**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRƯƠNG TRỌNG NGHĨA**

**PHÁT TRIỂN ỨNG DỤNG ĐỌC TRUYỆN TRỰC TUYẾN SỬ DỤNG SPRING BOOT VÀ ANGULAR**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**TP. HỒ CHÍ MINH, 2024BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRƯƠNG TRỌNG NGHĨA**

**PHÁT TRIỂN ỨNG DỤNG ĐỌC TRUYỆN TRỰC TUYẾN SỬ DỤNG SPRING BOOT VÀ ANGULAR**

**Mã số sinh viên: 2051052089**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Giảng viên hướng dẫn: DƯƠNG HỮU THÀNH**

**TP. HỒ CHÍ MINH, 2024**

LỜI CẢM ƠN

Chân thành cảm ơn sự hỗ trợ và sự tận tâm của Trường Đại học Mở thành phố Hồ Chí Minh, đặc biệt là thầy Dương Hữu Thành, giảng viên đã dành thời gian và công sức quý báu để hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đồ án ngành.

Hy vọng rằng trong tương lai, em sẽ tiếp tục nhận được sự hướng dẫn và khuyến khích từ thầy Thành, để có thể phát triển và thành công hơn trong sự nghiệp của mình. Chúc thầy luôn dồi dào sức khỏe và thành công trong mọi công việc.

Trân trọng.

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

TÓM TẮT ĐỒ ÁN NGÀNH

Mục tiêu hướng tới của đồ án là xây dựng một website đọc sách, giúp việc đọc trở nên thuận tiện và phù hợp hơn với môi trường 4.0. Từ đó tiếp tục nâng cao phát triển trải nghiệm của người dùng và nâng cao chất lượng các đầu sách.

Đồ án sẽ hỗ trợ phương tiện tiếp cận với các tác phẩm và các tác giả. Kết quả đạt được sẽ là thông tin về tính hiệu quả và tiện ích cho đọc giả, từ đó phát triển và khuyến khích văn hóa đọc của mọi người.

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc166626801)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 2](#_Toc166626802)

[TÓM TẮT ĐỒ ÁN NGÀNH 3](#_Toc166626803)

[DANH MỤC TỪ VIẾT TẮT 7](#_Toc166626804)

[DANH MỤC HÌNH VẼ 8](#_Toc166626805)

[DANH MỤC BẢNG 10](#_Toc166626806)

[MỞ ĐẦU 11](#_Toc166626807)

[Chương 1. GIỚI THIỆU ĐỀ TÀI 12](#_Toc166626808)

[1.1. Giới thiệu 12](#_Toc166626809)

[1.2. Mục tiêu đề tài 12](#_Toc166626810)

[1.2.1. Mục tiêu 12](#_Toc166626811)

[1.2.2. Đối tượng nghiên cứu 12](#_Toc166626812)

[1.3. Lý do chọn đề tài 13](#_Toc166626813)

[1.4. Bố cục báo cáo 13](#_Toc166626814)

[Chương 2. CƠ SỞ LÝ THUYẾT 14](#_Toc166626815)

[2.1. Angular 14](#_Toc166626816)

[2.1.1. Tổng quan Angular 14](#_Toc166626817)

[2.1.2. Kiến trúc Angular 14](#_Toc166626818)

[2.1.3. Các thành phần quan trọng Angular 15](#_Toc166626819)

[2.1.4. Lý do sử dụng Angular 17](#_Toc166626820)

[2.2. TypeScript 18](#_Toc166626821)

[2.2.1. Tổng quan TypeScript 18](#_Toc166626822)

[2.2.2. Một số thành phần quan trọng trong TypeScript 18](#_Toc166626823)

[2.2.3. Lý do sử dụng TypeScript 20](#_Toc166626824)

[2.3. RESTful API 20](#_Toc166626825)

[2.3.1. Giới thiệu về REST 20](#_Toc166626826)

[2.3.2. Giới thiệu về RESTful API 20](#_Toc166626827)

[2.3.3. Cách hoạt động của RESTful API 20](#_Toc166626828)

[2.3.4. Các phương thức HTTP phổ biến 21](#_Toc166626829)

[Chương 3. PHÁT TRIỂN ỨNG DỤNG 22](#_Toc166626830)

[3.1. Giới thiệu 22](#_Toc166626831)

[3.2. Kiến trúc hệ thống 22](#_Toc166626832)

[3.3. Phân tích và thiết kế 23](#_Toc166626833)

[3.3.1. Lược đồ Use case 23](#_Toc166626834)

[3.3.2. Thiết kế giao diện 28](#_Toc166626835)

[3.3.3. Thiết kế xử lý 38](#_Toc166626836)

[3.4. Các nghiệp vụ đã phát triển 44](#_Toc166626837)

[3.4.1. Đăng ký tài khoản 44](#_Toc166626838)

[3.4.2. Đăng nhập 45](#_Toc166626839)

[3.4.3. Tìm kiếm truyện 46](#_Toc166626840)

[3.4.4. Đọc truyện 47](#_Toc166626841)

[3.4.5. Tạo truyện mới 48](#_Toc166626842)

[3.4.6. Bình luận 49](#_Toc166626843)

[3.4.7. Thêm danh mục cho truyện 49](#_Toc166626844)

[3.4.8. Viết nội dung cho truyện 50](#_Toc166626845)

[3.4.9. Quản trị người dùng 50](#_Toc166626846)

[3.4.10. Quản trị danh mục 51](#_Toc166626847)

[3.4.11. Quản trị duyệt truyện 51](#_Toc166626848)

[Chương 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 52](#_Toc166626849)

[4.1. Kết luận 52](#_Toc166626850)

