

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI**  
**THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**XÂY DỰNG TRANG WEB VIẾT TRUYỆN CHỮ VÀ KIỂM DUYỆT NỘI  
DUNG ỨNG DỤNG THUẬT TOÁN COSINE SIMILARITY VÀ MÔ  
HÌNH THUẬT TOÁN NAIVE BAYES**

Giảng viên hướng dẫn: ThS. PHẠM THỊ MIÊN

Sinh viên thực hiện: NGUYỄN TUẤN ĐẠT

Lớp: CQ.60.CNTT

MSV: 6051071029

Khoá: 60

Tp. Hồ Chí Minh, năm 2023

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI**  
**THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**XÂY DỰNG TRANG WEB VIẾT TRUYỆN CHỮ VÀ KIỂM DUYỆT NỘI  
DUNG ỨNG DỤNG THUẬT TOÁN COSINE SIMILARITY VÀ MÔ  
HÌNH THUẬT TOÁN NAIVE BAYES**

Giảng viên hướng dẫn: ThS. PHẠM THỊ MIÊN

Sinh viên thực hiện: NGUYỄN TUẤN ĐẠT

Lớp: CQ.60.CNTT

MSV: 6051071029

Khoá: 60

Tp. Hồ Chí Minh, năm 2023

PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH  
do – Hạnh phúc

Độc lập – Tự

**NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP**  
**BỘ MÔN: CÔNG NGHỆ THÔNG TIN**

-----\*\*\*-----

**Mã sinh viên:** 6051071029

**Họ tên SV:** Nguyễn Tuấn Đạt

**Khóa:** 60

**Lớp:** CQ.60.CNTT

**1. Tên đề tài**

XÂY DỰNG TRANG WEB VIẾT TRUYỆN CHỮ VÀ KIỂM DUYỆT NỘI  
DUNG ỨNG DỤNG THUẬT TOÁN COSINE SIMILARITY VÀ MÔ HÌNH THUẬT  
TOÁN NAIVE BAYES

**2. Mục đích, yêu cầu**

**a. Mục đích:**

- Hiểu được phương pháp xây dựng một trang web hoàn chỉnh từ khâu chuẩn bị, phân tích, thiết kế cho đến thực thi.
- Xây dựng một trang web viết truyện chữ với mục đích áp dụng những kiến thức mà mình đã học cũng như đã tìm hiểu để tạo ra cho mọi người một nơi để mọi người có thể sáng tạo các câu truyện chữ với tiêu chí thư giãn, giải trí bên cạnh đó cũng là nơi tìm ra các nhà đọc giả khám phá những câu truyện thú vị mà những tác giả mang đến.

**b. Yêu cầu:**

**- Yêu cầu công nghệ**

- Sử dụng ngôn ngữ lập trình HTML, CSS, Javascript, Python.
- Công nghệ, Framework được áp dụng: ReactJS, Redux, MySQL, Django.
- Thư viện hỗ trợ thiết kế: Bootstrap.
- Thư viện hỗ trợ cho học máy: Scikit-learn
- Sử dụng công cụ Visual Studio Code.

- **Yêu cầu chức năng**

- Phần mềm cho phép người dùng:
  - + Đăng nhập, đăng ký tài khoản mới
  - + Tìm kiếm truyện theo tên, thể loại.
  - + Tạo truyện mới thêm tên, mô tả truyện.
  - + Viết truyện chữ.
  - + Quản lý các truyện mình đã tạo.
  - + Admin kiểm duyệt nội dung ngôn từ bằng thuật toán.
  - + Admin gửi phản hồi về truyện cho user qua gmail..
- Admin kiểm duyệt cho phép bên kiểm duyệt có thể:
  - + Quản lý kiểm duyệt xem thông tin truyện.
  - + Kiểm duyệt nội dung ngôn từ bằng thuật toán.
  - + Gửi phản hồi về truyện cho user qua gmail

- **Yêu cầu phi chức năng**

- Tốc độ: Nhận truy cập nhanh, truy xuất dữ liệu nhanh
- Giao diện: Thân thiện với người sử dụng.

- **Yêu cầu về tổ chức code**

- Tổ chức code ngay ngắn dễ đọc, lược bỏ các thành phần rườm rà.

### **3. Nội dung và phạm vi đề tài**

**a. Nội dung:**

- + Tổng quan về ngôn ngữ lập trình HTML, CSS, Javascript, Python.
- + Tổng quan về bài toán các công nghệ đang sử dụng.
- + Phân tích và thiết kế database.
- + Phân tích thiết kế giao diện người dùng.
- + Lập trình xây dựng trang web.
- + Kiểm thử và chạy thực nghiệm.
- + Kết quả thu được

**b. Phạm vi:**

- + Nghiên cứu và sử dụng kết hợp: Django, MySQL, ReactJS, Redux và thư viện

Bootstrap.

- + Nghiên cứu sử dụng công cụ Visual Studio Code.
- + Thư viện hỗ trợ cho học máy: Scikit-learn

#### **4. Công nghệ, công cụ và ngôn ngữ lập trình**

- + Công nghệ sử dụng: Django, MySQL, ReactJS, Redux.
- + Công cụ Visual Studio Code.
- + Ngôn ngữ lập trình: HTML, CSS, Javascript, Python.

#### **5. Các kết quả chính dự kiến sẽ đạt được và ứng dụng**

- + Quyền báo cáo đề tài tốt nghiệp.
- + Xây dựng được trang web với những chức năng đã nêu trên.
- + Sử dụng kết hợp được : Django, MySQL, ReactJS, Redux, Scikit-learn.

#### **6. Giáo viên và cán bộ hướng dẫn**

Họ tên: ThS. Phạm Thị Miên

Đơn vị công tác: Trường Đại học Giao thông Vận tải Phân hiệu tại Thành phố Hồ Chí Minh.

Điện thoại: 0961170638

Email: ptmien@utc2.edu.vn

Ngày ..... tháng ..... năm 2023  
Trưởng BM Công nghệ Thông tin

**Đã giao nhiệm vụ TKTN**  
**Giảng viên hướng dẫn**

Đã nhận nhiệm vụ TKTN

Sinh viên: Nguyễn Tuấn Đạt

Ký tên:

Điện thoại: 0965634186

Email: [6051071029@st.utc2.edu.vn](mailto:6051071029@st.utc2.edu.vn)

*Tp. Hồ Chí Minh, ngày ... tháng ... năm 2023*

## LỜI CẢM ƠN

Để hoàn thành đề tài tốt nghiệp này trước hết em xin gửi đến quý thầy, cô **Bộ môn Công nghệ thông tin – Trường Đại học Giao thông Vận tải Phân hiệu tại Thành phố Hồ Chí Minh** lời cảm ơn chân thành vì đã truyền đạt cho em những kiến thức không chỉ từ sách vở, mà còn những kinh nghiệm quý giá từ cuộc sống trong khoảng thời gian học tập tại trường. Đặc biệt em xin gửi đến cô Phạm Thị Miên lời cảm ơn sâu sắc nhất, cô đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình thực hiện đề tài. Tuy đề tài không được lớn lao nhưng nếu không được sự hướng dẫn chỉ bảo tận tình của thầy và cô thì đề tài tốt nghiệp này khó có thể hoàn thành được.

Vì thời gian làm đồ án có hạn cũng như hiểu biết cá nhân còn hạn chế, em cũng đã nỗ lực hết sức để hoàn thành bài báo cáo đồ án một cách tốt nhất, nhưng chắc chắn vẫn sẽ có những thiếu sót không thể tránh khỏi. Em kính mong nhận được sự thông cảm và những ý kiến đóng góp chân thành từ quý thầy cô.

Sau cùng, em xin kính chúc Quý Thầy Cô đang công tác tại Bộ môn Công nghệ thông tin cũng như những thầy cô đang công tác tại trường luôn mạnh khỏe, hạnh phúc và thành công hơn nữa trong công việc cũng như trong cuộc sống.

Em xin chân thành cảm ơn!

