1.5.3* Các công dụng của Bootstrap 4](#_Toc218818011)

[1.1.6 Tổng quan jQuery 5](#_Toc218818012)

[*1.1.6.1* Giới thiệu về jQuery 5](#_Toc218818013)

[*1.1.6.2* Các tính năng của jQuery 5](#_Toc218818014)

[1.2. PHP và Laravel 6](#_Toc218818015)

[1.2.1 Tổng quan về PHP 6](#_Toc218818016)

[*1.2.1.1* Giới thiệu về ngôn ngữ PHP 6](#_Toc218818017)

[*1.2.1.2* Ưu điểm của PHP 6](#_Toc218818018)

[*1.2.1.3* Nhược điểm của PHP 7](#_Toc218818019)

[1.2.2 Tổng quan về Laravel 7](#_Toc218818020)

[*1.2.2.1* Giới thiệu về Laravel 7](#_Toc218818021)

[*1.2.2.2* Ưu điểm của Laravel 8](#_Toc218818022)

[1.2.3 Nhược điểm của Laravel 9](#_Toc218818023)

[1.3. Hệ quản trị cơ sở dữ liệu MySQL 9](#_Toc218818024)

[1.3.1 Giới thiệu về hệ quản trị dữ liệu MySQL 9](#_Toc218818025)

[1.3.2 Các tính năng chính của MySQL 9](#_Toc218818026)

[1.3.3 Ứng dụng của MySQL 10](#_Toc218818027)

[1.4. Eloquent ORM trong Laravel 10](#_Toc218818028)

[1.4.1 Giới thiệu Eloquent ORM 10](#_Toc218818029)

[1.4.2 Đặc điểm cơ bản của Eloquent ORM 10](#_Toc218818030)

[1.5. Mô hình MVC (Model – View – Controller) 11](#_Toc218818031)

[1.5.1 Giới thiệu mô hình MVC 11](#_Toc218818032)

[1.5.2 Đặc điểm của MVC 11](#_Toc218818033)

[1.5.3 Các thành phần trong mô hình MVC 12](#_Toc218818034)

[*1.5.3.1* Model 12](#_Toc218818035)

[*1.5.3.2* View 12](#_Toc218818036)

[*1.5.3.3* Controller 12](#_Toc218818037)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 13](#_Toc218818038)

[2.1. Mô tả bài toán 13](#_Toc218818039)

[2.2. Phân tích và thiết kế hệ thống 14](#_Toc218818040)

[2.2.1 Mô hình dữ liệu mức quan niệm 14](#_Toc218818041)

[2.2.2 Mô hình dữ liệu mức luận lý 14](#_Toc218818042)

[2.2.3 Mô hình dữ liệu mức vật lý 15](#_Toc218818043)

[2.2.4 Mô hình cơ sở dữ liệu 15](#_Toc218818044)

[*2.2.4.1* Bảng users - Người dùng 16](#_Toc218818045)

[*2.2.4.2* Bảng roles - Vai trò người dùng 16](#_Toc218818046)

[*2.2.4.3* Bảng worlds - Thế giới hư cấu 16](#_Toc218818047)

[*2.2.4.4* Bảng entity\_types - Loại thực thể 17](#_Toc218818048)

[*2.2.4.5* Bảng entities - Thực thể trong thế giới 17](#_Toc218818049)

[*2.2.4.6* Bảng relationships - Quan hệ giữa các thực thể 18](#_Toc218818050)

[*2.2.4.7* Bảng tags - Thẻ gắn cho thực thể 18](#_Toc218818051)

[*2.2.4.8* Bảng entity\_tags - Liên kết thực thể và thẻ 19](#_Toc218818052)

[*2.2.4.9* Mô tả: Bảng entity\_tags (Gắn thẻ thực thể): Liên kết thực thể với các thẻ tương ứng. 19](#_Toc218818053)

[2.2.5 Sơ đồ Usecase hệ thống 19](#_Toc218818054)

[*2.2.5.1* Sơ đồ Usecase Người dùng 19](#_Toc218818055)

[*2.2.5.2* Sơ đồ Usecase Admin 20](#_Toc218818056)

[*2.2.5.3* Sơ đồ Usecase super Admin 21](#_Toc218818057)

[2.3. Phác thảo giao diện 22](#_Toc218818058)

[2.3.1 Trang Đăng nhập 22](#_Toc218818059)

[2.3.2 Trang Đăng ký 22](#_Toc218818060)

[2.3.3 Trang Bảng điều khiển 23](#_Toc218818061)

[2.3.4 Trang danh sách thế giới 24](#_Toc218818062)

[2.3.5 Trang chi tiết thế giới 24](#_Toc218818063)

[2.3.6 Trang sửa thế giới 25](#_Toc218818064)

[2.3.7 Trang danh sách thực thể 26](#_Toc218818065)

[2.3.8 Trang Chi tiết thực thể 26](#_Toc218818066)

[2.3.9 Trang Tạo thế giới bằng AI 27](#_Toc218818067)

[2.3.10 Trang Quản lý người dùng 28](#_Toc218818068)

[CHƯƠNG 3. XÂY DỰNG WEBSITE 29](#_Toc218818069)

[3.1. Giới thiệu giao diện 29](#_Toc218818070)

[3.1.1 Giao diện đăng nhập 29](#_Toc218818071)

[3.1.2 Giao diện đăng ký 29](#_Toc218818072)

[3.1.3 Giao diện bảng điều khiển 30](#_Toc218818073)

[3.1.4 Giao diện danh sách thế giới 30](#_Toc218818074)

[3.1.5 Giao diện chi tiết thế giới 31](#_Toc218818075)

[3.1.6 Giao diện sửa thế giới 31](#_Toc218818076)

[3.1.7 Giao diện danh sách thực thể 32](#_Toc218818077)

[3.1.8 Giao diện chi tiết thực thể 32](#_Toc218818078)

[3.1.9 Giao diện tạo thế giới bằng AI 33](#_Toc218818079)

[3.1.10 Giao diện quản lý người dung 33](#_Toc218818080)

[3.2. Các chức năng của website 33](#_Toc218818081)

[CHƯƠNG 4. KẾT LUÂN VÀ HƯỚNG PHÁT TRIỂN 34](#_Toc218818082)

[4.1. Kết quả đạt được 34](#_Toc218818083)

[4.2. Ưu điểm và hạn chế 35](#_Toc218818084)

[4.2.1 Ưu điểm 35](#_Toc218818085)

[4.2.2 Hạn chế 35](#_Toc218818086)

[4.3. Hướng phát triển 36](#_Toc218818087)

**DANH MỤC HÌNH ẢNH – BẢNG BIỂU**

[Hình 1.1 Laravel 7](#_Toc218818088)

[Hình 1.2 mô hình MVC 11](#_Toc218818089)

[Hình 2.1 Mô hình dữ liệu mức quan niệm 14](#_Toc218818090)

[Hình 2.2 Mô hình dữ liệu mức luận lý 14](#_Toc218818091)

[Hình 2.3 Mô hình dữ liệu mức vật lý 15](#_Toc218818092)

[Hình 2.4 Mô hình cơ sở dữ liệu 15](#_Toc218818093)

[Hình 2.5 Sơ đồ Usecase Người dùng 19](#_Toc218818094)

[Hình 2.6 Sơ đồ Usecase Admin 20](#_Toc218818095)

[Hình 2.7 Sơ đồ Usecase super Admin 21](#_Toc218818096)

[Hình 2.8 Trang Đăng nhập 22](#_Toc218818097)

[Hình.2.9 Trang Đăng ký 22](#_Toc218818098)

[Hình 2.10 Trang Bảng điều khiển 23](#_Toc218818099)

[Hình 2.11 Trang danh sách thế giới 24](#_Toc218818100)

[Hình 2.12 Trang Chi tiết thế giới 24](#_Toc218818101)

[Hình 2.13 Trang Sửa thế giới 25](#_Toc218818102)

[Hình 2.14 Trang danh sách thực thể 26](#_Toc218818103)

[Hình 2.15 Trang Chi tiết thực thể 26](#_Toc218818104)

[Hình 2.16 Trang Tạo thế giới bằng AI 27](#_Toc218818105)

[Hình 2.17 Trang Quản lý người dùng 28](#_Toc218818106)

[Hình 3.1 Giao diện đăng nhập 29](#_Toc218818107)

[Hình 3.2 Giao diện đăng ký 29](#_Toc218818108)

[Hình 3.3 Giao diện bảng điều khiển 30](#_Toc218818109)

[Hình 3.4 Giao diện danh sách thế giới 30](#_Toc218818110)

[Hình 3.5 Giao diện chi tiết thế giới 31](#_Toc218818111)

[Hình 3.6 Giao diên sửa thế giới 31](#_Toc218818112)

[Hình 3.7 Giao diện danh sách thực thể 32](#_Toc218818113)

[Hình 3.8 Giao diện chi tiết thực thể 32](#_Toc218818114)

[Hình 3.9 Giao diện tạo thế giới bằng AI 33](#_Toc218818115)

[Hình 3.10 Giao diện quản lý người dùng 33](#_Toc218818116)

[Bảng 2.1 Bảng users - Người dùng 16](#_Toc218818117)

[Bảng 2.2 Bảng roles - Vai trò người dùng 16](#_Toc218818118)

[Bảng 2.3 Bảng worlds - Thế giới hư cấu 16](#_Toc218818119)

[Bảng 2.4 Bảng entity\_types - Loại thực thể 17](#_Toc218818120)

[Bảng 2.5 Bảng entities - Thực thể trong thế giới 17](#_Toc218818121)

[Bảng 2.6 Bảng relationships - Quan hệ giữa các thực thể 18](#_Toc218818122)

[Bảng 2.7 Bảng tags - Thẻ gắn cho thực thể 18](#_Toc218818123)

[Bảng 2.8 Bảng entity\_tags - Liên kết thực thể và thẻ 19](#_Toc218818124)

**BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| **HỌ TÊN SINH VIÊN** | **CÔNG VIỆC THỰC HIỆN** | **THỜI GIAN**  **THỰC HIỆN** |
| **Ngô Huỳnh Quốc Khang** | Thiết kế cơ sở dữ liệu, xây dựng Models | Tuần 1 - 2 |
| **Ngô Huỳnh Quốc Khang** | Xây dựng chức năng xác thực (đăng nhập, đăng ký) | Tuần 2 - 3 |
| **Ngô Huỳnh Quốc Khang** | Xây dựng API quản lý World, Entity, Relationship, Tag | Tuần 3 - 5 |
| **Nguyễn Đinh Tuấn Khoa** | Thiết kế giao diện tổng thể (Layout, Dashboard) | Tuần 1 - 2 |
| **Nguyễn Đinh Tuấn Khoa** | Xây dựng giao diện quản lý World, Entity | Tuần 3 - 4 |
| **Nguyễn Đinh Tuấn Khoa** | Xây dựng giao diện Relationship, Tag, tìm kiếm | Tuần 4 - 5 |
| **Phạm Duy Tân** | Phân tích yêu cầu, thiết kế Use Case, ERD | Tuần 1 |
| **Phạm Duy Tân** | Tích hợp AI (Gemini) - tạo World tự động, AI Chat | Tuần 4 - 6 |
| **Phạm Duy Tân** | Kiểm thử, sửa lỗi, viết tài liệu hướng dẫn | Tuần 6 - 7 |

**BÁO CÁO KẾT QUẢ THỰC HIỆN TỪNG THÀNH VIÊN**

[Báo cáo tóm tắt kết quả thực hiện và minh chứng cho công việc được phân công của từng thành viên trong nhóm. Gợi ý: chụp hình thể hiện thành quả đóng góp của từng thành viên trên GitHub.]

# ****LÝ DO CHỌN ĐỀ TÀI****

Trong những năm gần đây, văn học mạng (truyện Online) tại Việt Nam và trên thế giới phát triển mạnh mẽ, thu hút đông đảo tác giả và độc giả tham gia trên các nền tảng trực tuyến. Đặc trưng của truyện online là nội dung kéo dài qua hàng trăm đến hàng nghìn chương, với hệ thống nhân vật phong phú, cốt truyện nhiều tuyến và thế giới giả tưởng ngày càng mở rộng. Chính quy mô lớn và thời gian sáng tác dài khiến các tác giả thường gặp khó khăn trong việc ghi nhớ, quản lý thông tin, dẫn đến tình trạng sai lệch chi tiết, thiếu nhất quán về mốc thời gian hoặc nhầm lẫn quan hệ giữa các nhân vật, ảnh hưởng trực tiếp đến chất lượng tác phẩm và trải nghiệm của người đọc.

Xuất phát từ thực tiễn đó, nhóm chúng em lựa chọn đề tài **“Xây dựng hệ thống quản lý thế giới ảo trong truyện Online”**. Mục tiêu của đề tài là xây dựng một nền tảng web cho phép tác giả quản lý tập trung các thực thể trong truyện như nhân vật, môn phái, địa danh, sự kiện và các yếu tố liên quan, đồng thời thiết lập mối quan hệ giữa chúng một cách rõ ràng và trực quan. Hệ thống hỗ trợ phân loại thông tin bằng thẻ (tag), giúp việc tìm kiếm và theo dõi dữ liệu trở nên thuận tiện hơn. Bên cạnh đó, hệ thống còn tích hợp trí tuệ nhân tạo Gemini nhằm hỗ trợ gợi ý và tạo mô tả tự động cho nhân vật hoặc bối cảnh, góp phần giảm tải công việc thủ công và hỗ trợ tác giả trong quá trình sáng tác.