[4.1.1. Kết quả đạt được 52](#_Toc166626851)

[4.1.2. Hạn chế 52](#_Toc166626852)

[4.2. Hướng phát triển 52](#_Toc166626853)

[TÀI LIỆU THAM KHẢO 53](#_Toc166626854)

[PHỤ LỤC 54](#_Toc166626855)

DANH MỤC TỪ VIẾT TẮT

DANH MỤC HÌNH VẼ

[Hình 1: Kiến trúc của một ứng dụng Angular 14](#_Toc166626018)

[Hình 2: File component.ts định nghĩa file html và file css cho component 15](#_Toc166626019)

[Hình 3: File AppModule.ts 16](#_Toc166626020)

[Hình 4: File route.ts điều hướng đường dẫn tới các component 16](#_Toc166626021)

[Hình 5: Mối quan hệ giữa TypeScript và JavaScript 18](#_Toc166626022)

[Hình 6: Khai báo biến age có kiểu dữ liệu number 19](#_Toc166626023)

[Hình 7: Interface Student kèm thuộc tính có kiểu dữ liệu 19](#_Toc166626024)

[Hình 8: Sơ đồ cách hoạt động của RESTful API 20](#_Toc166626025)

[Hình 9: Mô tả kiến trúc hệ thống 22](#_Toc166626026)

[Hình 10: Lược đồ Use case tổng quát 23](#_Toc166626027)

[Hình 11: Giao diện trang đăng nhập 28](#_Toc166626028)

[Hình 12: Giao diện trang đăng ký 29](#_Toc166626029)

[Hình 13: Giao diện trang chủ 30](#_Toc166626030)

[Hình 14: Giao diện cho người dùng tạo truyện 31](#_Toc166626031)

[Hình 15: Giao diện trang cá nhân 32](#_Toc166626032)

[Hình 16: Giao diện tìm kiếm 33](#_Toc166626033)

[Hình 17: Giao diện trang đọc truyện 34](#_Toc166626034)

[Hình 18: Giao diện trang quản trị người dùng 35](#_Toc166626035)

[Hình 19: Giao diện trang quản trị danh mục 36](#_Toc166626036)

[Hình 20: Giao diện trang Quản trị truyện 37](#_Toc166626037)

[Hình 21: Giao diện đăng ký tài khoản 44](#_Toc166626038)

[Hình 22: Giao diện đăng nhập 45](#_Toc166626039)

[Hình 23: Tìm kiếm truyện theo từ khóa 46](#_Toc166626040)

[Hình 24: Tìm kiếm truyện theo danh mục 46](#_Toc166626041)

[Hình 25: Giao diện đọc truyện 47](#_Toc166626042)

[Hình 26: Giao diện tạo truyện mới 48](#_Toc166626043)

[Hình 27: Giao diện chức năng đăng bài 49](#_Toc166626044)

[Hình 28: Giao diện thêm danh mục cho truyện 49](#_Toc166626045)

[Hình 29: Giao diện viết nội dung cho truyện 50](#_Toc166626046)

[Hình 30: Giao diện danh sách người dùng. 50](#_Toc166626047)

[Hình 31: Giao diện danh sách danh mục 51](#_Toc166626048)

[Hình 32: Giao diện danh sách truyện 51](#_Toc166626049)

DANH MỤC BẢNG

[Bảng 1: Use case đăng kí người dùng 24](#_Toc166626784)

[Bảng 2: Use case đăng nhập 25](#_Toc166626785)

[Bảng 3: Use case đăng xuất 25](#_Toc166626786)

[Bảng 4: Use case đọc truyện 26](#_Toc166626787)

[Bảng 5: Use case viết truyện 26](#_Toc166626788)

[Bảng 6: Use case bình luận 27](#_Toc166626789)

[Bảng 7: Use case tìm truyện 28](#_Toc166626790)

[Bảng 8: Thiết kế xử lý trang Đăng nhập 38](#_Toc166626791)

[Bảng 9: Thiết kế xử lý trang Đăng ký 38](#_Toc166626792)

[Bảng 10: Thiết kế xử lý trang Trang chủ 38](#_Toc166626793)

[Bảng 11:: Thiết kế xử lý tìm kiếm 39](#_Toc166626794)

[Bảng 12: Thiết kế xử lý trang trang cá nhân 39](#_Toc166626795)

[Bảng 13: Thiết kế xử lý trang đăng truyện 40](#_Toc166626796)

[Bảng 14: Thiết kế xử lý trang đọc truyện 41](#_Toc166626797)

[Bảng 15: Thiết kế xử lý trang Quản trị người dùng 42](#_Toc166626798)

[Bảng 16: Thiết kế xử lý trang quản trị danh mục 43](#_Toc166626799)

[Bảng 17: Thiết kế xử lý trang Quản trị duyệt chủ trọ 43](#_Toc166626800)

MỞ ĐẦU

Trong thời đại số hóa hiện nay, việc đọc sách trực tuyến trở nên vô cùng phổ biến song song cùng với sách giấy truyền thống. Do đo, việc xây dựng và phát triển một website đọc sách là một phương tiện để mọi người có thể truy cập mà không bị giới hạn bởi thời gian và không gian. Bên cạnh đó, website còn có khả năng tạo ra một cộng đồng bằng cách kết nối các độc giả với nhau.

Với tiêu chí là cung cấp một nơi để giải trí và truy cập tri thức, hy vọng website có thể góp phần vào việc tạo ra cộng đồng văn hóa đọc ngày càng phong phú và đa dạng hơn.