Tp. Hồ Chí Minh, ngày ..... tháng ..... năm 2023

Sinh viên thực hiện

**Nguyễn Tuấn Đạt**

## NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

*Tp. Hồ Chí Minh, ngày ..... tháng ..... năm 2023*

**Giảng viên hướng dẫn**

## MỤC LỤC

<b>NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP .....</b>	<b>i</b>
1. Tên đề tài .....	i
2. Mục đích, yêu cầu .....	i
3. Nội dung và phạm vi đề tài .....	ii
4. Công nghệ, công cụ và ngôn ngữ lập trình.....	iii
5. Các kết quả chính dự kiến sẽ đạt được và ứng dụng .....	iii
6. Giáo viên và cán bộ hướng dẫn .....	iii
<b>LỜI CẢM ƠN .....</b>	<b>iv</b>
<b>NHẬN XÉT CỦA GIÁNG VIÊN HƯỚNG DẪN .....</b>	<b>v</b>
<b>MỤC LỤC .....</b>	<b>vi</b>
<b>MỤC LỤC HÌNH ẢNH.....</b>	<b>ix</b>
<b>DANH MỤC TỪ VIẾT TẮT .....</b>	<b>xi</b>
<b>TỔNG QUAN .....</b>	<b>1</b>
1. Lý do chọn đề tài.....	1
2. Mục đích nghiên cứu .....	1
3. Phạm vi nghiên cứu .....	2
4. Phương pháp nghiên cứu .....	2
5. Cấu trúc báo cáo đồ án tốt nghiệp .....	2
<b>CHƯƠNG 1 CƠ SỞ LÝ THUYẾT .....</b>	<b>4</b>
1.1. Tổng quan về MySQL .....	4
1.2. Django.....	7
1.3. ReactJS .....	11
1.4. Thư viện Scikit-learn.....	15
1.5. Mô hình Naive Bayes.....	18
1.6. Thuật toán Cosine Similarity .....	21
<b>CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG. ....</b>	<b>25</b>



2.1. Tổng quan về một trang viết truyện chữ.....	25
2.2. Phân tích yêu cầu hệ thống.....	25
2.2.1. Khái quát chung về hệ thống.....	25
2.3. Các Use case của hệ thống.....	27
2.3.1. Biểu đồ Use case.....	27
2.3.2. Mô tả các Use case.....	30
2.4. Biểu đồ tuần tự.....	33
2.4.1. Biểu đồ tuần tự chức năng đăng nhập.....	34
2.4.2. Biểu đồ tuần tự chức năng đăng ký tài khoản mới.....	35
2.4.3. Biểu đồ tuần tự chức năng tạo truyện mới.....	36
2.4.4. Biểu đồ tuần tự chức năng tìm kiếm theo tên truyện.....	37
2.5. Biểu đồ hoạt động.....	37
2.5.1. Biểu đồ hoạt động chức năng cập nhật truyện.....	37
2.6. Cách tổ chức dữ liệu và thiết kế chương trình.....	39
2.6.1. Bảng user.....	39
2.6.2. Bảng story.....	39
2.6.3. Bảng chapterStory.....	40
2.6.4. Bảng typestory.....	40
2.6.5. Bảng comments.....	40
2.7. Mô hình database diagram.....	41
<b>CHƯƠNG 3 CÀI ĐẶT CHƯƠNG TRÌNH.....</b>	<b>42</b>
3.1. Các chức năng của chương trình.....	42
3.1.1. Giao diện đăng nhập.....	42
3.1.2. Giao diện đăng ký.....	43
3.1.3. Giao diện trang chủ.....	44
3.1.4. Giao diện viết thông tin truyện.....	45
3.1.5. Giao diện viết truyện.....	46
3.1.6. Giao diện quản lý truyện cá nhân.....	47

<b>3.1.7. Giao diện trang đọc thông tin truyện. ....</b>	<b>47</b>
<b>3.1.8. Giao diện đọc truyện. ....</b>	<b>49</b>
<b>3.1.9. Giao diện trang admin. ....</b>	<b>49</b>
<b>3.1.10. Giao diện hiển thị kiểm duyệt. ....</b>	<b>50</b>
<b>3.1.11. Gửi mail phản hồi ....</b>	<b>51</b>
<b>KẾT LUẬN VÀ KIẾN NGHỊ .....</b>	<b>51</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>53</b>

## MỤC LỤC HÌNH ẢNH

Hình 1 MySQL .....	5
Hình 2 Django .....	7
Hình 3 Scikit-learn.....	16
Hình 4 Mô hình ERD .....	26
Hình 5 Use case tổng quát .....	27
Hình 6 Use Case quản lý truyện .....	28
Hình 7 Use Case quản lý chương truyện.....	28
Hình 8 Use Case đăng nhập.....	29
Hình 9 Use Case lọc truyện theo thể loại .....	29
Hình 10 Use Case tìm kiếm truyện theo tên truyện .....	30
Hình 11 Use Case kiểm duyệt truyện .....	30
Hình 12 Biểu đồ tuần tự chức năng đăng nhập .....	34
Hình 13 Biểu đồ tuần tự chức năng đăng ký tài khoản mới .....	35
Hình 14 Biểu đồ tuần tự chức năng tạo truyện.....	36
Hình 15 Biểu đồ tuần tự chức năng tìm kiếm theo tên truyện .....	37
Hình 16 Hoạt động chức năng cập nhật truyện .....	38
Hình 17 Mô hình database diagram.....	41
Hình 18 Giao diện đăng nhập .....	42
Hình 19 Giao diện đăng ký.....	43
Hình 20 Giao diện trang chủ .....	44
Hình 21 Giao diện viết thông tin truyện .....	45
Hình 22 Giao diện viết chương truyện .....	46
Hình 23 Giao diện quản lý truyện cá nhân.....	47
Hình 24 Giao diện xem thông tin truyện.....	48
Hình 25 Giao diện đọc truyện .....	49
Hình 26 Giao diện trang đăng nhập admin.....	49
Hình 27 Giao diện trang kiểm duyệt của admin.....	50

Hình 28 Giao diện kiểm duyệt thông tin truyện .....	51
Hình 29 Phản hồi mail cho tác giả.....	51

## DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Thuật ngữ	Ý nghĩa
1	TCP	Transmission Control Protocol	là một giao thức mạng quan trọng được sử dụng trong việc truyền dữ liệu qua một mạng nào đó
2	HTTP	Hyper Text Transfer Protocol	Giao thức Truyền tải Siêu Văn Bản được sử dụng trong www
3	CSDL		Cơ sở dữ liệu
4	SQL	Structured Query Language	ngôn ngữ truy vấn có cấu trúc
5	RDBMS	Relational Database Management System	hệ quản trị cơ sở dữ liệu quan hệ
6	NPM	Node Package Manager	một công cụ tạo và quản lý các thư viện lập trình Javascript cho Node.js
7	IoT	Internet of Things	Internet vạn vật
8	SPA	Single-page application	đem đến người dùng sự trải nghiệm web tuyệt vời nhất không khác gì mobile
9	API	Application Programming Interface	là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác
10	MVC	Model-View-Controller	là một mẫu kiến trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller
11	URL	Uniform Resource Locator	Đây là một đường dẫn liên kết đến website, tham chiếu tới các tài nguyên trên mạng Internet
12	DOM	Document Object Model	được dùng để truy xuất và thao tác trên các tài liệu có cấu trúc dạng HTML hay XML
13	XML	Extensible Markup Language	là ngôn ngữ đánh dấu mở rộng

# TỔNG QUAN

## 1. Lý do chọn đề tài

Nhu cầu ngày càng tăng về việc viết và đọc truyện chữ trực tuyến: Trong thời đại kỹ thuật số ngày nay, việc đọc và viết truyện chữ trực tuyến đã trở thành một xu hướng phổ biến. Người đọc có thể tiếp cận với hàng ngàn tác phẩm khác nhau mà không cần mua sách giấy. Đồng thời, việc viết truyện chữ trực tuyến cũng giúp tác giả dễ dàng chia sẻ tác phẩm của mình với độc giả trên toàn cầu. Vì vậy, việc phát triển một trang web để viết truyện chữ sẽ đáp ứng nhu cầu ngày càng tăng của cả tác giả và người đọc.

Tạo một nền tảng cho các tác giả mới và không chuyên: Truyện chữ trực tuyến cung cấp cơ hội cho các tác giả mới và không chuyên để thể hiện sự sáng tạo và chia sẻ những câu chuyện của mình với cộng đồng. Một trang web cho phép tác giả dễ dàng tạo và quản lý các tác phẩm của mình, thu hút sự quan tâm và phản hồi từ người đọc. Việc phát triển một trang web như vậy sẽ đóng góp vào việc xây dựng một cộng đồng viết truyện chữ đa dạng và phong phú.

Tiện lợi và dễ sử dụng: Một trang web để viết truyện chữ phải được thiết kế để đáp ứng nhu cầu của cả tác giả và người đọc một cách thuận tiện. Các tính năng như giao diện trực quan, công cụ soạn thảo văn bản linh hoạt, khả năng lưu trữ và quản lý tác phẩm sẽ giúp tác giả dễ dàng thể hiện ý tưởng của mình và cập nhật tác phẩm theo nhu cầu. Đồng thời, độc giả cũng sẽ có trải nghiệm đọc truyện trực tuyến tiện lợi và dễ dàng tìm kiếm các tác phẩm mới và phổ biến.

## 2. Mục đích nghiên cứu

- Nghiên cứu và tìm hiểu cách sử dụng kết hợp giữa các công nghệ hỗ trợ xây dựng web viết truyện chữ .
- Nghiên cứu có thể tập trung vào việc phân tích các yếu tố như sở thích, độ tuổi, giới tính và xu hướng đọc của người đọc truyện chữ trên web. Điều này giúp tác giả hoặc người quản lý nền tảng truyện chữ có cái nhìn tổng quan về đối tượng đọc và từ đó phát triển nội dung và tính năng phù hợp.
- Nghiên cứu có thể tập trung vào cải thiện công nghệ và giao diện của nền tảng viết truyện chữ trên web. Điều này bao gồm việc tìm hiểu về các công nghệ phân tích ngôn ngữ tự nhiên.
- Nghiên cứu và tìm hiểu về database.
- Nghiên cứu và tìm hiểu cách kiểm duyệt ngôn từ xử lý ngôn ngữ Tiếng Việt.
- Áp dụng những kiến thức đã học để phân tích, thiết kế, xây dựng trang web.