Về mặt công nghệ, hệ thống được xây dựng theo mô hình **MVC với Laravel** làm nền tảng backend, giúp tổ chức và quản lý dữ liệu một cách khoa học, dễ mở rộng và bảo trì. Giao diện được thiết kế đơn giản, thân thiện với người dùng, tập trung vào trải nghiệm của tác giả khi sử dụng hệ thống. Thông qua đề tài này, nhóm không chỉ áp dụng kiến thức về lập trình web và cơ sở dữ liệu mà còn tiếp cận việc ứng dụng trí tuệ nhân tạo vào hỗ trợ sáng tạo nội dung. Sản phẩm có tính ứng dụng cao, hướng tới trở thành công cụ hỗ trợ hiệu quả cho cộng đồng sáng tác truyện online.

# NGHIÊN CỨU LÝ THUYẾT

## Tổng quan về thiết kế giao diện web

### Tổng quan HTML

### Giới thiệu ngôn ngữ HTML

HTML (Hypertext Markup Language-ngôn ngữ đánh dấu siêu văn bản) là ngôn ngữ được sử dụng để tạo và định dạng cấu trúc các thành phần trong trang web. Trang mã nguồn HTML là một tệp văn bản gồm các ký tự ASCII, chúng có thể được tạo ra từ bất kì trình soạn thảo nào. Các tệp mã nguồn này có phần mở rộng là “.htm” hoặc “.html” [1].

#### Cấu trúc của các thành phần HTML

Các cặp thẻ HTML thường gồm thẻ mở và thẻ đóng, 2 thẻ này có cấu trúc gần giống nhau:

**<tên thẻ> Nội dung </tên thẻ>**

Trong thẻ mở có thể có thêm tham số đặt phía sau tên thẻ, dữ liệu cần hiển thị nằm trong cặp thẻ mở và thẻ đóng. Tuy nhiên có một số thẻ đặc biệt sẽ không có thẻ đóng. Người soạn thảo có thể đặt các thẻ này lồng vào nhau.

#### Cấu trúc trang HTML

Cấu trúc trang HTML gồm 3 phần:

<!DOCTYPE html>

<html>

<head>

<title>Tiêu đề của trang sẽ nằm ở đây</title>

</head>

<body>

Nội dung của trang sẽ nằm ở đây

</body>

</html>

Phần khai báo loại chương trình: xuất hiện ở đầu trên cùng của file HTML, dùng để thông báo cho trình duyệt biết người dùng đang sử dụng phiên bản HTML nào. Ví dụ như thẻ khai báo <!DOCTYPE html> dùng để khai báo cho chuẩn HTML5.

Phần khai báo ban đầu: chứa tiêu đề trang web và các khai báo về bảng mã chữ cái, từ khóa tìm kiếm, …. Phần này được đặt giữa cặp thẻ <head></head>.

Phần chứa và hiển thị nội dung của trang web: đây là phần sẽ chứa các nội dung hiển thị trên màn hình khi người dùng truy cập vào trang web. Phần này được đặt giữa cặp thẻ <body></body>.

Phần khai báo ban đầu ,phần chứa và hiển thị nội dung lại được đặt trong một cặp thẻ nữa là <html></html>. Cặp thẻ này báo cho chương trình dịch biết điểm bắt đầu và kết thúc của trang HTML.

### Tổng quan về CSS

#### Giới thiệu ngôn ngữ CSS

CSS (Cascading Style Sheet) là một ngôn ngữ thiết kế đơn giản, được sử dụng để tìm và định dạng lại các phần tử tạo ra bởi ngôn ngữ đánh dấu (như HTML). Hay nói khác đi, CSS được sử dụng để tạo ra phong cách hiển thị trang web. Phương thức hoạt động của CSS là nó dựa vào các vùng chọn (có thể là thẻ HTML, tên ID, class, …) để thay đổi các thuộc tính hiển thị của các phần tử trên vùng chọn đó.

#### Chức năng của CSS

CSS cho phép các người lập trình web quản lý được hầu hết các thuộc tính của các thành phần có trong HTML. Các tính năng có trong CSS:

* Đặt màu, màu nền cho một đoạn văn bản.
* Căn chỉnh khoảng cách giữa các ký tự, giữa các dòng.
* Thay đổi font chữ và các thuộc tính của font chữ.

#### Chèn định dạng CSS vào tài liệu HTML

Có thể khai báo các thuộc tính định dạng trong CSS bằng nhiều cách khác nhau. Chẳng hạn đoạn đoạn CSS có thể được đặt bên trong thẻ cặp thẻ <head>...</head>, các lệnh định dạng sẽ được viết trong cặp thẻ <style>...</style>. Đoạn lệnh viết bằng CSS cũng có thể được nhúng vào trong từng thẻ HTML riêng biệt hoặc soạn thảo ra tập tin riêng với phần mở rộng là .css. Với từng cách đặt sẽ có độ ưu tiên khác nhau, thứ tự độ ưu tiên giảm dần từ trên xuống là:

* + Đoạn mã nằm trong từng thẻ HTML riêng biệt.
  + Đoạn mã đặt trong cặp thẻ <head>...</head>.
  + Đoạn mã đặt trong tập tin có phần mở rộng .css

### Tổng quan JavaScript

#### Giới thiệu ngôn ngữ JavaScript

JavaScript (JS) là một ngôn ngữ lập trình thông dịch với khả năng hướng đối tượng. JavaScript là một ngôn ngữ lập trình web phổ biến hiện nay, dùng để tích hợp vào HTML giúp trang web trở nên sống động hơn. JavaScript có mối liên hệ chặt chẽ với HTML và CSS: trong khi HTML định dạng các thành phần của trang web, CSS quy định phong cách hiển thị trang web, thì JavaScript có thể xử lý các thuộc tính đối tượng do HTML và CSS quy định, tạo nên sự tương tác giữa người dùng với trang web [3].

#### Các tính năng của JavaScript

JavaScript có các tính năng gần giống với các ngôn ngữ lập trình thông thường, người lập trình có thể khai báo biến, hằng và các thao tác với các biến và hằng này, có thể xây dựng các hàm thực thi các chức năng khác nhau. Việc xây dựng chương trình JavaScript được dựa trên mục đích riêng biệt của người lập trình với trang mã nguồn HTML, một số chức năng mà JavaScript cung cấp:

* + Thay đổi nội dung các thành phần HTML.
  + Thay đổi giá trị thuộc tính của thành phần HTML.
  + Ẩn, hiện các phần tử HTML.

#### Sử dụng JavaScript trong tài liệu HTML

Đoạn chương trình viết bằng JavaScript được nhúng trực tiếp vào trang mã nguồn HTML thông qua cặp thẻ <script>...</script> ở trong phần <body> của trang. Các đoạn lệnh này có thể được viết trực tiếp hoặc tham chiếu qua đường dẫn đến tập tin có phần mở rộng là .js.

### Tổng quan Bootstrap

#### Giới thiệu về framework Bootstrap

#### Bootstrap

Bootstrap là một front-end framework mã nguồn mở miễn phí giúp quá trình phát triển web được nhanh và dễ dàng hơn. Nó cung cấp một bộ công cụ và các lớp CSS có sẵn để tạo giao diện người dùng, bao gồm hệ thống lưới, nút, biểu mẫu và nhiều thành phần khác.

Với Bootstrap, bạn không cần phải viết CSS từ đầu, mà chỉ cần sử dụng các lớp CSS đã được định nghĩa sẵn. Điều này giúp bạn tiết kiệm thời gian và công sức trong việc tạo kiểu trong trang web của mình. Bạn cũng có thể tùy chỉnh Bootstrap để phù hợp với nhu cầu cụ thể của dự án.

Một điểm mạnh của Bootstrap là khả năng tương thích với đa dạng các thiết bị và kích thước màn hình khác nhau. Với việc sử dụng các lớp CSS và các thành phần đáp ứng sẵn có trong Bootstrap, trang web của bạn sẽ tự động thích ứng và hiển thị tốt trên điện thoại di động, máy tính bảng và máy tính để bàn.

Với sự phổ biến và mã nguồn mở của nó, Bootstrap đã trở thành một trong những framework CSS phổ biến nhất trên thế giới, được sử dụng rộng rãi trong cộng đồng phát triển web.

#### Các công dụng của Bootstrap

Tiết kiệm thi gian và công sức: Cung cấp một bộ công cụ và các lớp CSS có sẵn, giúp bạn nhanh chóng tạo giao diện người dùng mà không cần viết CSS từ đầu. Bạn có thể sử dụng các thành phần và kiểu dáng đã được chuẩn hóa sẵn, giảm bớt công việc lặp lại và tăng tốc quá trình phát triển.

Giao diện đáp ứng (responsive design): Hỗ trợ xây dựng trang web có giao diện đáp ứng, tức là giao diện sẽ tự động thích ứng và hiển thị tốt trên các thiết bị và kích thước màn hình khác nhau. Điều này giúp đảm bảo trải nghiệm người dùng tốt trên di động, máy tính bảng và máy tính để bàn mà không cần phải viết mã CSS phức tạp.

Hệ thống lưới linh hoạt: Hệ thống lưới (grid system) mạnh mẽ, giúp xây dựng cấu trúc trang linh hoạt và tương thích với các kích thước màn hình khác nhau. Có thể dễ dàng điều chỉnh vị trí và kích thước của các phần tử trên trang web.

Tính nhất quán và tương thích trên nhiều trình duyệt: Bootstrap được kiểm thử và tối ưu để hoạt động tốt trên nhiều trình duyệt phổ biến như Chrome, Firefox, Safari và Internet Explorer. Điều này giúp đảm bảo rằng trang web của bạn sẽ có trải nghiệm đồng nhất trên các nền tảng khác nhau.

Cộng đồng phát triển mạnh mẽ: Bootstrap là một framework phổ biến với cộng đồng lớn. Bạn có thể tìm thấy nhiều tài liệu, ví dụ và giải pháp sẵn có từ cộng đồng, giúp bạn giải quyết các vấn đề và tận dụng tối đa khả năng của Bootstrap [4].

### Tổng quan jQuery

#### Giới thiệu về jQuery

jQuery là một thuật ngữ được dùng rất nhiều trong thời đại công nghệ hiện nay. Kể từ khi phát triển 2.0, cụm từ này đã được dùng khá phổ biến. Nói một cách dễ hiểu, jQuery là một thư viện được xây dựng trên Javascript. Chúng được tạo ra mới mục đích hỗ trợ các lập trình viên giải quyết các khó khăn khi sử dụng Javascript. jQuery được tích hợp từ nhiều module chuyên biệt, bao gồm cả module hiệu ứng đến module truy vấn selector.

#### Các tính năng của jQuery

Thao tác DOM: jQuery cho phép người dùng đơn giản hóa việc lựa chọn các phần tử DOM để duyệt như khi dùng CSS. Người dùng cũng có thể sử dụng phương tiện selector mã nguồn mở để chỉnh sửa nội dung của chúng.

Xử lý sự kiện: thư viện jQuery xử lý các sự kiện một cách nhanh chóng chính xác và hiệu quả mà không làm HTML code rối với các Event Handler. Nhờ đó nâng cao khả năng tương tác với người dùng một cách tối đa.

Hỗ trợ AJAX: bằng việc sử dụng công nghệ AJAX, jQuery cho phép lập trình viên phát triển website với đa dạng các tính năng và phản hồi tích cực.

Tạo hiệu ứng động: jQuery cho phép cung cấp đa dạng các hiệu ứng động đẹp mắt, độc đáo cho website, giúp trở nên sinh động và chuyên nghiệp hơn.

Kích thước cực kỳ gọn nhẹ: thư viện này vô cùng gọn nhẹ, vì vậy chạy rất mượt và nhanh. jQuery chỉ có 19KB.

Cập nhật và hỗ trợ các ứng dụng mới nhất: hỗ trợ CSS3 Selector và cú pháp XPath cơ bản giúp lập trình viên có thể tiếp cận với những công nghệ tân tiến nhất của thời đại khoa học – máy tính 4.0 [5].

## PHP và Laravel

### Tổng quan về PHP

#### Giới thiệu về ngôn ngữ PHP

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình phía máy chủ, được thiết kế chủ yếu để phát triển các ứng dụng web động và tương tác. PHP rất phổ biến nhờ tính đơn giản, linh hoạt và khả năng tích hợp tốt với các cơ sở dữ liệu như MySQL, PostgreSQL. Ngoài ra, nó còn có thể chạy trên nhiều nền tảng như Windows, Linux, macOS.

PHP là một ngôn ngữ lập trình web động mã nguồn mở. Tập tin PHP có phần mở rộng là .php. Về tổng quan PHP có cú pháp khá tương đồng với một số ngôn ngữ như C, java. Tuy nhiên, tự bản thân chúng cũng có những điểm rất riêng biệt. Là một ngôn ngữ mã nguồn mở, PHP được một số lớn các nhà lập trình Web chọn làm ngôn ngữ để phát triển Website.

#### Ưu điểm của PHP

Đầu tiên phải kể đến đó là PHP chính là một mã nguồn mở. Đặc biệt PHP là một mã nguồn miễn phí và được chia sẻ nhiều trên các diễn đàn, cộng đồng và bạn có thể dễ dàng sao chép và cài đặt sử dụng các website hay các ngữ cảnh có sẵn. PHP là ngôn ngữ dễ học và dễ sử dụng, nó có tính linh hoạt cao để giải quyết các vấn đề. Đây là một ngôn ngữ có tính công đồng lớn nhất hiện nay vì đây là một mã nguồn mở và cũng rất dễ sử dụng cho nên PHP luôn được ưa chuộng và tạo nên một cộng đồng lớn và chất lượng với các chuyên gia trên toàn thế giới.