# GIỚI THIỆU ĐỀ TÀI

## Giới thiệu

Đề tài PHÁT TRIỂN ỨNG DỤNG ĐỌC TRUYỆN TRỰC TUYẾN SỬ DỤNG SPRING BOOT VÀ ANGULAR được lựa chọn để xây dựng, phát triển thành một hệ thống đọc sách trực truyến. Dự án có thể giúp cho người dùng: đọc truyện, đăng truyện, tìm kiếm truyện, …

## Mục tiêu đề tài

### Mục tiêu

Mục tiêu chính là xây dựng và phát triển website đọc sách trực tuyến. Hệ thống ưu tiên vào trải nghiệm của người dùng và hiệu năng truy cập. Bao gồm các chức năng đọc sách, tìm kiếm theo từ khóa, tác giả và các yếu tố khác.

Mục tiêu xây dựng các chức năng bao gồm:

* Đăng ký tài khoản với vai trò là người dùng.
* Đăng truyện chính người dùng tự viết.
* Bình luận, trao đổi về truyện.
* Tìm kiếm linh hoạt truyện theo từ khóa, tác giả, …
* Lưu trữ tiến trình đọc.
* Đăng nhập bằng Google.
* Tích hợp VNPay.
* …

### Đối tượng nghiên cứu

Dự án có hai đối tượng:

* Người dùng: Bao gồm mọi người mọi lứa tuổi có nhu cầu tìm kiếm một cộng đồng để phát triển văn hóa đọc.
* Người quản trị: Người quản trị là một phần quan trọng trong hệ thống, có vai trò giữ cho hệ thống hoạt động đúng với các tiêu chí của website.

## Lý do chọn đề tài

Đề tài được chọn là do văn hóa đọc rất quan trọng nhưng trong thời đại hiện nay, thói quen truy cập mạng và mạng xã hội trở nên phổ biến, điều đó khiến giới trẻ dường như ít tìm tới sách báo hơn. Do đó thói quen đọc cũng dần bị suy thoái đi.

Theo một khảo sát gần đây, Việt Nam chỉ có 30% số người đọc sách thường xuyên, 26% không đọc sách và 44% thỉnh thoảng đọc sách [1]. Nếu có thể sử dụng công nghệ hiện đại để tạo một môi trường hứng thú cho người đọc thì có thể góp phần xây dựng phong trào đọc lành mạnh cho mọi lứa tuổi. Để làm được điều đó thì việc sáng tạo, đổi mới nội dung liên tục cần được chú trọng để thu hút đọc giả. Như vậy thói quen đọc của người Việt sẽ ngày càng được phát triển và hoàn thiện hơn.

## Bố cục báo cáo

Đề tài sẽ được chia thành các phần để trình bày chi tiết về quá trình xây dựng và phát triển hệ thống. Bao gồm:

* Chương 1: Giới thiệu đề tài
* Chương 2: Cơ sở lý thuyết
* Chương 3: Phát triển Ứng dụng
* Chương 4: Kết luận và hướng phát triển

# CƠ SỞ LÝ THUYẾT

## Angular

### Tổng quan Angular

Angular là một framework JavaScript mã nguồn mở, được phát triển và duy trì bởi Google. Angular dùng để xây dựng và phát triển ứng dụng web SPA (Single Page Application) và các ứng dụng web động.

### Kiến trúc Angular

Angular sử dụng kiến trúc component-based: Kiến trúc được phát triển dưới dạng từng phần (component) với khả năng tái sử dụng. Các thành phần này có trách nhiệm xác định giao diện người dùng, xử lý tương tác và xử lý dữ liệu cho các phần cụ thể của ứng dụng.



Hình 1: Kiến trúc của một ứng dụng Angular

Các thành phần này có trách nhiệm xác định giao diện người dùng, xử lý tương tác và xử lý dữ liệu cho các phần cụ thể của ứng dụng [2].

### Các thành phần quan trọng Angular

#### Component

Component là thành phần xây dựng giao diện người dùng và quản lý logic trong Angular. Chúng có trách nhiệm tạo cấu trúc giao diện người dùng, quản lý dữ liệu và logic ứng dụng.

Một component sẽ gồm các file chính sau:

* Component.ts: Quản lý dữ liệu và logic ứng dụng.
* Component.html: Quản lý giao diện người dùng.
* Component.css: Quản lý định dạng giao diện người dùng.

A screen shot of a computer

Description automatically generated

Hình 2: File component.ts định nghĩa file html và file css cho component

#### Service

Là các đối tượng có thể được chia sẻ trong toàn ứng dụng, nó dùng để thực hiện các chức năng không phụ thuộc vào giao diện người dùng (gọi API, xử lý logic, …).

#### Module

Là cách tổ chức ứng dụng trong Angular, nó là các thành phần độc lập. Mỗi ứng dụng Angular có ít nhất một module gốc được gọi là AppModule và có thể có nhiều module con.

A screen shot of a computer program

Description automatically generated

Hình 3: File AppModule.ts

#### Router

Router là phần quan trọng trong việc điều hướng trong ứng dụng Angular. Router cho phép xác định các đường dẫn URL và ánh xạ chúng vào các thành phần để hiển thị

A screen shot of a computer

Description automatically generated

Hình 4: File route.ts điều hướng đường dẫn tới các component

#### Directive

Có 2 loại Directive là built-in Directive và custom Directive do người dùng tạo ra. Chức năng của chúng là thêm vào các phần tử html để chỉ thị các thành phần hoạt động hoặc xuất hiện trong ứng dụng.

### Lý do sử dụng Angular

Angular đặc biệt chú trọng vào việc phát triển ứng dụng web phức tạp. Nó cung cấp một cơ sở vững chắc để xây dựng các ứng dụng lớn và phức tạp. Angular cho phép tích hợp linh hoạt với các thư viện và framework khác, giúp tạo ra các ứng dụng linh hoạt theo ý muốn. Điều này giúp cho việc phát triển và quản lý ứng dụng trở nên hiệu quả hơn, đặc biệt là trong các dự án lớn.