### 3. Phạm vi nghiên cứu

- Tìm hiểu Django, MySQL, ReactJS, Redux, và Scikit-learn.
- Thuật toán Cosine Similarity .
- Mô hình thuật toán Naïve Bayes.

### 4. Phương pháp nghiên cứu

- Xác định rõ mục tiêu và phạm vi của nghiên cứu. Có thể là phân tích yêu cầu của người đọc, tìm hiểu về tương tác giữa tác giả và độc giả, nghiên cứu về công nghệ và giao diện, tối ưu hóa nội dung, hoặc các khía cạnh khác liên quan đến viết truyện chữ trên web.
- Tiến hành thu thập dữ liệu về truyện chữ trên web, có thể là các bài viết, truyện ngắn, hoặc tài liệu từ các nền tảng viết truyện chữ trực tuyến. Dữ liệu này sẽ là cơ sở để phân tích và thực hiện nghiên cứu.
- Phương pháp tham khảo từ thực tế: truy cập, quan sát, phân tích những trang viết truyện đã và đang thịnh hành trên internet.
- Phương pháp thực nghiệm: Ứng dụng kết hợp các công nghệ mới để giải quyết những vấn đề gặp phải trong quá trình xây dựng.

### 5. Cấu trúc báo cáo đồ án tốt nghiệp

Cấu trúc đồ án được chia thành các chương như sau:

#### **Tổng quan:**

- Lý do chọn đề tài
- Mục đích nghiên cứu
- Phạm vi nghiên cứu
- Phương pháp nghiên cứu

#### **Chương 1: Cơ sở lý thuyết**

- Giới thiệu về các công nghệ đã dùng
- Trình bày về các thuật toán đã sử dụng

#### **Chương 2: Phân tích thiết kế hệ thống**

- Tổng quan về một trang viết truyện chữ.
- Phân tích yêu cầu hệ thống.

- Các Use case của hệ thống.
- Biểu đồ tuần tự.
- Biểu đồ hoạt động.
- Cách tổ chức dữ liệu và thiết kế chương trình.

### **Chương 3: Cài đặt chương trình**

- Các chức năng của chương trình
- Kết quả và đánh giá



# CHƯƠNG 1 CƠ SỞ LÝ THUYẾT

## 1.1. Tổng quan về MySQL

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.

MySQL đang tạo ra bảng để có thể lưu trữ dữ liệu và định nghĩa về sự liên quan giữa những bảng đó. Client sẽ trực tiếp gửi yêu cầu SQL bằng 1 lệnh đặc biệt có trên MySQL. Ứng dụng tại server sẽ tiến hành phản hồi thông tin cũng như trả về những kết quả trên máy client.

Ngôn ngữ truy vấn SQL: MySQL sử dụng ngôn ngữ truy vấn SQL (Structured Query Language) để thao tác và truy vấn cơ sở dữ liệu. SQL cung cấp các câu lệnh cho phép tạo, truy vấn, cập nhật và xóa dữ liệu trong cơ sở dữ liệu MySQL.

Tích hợp và mở rộng: MySQL có khả năng tích hợp với các ngôn ngữ lập trình phổ biến như PHP, Python, Java và nhiều ngôn ngữ khác. Ngoài ra, nó cũng hỗ trợ các công cụ quản lý cơ sở dữ liệu và giao diện lập trình ứng dụng (API) cho phép mở rộng chức năng và tích hợp dễ dàng vào các ứng dụng hiện có.

Tính năng đa nền tảng: MySQL hỗ trợ trên nhiều nền tảng hệ điều hành như Windows, Linux, macOS và các nền tảng khác. Điều này cho phép phát triển và triển khai ứng dụng MySQL trên nhiều môi trường khác nhau.

### Định nghĩa về MySQL

Được phát hành chính thức từ thập niên 90s, MySQL hiện đang quản lý dữ liệu qua những cơ sở dữ liệu, với mỗi một cơ sở dữ liệu hoàn toàn có thể có rất nhiều những bản quan hệ có chứa dữ liệu. Ngoài ra, MySQL cũng có cùng 1 cách thức truy xuất cũng như mã lệnh tương tự cùng với ngôn ngữ SQL.

MySQL cung cấp một loạt các tính năng và chức năng như:

- Quản lý cơ sở dữ liệu: MySQL cho phép tạo và quản lý các cơ sở dữ liệu, bao gồm các bảng, khóa, chỉ mục và quan hệ giữa chúng.
- Ngôn ngữ truy vấn SQL: MySQL hỗ trợ ngôn ngữ truy vấn SQL để thực hiện các truy vấn và thao tác dữ liệu trong cơ sở dữ liệu.
- Bảo mật và quyền truy cập: MySQL cung cấp các tính năng bảo mật như quản lý người dùng, phân quyền truy cập dựa trên vai trò và quản lý mật khẩu để bảo vệ dữ liệu trong cơ sở dữ liệu.

- Sao lưu và phục hồi dữ liệu: MySQL cung cấp các công cụ để sao lưu và phục hồi dữ liệu, đảm bảo an toàn và khả năng khôi phục dữ liệu khi có sự cố.
- Hiệu suất và tối ưu hóa: MySQL được tối ưu hóa để xử lý tải lớn và có hiệu suất cao. Nó cung cấp các cơ chế tối ưu hóa truy vấn, chỉ mục, và phân vùng dữ liệu để tăng cường hiệu suất và khả năng mở rộng.
- Tích hợp và mở rộng: MySQL có khả năng tích hợp với các ngôn ngữ lập trình và công cụ phát triển phổ biến, cho phép phát triển ứng dụng linh hoạt và mở rộng chức năng theo nhu cầu.



*Hình 1 MySQL*

## **Ứng dụng của MySQL**

**Phát triển ứng dụng web:** MySQL thường được sử dụng làm hệ quản trị cơ sở dữ liệu cho các ứng dụng web. Nó cung cấp khả năng lưu trữ và truy xuất dữ liệu hiệu quả, cùng với tính bảo mật và khả năng mở rộng. MySQL thường được kết hợp với các ngôn ngữ lập trình phổ biến như PHP, Python và Java để xây dựng các ứng dụng web động và tương tác.

**Hệ thống quản lý nội dung (CMS):** MySQL là một lựa chọn phổ biến cho các hệ thống quản lý nội dung như WordPress, Drupal và Joomla. CMS sử dụng MySQL để lưu trữ và quản lý nội dung trang web, bao gồm bài viết, trang, danh mục và thông tin người dùng.

Ứng dụng di động: MySQL cũng được sử dụng trong phát triển ứng dụng di động. Các ứng dụng di động có thể sử dụng MySQL để lưu trữ và đồng bộ dữ liệu người dùng, bài viết, hình ảnh và thông tin khác. MySQL có thể tích hợp với các framework và công cụ phát triển di động như Android và iOS để xây dựng ứng dụng di động đa nền tảng.

Dịch vụ lưu trữ dữ liệu: MySQL được sử dụng trong các dịch vụ lưu trữ dữ liệu trực tuyến (cloud storage). Các công ty cung cấp dịch vụ lưu trữ dữ liệu như Dropbox, Google Drive và Box thường sử dụng MySQL để quản lý và lưu trữ các tệp tin và dữ liệu của người dùng.

Hệ thống phân tích dữ liệu: MySQL cung cấp các tính năng phân tích dữ liệu và truy vấn mạnh mẽ, đi kèm với khả năng xử lý tải lớn. Do đó, nó thường được sử dụng trong các hệ thống phân tích dữ liệu và báo cáo kinh doanh. MySQL có thể lưu trữ và truy xuất dữ liệu phục vụ cho việc phân tích, truy xuất thông tin và tạo báo cáo.

Ứng dụng IoT (Internet of Things): Với khả năng xử lý và quản lý dữ liệu linh hoạt, MySQL có thể được sử dụng trong các ứng dụng IoT. Nó có thể lưu trữ và phân tích dữ liệu từ các thiết bị IoT và hỗ trợ tích hợp với các hệ thống IoT khác.