#### Nhược điểm của PHP

Bên cạnh những ưu điểm đã trình bày ở trên, PHP cũng có nhược điểm không thể chối bỏ. Đó là PHP được tạo ra chỉ để phục vụ một mục đích duy nhất là phát triển ứng dụng web. Do đó, bạn không thể dùng ngôn ngữ này để phát triển phần mềm ứng dụng. Nhược điểm này làm cho PHP mất đi một số lượng khá lớn khách hàng khi họ chỉ muốn phát triển ứng dụng chạy trên window-form [6].

### Tổng quan về Laravel

#### Giới thiệu về Laravel

Laravel là một framework PHP hiện đại và mạnh mẽ, được phát triển nhằm hỗ trợ lập trình viên xây dựng các ứng dụng web một cách nhanh chóng, rõ ràng và dễ bảo trì. Với cú pháp đẹp, tinh gọn, Laravel giúp tối ưu hóa năng suất lập trình và giảm thiểu các thao tác lặp lại nhờ việc áp dụng nhiều mô hình.



Hình 1.1 Laravel

Laravel sử dụng kiến trúc MVC (Model – View – Controller) giúp phân tách rõ ràng giữa giao diện người dùng, xử lý logic và quản lý dữ liệu, tạo điều kiện thuận lợi cho việc mở rộng và bảo trì hệ thống. Ngoài ra, Laravel tích hợp nhiều công nghệ và thư viện mạnh mẽ như hệ thống routing linh hoạt, ORM Eloquent, cơ chế migration, hệ thống queue, scheduler, bảo mật CSRF, middleware,… giúp việc phát triển ứng dụng web trở nên nhanh chóng và an toàn hơn.

Laravel hiện đang là một trong những framework PHP phổ biến nhất thế giới, được cộng đồng rộng lớn hỗ trợ và cập nhật liên tục, phù hợp với các dự án từ nhỏ đến phức tạp.

#### Ưu điểm của Laravel

* Laravel cung cấp nhiều tính năng mạnh mẽ giúp lập trình viên phát triển ứng dụng một cách thuận tiện và hiệu quả:
* Cú pháp rõ ràng, dễ đọc dễ bảo trì: Laravel được thiết kế hướng đến sự đơn giản và dễ hiểu, giúp lập trình viên tăng tốc độ phát triển và giảm lỗi trong quá trình viết mã.
* Hỗ trợ kiến trúc MVC: Eloquent giúp thao tác cơ sở dữ liệu dễ dàng, trực quan thông qua các mô hình (Model) mà không cần viết nhiều câu lệnh SQL phức tạp.
* ORM Eloquent mạnh mẽ: Eloquent giúp thao tác cơ sở dữ liệu dễ dàng, trực quan thông qua các mô hình (Model) mà không cần viết nhiều câu lệnh SQL phức tạp.
* Hệ thống Routing linh hoạt: cho phép định tuyến rõ ràng, dễ quản lý, hỗ trợ RESTful route và middleware giúp kiểm soát truy cập tốt hơn.
* Blade Template Engine: Blade là bộ máy template tích hợp của Laravel, hỗ trợ: kế thừa giao diện, tách các phần header/footer/sidebar, chèn mã PHP đơn giản và an toàn.
* Hỗ trợ Migration & Seeder: Giúp quản lý, tạo và cập nhật cấu trúc cơ sở dữ liệu nhanh chóng, đảm bảo đồng bộ giữa các môi trường.

### Nhược điểm của Laravel

* Hiệu năng không cao bằng các ngôn ngữ biên dịch: Do sử dụng PHP, Laravel có tốc độ chậm hơn so với Node.js hoặc Golang trong những hệ thống cực lớn cần xử lý realtime.
* Đòi hỏi người lập trình có kiến thức nền tảng: Laravel yêu cầu người học nắm vững: PHP OOP, Composer, MVC, cấu trúc dự án.
* Tiêu tốn tài nguyên hơn các framework PHP truyền thống: Laravel sử dụng nhiều lớp và thư viện, do đó cần hosting/VPS cấu hình tốt hơn so với các framework nhẹ như CodeIgniter [8].

## Hệ quản trị cơ sở dữ liệu MySQL

### Giới thiệu về hệ quản trị dữ liệu MySQL

MySQL là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) phổ biến hàng đầu thế giới, sử dụng ngôn ngữ SQL để lưu trữ và truy vấn dữ liệu. MySQL có ưu điểm nổi bật như tốc độ nhanh, ổn định, dễ sử dụng, mã nguồn mở và hỗ trợ tốt cho các ứng dụng web [9].

### Các tính năng chính của MySQL

Hiệu năng cao và tốc độ xử lý nhanh: MySQL tối ưu tốt cho các ứng dụng web với số lượng truy vấn lớn.

Hỗ trợ nhiều kiểu dữ liệu: bao gồm số, chuỗi, ngày giờ, JSON,… phù hợp cho nhiều loại ứng dụng.

Bảo mật tốt, dễ mở rộng: hỗ trợ phân quyền người dùng, mã hóa kết nối và cơ chế xác thực an toàn, hỗ trợ cơ chế replication, sharding và clustering giúp mở rộng quy mô hệ thống.

Tương thích cao: hoạt động tốt với hầu hết framework web, đặc biệt là Laravel thông qua Eloquent ORM và Query Builder.

Dễ quản lý:có nhiều công cụ phổ biến như phpMyAdmin, MySQL Workbench hỗ trợ quản trị trực qua

### Ứng dụng của MySQL

MySQL được sử dụng rộng rãi trong nhiều hệ thống nhờ tính ổn định và khả năng xử lý tốt:

* Quản lý dữ liệu cho các website thương mại điện tử.
* Lưu trữ dữ liệu người dùng, sản phẩm, đơn hàng, thông tin liên hệ.
* Ứng dụng trong các hệ thống web và mobile có lượng truy cập lớn.
* Xây dựng các hệ thống thông tin doanh nghiệp, CRM, quản lý bán hàng.
* Làm cơ sở dữ liệu cho các ứng dụng API, hệ thống microservices và backend
* Ứng dụng các hệ thống đặt vé, đặt phòng, quản lý lịch và dịch vụ trực tuyến.

## Eloquent ORM trong Laravel

### Giới thiệu Eloquent ORM

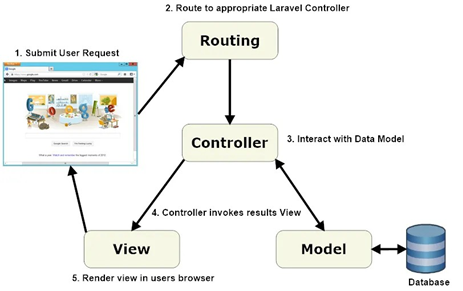
Eloquent ORM là hệ thống ánh xạ quan hệ – đối tượng (ORM) được tích hợp sẵn trong Laravel. Eloquent cho phép thao tác với cơ sở dữ liệu MySQL thông qua mô hình (Model) một cách trực quan, thay vì viết câu lệnh SQL thủ công. Mỗi bảng trong cơ sở dữ liệu được ánh xạ thành một Model tương ứng giúp việc truy vấn, thêm, sửa, xóa dữ liệu trở nên đơn giản và dễ quản lý [10].

### Đặc điểm cơ bản của Eloquent ORM

* **Truy vấn dữ liệu đơn giản:** cho phép thao tác CRUD bằng cú pháp hướng đối tượng, không cần viết SQL phức tạp.
* **Hỗ trợ quan hệ mạnh mẽ:** như one-to-one, one-to-many, many-to-many, polymorphic,…
* Hỗ trợ Migration: giúp tạo bảng, cập nhật cấu trúc dữ liệu bằng code, dễ bảo trì và theo dõi phiên bản.
* Eager Loading: tối ưu hiệu năng truy vấn khi làm việc với quan hệ dữ liệu.
* Dễ tích hợp với MySQL: Eloquent làm việc trực tiếp và tối ưu với MySQL, phù hợp cho các ứng dụng thương mại điện tử.

## Mô hình MVC (Model – View – Controller)

### Giới thiệu mô hình MVC



Hình 1.2 mô hình MVC

MVC là viết tắt của Model – View – Controller, là một mẫu kiến trúc (design pattern) được sử dụng phổ biến trong phát triển ứng dụng web hiện đại. Mô hình MVC chia ứng dụng thành ba thành phần chính gồm Model, View và Controller, mỗi thành phần đảm nhiệm một vai trò riêng biệt nhằm tách biệt rõ ràng giữa xử lý dữ liệu, xử lý nghiệp vụ và hiển thị giao diện.

Việc áp dụng mô hình MVC giúp cho mã nguồn trở nên rõ ràng, dễ hiểu và dễ bảo trì, đồng thời hỗ trợ làm việc nhóm hiệu quả khi mỗi thành viên có thể đảm nhận một phần riêng trong hệ thống. Mô hình này cũng giúp tăng khả năng mở rộng, cho phép nâng cấp hoặc thay đổi từng thành phần mà không ảnh hưởng lớn đến các phần còn lại [8].

### Đặc điểm của MVC

* Cung cấp sự phân tách rõ ràng giữa logic nghiệp vụ, logic giao diện người dùng và logic xử lý yêu cầu giúp mã nguồn dễ hiểu và dễ bảo trì.
* Cung cấp toàn quyền kiểm soát HTML và URL, giúp bạn dễ dàng thiết kế kiến trúc ứng dụng web.
* Hỗ trợ Lập trình dựa trên thử nghiệm (Test-driven Development).

MVC là một mẫu thiết kế tiêu chuẩn được nhiều lập trình viên quen thuộc nhờ vào khả năng mở rộng và có thể mở rộng. MVC thường được sử dụng để làm framework phát triển web tiêu chuẩn cũng như các ứng dụng di động.

### Các thành phần trong mô hình MVC

#### Model

Model là các thành phần của ứng dụng tương ứng với tất cả logic liên quan đến miền dữ liệu (data domain) hoặc nói ngắn gọn đây là phần back-end chứa tất cả logic dữ liệu của ứng dụng.

Dữ liệu ở đây có thể là dữ liệu đang được truyền giữa các thành phần View và Controller hoặc bất kỳ dữ liệu nào khác liên quan đến logic của doanh nghiệp.

#### View

View là các thành phần hiển thị giao diện người dùng (UI) của ứng dụng. Thông thường, giao diện người dùng này được tạo từ dữ liệu Model.

#### Controller

Controller là các thành phần xử lý tương tác của người dùng để làm việc với Model (cập nhật logic dữ liệu) hoặc/ và với View (cập nhật hiển thị giao diện người dùng). Trong ứng dụng MVC, Controller xử lý các giá trị chuỗi truy vấn và chuyển các giá trị này cho Model.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Mô tả bài toán

Trong những năm gần đây, cùng với sự phát triển mạnh mẽ của công nghệ số, các sản phẩm sáng tạo như trò chơi điện tử, truyện online, phim ảnh và các nội dung giải trí số ngày càng được đầu tư kỹ lưỡng về nội dung và chiều sâu. Một trong những yếu tố quan trọng góp phần tạo nên sức hấp dẫn và tính bền vững của các sản phẩm này chính là việc xây dựng thế giới hư cấu (World Building). World Building không chỉ đơn thuần là tạo ra bối cảnh cho câu chuyện mà còn bao gồm việc xây dựng hệ thống nhân vật, địa danh, sự kiện, quy luật vận hành và mối quan hệ giữa các yếu tố trong thế giới đó.

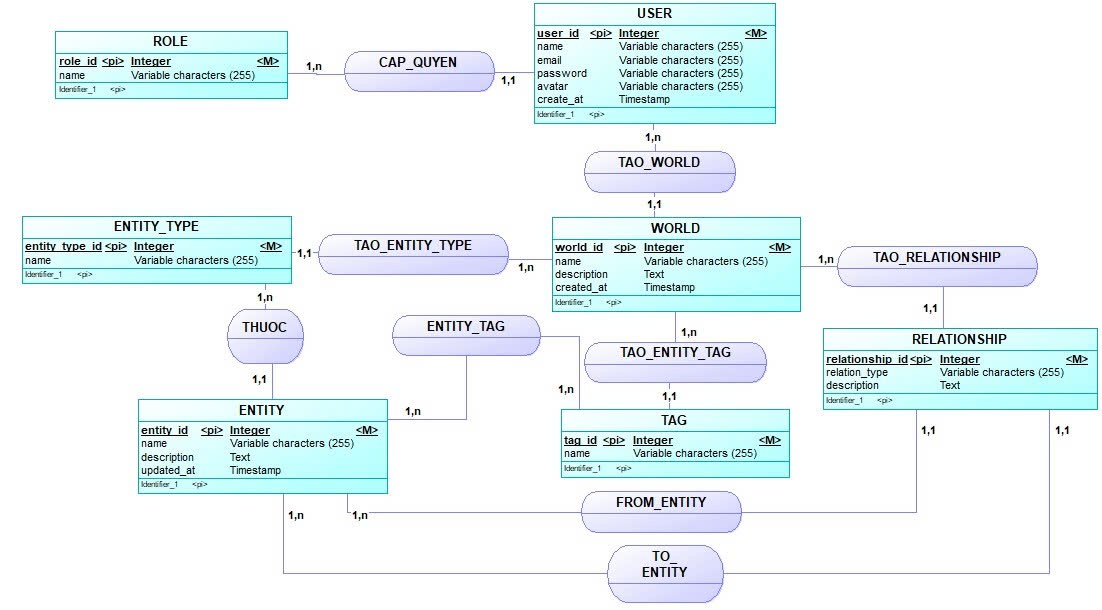
Tuy nhiên, trên thực tế, quá trình xây dựng và quản lý thế giới hư cấu thường gặp nhiều khó khăn, đặc biệt đối với các dự án có quy mô lớn và thời gian phát triển kéo dài. Phần lớn tác giả và nhà sáng tạo nội dung vẫn quản lý thông tin bằng các phương pháp thủ công như ghi chép sổ tay, tệp văn bản hoặc bảng tính rời rạc Cách làm này thiếu tính liên kết, gây khó khăn trong việc tra cứu, cập nhật và kiểm soát tính nhất quán của nội dung, đồng thời dễ phát sinh sai sót khi số lượng nhân vật và sự kiện ngày càng tăng.