## TypeScript

### Tổng quan TypeScript

TypeScript là ngôn ngữ lập trình mã nguồn mở, được phát triển bởi Microsoft dựa trên JavaScript. Nổi bật với việc mở rộng JavaScript bằng cách cung cấp tính năng kiểu dữ liệu tĩnh tại thời điểm biên dịch. Thiết kế của TypeScript nhằm mục đích hỗ trợ xây dựng các ứng dụng có quy mô lớn và phức tạp một cách hiệu quả.

TypeScript thêm các cú pháp bổ sung vào JavaScript để hỗ trợ kiểm soát chặt chẽ trong quá trình viết mã giúp phát hiện lỗi sớm ở trong trình soạn thảo [3].

A diagram of a type of script

Description automatically generated

Hình 5: Mối quan hệ giữa TypeScript và JavaScript

### Một số thành phần quan trọng trong TypeScript

#### Kiểu dữ liệu

TypeScript mang lại khả năng định nghĩa kiểu dữ liệu cho biến, tham số của hàm và kiểu dữ liệu trả về, giúp phát hiện lỗi sớm hơn trong quá trình viết mã và cải thiện khả năng đọc mã.

A screen shot of a number

Description automatically generated

Hình 6: Khai báo biến age có kiểu dữ liệu number

#### Interface

Interface trong TypeScript được sử dụng để mô tả cấu trúc của một đối tượng, chỉ định tên và kiểu dữ liệu của các thuộc tính nhằm đảm bảo tính đồng nhất của dữ liệu.

A screen shot of a computer code

Description automatically generated

Hình 7: Interface Student kèm thuộc tính có kiểu dữ liệu

#### Kiểm tra tại thời điểm biên dịch

TypeScript kiểm tra cú pháp tại thời điểm biên dịch nhằm mục đích đảm bảo tính đúng đắn và an toàn của mã nguồn. Quá trình này giúp phát hiện lỗi trước khi chương trình chạy giúp tăng tính ổn định và độ tin cậy của ứng dụng.

### Lý do sử dụng TypeScript

TypeScript có hệ thống kiểm tra cú pháp tại thời điểm biên dịch. Quá trình này giúp phát hiện lỗi trước khi chương trình chạy giúp tăng tính ổn định và độ tin cậy của ứng dụng.

TypeScript cũng hỗ trợ ECMAScript 6 và các phiên bản JavaScript mới nhất nhằm tận dụng và phát triển dựa trên các tính năng mới nhất của ngôn ngữ.

## RESTful API

### Giới thiệu về REST

REST (Representational State Transfer) là kiến trúc cung cấp tiêu chuẩn cho việc giao tiếp giữa các hệ thống máy tính trên môi trường mạng. Nó giúp các hệ thống có khả năng giao tiếp với nhau dễ dàng và hiệu quả [5].

### Giới thiệu về RESTful API

RESTful API là một loại API tuân thủ các nguyên tắc và ràng buộc của kiến trúc REST. Nó sử dụng các phương thức HTTP như GET, POST, PUT, DELETE để thực hiện các hoạt động với tài nguyên và sử dụng các URI một cách logic nhằm đại diện cho các tài nguyên.

### Cách hoạt động của RESTful API

Bằng cách xử lý các yêu cầu (request) REST trả về các thông tin về tài nguyên.

A diagram of a software application

Description automatically generatedHình 8: Sơ đồ cách hoạt động của RESTful API

Mỗi tài nguyên đều phải được gán một URI (Uniform Resource Identifier) duy nhất.

Dữ liệu được trao đổi thông qua client và server bởi các yêu cầu (request) HTTP và phản hồi (response) tương ứng. Dữ liệu được truyền đi thường dưới dạng JSON (JavaScript Object Notation) hoặc XML (Extensible Markup Language).

### Các phương thức HTTP phổ biến

#### GET

Được sử dụng để lấy thông tin từ server.

#### POST

Được dùng để tạo một tài nguyên. Dữ liệu được gửi từ client đến server và tạo tài nguyên trên server.

#### PUT

Được sử dụng để cập nhật, chỉnh sửa một tài nguyên đã tồn tại ở server. Client sẽ gửi dữ liệu mới từ client để cập nhật tài nguyên đó trên server.

#### PATCH

Được sử dụng để cập nhật, chỉnh sửa một phần của tài nguyên đã tồn tại ở server.