### **Ưu điểm của MySQL**

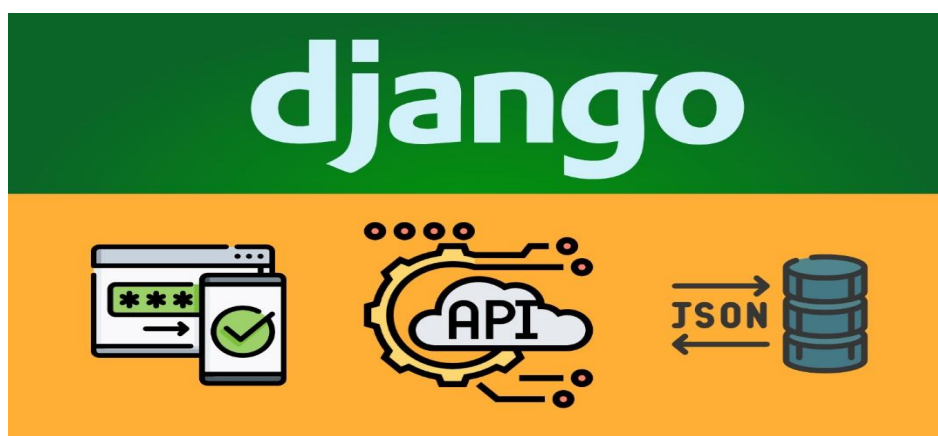
- Tính linh hoạt: MySQL là một hệ quản trị cơ sở dữ liệu linh hoạt và có thể được triển khai trên nhiều nền tảng hệ điều hành như Windows, Linux và macOS. Điều này cho phép bạn chọn môi trường phát triển và triển khai phù hợp với yêu cầu của bạn.
- Hiệu suất cao: MySQL được tối ưu hóa để đạt hiệu suất cao, cho phép xử lý tải lớn và truy vấn dữ liệu nhanh chóng. Cơ chế tối ưu hóa truy vấn, chỉ mục và bộ nhớ đệm giúp cải thiện hiệu suất và thời gian đáp ứng của cơ sở dữ liệu.
- Bảo mật mạnh mẽ: MySQL cung cấp các tính năng bảo mật để bảo vệ dữ liệu, bao gồm quản lý người dùng, quyền truy cập và mã hóa dữ liệu. Nó hỗ trợ các phương pháp xác thực và kiểm soát truy cập, đảm bảo rằng chỉ có người dùng có quyền được truy cập vào dữ liệu.
- Độ tin cậy và ổn định: MySQL đã được thử nghiệm và sử dụng rộng rãi trong các môi trường sản xuất. Nó cung cấp các tính năng như sao lưu và phục hồi dữ liệu, kiểm soát xung đột và khóa để đảm bảo tính toàn vẹn và độ tin cậy của cơ sở dữ liệu.
- Cộng đồng lớn và hỗ trợ mạnh mẽ: MySQL có một cộng đồng lớn và tích cực phát triển. Có sẵn nhiều tài liệu, tài liệu hướng dẫn, diễn đàn và nhóm người dùng để hỗ trợ và chia sẻ kiến thức về MySQL. Bạn có thể tìm thấy giải pháp và sự hỗ trợ cho các vấn đề cụ thể mà bạn gặp phải.
- Tích hợp và mở rộng: MySQL có khả năng tích hợp với các ngôn ngữ lập trình và công cụ phát triển phổ biến. Nó hỗ trợ các giao diện lập trình ứng dụng (API) và các kết nối chuẩn như JDBC và ODBC và phát triển ứng dụng một cách dễ dàng..

## Nhược điểm của MySQL

- Theo thiết kế, MySQL không nhằm mục đích làm mọi thứ và nó có những hạn chế về chức năng mà ứng dụng có thể yêu cầu.
- Cách MySQL xử lý các chức năng cụ thể (ví dụ: tài liệu tham khảo, giao dịch, kiểm tra...) làm cho nó kém tin cậy hơn so với một số RDBMS khác.
- Nếu số lượng bản ghi trong cơ sở dữ liệu của bạn tăng lên, việc truy xuất dữ liệu của bạn sẽ trở nên khó khăn hơn. Do đó, chúng ta sẽ phải thực hiện nhiều biện pháp để tăng tốc độ truy xuất dữ liệu, chẳng hạn như tải chia sẻ cơ sở dữ liệu này qua nhiều máy chủ hoặc tạo bộ nhớ cache MySQL.

## 1.2. Django

Django là một Web Framework bậc cao miễn phí, sử dụng mã nguồn mở được lập trình bằng Ngôn ngữ Python. Django cho phép người dùng kiểm soát chặt chẽ quá trình phát triển website thông qua một nền tảng duy nhất, có khả năng xử lý các thao tác phức tạp khi lập trình web. Django hiện nay sở hữu một cộng đồng đông đảo người sử dụng và có nhiều tài liệu hỗ trợ cho các developer mới tìm hiểu.



*Hình 2 Django*

### Lịch sử ra đời của Django.

Django là một framework phát triển ứng dụng web mạnh mẽ và phổ biến được viết bằng ngôn ngữ Python. Nó ra đời từ một dự án được phát triển bởi cộng đồng báo chí online Lawrence Journal-World vào năm 2003. Dự án ban đầu được gọi là "World Online" và được xây dựng để quản lý các ứng dụng web báo chí.

Trong quá trình phát triển dự án, nhóm phát triển đã nhận ra rằng nhiều phần mềm và công nghệ có thể được tái sử dụng trong các dự án web khác. Vì vậy, họ đã quyết định tách

riêng phần framework và đặt tên nó là Django. Tên Django được lấy cảm hứng từ tên của nhà soạn nhạc jazz người Mỹ Django Reinhardt.

Django được công bố công khai và phát hành vào tháng 7 năm 2005 bởi Adrian Holovaty và Simon Willison, hai thành viên chính trong nhóm phát triển của dự án Lawrence Journal-World. Sau đó, Django đã thu hút sự quan tâm và đóng góp từ cộng đồng phát triển mở rộng và trở thành một trong những framework phát triển web phổ biến nhất trên thế giới.

Django được đánh giá cao với các tính năng như quản lý cơ sở dữ liệu, xử lý URL, quản lý biểu mẫu, bảo mật, giao diện quản trị tự động và nhiều tính năng khác. Nó giúp lập trình viên xây dựng ứng dụng web nhanh chóng, dễ dàng và bảo mật.

Kể từ khi ra mắt, Django đã trải qua nhiều phiên bản và cải tiến liên tục. Hiện tại, Django vẫn là một trong những framework phát triển web hàng đầu và được sử dụng rộng rãi trên toàn thế giới.

### **Ưu điểm của Django**

- Nguyên tắc phát triển nhanh chóng: tức là các nhà phát triển có thể thực hiện nhiều iteration một lúc thay vì phải bắt đầu lại từ đầu.
- Nguyên lý DRY (Don't Repeat Yourself – không lặp lại)- có nghĩa là các nhà phát triển có thể tái sử dụng các đoạn code hiện có và tập trung vào nó.
- Các đoạn mã trùng lặp không cần thiết sẽ được framework này loại bỏ, từ đó giảm số lượng code. Nhờ đó, bạn sẽ tiết kiệm được nhiều thời gian hơn.
- Bảo mật cũng là ưu tiên hàng đầu của Django. Nó là một trong những framework có hệ thống bảo mật tốt nhất hiện nay.
- Nhờ hệ thống đó, các nhà phát triển có thể tránh các vấn đề về bảo mật phổ biến như: Clickjacking; Cross-site scripting; Cross-site request forgery; SQL injection.
- Framework này còn thường xuyên phát hành các bản vá bảo mật. Django được xem là framework đi đầu trong việc phản hồi các lỗ hổng bảo mật và cách báo các framework khác.
- Django có nhiều tính năng bổ sung và có thể mở rộng. Bạn có thể tạo các ứng dụng xử lý lưu lượng truy cập lớn hoặc khối lượng lớn thông tin thông qua Django.
- Django đa nền tảng (cross-platform). Bạn có thể chạy dự án trên Mac, Linux hoặc PC đều được.
- Django hoạt động với hầu hết các database chính và cho phép sử dụng cơ sở dữ liệu khác nhau tùy thuộc vào dự án. Thậm chí, bạn có thể sử dụng nhiều cơ sở dữ liệu cùng lúc.

- Django set up khá tốt. Điều này đã được chứng nhận bởi thời gian và một số người sử dụng khung công tác này.
- Nó có một big user cộng đồng, được hỗ trợ truy cập thông qua nhiều diễn đàn, kênh và các chuyên trang web.
- Bạn sẽ dễ dàng nhận được sự trợ giúp khi các vấn đề liên quan đến Django cũng như dễ dàng tìm thấy nhà phát triển chạy dự án cho công ty mình (nếu bạn có ý định chạy dự án trên Django).
- Nhờ đó bạn hoàn toàn có thể thích ứng nhanh chóng với các thay đổi của khung này (nếu có).
- Các vấn đề liên quan sẽ luôn được giải quyết ngay khi chúng tôi phát sinh. Phần mềm được cập nhật liên tục và các gói mới được phát hành để bạn làm việc với Django ngày càng thuận lợi.

### **Nhược điểm của Django**

- Chỉ định URL bằng quy tắc biểu thức không dễ thực hiện (đặc biệt là người mới bắt đầu). Nó cũng gây cảm giác cồng kềnh đối với các dự án nhỏ.
- Mẫu lỗi sẽ không phải là lỗi thông báo. Nếu bạn không biết, bạn sẽ mất rất nhiều thời gian để tìm ra vấn đề ở đâu hoặc tệ hơn, bạn sẽ không biết ứng dụng của mình đối với vấn đề.
- Django là một khuôn khổ có một khổ, bởi vì thế giới, sẽ có một số điều quan trọng nhất mà bạn cần tuân theo khi sử dụng nó..