Xuất phát từ nhu cầu thực tiễn đó, đề tài World Building được xây dựng dưới dạng một ứng dụng web hỗ trợ người dùng tạo lập và quản lý các thế giới hư cấu phục vụ cho game, truyện, phim và các dự án sáng tạo khác. Ứng dụng cho phép người dùng tổ chức và lưu trữ thông tin về các thành phần của thế giới như nhân vật, bối cảnh, sự kiện và quy luật vận hành một cách tập trung và có hệ thống, giúp quá trình sáng tác trở nên khoa học và hiệu quả hơn.

Bên cạnh chức năng quản lý dữ liệu, ứng dụng còn tích hợp trí tuệ nhân tạo (AI) nhằm hỗ trợ người dùng trong quá trình sáng tạo nội dung.

## Phân tích và thiết kế hệ thống

### Mô hình dữ liệu mức quan niệm



Hình 2.1 Mô hình dữ liệu mức quan niệm

### Mô hình dữ liệu mức luận lý

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, biểu đồ

Mô tả được tạo tự động

Hình 2.2 Mô hình dữ liệu mức luận lý

### Mô hình dữ liệu mức vật lý

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, biểu đồ

Mô tả được tạo tự động

Hình 2.3 Mô hình dữ liệu mức vật lý

### Mô hình cơ sở dữ liệu

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

Hình 2.4 Mô hình cơ sở dữ liệu

#### Bảng users - Người dùng

Bảng 2.1 Bảng users - Người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã người dùng | BIGINT(20) | Khóa chính (PK), Tự tăng |
| name | Tên người dùng | VARCHAR(255) | Không được rỗng |
| email | Email đăng nhập | VARCHAR(255) | Không được rỗng, Duy nhất |
| password | Mật khẩu | VARCHAR(255) | Không được rỗng |
| avatar | Ảnh đại diện | VARCHAR(255) | Có thể rỗng |
| role\_id | Mã vai trò | BIGINT(20) | Khóa ngoại (FK) |
| created\_at | Ngày tạo | TIMESTAMP | Tự động |

Mô tả: Bảng users (Người dùng): Lưu trữ thông tin tài khoản người dùng trong hệ thống.

#### Bảng roles - Vai trò người dùng

Bảng 2.2 Bảng roles - Vai trò người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã vai trò | BIGINT(20) | Khóa chính (PK), Tự tăng |
| name | Tên vai trò | VARCHAR(50) | Không được rỗng |

**Mô tả: Bảng** roles **(Vai trò):** Quản lý các vai trò để phân quyền người dùng.

#### Bảng worlds - Thế giới hư cấu

Bảng 2.3 Bảng worlds - Thế giới hư cấu

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã thế giới | BIGINT(20) | Khóa chính (PK), Tự tăng |
| owner\_id | Người sở hữu | BIGINT(20) | Khóa ngoại (FK → users) |
| name | Tên thế giới | VARCHAR(255) | Không được rỗng |
| description | Mô tả thế giới | TEXT | Có thể rỗng |
| created\_at | Ngày tạo | TIMESTAMP | Tự động |

**Mô tả: Bảng** worlds **(Thế giới):** Lưu trữ thông tin các thế giới hư cấu do người dùng tạo.

#### Bảng entity\_types - Loại thực thể

Bảng 2.4 Bảng entity\_types - Loại thực thể

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã thế giới | BIGINT(20) | Khóa chính (PK), Tự tăng |
| owner\_id | Người sở hữu | BIGINT(20) | Khóa ngoại (FK → users) |
| name | Tên thế giới | VARCHAR(255) | Không được rỗng |
| description | Mô tả thế giới | TEXT | Có thể rỗng |
| created\_at | Ngày tạo | TIMESTAMP | Tự động |

**Mô tả: Bảng** entity\_types **(Loại thực thể):** Phân loại thực thể trong thế giới (nhân vật, địa điểm, vật phẩm…).

#### Bảng entities - Thực thể trong thế giới

Bảng 2.5 Bảng entities - Thực thể trong thế giới

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã loại thực thể | BIGINT(20) | Khóa chính (PK), Tự tăng |
| world\_id | Mã thế giới | BIGINT(20) | Khóa ngoại (FK → worlds) |
| name | Tên loại thực thể | VARCHAR(100) | Không được rỗng |

#### Bảng relationships - Quan hệ giữa các thực thể

Bảng 2.6 Bảng relationships - Quan hệ giữa các thực thể

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã quan hệ | BIGINT(20) | Khóa chính (PK), Tự tăng |
| world\_id | Mã thế giới | BIGINT(20) | Khóa ngoại (FK → worlds) |
| from\_entity\_id | Thực thể nguồn | BIGINT(20) | Khóa ngoại (FK → entities) |
| to\_entity\_id | Thực thể đích | BIGINT(20) | Khóa ngoại (FK → entities) |
| relation\_type | Loại quan hệ | VARCHAR(100) | Không được rỗng |
| description | Mô tả quan hệ | TEXT | Có thể rỗng |

**Mô tả: Bảng** relationships **(Quan hệ):** Mô tả mối quan hệ giữa các thực thể trong thế giới.

#### Bảng tags - Thẻ gắn cho thực thể

Bảng 2.7 Bảng tags - Thẻ gắn cho thực thể

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã quan hệ | BIGINT(20) | Khóa chính (PK), Tự tăng |
| world\_id | Mã thế giới | BIGINT(20) | Khóa ngoại (FK → worlds) |
| from\_entity\_id | Thực thể nguồn | BIGINT(20) | Khóa ngoại (FK → entities) |
| to\_entity\_id | Thực thể đích | BIGINT(20) | Khóa ngoại (FK → entities) |
| relation\_type | Loại quan hệ | VARCHAR(100) | Không được rỗng |
| description | Mô tả quan hệ | TEXT | Có thể rỗng |

**Mô tả: Bảng** tags (**Thẻ):** Phân loại và hỗ trợ tìm kiếm thực thể.

#### Bảng entity\_tags - Liên kết thực thể và thẻ

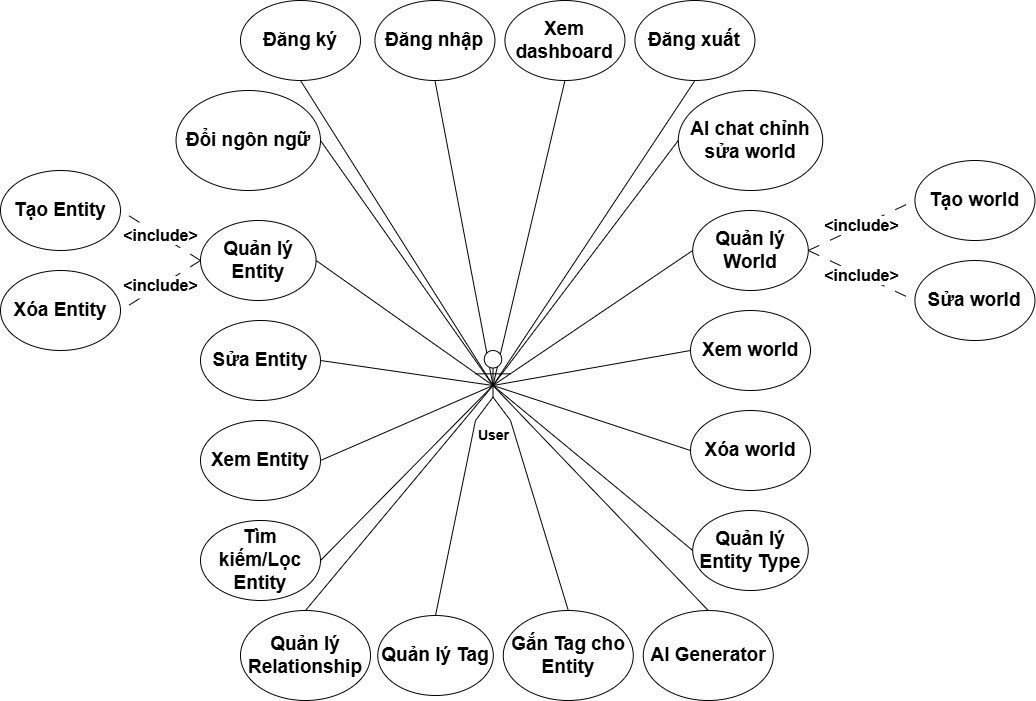
Bảng 2.8 Bảng entity\_tags - Liên kết thực thể và thẻ

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã thẻ | BIGINT(20) | Khóa chính (PK), Tự tăng |
| world\_id | Mã thế giới | BIGINT(20) | Khóa ngoại (FK → worlds) |
| name | Tên thẻ | VARCHAR(100) | Không được rỗng |

#### ****Mô tả: Bảng**** entity\_tags ****(Gắn thẻ thực thể):**** Liên kết thực thể với các thẻ tương ứng.

### Sơ đồ Usecase hệ thống

#### Sơ đồ Usecase Người dùng



Hình 2.5 Sơ đồ Usecase Người dùng

Người dùng có thể đăng ký, đăng nhập, đăng xuất, xem dashboard và thay đổi ngôn ngữ. Hệ thống cho phép người dùng quản lý thế giới (tạo, xem, sửa, xóa world), quản lý thực thể (tạo, xem, sửa, xóa, tìm kiếm/lọc entity), quản lý loại thực thể, thẻ (tag) và mối quan hệ giữa các thực thể. Ngoài ra, hệ thống tích hợp AI Generator và AI chat chỉnh sửa world nhằm hỗ trợ tự động tạo và chỉnh sửa nội dung thế giới hư cấu.

#### Sơ đồ Usecase Admin

Ảnh có chứa văn bản, biểu đồ, vòng tròn, hàng

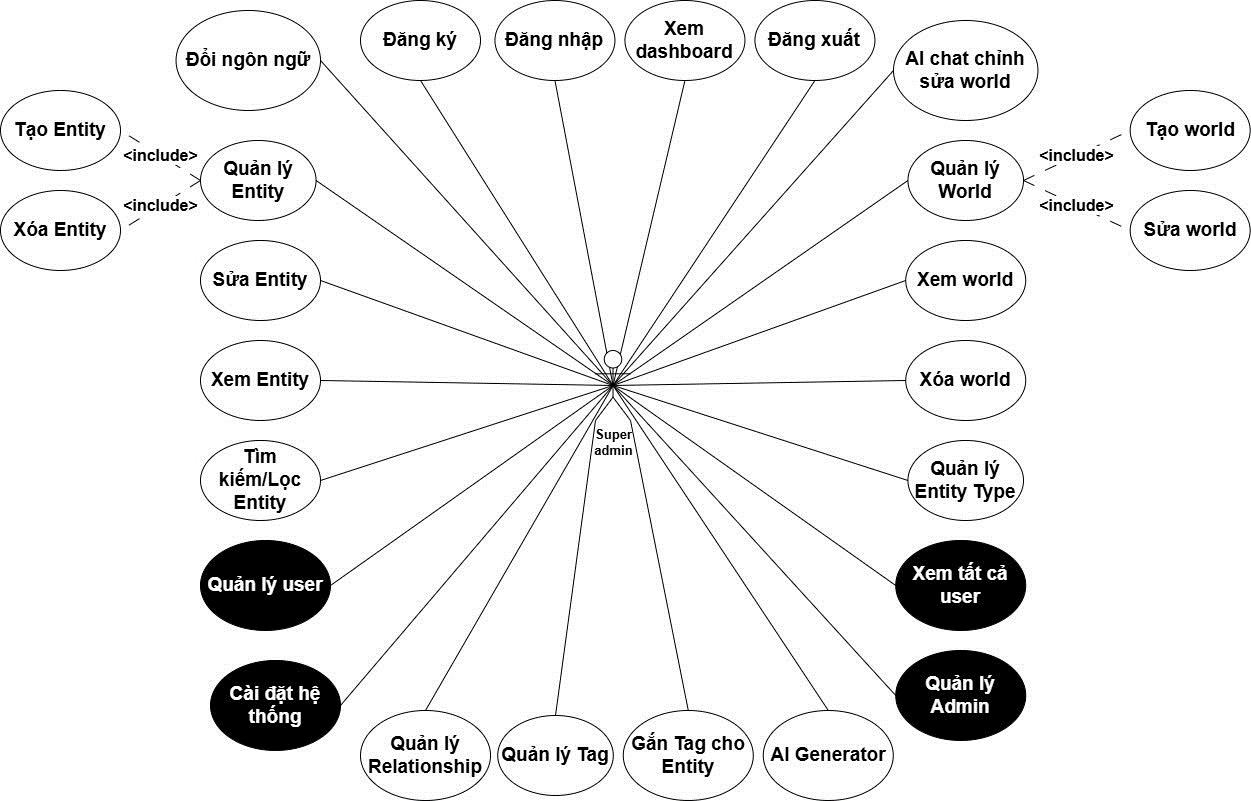
Mô tả được tạo tự động

Hình 2.6 Sơ đồ Usecase Admin

Sơ đồ Use Case của **Admin** mô tả chi tiết các chức năng quản trị nội dung trong hệ thống **World Building**. Admin là tác nhân đóng vai trò trung gian giữa người dùng thông thường và hệ thống, chịu trách nhiệm quản lý, duy trì và cập nhật dữ liệu nhằm đảm bảo hệ thống vận hành ổn định, chính xác và hiệu quả. Thông qua sơ đồ, có thể thấy Admin được cấp quyền truy cập vào hầu hết các chức năng cốt lõi của hệ thống.