#### DELETE

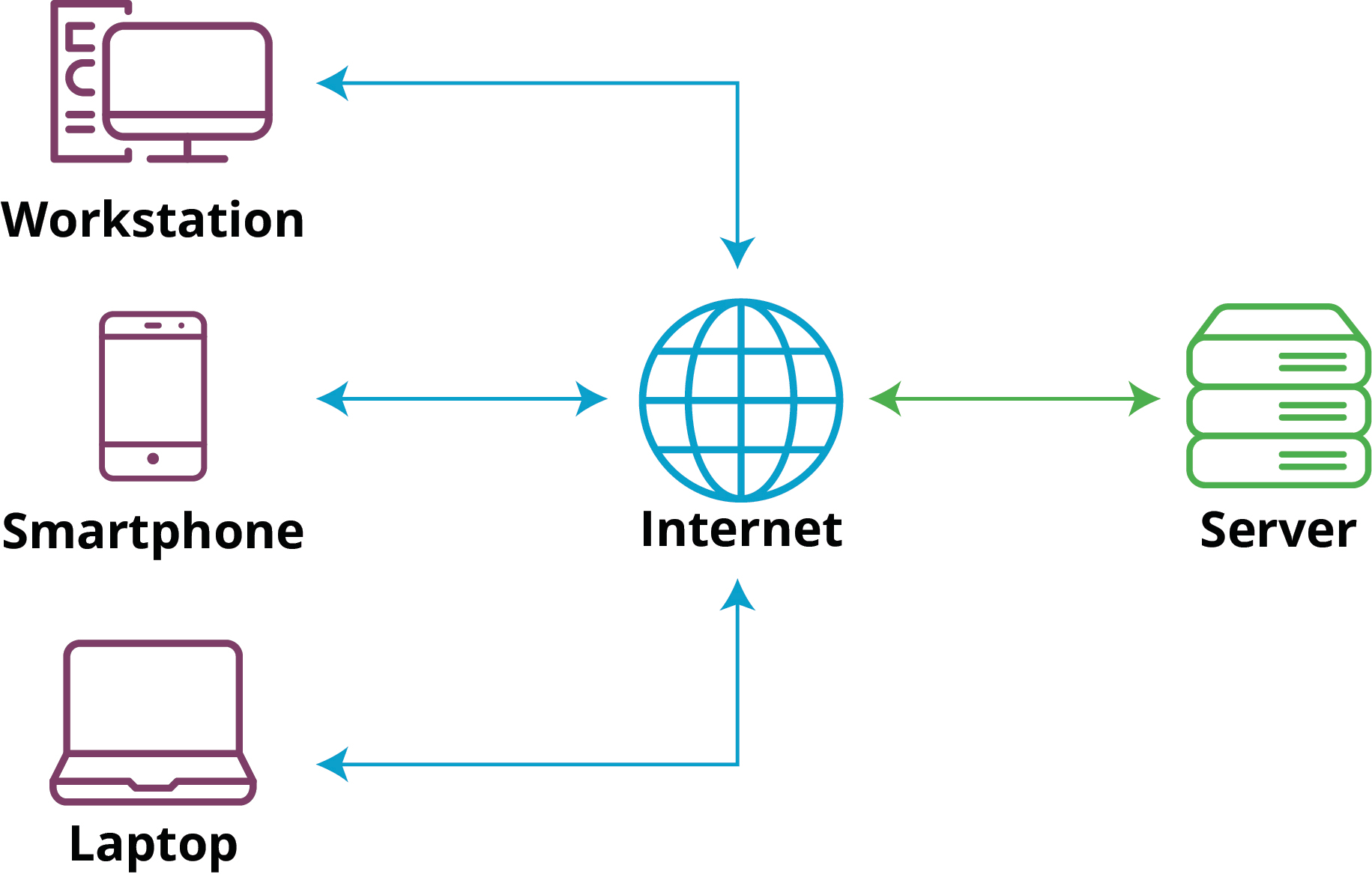
Được sử dụng để xoá một tài nguyên ở server. Phương thức này không cần dữ liệu từ client gửi kèm theo.

# PHÁT TRIỂN ỨNG DỤNG

## Giới thiệu

Trang web đọc sách cung cấp một giao diện thân thiện, dễ dùng với người sử dụng, với các tính năng hữu ích. Người dùng có thể tìm kiếm dựa trên nhiều tiêu chí như tên tác giả, thể loại, .... Trang web cung cấp thông tin chi tiết về từng cuốn sách, bao gồm tóm tắt nội dung và thông tin về tác giả. Đồng thời, người dùng cũng có thể tương tác với cộng đồng thông qua việc bình luận và chia sẻ những cuốn sách họ yêu thích.

## Kiến trúc hệ thống



Hình 9: Mô tả kiến trúc hệ thống

Client: Giao diện của ứng dụng được xây dựng và thiết kế bằng Angular. Người dùng sẽ tương tác với ứng dụng bằng các trình duyệt web như Chrome, Cốc Cốc, Edge, …. Trong Angular xử dụng HttpClient để gửi các yêu cầu (request) tới server được phản hồi (response) dữ liệu dưới dạng JSON.

Server: Spring Boot sẽ nhận các yêu cầu (request) từ phía client và xử lý, sau đó sẽ trả về phản hồi (response) cho client.

Database: MySql dùng để lưu trữ và quản lý toàn bộ tài nguyên của hệ thống.

Bên cạnh đó, để phát triển hiệu suất hệ thống cũng như cải thiện trải nghiệm người dùng, ứng dụng sử dụng Cloudinary như một giải pháp quản lý hình ảnh, điều này tạo thuận lợi cho việc dễ dàng upload hình ảnh và lưu trữ nhằm mục đích có thể lấy tài nguyên nhanh chóng và tiện lợi.

## Phân tích và thiết kế

### Lược đồ Use case

Dưới đây là lược đồ use case tổng quát của hệ thống.

A diagram of a person with text

Description automatically generated

Hình 10: Lược đồ Use case tổng quát

Đặc tả use case:

|  |  |
| --- | --- |
| Mô tả | Người dùng có thể đăng ký để sử dụng các tính năng của ứng dụng. |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng chưa có tài khoản hơặc có nhu cầu tạo tài khoản khác. |
| Hậu điền kiện | Nếu đăng ký thành công, người dùng sẽ được thông báo thành công. Nếu thất bại sẽ nhận thông báo lỗi. |
| Luồng hoạt động | 1. Truy cập vào trang đăng ký.  2. Nhập đầy đủ các thông tin.  3. Hệ thống thông báo đăng ký thành công |
| Luồng ngoại lệ | 3.a. Hệ thống thông báo thất bại.  Use case quay lại bước 2. |