### **Những ứng dụng nên viết bằng Django**

- Trang web thương mại điện tử: Django cung cấp các tính năng và công cụ hỗ trợ xây dựng các trang web thương mại điện tử. Bạn có thể quản lý sản phẩm, đơn hàng, thanh toán, giỏ hàng và chức năng khác. Các trang web thương mại điện tử như Shopify, Fiverr và Pinterest đã sử dụng Django để phát triển hệ thống của họ.
- Mạng xã hội: Django cung cấp các tính năng cần thiết để xây dựng mạng xã hội, bao gồm quản lý người dùng, tạo và quản lý hồ sơ cá nhân, kết nối người dùng và chia sẻ nội dung. Instagram là một ví dụ điển hình về ứng dụng mạng xã hội được xây dựng bằng Django.
- Hệ thống quản lý nội dung (CMS): Django cung cấp giao diện quản trị tự động và các tính năng quản lý nội dung tốt. Với Django, bạn có thể xây dựng các CMS cho

các trang web và ứng dụng web, giúp quản lý bài viết, hình ảnh, video và nội dung khác. Django CMS là một ví dụ tiêu biểu về CMS được phát triển bằng Django.

- Ứng dụng phân tích dữ liệu: Django cung cấp các tính năng và thư viện hỗ trợ cho việc xử lý và phân tích dữ liệu. Bạn có thể xây dựng các ứng dụng phân tích dữ liệu để tạo báo cáo, biểu đồ và visualizations dữ liệu. Django Rest Framework cung cấp các công cụ mạnh mẽ để phát triển các API phân tích dữ liệu.
- Ứng dụng IoT (Internet of Things): Django có thể được sử dụng để phát triển ứng dụng quản lý và giám sát các thiết bị IoT. Bạn có thể kết hợp Django với các công nghệ như Raspberry Pi, Arduino để xây dựng các hệ thống IoT phức tạp.
- Ứng dụng học máy và trí tuệ nhân tạo: Django có thể được sử dụng để phát triển các ứng dụng học máy và trí tuệ nhân tạo. Bạn có thể tích hợp các thư viện như TensorFlow hoặc scikit-learn để xây dựng và triển khai mô hình học máy trong ứng dụng Django.

### **Một số lý do nên sử dụng Django**

- Hoàn thiện: Django phát triển theo tư tưởng "Batteries included" (có thể hiểu ý nghĩa là tích hợp toàn bộ, chỉ cần gọi ra mà dùng). Nó cung cấp mọi thứ cho developer không cần phải nghĩ phải dùng cái ngoài. Chúng ta chỉ cần tập trung vào sản phẩm, tất cả đều hoạt động liền mạch với nhau.
- Đa năng: Django có thể được dùng để xây dựng hầu hết các loại website, từ hệ thống quản lý nội dung, cho đến các trang mạng xã hội hay web tin tức. Nó có thể làm việc với framework client-side, và chuyển nội dung hầu hết các loại format (HTML, RESS, JSON, XML, ...)
- Bảo mật: Django giúp các developer trang các lỗi bảo mật thông thường bằng cách cung cấp framework rằng có những kỹ thuật "phải làm như vậy" để bảo vệ website. Ví dụ: Django cung cấp bảo mật quản lý tên tài khoản và mật khẩu, tránh các lỗi cơ bản như để thông tin session lên cookie, mã hóa mật khẩu thay vì lưu thẳng.
- Dễ Scale: Django sử dụng kiến trúc shared-nothing dựa vào component (mỗi phần của kiến trúc sẽ độc lập với nhau, và có thể thay thế hoặc sửa đổi nếu cần thiết). Có sự chia tách rõ ràng giữa các phần nghĩa là nó có thể scale cho việc gia tăng traffic bằng cách thêm phần cứng ở mỗi cấp độ: caching, servers, database servers, hoặc application servers. Nhiều web về kinh doanh đã thành công khi Django được scale đáp ứng yêu cầu của họ
- Dễ maintain: code Django được viết theo nguyên tắc thiết kế và pattern có thể khuyến khích ý tưởng bảo trì và tái sử dụng code. Trên thực tế, nó sự theo khái

niệm Don't Repeat Yourself làm cho không có sự lặp lại không cần thiết, giảm một lượng code.

- Tính linh động: Django được viết bằng Python, nó có thể chạy đa nền tảng. Nó có nghĩa rằng bạn không ràng buộc một platform server cụ thể. Django được hỗ trợ tốt ở nhiều nhà cung cấp hosting, họ sẽ cung cấp hạ tầng và tài liệu cụ thể cho hosting web Django.

### **1.3. ReactJS**

ReactJS là một thư viện JavaScript có tính hiệu quả và linh hoạt để xây dựng các thành phần giao diện người dùng (UI) có thể sử dụng lại. ReactJS giúp phân chia các UI phức tạp thành các thành phần nhỏ (được gọi là component). Nó được tạo ra bởi Jordan Walke, một kỹ sư phần mềm tại Facebook. ReactJS ban đầu được phát triển và duy trì bởi Facebook và sau đó được sử dụng trong các sản phẩm của mình như WhatsApp & Instagram.

ReactJS được dùng để xây dựng các ứng dụng [single page application] (SPA). Một trong những điểm hấp dẫn của ReactJS là nó không chỉ được xây dựng bên phía clients mà còn sử dụng được bên phía server.

#### **Các khái niệm cơ bản về ReactJS**

##### **Virtual DOM**

Để hiểu rõ khái niệm về Virtual DOM, chúng ta cùng nhau đi tìm hiểu về DOM trước. DOM là một Document Object Model và là một cấu trúc trừu tượng của text. Các đoạn mã HTML được gọi là HTML DOM. Mỗi elements trong HTML là các nodes của DOM đó.

Điều này là không cần thiết, mặc dù tốc độ xử lý của DOM khá nhanh nhưng đối với các ứng dụng SPA việc thay đổi các DOM này là liên tục nên nó sẽ xảy ra khá chậm và không khả thi để xây dựng ứng dụng lớn. Lúc này Virtual DOM sẽ được dùng để thay thế. Nó được xây dựng dựa trên DOM thật, có một vài thuộc tính của DOM thật nhưng khi thay đổi Virtual DOM sẽ không thực hiện thay đổi trên màn hình giống như DOM thật.

##### **JSX**

JSX là viết tắt là Javascript XML, nó cho phép bạn viết các đoạn mã HTML trong React một cách dễ dàng và có cấu trúc hơn. Về cú pháp cũng gần tương tự như HTML, giả sử mình có 1 đoạn mã HTML như sau:



```
1 | <p class="text">freetuts.net</p>
```

Thì trong JSX thì sẽ được viết là :

```
1 | <p className="text">freetuts.net</p>
```

## Components

Khi bạn làm việc với một dự án lớn, UI có độ phức tạp cao chia thành các phần khác nhau. Việc chia nhỏ các thành phần trong UI là một điều cần thiết, các phần nhỏ này được gọi là các components, cho phép render các đoạn mã HTML,... Trong ReactJS cách viết components được chia thành 2 loại:

- Function components.

```
1 | //Function component
2 | function Clock(props) {
3 |   return (
4 |     <div>
5 |       <h1>Hello, world!</h1>
6 |     </div>
7 |   );
8 | }
```

## Class components

```
1 | //Class component
2 | class Clock extends React.Component {
3 |   render() {
4 |     return (
5 |       <div>
6 |         <h1>Hello, world!</h1>
7 |       </div>
8 |     );
9 |   }
10 | }
```

## Props và State

Props là một tham số được chuyển qua lại giữa các React Components, các props này được truyền qua các component với cú pháp giống như là HTML attributes.

State là một object mà lưu trữ giá trị của các thuộc tính bên trong components và chỉ tồn tại trong phạm vi của component đó. Mỗi khi bạn thay đổi giá trị của một state thì component đó sẽ được render lại.

```

1  class Car extends React.Component {
2      constructor(props) {
3          super(props);
4          this.state = {brand: "Ford"};
5      }
6      render() {
7          return (
8              <div>
9                  <h1>My Car</h1>
10             </div>
11         );
12     }
13 }

```

## React Lifecycle

React Lifecycle là một vòng đời của component, khi chúng ta tiến hành render một component thì ReactJS thực hiện nhiều tiến trình khác nhau, các tiến trình này được lặp đi lặp lại đối với các component.

Giả sử khi một component được gọi trước tiên nó sẽ cài đặt props và state, sau đó tiến hành mounting, update, unmounting,...việc tham gia vào quá trình này bạn cần sử dụng đến các hàm hỗ trợ của lifecycle.