Về các chức năng cơ bản, Admin có thể thực hiện đăng ký, đăng nhập, đăng xuất, xem dashboard và thay đổi ngôn ngữ sử dụng hệ thống. Chức năng dashboard giúp Admin theo dõi tổng quan tình trạng hoạt động của hệ thống như số lượng world, entity, người dùng đang hoạt động và các tác vụ liên quan đến trí tuệ nhân tạo. Việc hỗ trợ đổi ngôn ngữ giúp hệ thống thân thiện hơn với nhiều đối tượng người dùng khác nhau.

#### Sơ đồ Usecase super Admin



Hình 2.7 Sơ đồ Usecase super Admin

Sơ đồ Use Case của Super Admin mô tả các chức năng quản trị cao nhất trong hệ thống World Building. Super Admin là tác nhân có quyền kiểm soát toàn diện hệ thống, chịu trách nhiệm giám sát hoạt động, phân quyền và cấu hình các thành phần quan trọng nhằm đảm bảo hệ thống vận hành ổn định, an toàn và đúng định hướng. Ngoài các chức năng cơ bản như đăng nhập, đăng xuất, xem dashboard và thay đổi ngôn ngữ, Super Admin có quyền truy cập vào toàn bộ chức năng quản trị.

Trong nghiệp vụ quản lý hệ thống, Super Admin có thể quản lý người dùng, xem danh sách tất cả người dùng và quản lý tài khoản Admin, bao gồm tạo mới, chỉnh sửa hoặc thu hồi quyền quản trị khi cần thiết. Các chức năng này giúp kiểm soát chặt chẽ quyền truy cập và phân cấp người dùng trong hệ thống, hạn chế rủi ro về bảo mật và sai sót trong quá trình vận hành. Bên cạnh đó, Super Admin thực hiện cài đặt và cấu hình hệ thống, đảm bảo các thông số kỹ thuật, chức năng và tích hợp hoạt động đúng yêu cầu.

Ngoài các quyền quản trị nâng cao, Super Admin kế thừa toàn bộ chức năng nghiệp vụ của Admin như quản lý world, thực thể, loại thực thể, thẻ, mối quan hệ và gán thẻ cho thực thể. Super Admin cũng có thể sử dụng các chức năng AI Generator và AI chat chỉnh sửa world để hỗ trợ xây dựng và hoàn thiện nội dung thế giới hư cấu. Nhờ đó, Super Admin vừa đảm bảo vai trò quản lý hệ thống, vừa trực tiếp tham gia phát triển và kiểm soát chất lượng dữ liệu trong hệ thống World Building.

## Phác thảo giao diện

### Trang Đăng nhập

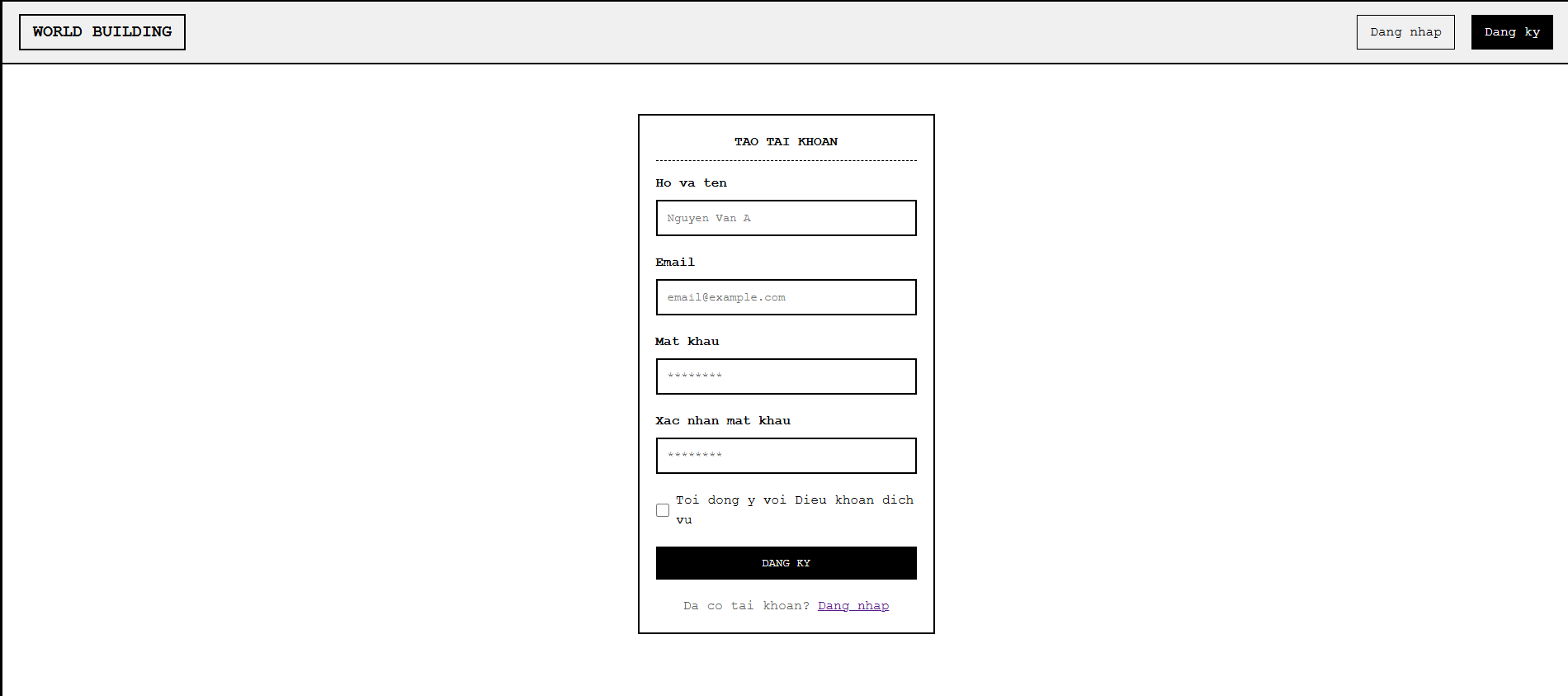
Ảnh có chứa ảnh chụp màn hình, văn bản, thiết kế

Mô tả được tạo tự động

Hình 2.8 Trang Đăng nhập

Trang đăng nhập cho phép người dùng nhập email và mật khẩu để truy cập hệ thống. Người dùng điền email vào ô "Email", nhập mật khẩu vào ô "Mật khẩu", có thể tích chọn "Ghi nhớ đăng nhập" để không phải đăng nhập lại lần sau, sau đó nhấn nút "ĐĂNG NHẬP" để vào hệ thống. Nếu chưa có tài khoản, người dùng có thể click vào link "Đăng ký" để chuyển sang trang đăng ký.

### Trang Đăng ký



Hình.2.9 Trang Đăng ký

Trang đăng ký cho phép người dùng tạo tài khoản mới. Người dùng nhập họ và tên vào ô "Họ và tên", nhập email vào ô "Email", nhập mật khẩu vào ô "Mật khẩu", nhập lại mật khẩu vào ô "Xác nhận mật khẩu" để đảm bảo không nhập sai, tích chọn checkbox "Tôi đồng ý với Điều khoản dịch vụ", sau đó nhấn nút "ĐĂNG KÝ" để tạo tài khoản. Nếu đã có tài khoản, người dùng có thể click "Đăng nhập" để quay lại trang đăng nhập.

### Trang Bảng điều khiển

Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, thiết kế

Mô tả được tạo tự động

Hình 2.10 Trang Bảng điều khiển

Sau khi đăng nhập thành công, người dùng được chuyển đến trang Dashboard trang chủ của hệ thống. Trang hiển thị lời chào "Chào mừng trở lại, [Tên]!" và 4 thẻ thống kê cho biết số lượng: Thế giới (5), Thực thể (48), Mối quan hệ (32), và Users (15). Phía dưới là phần "Thế giới của tôi" hiển thị danh sách các thế giới đã tạo dạng lưới 3 cột, mỗi thẻ có hình ảnh, tên thế giới, mô tả ngắn và số thực thể.

Người dùng có thể click nút "Khám phá" trên mỗi thẻ để xem chi tiết thế giới, hoặc click nút "+ Tạo thế giới" hoặc "Tạo bằng AI" ở góc trên để tạo thế giới mới. Menu điều hướng ở header cho phép chuyển sang các trang khác: Dashboard, Thế giới, Tạo bằng AI, Users.

### Trang danh sách thế giới

Ảnh có chứa văn bản, ảnh chụp màn hình, số, Song song

Mô tả được tạo tự động

Hình 2.11 Trang danh sách thế giới

Trang này hiển thị tất cả các thế giới dạng bảng với đầy đủ thông tin. Người dùng có thể xem danh sách các thế giới với các cột: Tên thế giới, Mô tả, Số thực thể, Số loại. Mỗi dòng có 2 nút: "Xem" để xem chi tiết thế giới và "Sửa" để chỉnh sửa. Người dùng có thể click nút "+ Tạo thế giới" hoặc "Tạo bằng AI" ở góc trên để tạo thế giới mới. Nếu có nhiều thế giới, người dùng dùng các nút phân trang "Trước", "1", "2", "3", "Sau" ở cuối trang để xem thêm.

### Trang chi tiết thế giới

Ảnh có chứa văn bản, ảnh chụp màn hình, số, biểu đồ

Mô tả được tạo tự động

Hình 2.12 Trang Chi tiết thế giới

Khi người dùng click "Xem" hoặc "Khám phá" một thế giới, họ được chuyển đến trang chi tiết. Trang hiển thị đầy đủ thông tin về thế giới: tên, mô tả chi tiết, và 4 thẻ thống kê (Thực thể: 12, Loại thực thể: 5, Mối quan hệ: 10, Nhãn: 10). Người dùng có thể click nút "Sửa" để chỉnh sửa thế giới hoặc "Quay lại" để về danh sách. Có 4 nút quick action: "Thêm thực thể", "Thêm loại", "Thêm quan hệ", "Thêm nhãn" để nhanh chóng thêm các thành phần mới. Phần "Thực thể gần đây" hiển thị 3 thực thể mới nhất dạng thẻ, mỗi thẻ có tên, loại, mô tả ngắn và các nhãn. Người dùng click "Xem tất cả" để xem danh sách đầy đủ các thực thể.

### Trang sửa thế giới

Ảnh có chứa văn bản, ảnh chụp màn hình, số, phần mềm

Mô tả được tạo tự động

Hình 2.13 Trang Sửa thế giới

Trang cho phép người dùng chỉnh sửa thông tin thế giới. Người dùng có thể sửa tên thế giới trong ô "Tên thế giới", sửa mô tả trong ô "Mô tả". Đặc biệt, bên cạnh ô mô tả có nút "Sửa bằng AI" - khi click vào, một cửa sổ popup hiện ra cho phép người dùng nhập yêu cầu chỉnh sửa (ví dụ: "Thêm chi tiết hơn", "Dịch sang tiếng Anh", "Rút gọn lại"), sau đó nhấn "Áp dụng" để AI tự động sửa nội dung mô tả. Trang cũng hiển thị thống kê thế giới (12 Thực thể, 5 Loại, 10 Mối quan hệ). Sau khi sửa xong, người dùng nhấn "Lưu thay đổi" để lưu hoặc "Hủy" để bỏ qua. Nếu muốn xóa thế giới, người dùng nhấn nút "Xóa thế giới" ở góc trái.

### Trang danh sách thực thể

Ảnh có chứa văn bản, ảnh chụp màn hình, số, Song song

Mô tả được tạo tự động

Hình 2.14 Trang danh sách thực thể

Trang hiển thị tất cả thực thể trong một thế giới. Người dùng có thể tìm kiếm bằng cách nhập tên vào ô "Tìm kiếm thực thể...", lọc theo loại bằng dropdown "Tất cả loại" (Nhân vật, Địa điểm, Vật phẩm, Sự kiện), lọc theo nhãn bằng dropdown "Tất cả nhãn", sau đó nhấn "Lọc" để áp dụng.

Danh sách thực thể hiển thị dạng lưới 3 cột, mỗi thẻ có tên, loại (tag), mô tả ngắn, các nhãn và 2 nút "Xem" (xem chi tiết) và "Sửa" (chỉnh sửa). Người dùng click "+ Thêm thực thể" ở góc trên để tạo thực thể mới. Có phân trang ở cuối để xem thêm thực thể.

### Trang Chi tiết thực thể

Ảnh có chứa văn bản, ảnh chụp màn hình, số, Phông chữ

Mô tả được tạo tự động

Hình 2.15 Trang Chi tiết thực thể

Khi click "Xem" một thực thể, người dùng được chuyển đến trang chi tiết với đầy đủ thông tin. Phần chính hiển thị mô tả chi tiết về thực thể (có thể nhiều đoạn văn) và danh sách "Mối quan hệ".

liệt kê các thực thể khác có liên quan (ví dụ: Marcus Chen - Đồng minh với Sarah Kim, Thuộc về Trại Hy Vọng, Tạo ra Serum X-7). Cột bên phải hiển thị các nhãn gắn với thực thể và thông tin meta (Loại, Thế giới, Ngày tạo, Ngày cập nhật). Người dùng có thể click "Sửa" để chỉnh sửa, "Xóa" để xóa thực thể, hoặc "Quay lại" để về danh sách.