Bảng 1: Use case đăng kí người dùng

|  |  |
| --- | --- |
| Mô tả | Người dùng đăng nhập bằng tài khoản và mật khẩu |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã có tài khoản hoặc đã đăng ký tài khoản thành công |
| Hậu điền kiện | Nếu đăng nhập thành công, hệ thống sẽ đưa người dùng đến trang chủ. Nếu thất bại sẽ thì sẽ nhận được thông báo đăng nhập bị lỗi. |
| Luồng hoạt động | 1. Truy cập vào trang đăng nhập.  2. Nhập đầy đủ các thông tin.  3. Hệ thống xác thực người dùng.  4. Chuyển hướng đến trang chính. |
| Luồng ngoại lệ | 3.a. Hệ thống thông báo thất bại.  Use case quay lại bước 2. |

Bảng 2: Use case đăng nhập

|  |  |
| --- | --- |
| Mô tả | Người dùng đăng xuất cách tương tác với giao diện |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã đăng nhập thàng công. |
| Hậu điền kiện | Nếu đăng xuất thành công, hệ thống sẽ đưa người dùng đến trang đăng nhập. Nếu thất bại sẽ thì sẽ nhận được thông báo đăng nhập bị lỗi. |
| Luồng hoạt động | 1. Bấm nút đăng xuất.  2. Hệ thống đưa người dùng về trang đăng xuất. |
| Luồng ngoại lệ | 1.a. Hệ thống thông báo thất bại.  Use case quay lại bước 1. |

Bảng 3: Use case đăng xuất

|  |  |
| --- | --- |
| Mô tả | Người dùng chọn 1 truyện đã được đăng để đọc. |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã đăng nhập thàng công. |
| Hậu điền kiện | Nếu người dùng vào thành công sẽ hiển thị chi tiết các thông tin về truyện  và thông tin về tác giả. Đọc giả có thể chọn chương để đọc. |
| Luồng hoạt động | 1. Đăng nhập thành công.  2. Nhấp chuột vào 1 chuyện đã được đăng.  3. Bấm vào 1 chương  bất kì trong truyện để đọc. 4. Hệ thống sẽ chuyển người dùng đến trang chứa nội dung chương được chọn. |
| Luồng ngoại lệ | 2.a. Hệ thống thông báo thất bại.  Use case quay lại bước 2.  3.a. Hệ thống thông báo thất bại.  Use case quay lại bước 3. |

Bảng 4: Use case đọc truyện

|  |  |
| --- | --- |
| Mô tả | Người dùng tự viết truyện và đăng lên website. |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã đăng nhập thàng công. |
| Hậu điền kiện | Nếu người dùng tạo thành công 1 truyện mới, hệ thống sẽ đăng truyện lên website. |
| Luồng hoạt động | 1. Đăng nhập thành công.  2. Nhấp chuột vào nút viết truyện.  3. Điền đầy đủ thông tin cần thiết. 4. Bấm nút lưu. 5. Hệ thống sẽ chuyển về trang chủ nếu tạo thành công. |
| Luồng ngoại lệ | 4.a. Hệ thống thông báo thất bại.  Use case quay lại bước 3. |

Bảng 5: Use case viết truyện

|  |  |
| --- | --- |
| Mô tả | Người dùng bình luận 1 truyện được đăng trên website. |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã đăng nhập thàng công. |
| Hậu điền kiện | Nếu người dùng bình luận thành công thì bình luận sẽ hiện ở phần bình luận của truyện. |
| Luồng hoạt động | 1. Đăng nhập thành công.  2. Chọn 1 truyện được đăng trên website.  3. Điền nội dung vào ô bình luận. 4. Bấm nút gửi. 5. Hệ thống sẽ cập nhật bình luận nếu thành công. |
| Luồng ngoại lệ | 4.a. Hệ thống thông báo thất bại.  Use case quay lại bước 3. |

Bảng 6: Use case bình luận

|  |  |
| --- | --- |
| Mô tả | Người dùng tìm kiếm truyện bằng cách nhập từ khóa hoặc chọn 1 danh mục truyện. |
| Tác nhân chính | Người dùng. |
| Tác nhân phụ | Không có. |
| Tiền điều kiện | Người dùng đã đăng nhập thàng công. |
| Hậu điền kiện | Nếu người dùng tạo tìm kiếm thành công, trang chủ sẽ lọc các truyện thích hợp với từ khóa được nhập, hoặc với danh mục được chọn, hoặc cả 2. |
| Luồng hoạt động | 1. Đăng nhập thành công.  2. Điền từ khóa vào ô search, hoặc chọn 1 danh mục truyện, hoặc cả 2.  3. Hệ thống sẽ lọc các truyện phù hợp với yêu cầu của người dùng |
| Luồng ngoại lệ | 3.a. Hệ thống thông báo thất bại.  Use case quay lại bước 2. |

Bảng 7: Use case tìm truyện

### Thiết kế giao diện

#### Đăng nhập

Giao diện trang đăng nhập.

A screenshot of a login form

Description automatically generated

Hình 11: Giao diện trang đăng nhập

#### Đăng ký

Giao diện trang đăng ký.

A screenshot of a sign up form

Description automatically generated

Hình 12: Giao diện trang đăng ký

#### Trang chủ

Giao diện trang chủ.

A screenshot of a computer screen

Description automatically generated

Hình 13: Giao diện trang chủ

#### Viết truyện

Giao diện cho người dùng tạo truyện.

A screenshot of a computer

Description automatically generated

Hình 14: Giao diện cho người dùng tạo truyện

#### Trang cá nhân

Giao diện trang cá nhân.

A screenshot of a computer

Description automatically generated

Hình 15: Giao diện trang cá nhân

#### Tìm kiếm truyện

Giao diện tìm kiếm.