## Một số điều cần biết Reactjs

- ReactJS chỉ là một view Library của Facebook nên nó giúp Render ra phần view và không phải là một framework JS nào. Bên cạnh đó, nó hỗ trợ xây dựng các thành phần UI tính tương tác cao, trạng thái và có thể sử dụng lại. Cả hai phần này đều là nơi xây dựng các Component và có thể kết nối với nhau khi hoạt động React trên cả Client và Render trên Server. Hơn nữa, react không phải là một MVC Framework nên khi thao tác có thể không cần Model và Controller mà cần phải kết hợp với các thư viện khác.
- State Components được sử dụng ít do nó gây khó khăn cho quá trình làm test và liên quan tới việc có Render lại hay không. Bên cạnh đó, State chỉ tồn tại trong một Component với khả năng trao đổi các dữ liệu với bên ngoài và việc sử dụng state là không cần thiết. Bạn cần hiểu rằng chỉ nên sử dụng state khi cần thiết và phản ánh đúng các trạng thái của Component.
- Component phải luôn luôn nhỏ để phần mềm dễ hiểu và dễ bảo trì hơn: Với các ReactJS, việc giữ cho Component nhỏ nhất có thể tái sử dụng và đạt được hiệu

năng cao nhất cùng với độ chia nhỏ phù hợp theo mức độ của team.

- Kết hợp ReactJS với Redux.JS, flux hay bất cứ luồng dữ liệu nào. Trong đó, Redux là một trong những nguồn dữ liệu được sử dụng nhiều nhất và tư duy của React tương đối hay.
- Bên cạnh đó có thể sử dụng JSX, ES6, Babel, Webpack và NPM. JSX là một trong những tính năng thú vị nhất của React và toàn bộ những thứ mà người dùng viết đều được hiển thị lên nếu bạn kết hợp cùng Babel biên dịch và tận dụng được tính năng mới của ES6. Hai nhân tố NPM và webpack đóng vai trò hỗ trợ quá trình đóng gói cũng như tận dụng các thư viện triệt để.
- Các trình Dev Tools của Redux và React được sử dụng để giúp lập trình viên Debug nhanh chóng và tìm ra các lỗi trong ứng dụng. Thậm chí nó còn có thể Inspect ngược lại các Component của React trong trạng thái Props và State của từng Component. Hay bạn cũng có thể quan sát trạng thái action và các lý do thay đổi State và back lại state trước.

### **Ưu điểm:**

- Rất dễ dàng để tiếp cận: Nhờ vào JSX, bạn có thể dễ dàng lồng các đoạn HTML vào trong JS.
- Reactjs cực kì hiệu quả: Reactjs tạo ra cho chính nó DOM ảo – nơi mà các component thực sự tồn tại trên đó. Điều này sẽ giúp cải thiện hiệu suất rất nhiều.
- Reactjs cũng tính toán những thay đổi nào cần cập nhật lên DOM và chỉ thực hiện chúng. Điều này giúp Reactjs tránh những thao tác cần trên DOM mà nhiều chi phí.
- Reactjs giúp việc viết các đoạn code JS dễ dàng hơn: Nó dung cú pháp đặc biệt là JSX (Javascript mở rộng) cho phép ta trộn giữa code HTML và Javascript.
- Ta có thể thêm vào các đoạn HTML vào trong hàm render mà không cần phải nối chuỗi. Đây là đặc tính thú vị của Reactjs. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.
- Nó có nhiều công cụ phát triển: Khi bạn bắt đầu Reactjs, đừng quên cài đặt ứng dụng mở rộng của Chrome dành cho Reactjs. Nó giúp bạn debug code dễ dàng hơn. Sau khi bạn cài đặt ứng dụng này, bạn sẽ có cái nhìn trực tiếp vào virtual DOM như thể bạn đang xem cây DOM thông thường.
- Render tăng server: Một trong những vấn đề với các ứng dụng đơn trang là tối ưu SEO và thời gian tải trang. Nếu tất cả việc xây dựng và hiển thị trang đều thực hiện ở client, thì người dùng sẽ phải chờ cho trang được khởi tạo và hiển thị lên.

- Duyệt sử dụng DOM và cũng có thể render bằng các chuỗi HTML mà server trả về.
- Làm việc với vấn đề test giao diện: Nó cực kì dễ để viết các test case giao diện vì virtual DOM được cài đặt hoàn toàn bằng JS.
- Reactjs là chỉ là một thư viện: React là thư viện cung cấp các hàm khai báo
- Các component trên UI. ReactDOM là thư viện tổng hợp cho phép render và so sánh sự khác biệt trên DOM. Redux là thư viện dùng để lưu trữ dữ liệu và React-redux dùng để kết dính React và Redux lại với nhau. Nó tổng hợp lại tạo nên ứng dụng Reactjs. Hiển nhiên những cái trên không phải là 1 framework chính thức để ép buộc bạn phải tuân theo. Nó giảm tải những ràng buộc với nhau giữa các thư viện và bạn có thể thay thế tầng hiển thị mà không ảnh hưởng đến những phần còn lại.
- Thay đổi tầng data thì khó nhưng hoàn toàn khả thi.

#### **Nhược điểm.**

- Reactjs chỉ phục vụ cho tầng View. React chỉ là View Library nó không phải là một MVC framework như những framework khác. Đây chỉ là thư viện của Facebook giúp render ra phần view. Vì thế React sẽ không có phần Model và Controller, mà phải kết hợp với các thư viện khác. React cũng sẽ không có 2-way binding hay là Ajax
- Tích hợp Reactjs vào các framework MVC truyền thống yêu cầu cần phải cấu hình lại.
- React khá nặng nếu so với các framework khác React có kích thước tương đương với Angular (Khoảng 35kb so với 39kb của Angular). Trong khi đó Angular là một framework hoàn chỉnh
- Khó tiếp cận cho người mới học Web

#### **1.4. Thư viện Scikit-learn**

Scikit-learn (Sklern) là thư viện mạnh mẽ nhất dành cho các thuật toán học máy được viết trên ngôn ngữ Python. Thư viện cung cấp một tập các công cụ xử lý các bài toán machine learning và statistical modeling gồm: classification, regression, clustering, và dimensionality reduction.

Thư viện được cấp phép bản quyền chuẩn FreeBSD và chạy được trên nhiều nền tảng Linux. Scikit-learn được sử dụng như một tài liệu để học tập.

Để cài đặt scikit-learn trước tiên phải cài thư viện SciPy (Scientific Python). Những thành phần gồm:

Để cài đặt scikit-learn trước tiên phải cài thư viện SciPy (Scientific Python). Những thành phần gồm:

- Numpy: Gói thư viện xử lý dãy số và ma trận nhiều chiều
- SciPy: Gói các hàm tính toán logic khoa học
- Matplotlib: Biểu diễn dữ liệu dưới dạng đồ thị 2 chiều, 3 chiều
- IPython: Notebook dùng để tương tác trực quan với Python
- SymPy: Gói thư viện các kí tự toán học
- Pandas: Xử lý, phân tích dữ liệu dưới dạng bảng

Những thư viện mở rộng của SciPy thường được đặt tên dạng SciKits. Như thư viện này là gói các lớp, hàm sử dụng trong thuật toán học máy thì được đặt tên là scikit-learn.

Scikit-learn hỗ trợ mạnh mẽ trong việc xây dựng các sản phẩm. Nghĩa là thư viện này tập trung sâu trong việc xây dựng các yếu tố: dễ sử dụng, dễ code, dễ tham khảo, dễ làm việc, hiệu quả cao.

Mặc dù được viết cho Python nhưng thực ra các thư viện nền tảng của scikit-learn lại được viết dưới các thư viện của C để tăng hiệu suất làm việc. Ví dụ như: Numpy(Tính toán ma trận), LAPACK, LibSVM và Cython.



*Hình 3 Scikit-learn*

### **Lí do nên sử dụng Scikit-learn**

Scikit-learn là một thư viện Python phổ biến và mạnh mẽ trong lĩnh vực học máy và khoa học dữ liệu. Dưới đây là một số lý do nên sử dụng Scikit-learn trong các dự án học máy:

- Dễ sử dụng: Scikit-learn được thiết kế với nguyên tắc "dễ học, dễ sử dụng". Nó cung cấp các API đơn giản và rõ ràng cho việc xây dựng và đánh giá các mô hình học máy. Cú pháp dễ hiểu và tài liệu phong phú của Scikit-learn giúp người dùng mới và chuyên gia đều có thể nhanh chóng áp dụng và tận dụng các tính năng của thư viện.
- Tích hợp và tương thích: Scikit-learn tương thích tốt với các thư viện và công cụ khác trong hệ sinh thái Python. Bạn có thể kết hợp Scikit-learn với NumPy, Pandas và Matplotlib để xử lý dữ liệu, khám phá và trực quan hóa. Ngoài ra, Scikit-learn cũng tích hợp với các thư viện học máy khác như TensorFlow và Keras, mở rộng khả năng xử lý và đào tạo mô hình.
- Đa dạng các thuật toán: Scikit-learn cung cấp một loạt các thuật toán học máy phổ biến và mạnh mẽ. Từ các thuật toán phân loại (classification), hồi quy (regression), gom nhóm (clustering), phân tích thành phần chính (principal component analysis) đến xử lý dữ liệu và trích xuất đặc trưng (data preprocessing, feature extraction), Scikit-learn đáp ứng nhiều nhu cầu và phù hợp với nhiều bài toán khác nhau.
- Tự động hóa và đa luồng: Scikit-learn cho phép tự động hóa quá trình lựa chọn mô hình và tinh chỉnh siêu tham số (hyperparameter tuning) thông qua các công cụ như GridSearchCV và RandomizedSearchCV. Nó cũng hỗ trợ tính toán đa luồng (multithreading) để tăng tốc quá trình huấn luyện và dự đoán.
- Hiệu năng và mở rộng: Scikit-learn được tối ưu hóa để có hiệu năng tốt và khả năng mở rộng cao. Nó có thể xử lý các tập dữ liệu lớn và hỗ trợ phân tán trên nhiều CPU hoặc máy tính.
- Tài liệu phong phú và cộng đồng phát triển mạnh mẽ: Scikit-learn có một tập tài liệu phong phú và hướng dẫn chi tiết, bao gồm các ví dụ và hướng dẫn sử dụng. Ngoài ra, có một cộng đồng phát triển đông đảo và sẵn lòng hỗ trợ, giúp bạn giải quyết các vấn đề và tìm hiểu thêm về Scikit-learn.