### Trang Tạo thế giới bằng AI

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Trang web

Mô tả được tạo tự động

Hình 2.16 Trang Tạo thế giới bằng AI

Trang cho phép người dùng tạo thế giới tự động bằng AI. Người dùng nhập mô tả chi tiết về thế giới muốn tạo vào ô textarea lớn (ví dụ: "Một thế giới fantasy thời trung cổ với rồng, hiệp sĩ và phép thuật. Có nhiều vương quốc đang chiến tranh..."), sau đó nhấn nút "Tạo thế giới".

AI sẽ tự động tạo ra thế giới hoàn chỉnh với các thực thể, mối quan hệ, loại và nhãn. Nếu không biết viết gì, người dùng có thể click vào 1 trong 4 ví dụ nhanh (Cyberpunk, Trường phép thuật, Space Opera, Hậu tận thế Zombie) để tự động điền mô tả mẫu. Phần "Mẹo để có kết quả tốt hơn" gợi ý 4 điểm: mô tả bối cảnh/thời kỳ, đề cập nhân vật/phe phái, bao gồm xung đột/bí ẩn, chỉ định thể loại.

### Trang Quản lý người dùng

Ảnh có chứa văn bản, ảnh chụp màn hình, số, Phông chữ

Mô tả được tạo tự động

Hình 2.17 Trang Quản lý người dùng

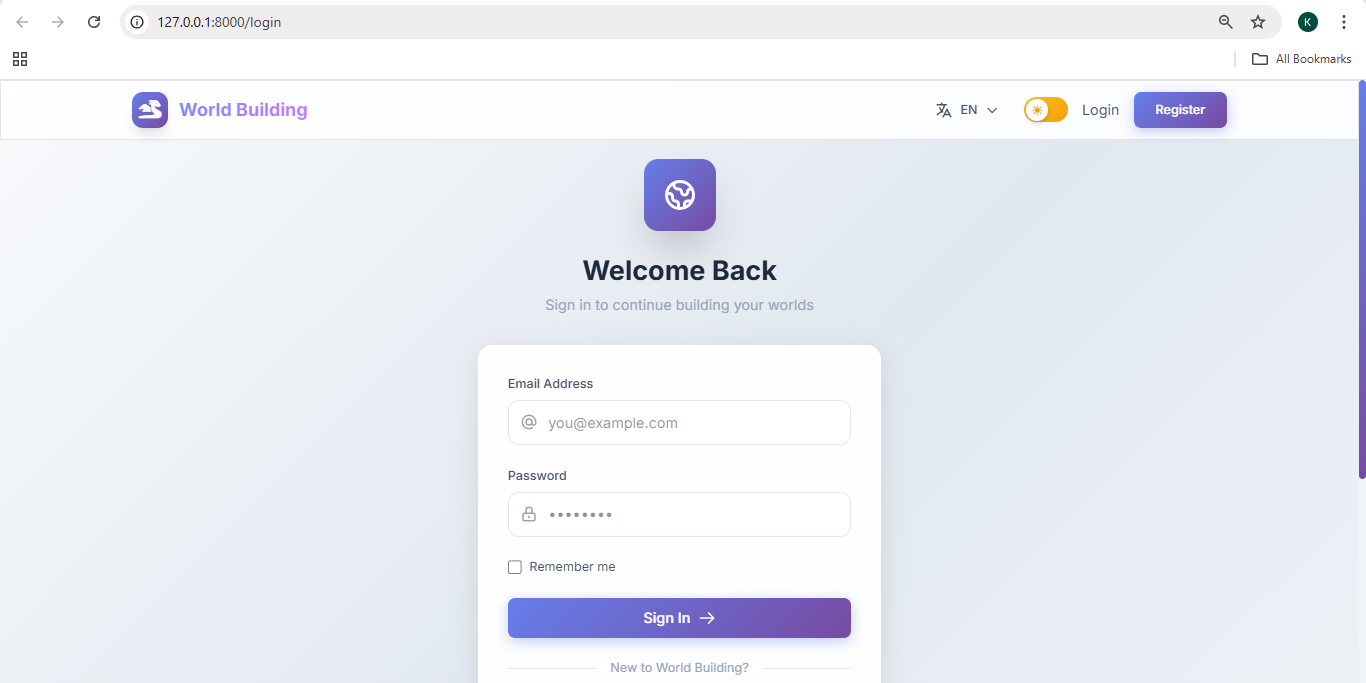
Trang dành cho Admin để quản lý người dùng. Hiển thị 4 thẻ thống kê: Tổng Users (15), Admins (2), Users thường (13), Tổng Worlds (28). Bảng danh sách users có các cột: Avatar, Tên, Email, Vai trò (super\_admin/admin/user), Số Worlds, Ngày tham gia. Mỗi dòng có nút "Sửa" để chỉnh sửa thông tin user, và nút "Xóa" để xóa user (trừ chính mình và super\_admin).

Admin click "+ Thêm người dùng" để mở popup, nhập Họ và tên, Email, Mật khẩu, chọn Vai trò từ dropdown (User/Admin/Super Admin), sau đó nhấn "Lưu" để tạo user mới hoặc "Hủy" để đóng popup. Có phân trang ở cuối để xem thêm users.