A screenshot of a person's profile

Description automatically generated

Hình 16: Giao diện tìm kiếm

#### Đọc truyện

Giao diện trang đọc truyện.

A text on a white background

Description automatically generated

Hình 17: Giao diện trang đọc truyện

#### Quản trị người dùng

Giao diện trang quản trị người dùng.

A screenshot of a computer

Description automatically generated

Hình 18: Giao diện trang quản trị người dùng

#### Quản trị danh mục

Giao diện trang quản trị danh mục.

A white screen with black text

Description automatically generated

Hình 19: Giao diện trang quản trị danh mục

#### Quản trị truyện

Giao diện trang quản trị truyện.

A screenshot of a computer

Description automatically generated

Hình 20: Giao diện trang Quản trị truyện

### Thiết kế xử lý

#### Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | isLogin | Thay đổi thông tin đăng nhập của người dùng. | Xử lý thông tin đăng nhập của người dùng. |
| 2 | onSubmit | Nhấp chuột vào nút Đăng nhập | Hệ thống xác thực người dùng. |

Bảng 8: Thiết kế xử lý trang Đăng nhập

#### Đăng ký

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | validation | Nhập đầy đủ thông tin cần thiết. | Hệ thống kiểm tra thông tin được người dùng nhập. |
| 2 | onSubmit | Click vào Đăng ký. | Hệ thống thực hiện đăng ký tài khoản. |

Bảng 9: Thiết kế xử lý trang Đăng ký

#### Trang chủ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | getListFiction | Trang được tải hoặc được lọc bởi ô tìm kiếm và danh mục. | Lấy tất cả truyện phù hợp. |
| 2 | getListTag | Trang được tải. | Lấy tất cả danh mục truyện có sẵn. |

Bảng 10: Thiết kế xử lý trang Trang chủ

#### Tìm kiếm truyện

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | searchInput | Nhập từ khóa và đợi 0,5 giây | Lấy các truyện có tên phù hợp với từ khóa được nhập vào. |
| 2 | tagId | Chọn 1 danh mục có sẵn ở trang chủ | Lấy các truyện có danh mục được chọn. |

Bảng 11:: Thiết kế xử lý tìm kiếm

#### Trang cá nhân

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | userById | Nhấn vào avatar ở header sau khi đăng nhập thành công | Vào trang cá nhân. |
| 2 | fictionsByUserId | Nhấn vào avatar ở header sau khi đăng nhập thành công | Lấy tất cả các truyện đã được người dùng đó đăng. |

Bảng 12: Thiết kế xử lý trang trang cá nhân

#### Đăng truyện

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | routerLink | Nhấp vào nút tạo truyện mới | Chuyển đến trang tạo truyện mới. |
| 2 | onSubmitFiction | Nhấp vào nút Lưu sau khi đã nhập đầy đủ thông tin cần thiết | Lưu truyện và đăng lên website. |
| 3 | getAvailableTags | Vào phần Tag và chọn những danh mục phù hợp với truyện | Thêm danh mục cho truyện để dễ dàng cho việc tìm kiếm. |
| 4 | onDeleteComment | Vào phần admin, vào 1 truyện có bình luận, bấm Xóa 1 bình luận | Xóa bình luận không phù hợp với tiêu chí cộng đồng. |
| 7 | validation | Nhập vào ô nhập liệu | Hệ thống sẽ kiểm tra thông tin có đúng hợp lệ hay không. |

Bảng 13: Thiết kế xử lý trang đăng truyện

#### Đọc truyện

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | getChapterById | Nhấp vào 1 chương trong danh sách chương của truyện | Chuyển qua trang đọc truyện, lấy nội dung của chương đó hiển thị. |
| 2 | safeHtmlContent | Trang đọc truyện được tải | Xử lý các chuỗi html một cách an toàn. |

Bảng 14: Thiết kế xử lý trang đọc truyện

#### Quản trị người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | getAllUser | Trang vừa được tải. | Lấy tất cả người dùng. |
| 2 | createUser | Nhấp vào nút tạo mới. | Tạo mới người dùng. |
| 3 | updateUser | Thay đổi thông tin trên ô nhập liệu và bấm nút lưu. | Cập nhật người dùng. |
| 4 | validation | Điền thông tin vào các ô nhập liệu | Hệ thống sẽ kiểm tra thông tin có hợp lệ hay không. |
| 5 | handleDeleteUser | Nhấp vào icon xoá. | Xoá người dùng đã chọn. |

Bảng 15: Thiết kế xử lý trang Quản trị người dùng

#### Quản trị danh mục

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | createTag | Bấm vào nút tạo mới. | Tạo mới một danh mục |
| 2 | editTag | Thay đổi thông tin và bấm nút lưu | Cập nhật một danh mục |
| 3 | getAllTags | Trang được tải | Lấy tất cả các danh mục |
| 4 | validation | Nhập vào các ô nhập liệu | Hệ thống sẽ kiểm tra thông tin có hợp lệ hay không |
| 5 | deleteTag | Bấm vào icon xóa | Xóa một danh mục được chọn. |

Bảng 16: Thiết kế xử lý trang quản trị danh mục

#### Quản trị truyện

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên xử lý | Điều kiện gọi thực hiện | Ý nghĩa |
| 1 | createFiction | Bấm vào nút tạo mới. | Tạo mới một truyện |
| 2 | editFiction | Thay đổi thông tin và bấm nút lưu | Cập nhật một truyện |
| 3 | getAllTags | Trang được tải | Lấy tất cả các danh mục có trong truyện |
| 4 | validation | Nhập vào các ô nhập liệu | Hệ thống sẽ kiểm tra thông tin có hợp lệ hay không |
| 5 | deleteFiction | Bấm vào icon xóa | Xóa một truyện được chọn. |

Bảng 17: Thiết kế xử lý trang Quản trị duyệt chủ trọ

## Các nghiệp vụ đã phát triển

### Đăng ký tài khoản

Chức năng đăng ký tài khoản cho người dùng chưa có tài khoản hoăc có nhu cầu tạo mới tài khoản.