### **Nhóm thuật toán**

- Clustering: Nhóm thuật toán Phân cụm dữ liệu không gán nhãn. Ví dụ thuật toán KMeans
- Cross Validation: Kiểm thử chéo, đánh giá độ hiệu quả của thuật toán học giám sát sử dụng dữ liệu kiểm thử (validation data) trong quá trình huấn luyện mô hình.
- Datasets: Gồm nhóm các Bộ dữ liệu được tích hợp sẵn trong thư viện. Hầu như các bộ dữ liệu đều đã được chuẩn hóa và mang lại hiệu suất cao trong quá trình

huấn luyện như iris, digit, ...

- Dimensionality Reduction: Mục đích của thuật toán này là để Giảm số lượng thuộc tính quan trọng của dữ liệu bằng các phương pháp như tổng hợp, biểu diễn dữ liệu và lựa chọn đặc trưng. Ví dụ thuật toán PCA (Principal component analysis).
- Ensemble methods: Các Phương pháp tập hợp sử dụng nhiều thuật toán học tập để có được hiệu suất dự đoán tốt hơn so với bất kỳ thuật toán học cấu thành nào.
- Feature extraction: Trích xuất đặc trưng. Mục đích là để định nghĩa các thuộc tính với dữ liệu hình ảnh và dữ liệu ngôn ngữ.
- Feature selection: Trích chọn đặc trưng. Lựa chọn các đặc trưng có ý nghĩa trong việc huấn luyện mô hình học giám sát.
- Parameter Tuning: Tinh chỉnh tham số. Các thuật toán phục vụ việc lựa chọn tham số phù hợp để tối ưu hóa mô hình.
- Manifold Learning: Các thuật toán học tổng hợp và Phân tích dữ liệu đa chiều phức tạp.
- Supervised Models: Học giám sát. Mảng lớn các thuật toán học máy hiện nay. Ví dụ như linear models, discriminate analysis, naive bayes, lazy methods, neural networks, support vector machines và decision trees.

## 1.5. Mô hình Naive Bayes

### Nguyên lý cơ bản của mô hình Naive Bayes

Sử dụng công thức Bayes để tính xác suất của một lớp (class) dựa trên các đặc trưng đầu vào. Công thức Bayes được biểu diễn như sau:

$$P(\text{class} \mid \text{features}) = (P(\text{class}) * P(\text{features} \mid \text{class})) / P(\text{features})$$

Trong đó:

- $P(\text{class} \mid \text{features})$  là xác suất của lớp dựa trên các đặc trưng đầu vào.
- $P(\text{class})$  là xác suất tiên nghiệm của lớp.
- $P(\text{features} \mid \text{class})$  là xác suất của các đặc trưng đầu vào dựa trên lớp.
- $P(\text{features})$  là xác suất của các đặc trưng đầu vào.

Mô hình Naive Bayes sử dụng giả định độc lập để ước lượng  $P(\text{features} \mid \text{class})$ . Dựa trên đặc trưng của từng lớp, mô hình tính toán xác suất của mỗi lớp và dựa vào xác suất cao nhất để phân loại.

Ví dụ minh họa ứng dụng mô hình học máy Naive Bayes để phân biệt từ ngữ thô tục sau đây:

```

# print(result)

import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score

# Dữ liệu huấn luyện
sentences = [
    .... "This is a clean sentence.",
    .... "Don't use foul language!",
    .... "The weather is nice today.",
    .... "You're an idiot.",
    .... "Please be polite."
    ..
]

```

```

# Nhãn cho dữ liệu huấn luyện
# (1: từ ngữ thô tục, 0: không phải từ ngữ thô tục)
labels = np.array([0, 1, 0, 1, 0])

# Tiền xử lý dữ liệu văn bản
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(sentences)

# Chia dữ liệu thành tập huấn luyện và tập kiểm tra
X_train, X_test, y_train, y_test = train_test_split(X, labels, test_size=0.2, random_state=42)

# Xây dựng mô hình mạng nơ-ron
model = MLPClassifier(hidden_layer_sizes=(100,), max_iter=1000, random_state=42)

# Huấn luyện mô hình
model.fit(X_train, y_train)

# Dự đoán nhãn cho dữ liệu kiểm tra
y_pred = model.predict(X_test)

```

```

# Đánh giá mô hình
accuracy = accuracy_score(y_test, y_pred)

def is_offensive(sentence):
    .... # Tiền xử lý câu văn
    .... preprocessed_sentence = vectorizer.transform([sentence])

    .... # Dự đoán nhãn của câu văn
    .... label = model.predict(preprocessed_sentence)[0]

    .... # Trả về kết quả
    .... if label == 1:
    ....     return True
    .... else:
    ....     return False

```



Import các thư viện cần thiết:

- `numpy` (được import dưới tên `np`): Thư viện để làm việc với mảng và ma trận.
- `train_test_split` từ `sklearn.model_selection`: Hàm để chia dữ liệu thành tập huấn luyện và tập kiểm tra.
- `CountVectorizer` từ `sklearn.feature_extraction.text`: Class để biểu diễn văn bản dưới dạng vector đếm từ.
- `MLPClassifier` từ `sklearn.neural_network`: Class để xây dựng mô hình mạng nơ-ron đa tầng.
- `accuracy_score` từ `sklearn.metrics`: Hàm để tính độ chính xác của mô hình.

Khai báo dữ liệu huấn luyện:

- `sentences`: Một danh sách các câu văn để huấn luyện mô hình.
- `labels`: Một mảng `numpy` chứa nhãn tương ứng cho từng câu văn. Nhãn 1 đại diện cho từ ngữ thô tục, nhãn 0 đại diện cho không phải từ ngữ thô tục.

Tiền xử lý dữ liệu văn bản:

- Sử dụng `CountVectorizer` để biểu diễn câu văn dưới dạng vector đếm từ. Mỗi từ trong câu văn sẽ được biểu diễn bằng một feature và giá trị của feature là số lần xuất hiện của từ đó trong câu văn.
- Chia dữ liệu thành tập huấn luyện và tập kiểm tra:
- Sử dụng hàm `train_test_split` để chia dữ liệu và nhãn thành tập huấn luyện và tập kiểm tra. Trong ví dụ này, tập kiểm tra chiếm 20% của dữ liệu ban đầu.

Xây dựng mô hình mạng nơ-ron:

- Sử dụng `MLPClassifier` để xây dựng mô hình mạng nơ-ron đa tầng với một lớp ẩn gồm 100 nơ-ron.
- Huấn luyện mô hình:
- Sử dụng phương thức `fit` để huấn luyện mô hình trên tập huấn luyện (`X_train` và `y_train`).

Dự đoán nhãn cho dữ liệu kiểm tra:

- Sử dụng phương thức `predict` để dự đoán nhãn cho tập kiểm tra (`X_test`). Kết quả dự đoán được lưu trong mảng `y_pred`.

Đánh giá mô hình:

- Sử dụng hàm `accuracy_score` để tính toán độ chính xác của mô hình bằng cách so sánh nhãn thực tế (`y_test`) với nhãn dự đoán (`y_pred`).

Hàm `is_offensive(sentence)`:

- Hàm này nhận một câu văn và kiểm tra xem câu văn có chứa từ ngữ thô tục hay không.
- Tiền xử lý câu văn bằng cách biểu diễn nó dưới dạng vector đếm từ sử dụng `CountVectorizer`.
- Dùng mô hình đã huấn luyện để dự đoán nhãn của câu văn (1 nếu là từ ngữ thô tục, 0 nếu không).
- Trả về kết quả kiểm tra (True nếu câu văn chứa từ ngữ thô tục, False nếu không).

Kiểm tra một câu văn:

- Sử dụng hàm `is_offensive` để kiểm tra xem câu văn "This is a clean sentence." có chứa từ ngữ thô tục hay không.
- Kết quả kiểm tra sẽ được in ra màn hình.

### **Ưu điểm**

- Đơn giản và nhanh chóng để huấn luyện và dự đoán.
- Hiệu suất tốt trong các bài toán với số lượng đặc trưng lớn.
- Khả năng làm việc tốt với các bài toán phân loại văn bản và phân loại dựa trên từ khóa.