# XÂY DỰNG WEBSITE

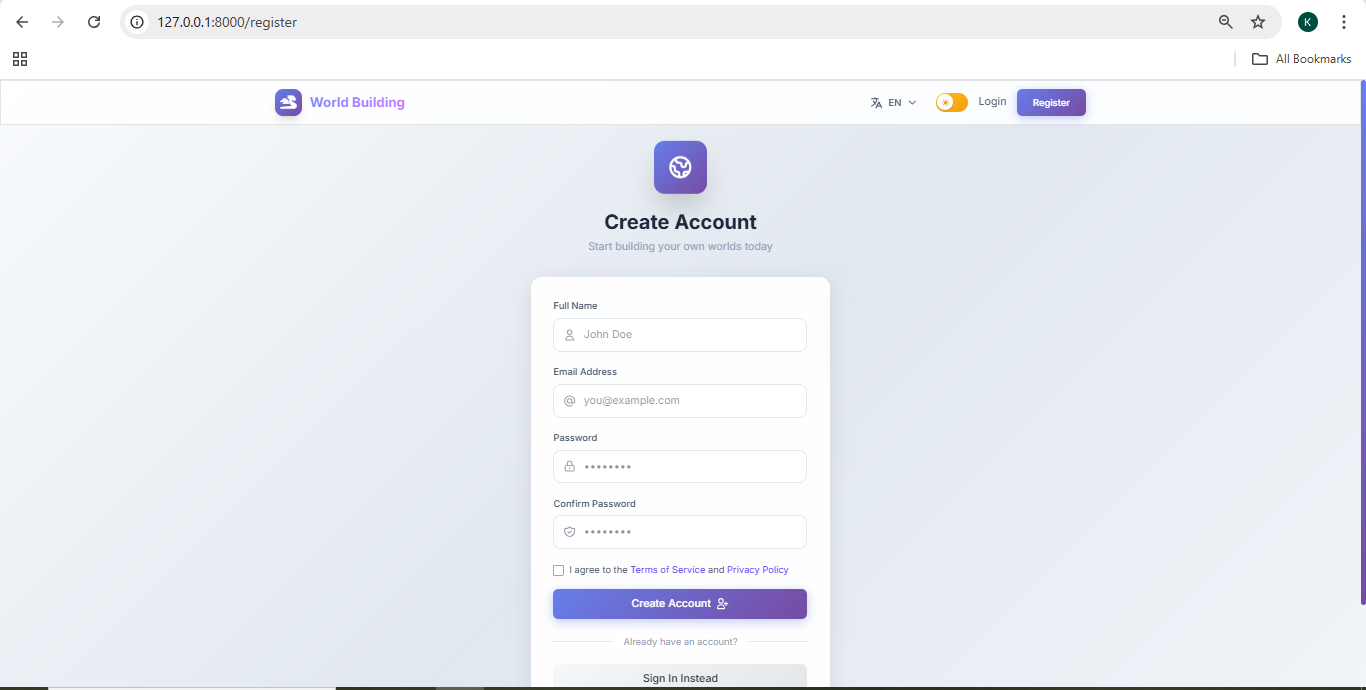
## Giới thiệu giao diện

### Giao diện đăng nhập



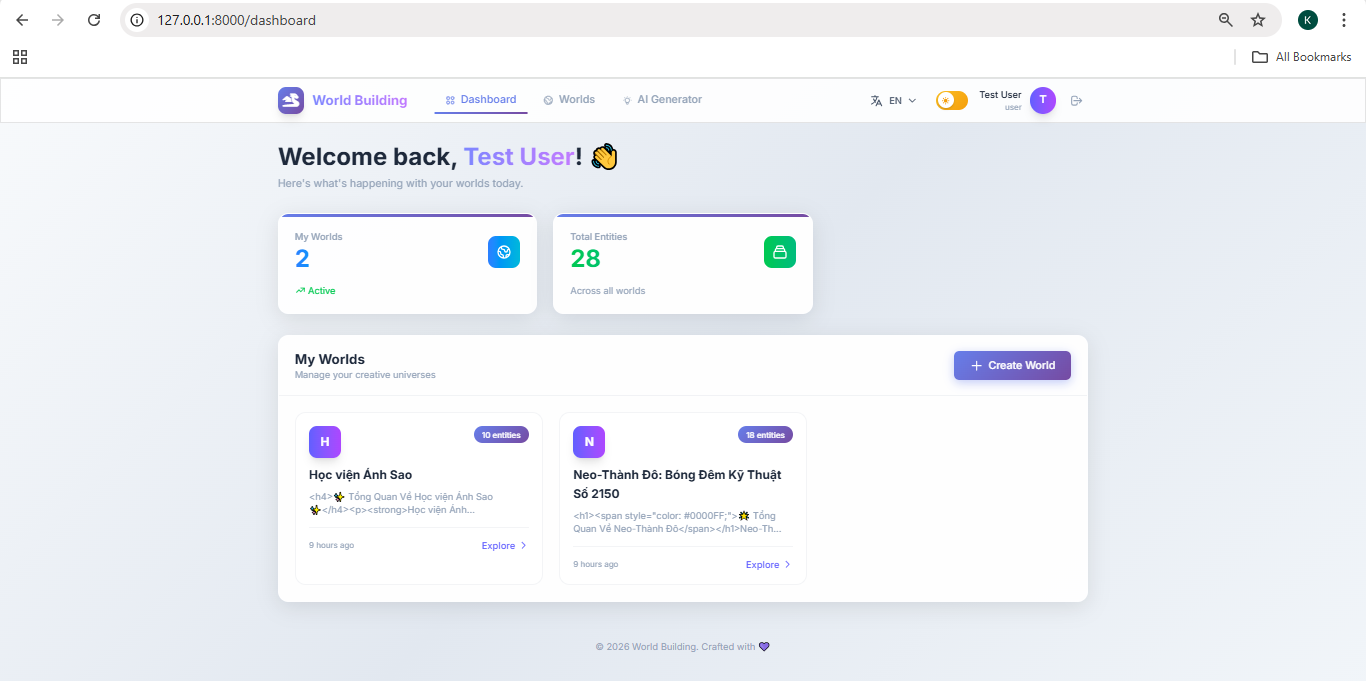
Hình 3.1 Giao diện đăng nhập

### Giao diện đăng ký



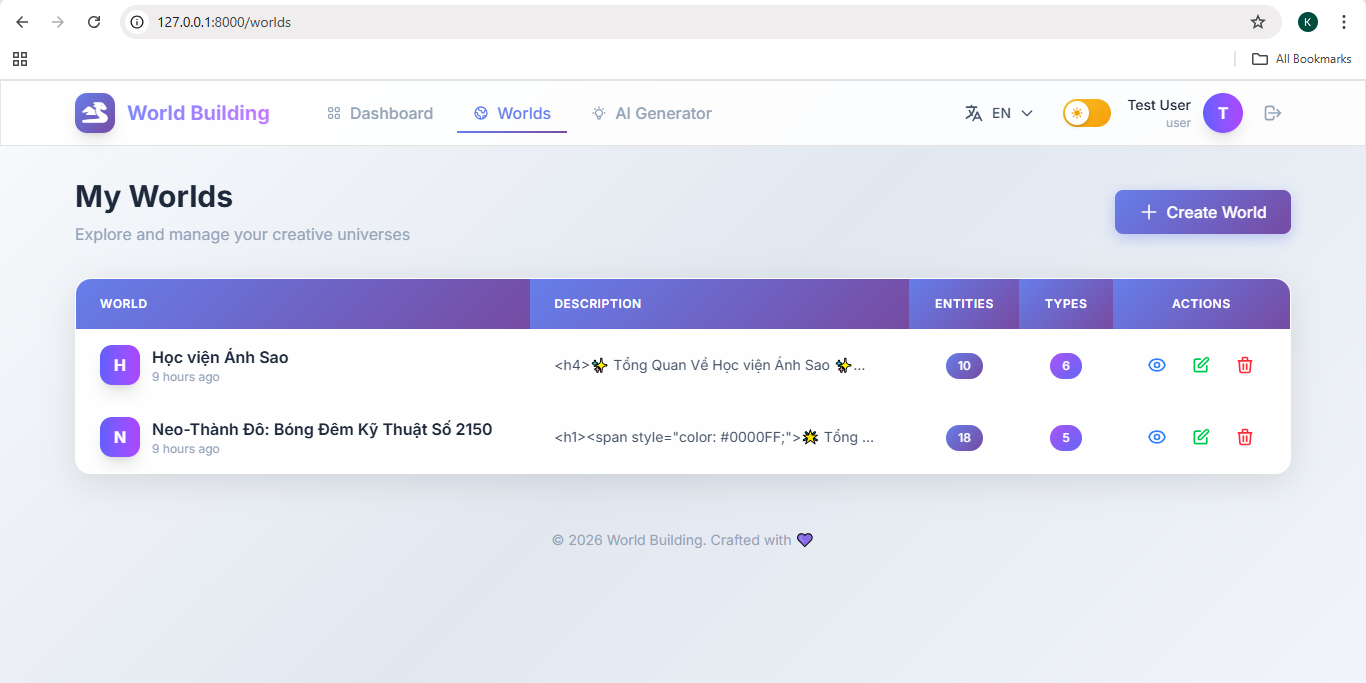
Hình 3.2 Giao diện đăng ký

### Giao diện bảng điều khiển



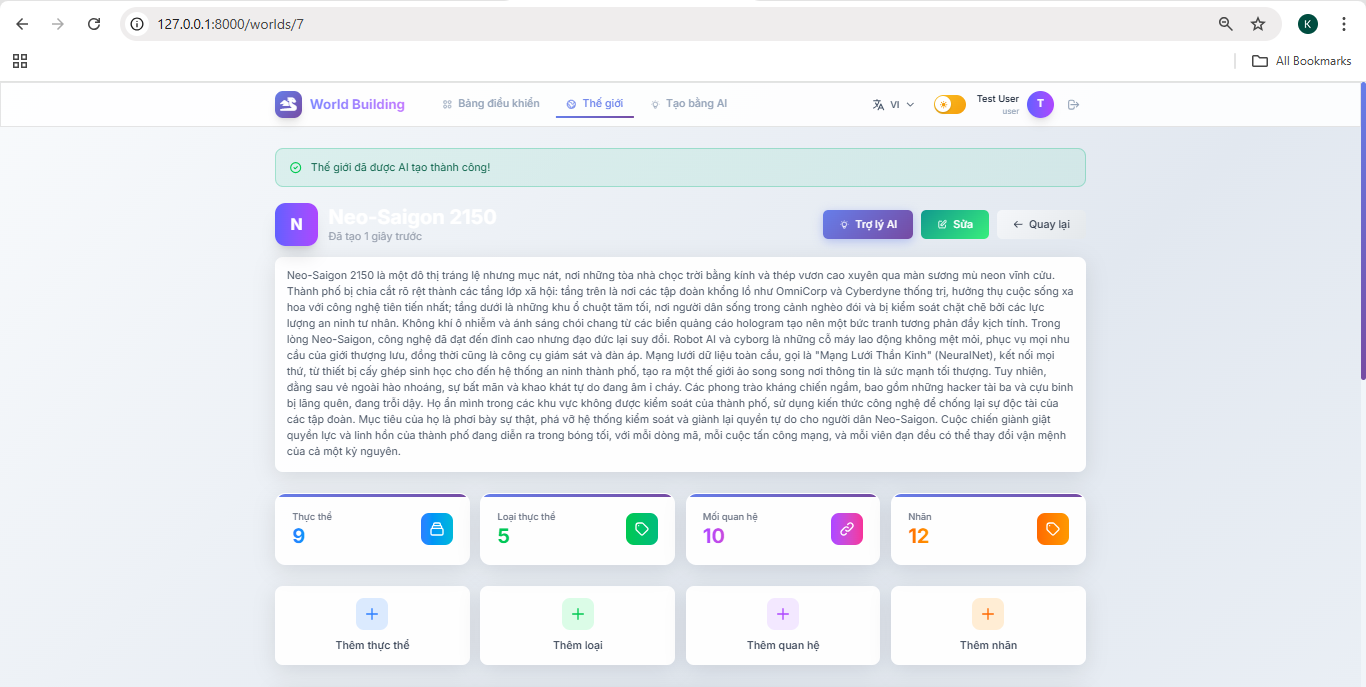
Hình 3.3 Giao diện bảng điều khiển

### Giao diện danh sách thế giới



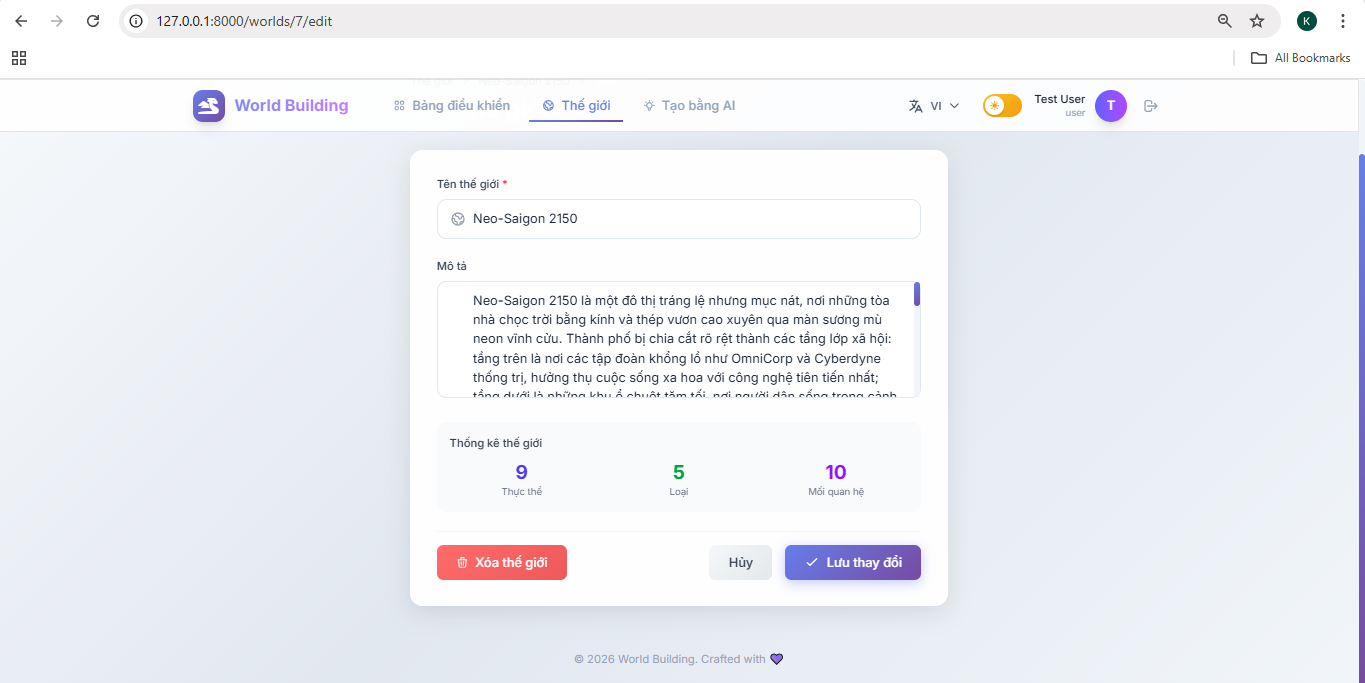
Hình 3.4 Giao diện danh sách thế giới

### Giao diện chi tiết thế giới



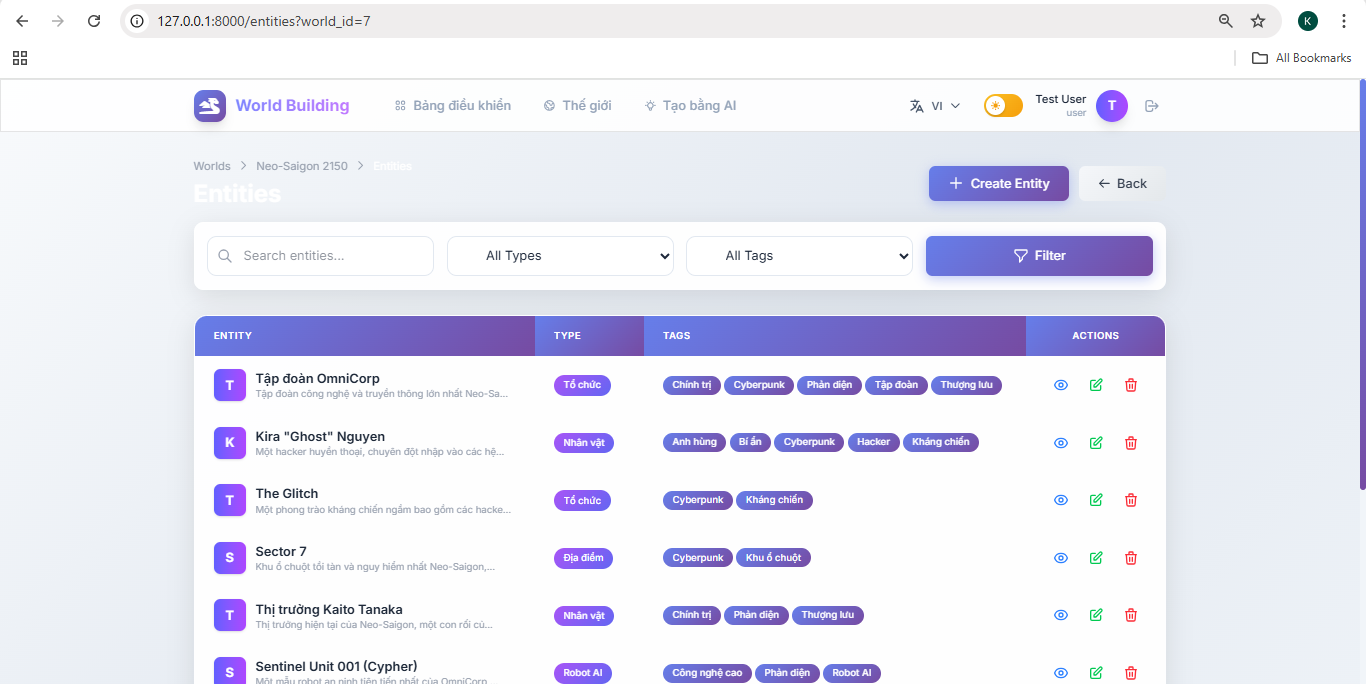
Hình 3.5 Giao diện chi tiết thế giới

### Giao diện sửa thế giới



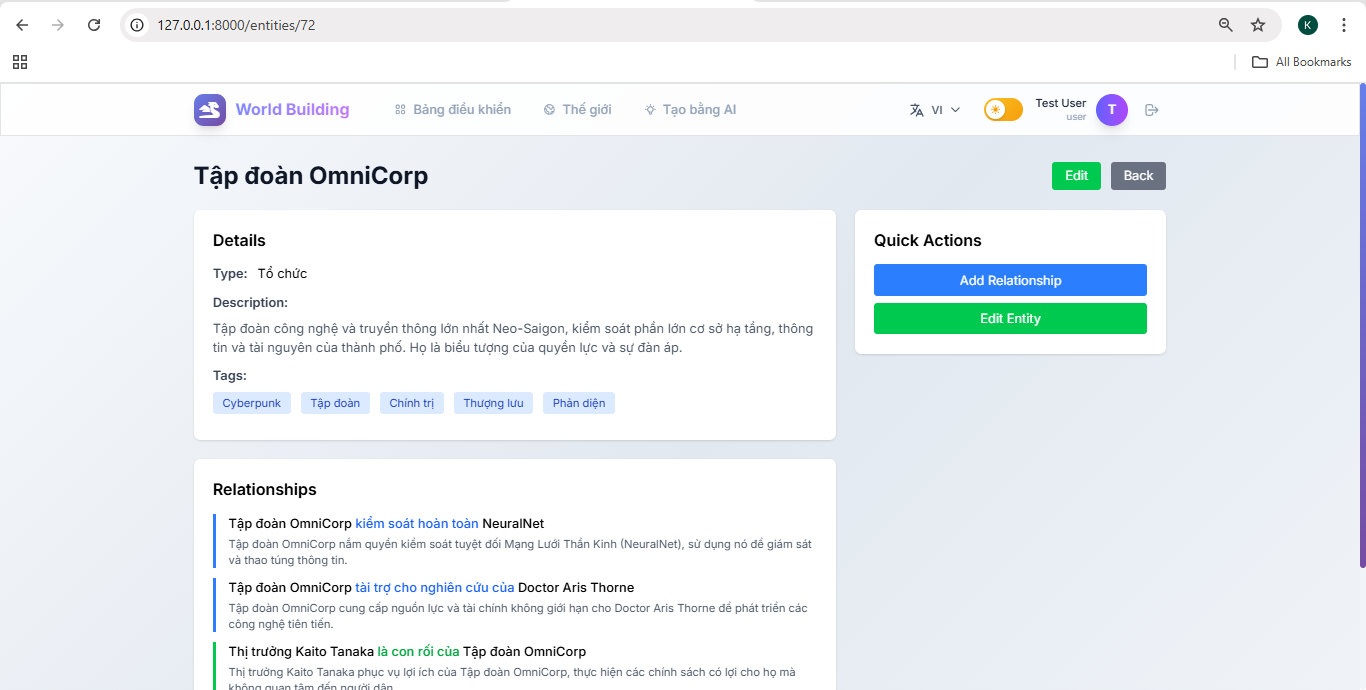
Hình 3.6 Giao diên sửa thế giới

### Giao diện danh sách thực thể



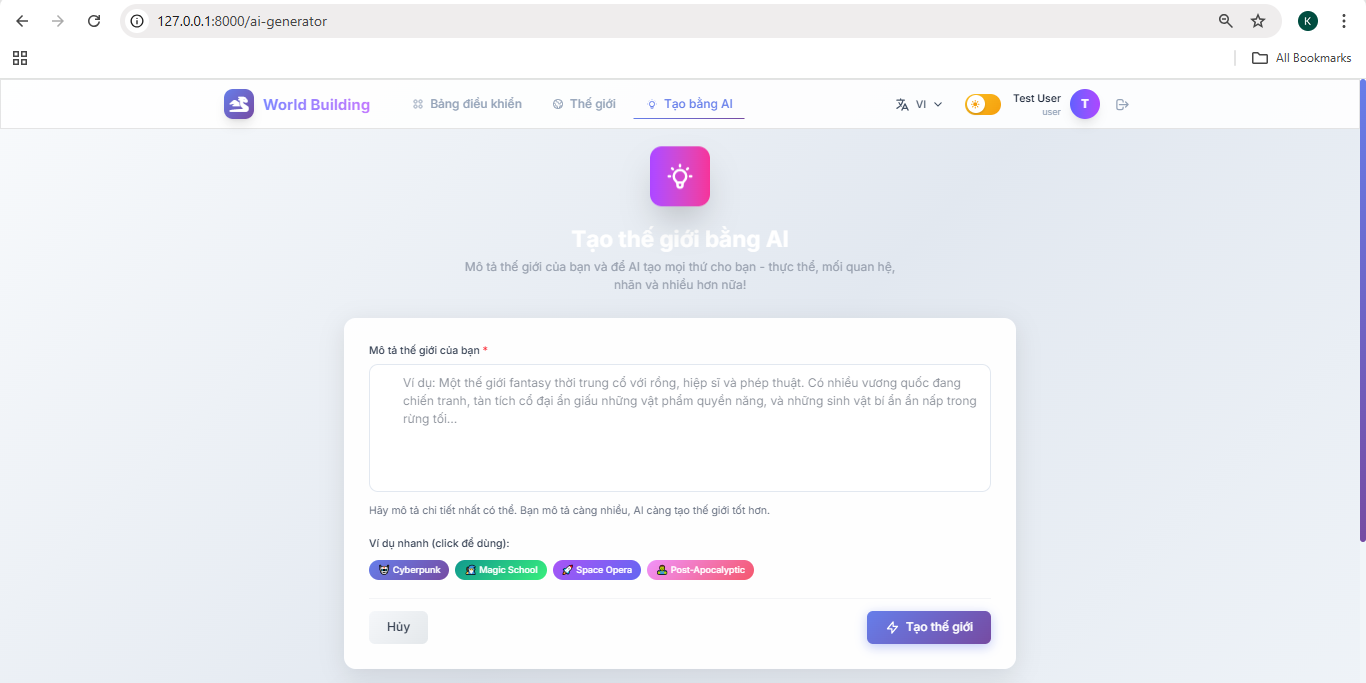
Hình 3.7 Giao diện danh sách thực thể

### Giao diện chi tiết thực thể



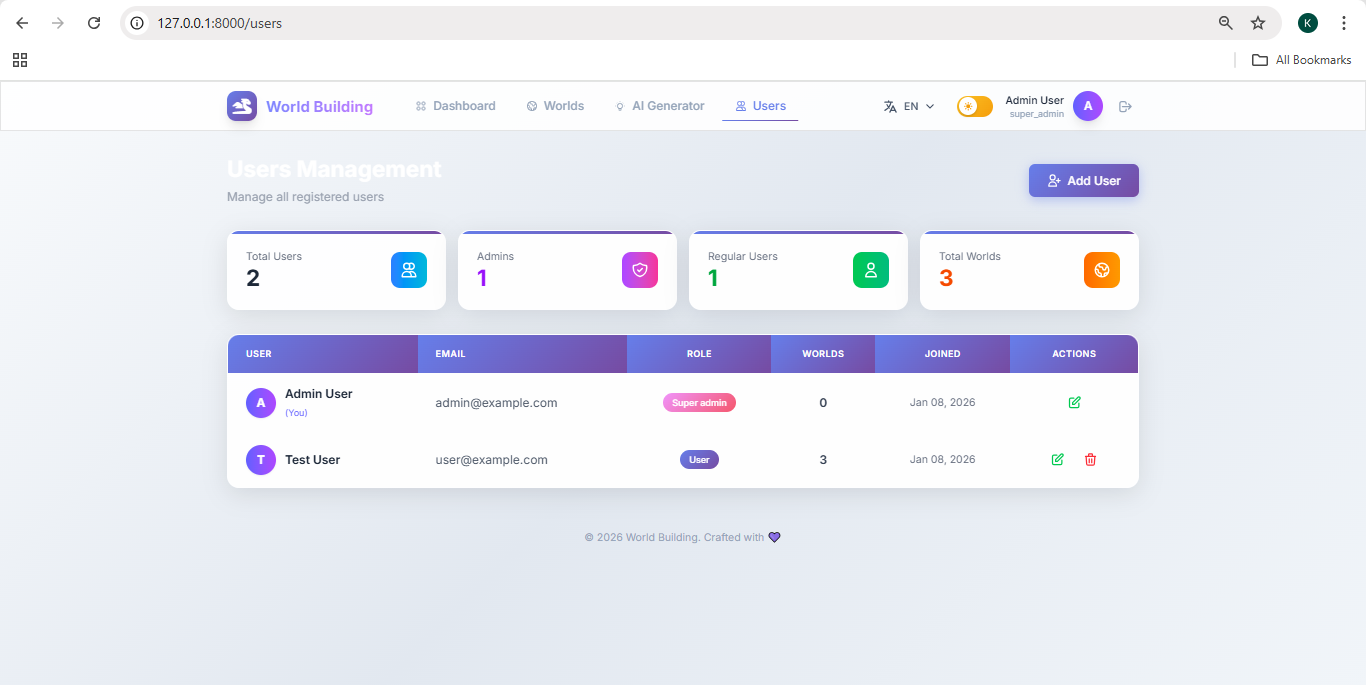
Hình 3.8 Giao diện chi tiết thực thể

### Giao diện tạo thế giới bằng AI



Hình 3.9 Giao diện tạo thế giới bằng AI

### Giao diện quản lý người dung



Hình 3.10 Giao diện quản lý người dùng

## Các chức năng của website

# KẾT LUÂN VÀ HƯỚNG PHÁT TRIỂN

## Kết quả đạt được

Sau quá trình nghiên cứu, phân tích yêu cầu và triển khai thực nghiệm, đề tài “Xây dựng hệ thống quản lý thế giới ảo trong truyện Online”đã hoàn thành các mục tiêu đặt ra ban đầu và xây dựng thành công một hệ thống web ứng dụng tương đối hoàn chỉnh. Hệ thống không chỉ đáp ứng được các yêu cầu nghiệp vụ cốt lõi của bài toán mà còn đảm bảo tính ổn định, dễ sử dụng và có khả năng mở rộng trong tương lai.

Về mặt kiến trúc hệ thống, nhóm đã xây dựng phần mềm dựa trên nền tảng Laravel Framework kết hợp với hệ quản trị cơ sở dữ liệu MySQL. Hệ thống được thiết kế và triển khai theo mô hình kiến trúc MVC (Model – View – Controller), giúp phân tách rõ ràng giữa các thành phần xử lý dữ liệu, giao diện và điều khiển luồng xử lý. Cách tổ chức này không chỉ giúp mã nguồn rõ ràng, dễ bảo trì mà còn tạo điều kiện thuận lợi cho việc mở rộng và nâng cấp hệ thống trong các giai đoạn tiếp theo.

Về giao diện người dùng, hệ thống sử dụng các công nghệ web hiện đại như HTML5, CSS3 và đặc biệt là Tailwind CSS để xây dựng giao diện. Kết quả đạt được là một giao diện trực quan, hiện đại và thân thiện với người dùng. Các chức năng được bố trí hợp lý, dễ tiếp cận, giúp tác giả nhanh chóng làm quen và thao tác với hệ thống. Ngoài ra, giao diện còn đáp ứng tốt yêu cầu Responsive Design, cho phép hệ thống hoạt động ổn định trên nhiều loại thiết bị khác nhau như máy tính để bàn, máy tính bảng và điện thoại di động.

Về mặt chức năng nghiệp vụ, hệ thống đã đáp ứng đầy đủ các yêu cầu cơ bản của bài toán quản lý thế giới truyện online. Người dùng có thể khởi tạo và quản lý nhiều bộ truyện hoặc thế giới khác nhau trong cùng một tài khoản. Hệ thống cho phép xây dựng và quản lý chi tiết các thành phần của thế giới ảo như nhân vật, địa danh, vật phẩm, luật lệ, sự kiện và các yếu tố liên quan khác. Các thực thể này được liên kết với nhau một cách có hệ thống, giúp tác giả dễ dàng theo dõi mối quan hệ và lịch sử phát triển của thế giới truyện.