A screenshot of a login form

Description automatically generated

Hình 21: Giao diện đăng ký tài khoản

### Đăng nhập

Chức năng đăng nhập để có thể truy cập vào hệ thống.

A screenshot of a login form

Description automatically generated

Hình 22: Giao diện đăng nhập

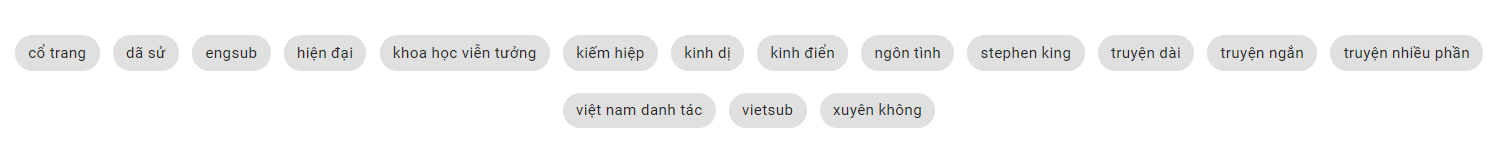
### Tìm kiếm truyện

Chức năng này giúp dọc giả lọc ra các truyện theo nhu cầu.

A white rectangular object with black edges

Description automatically generated

Hình 23: Tìm kiếm truyện theo từ khóa



Hình 24: Tìm kiếm truyện theo danh mục

### Đọc truyện

Chức năng giúp người dùng tiếp cận với nội dung của các chương của truyện.

A close up of a document

Description automatically generated

Hình 25: Giao diện đọc truyện

### Tạo truyện mới

Chức năng này giúp người dùng có thể tự do viết ra những ý tưởng của chính mình.

A screenshot of a computer

Description automatically generated

Hình 26: Giao diện tạo truyện mới

### Bình luận

Chức năng bình luận cho phép người dùng bày tỏ suy nghĩ của mình với cộng đồng về truyện.

A close-up of a computer screen

Description automatically generated

Hình 27: Giao diện chức năng đăng bài

### Thêm danh mục cho truyện

Chức năng bình luận giúp tác giả thêm danh mục cho truyện nhằm giúp đọc giả có thể tìm truyện bằng bộ lọc dễ dàng hơn.

**A screenshot of a computer

Description automatically generated**

Hình 28: Giao diện thêm danh mục cho truyện

### Viết nội dung cho truyện

Chức năng nhắn tin giúp người dùng tạo nội dung dễ dàng truyền tải đến người đọc

A screenshot of a computer

Description automatically generated

Hình 29: Giao diện viết nội dung cho truyện

### Quản trị người dùng

Chức năng quản trị người dùng cho phép quản trị viên có thể xem, thêm, sửa, xóa người dùng.

A screenshot of a computer

Description automatically generated

Hình 30: Giao diện danh sách người dùng.

### Quản trị danh mục

Chức năng quản trị danh mục cho phép quản trị viên có thể xem, thêm, sửa, xóa danh mục.

A screenshot of a notebook

Description automatically generated

Hình 31: Giao diện danh sách danh mục

### Quản trị duyệt truyện

Chức năng quản trị truyện cho phép quản trị viên có thể xem, thêm, sửa, xóa truyện.

A screenshot of a computer

Description automatically generated

Hình 32: Giao diện danh sách truyện

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

### Kết quả đạt được

Học được cách tổ chức và kiến trúc của một hệ thống website.

Xây dựng ứng dụng bằng các tính năng như tìm kiếm, bình luận, đăng truyện,…

Học được các công nghệ mới: Spring Boot, Angular, …

### Hạn chế

Ứng dụng vẫn còn rất nhiều thiếu sót và hạn chế về mặt giao diện.

Hệ thống còn một số chức năng chưa hoàn chỉnh.

Chưa tối ưu được hiệu suất ứng dụng.

## Hướng phát triển

Tích hợp FireBase để người dùng có thể nhắn tin với nhau.

Tích hợp VNPay.

TÀI LIỆU THAM KHẢO

[1] Nguyễn Thị Hải Yến, “Thực trạng văn hoá đọc hiện nay ở Việt Nam”, 2023. [Trực tuyến]. Địa chỉ: [https://iper.org.vn/thuc-trang-van-hoa-doc-hien-nay-o-viet-nam/](https://iper.org.vn/thuc-trang-van-hoa-doc-hien-nay-o-viet-nam/%20) [Truy cập 10/05/2024]

[2] Alka Vats, “Role of Components in Angular Architecture”, 2023. [Trực tuyến]. Địa chỉ: <https://blog.nashtechglobal.com/role-of-components-in-angular-architecture/#:~:text=Angular%20is%20a%20powerful%20front,specific%20parts%20of%20the%20application> [Truy cập 10/05/2024].

[3] Microsoft, “TypeScript: JavaScript With Syntax For Types.”, 2023. [Trực tuyến]. Địa chỉ: <https://www.typescriptlang.org/> [Truy cập 2/10/2023].

[4] Codecademy Team, “What is REST?”, 2023. [Trực tuyến]. Địa chỉ: <https://www.codecademy.com/article/what-is-rest> [Truy cập 2/10/2023].

PHỤ LỤC