Bên cạnh đó, hệ thống còn hỗ trợ các chức năng tương tác cơ bản như hiển thị thông tin cho người đọc, tìm kiếm nội dung và tham gia bình luận, thảo luận. Phân hệ quản trị (Admin Panel) được xây dựng nhằm hỗ trợ quản lý người dùng, kiểm duyệt nội dung và theo dõi hoạt động chung của hệ thống. Nhìn chung, hệ thống đã hoàn thành tốt vai trò là một công cụ hỗ trợ tác giả trong việc quản lý và kiểm soát toàn bộ cấu trúc thế giới truyện một cách khoa học và hiệu quả.

## Ưu điểm và hạn chế

### Ưu điểm

Ưu điểm nổi bật nhất của hệ thống là trải nghiệm người dùng (UX/UI). Nhờ áp dụng Tailwind CSS, giao diện có tính thẩm mỹ cao, hiện đại và dễ tùy chỉnh, phù hợp với đối tượng người dùng là các tác giả truyện online. Việc thiết kế giao diện theo hướng đơn giản, trực quan giúp giảm bớt sự phức tạp khi quản lý dữ liệu lớn.

Bên cạnh đó, hệ thống thể hiện tốt khả năng quản lý dữ liệu phức tạp, đặc biệt là cãc mối quan hệ giữa nhân vật, sự kiện và các yếu tố trong thế giới truyện. Việc tổ chức dữ liệu tập trung giúp tác giả dễ dàng tra cứu, chỉnh sửa và kiểm soát tính nhất quán của nội dung.

Ngoài ra, nhờ sử dụng nền tảng Laravel, hệ thống có khả năng mở rộng tốt, cho phép phát triển thêm các chức năng mới mà không ảnh hưởng đến cấu trúc hiện tại. Các cơ chế bảo mật cơ bản như xác thực người dùng, phân quyền truy cập và chống tấn công CSRF, XSS cũng đã được áp dụng, góp phần đảm bảo an toàn cho hệ thống.

### Hạn chế

Bên cạnh những ưu điểm đạt được, hệ thống vẫn còn một số hạn chế nhất định. Trước hết, các tính năng mạng xã hội chưa được phát triển đầy đủ, hệ thống hiện chỉ hỗ trợ hình thức đăng nhập truyền thống, chưa tích hợp đăng nhập thông qua các nền tảng phổ biến như Google hoặc Facebook.

Ngoài ra, hệ thống chưa khai thác sâu sức mạnh của trí tuệ nhân tạo. Việc thiếu các chức năng gợi ý cốt truyện, phân tích mối quan hệ hay hỗ trợ sáng tác thông minh khiến tiềm năng của hệ thống chưa được phát huy tối đa. Hiệu năng xử lý trong các tình huống có lượng truy cập lớn đồng thời cũng chưa được kiểm chứng và tối ưu sâu.

Bên cạnh đó, việc trực quan hóa dữ liệu vẫn còn đơn giản, các mối quan hệ giữa nhân vật và sự kiện chủ yếu được hiển thị dưới dạng danh sách, chưa có sơ đồ hay biểu đồ trực quan. Một số cơ chế bảo mật nâng cao như xác thực hai lớp (2FA) cũng chưa được tích hợp.

## Hướng phát triển

Để hoàn thiện và nâng cao giá trị ứng dụng của hệ thống trong tương lai, nhóm đề xuất một số hướng phát triển như sau:

* Tích hợp trí tuệ nhân tạo nâng cao hơn (sử dụng Gemini hoặc OpenAI) để hỗ trợ gợi ý ý tưởng cốt truyện, phân tích tính logic của thế giới truyện, tự động tạo hoặc tóm tắt mô tả nhân vật/bối cảnh.
* Phát triển các công cụ trực quan hóa dữ liệu phức tạp như sơ đồ mối quan hệ giữa các thực thể, dòng thời gian sự kiện, bản đồ thế giới giả tưởng.
* Tăng cường bảo mật và tiện ích người dùng: hỗ trợ đăng nhập qua mạng xã hội (Google, Facebook), tích hợp xác thực hai yếu tố (2FA).
* Xây dựng tính năng cộng đồng: cho phép tác giả chia sẻ thế giới truyện, thảo luận, hợp tác sáng tác với nhau.
* Ngoài ra, hệ thống có thể được mở rộng thành ứng dụng di động (mobile app) và tối ưu hiệu năng để xử lý lượng dữ liệu lớn, mang lại trải nghiệm linh hoạt hơn, đáp ứng tốt nhu cầu sáng tác lâu dài của cộng đồng tác giả truyện online.

**TÀI LIỆU THAM KHẢO**