### **Nhược điểm**

- Giả định độc lập của các đặc trưng có thể không thực tế trong nhiều tình huống thực tế.
- Dễ bị ảnh hưởng bởi đặc trưng dư thừa hoặc không có tác động lớn đến kết quả.
- Tính chất xác suất của mô hình có thể dẫn đến kết quả không chính xác khi dữ liệu huấn luyện không đủ.

### **Ứng dụng**

Mô hình Naive Bayes thường được sử dụng trong các bài toán phân loại văn bản, lọc thư rác (spam detection), phân loại tin tức, phân loại tài liệu, và các bài toán dự đoán xác suất.

## **1.6. Thuật toán Cosine Similarity**

Thuật toán Cosine Similarity (tương đồng cosin) là một phương pháp để đo độ tương đồng giữa hai vector dựa trên góc cosin giữa chúng. Đây là một phương pháp phổ biến được sử dụng trong lĩnh vực xử lý ngôn ngữ tự nhiên và khám phá dữ liệu.

### **Công thức**

Công thức tích vô hướng của hai vector (dot product) như sau:

$$\mathbf{A} \cdot \mathbf{B} = \|\mathbf{A}\| \|\mathbf{B}\| \cos(\theta),$$

Công thức trên được phát biểu như sau: Tích vô hướng của vector A và vector B bằng tích độ lớn của vector A và vector B với cosine góc tạo bởi vector A và vector B.

Từ đó ta rút ra:

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}},$$

Trong đó:

$\mathbf{A} \cdot \mathbf{B}$  là tích vô hướng (dot product) của hai vector A và B.

$\|\mathbf{A}\|$  và  $\|\mathbf{B}\|$  lần lượt là độ dài (norm) của vector A và B.

Giá trị Cosine Similarity nằm trong khoảng  $[-1, 1]$ . Giá trị 1 thể hiện hai vector hoàn toàn giống nhau, giá trị -1 thể hiện hai vector hoàn toàn trái ngược nhau và giá trị 0 thể hiện hai vector không có sự tương đồng góc.

Ví dụ ứng dụng thuật toán Cosine Similarity để so sánh hai đoạn văn bản Tiếng Việt

```
from pyvi import ViTokenizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

def preprocess_text_vietnamese(text):
    ... # Tiền xử lý văn bản tiếng Việt
    ... tokenized_text = ViTokenizer.tokenize(text)
    ... # Các bước tiền xử lý khác (loại bỏ stopwords, chuẩn hóa văn bản, ...)
    ... return tokenized_text
```

Hàm `preprocess_text_vietnamese(text)`: Đây là hàm để tiền xử lý văn bản tiếng Việt trước khi tính toán Cosine Similarity. Trong ví dụ này, hàm sử dụng thư viện `pyvi` để tách từ (`tokenize`) và chuẩn hóa văn bản tiếng Việt. Bạn có thể thêm các bước tiền xử lý khác như loại bỏ stopwords, chuẩn hóa văn bản theo nhu cầu.

```
def calculate_similarity(text1, text2):
    """# Tiền xử lý văn bản
    preprocessed_text1 = preprocess_text_vietnamese(text1)
    preprocessed_text2 = preprocess_text_vietnamese(text2)

    # Biểu diễn văn bản dưới dạng vector TF-IDF
    vectorizer = TfidfVectorizer()
    vectors = vectorizer.fit_transform([preprocessed_text1, preprocessed_text2])

    # Tính toán Cosine Similarity giữa hai vector văn bản
    cosine_similarity_score = cosine_similarity(vectors[0], vectors[1])[0][0]

    return cosine_similarity_score

# Sử dụng chương trình kiểm duyệt truyện
new_story = "Nội dung truyện mới"
existing_story = "Nội dung truyện mới"

similarity = calculate_similarity(new_story, existing_story)
print(f"Truyện bị trùng {round(similarity * 100, 2)}% với truyện trong kho truyện.")
```

Hàm `calculate_similarity(text1, text2)`: Đây là hàm để tính toán Cosine Similarity giữa hai văn bản. Trước tiên, hàm gọi `preprocess_text_vietnamese` để tiền xử lý hai văn bản đầu vào. Sau đó, hàm sử dụng `TfidfVectorizer` từ thư viện `sklearn` để biểu diễn văn bản dưới dạng vector TF-IDF. Vectorizer này sẽ tính toán chỉ số TF-IDF cho các từ trong văn bản và biểu diễn văn bản thành các vector. Cuối cùng, hàm sử dụng `cosine_similarity` để tính toán Cosine Similarity giữa hai vector văn bản và trả về kết quả.

## Ưu điểm

- Đơn giản và hiệu quả: Cosine Similarity là một phương pháp tính toán đơn giản và nhanh chóng. Nó không yêu cầu nhiều phép tính phức tạp và thích hợp cho việc xử lý các tập dữ liệu lớn.
- Độc lập với tỷ lệ: Cosine Similarity không bị ảnh hưởng bởi tỷ lệ của các vector đầu vào. Điều này có nghĩa là nó có thể tìm ra sự tương đồng giữa các vector mà không phụ thuộc vào kích thước hay đơn vị đo lường.
- Xử lý các văn bản dài: Trong xử lý ngôn ngữ tự nhiên, Cosine Similarity thường được sử dụng để so sánh các văn bản dựa trên vector đặc trưng. Nó thích hợp cho việc xử lý các văn bản dài và có khả năng ứng dụng cho các hệ thống tìm kiếm và gợi ý.

## Nhược điểm

- Không xử lý được ngữ cảnh: Cosine Similarity không xem xét ngữ cảnh và sự liên quan giữa các từ hoặc đặc trưng trong văn bản. Điều này có thể dẫn đến việc bỏ qua thông tin quan trọng và làm giảm chất lượng của kết quả.
- Không xử lý được ý nghĩa ngữ nghĩa: Cosine Similarity không hiểu được ý nghĩa ngữ nghĩa của từng từ hay đặc trưng. Điều này có thể dẫn đến việc không phân biệt được giữa các từ đồng nghĩa hoặc từ trái nghĩa, gây sai sót trong việc so sánh và phân loại.
- Yếu tố giới hạn không gian: Cosine Similarity chỉ xem xét mặt phẳng không gian vector và tính toán góc cosin. Điều này có nghĩa là nó không thể hiểu được các mối quan hệ phức tạp và không gian không gian giữa các vector, giới hạn khả năng của nó trong việc đo đặc độ tương đồng.

## Ứng dụng của thuật toán Cosine Similarity

- Xử lý ngôn ngữ tự nhiên: Cosine Similarity được sử dụng để so sánh văn bản, tính toán độ tương đồng giữa các văn bản, và thực hiện các tác vụ như tìm kiếm tương đồng, phân loại văn bản, gom nhóm (clustering) văn bản.
- Khám phá dữ liệu: Cosine Similarity được sử dụng để đo độ tương đồng giữa các mẫu dữ liệu trong không gian nhiều chiều. Điều này có thể áp dụng trong việc gom nhóm dữ liệu, tìm kiếm gần nhất (nearest neighbor search), và hệ thống gợi ý (recommendation systems).
- Xử lý hình ảnh và âm thanh: Cosine Similarity có thể được sử dụng để so sánh các đặc trưng trích xuất từ hình ảnh và âm thanh. Điều này hữu ích trong việc tìm kiếm hình ảnh tương tự, nhận dạng mẫu âm thanh và nhiều tác vụ khác trong xử lý tín hiệu.

## Kết luận

Thuật toán Cosine Similarity là một công cụ mạnh mẽ để đo đặc độ tương đồng giữa các vector. Nó thường được sử dụng trong nhiều lĩnh vực khác nhau và mang lại những kết quả hiệu quả trong việc so sánh, phân loại và gom nhóm dữ liệu.

## **CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG.**

### **2.1. Tổng quan về một trang viết truyện chữ.**

Tạo và quản lý tác phẩm: Người dùng có thể tạo mới các tác phẩm văn bản, cấu trúc chương, thêm và chỉnh sửa nội dung, và quản lý các tác phẩm đã xuất bản. Giao diện dễ sử dụng và trực quan giúp tác giả tạo và chỉnh sửa nội dung một cách thuận tiện.

Tìm kiếm và khám phá: Trang web cung cấp tính năng tìm kiếm để người đọc có thể tìm kiếm và khám phá các tác phẩm theo từ khóa, thể loại. Điều này giúp người đọc tìm được các tác phẩm phù hợp với sở thích của mình.

Tài khoản và quản lý người dùng: Trang web có hệ thống đăng ký và đăng nhập, cho phép người dùng tạo và quản lý tài khoản của mình. Người dùng có thể theo dõi các tác phẩm yêu thích, theo dõi tác giả, và nhận thông báo về các bài viết mới. Họ có thể biết được mình là ai.

Trang web viết truyện chữ có thể mang lại lợi ích cho cả tác giả và độc giả. Tác giả có thể chia sẻ và xuất bản tác phẩm của mình một cách dễ dàng, thu hút độc giả và nhận phản hồi từ độc giả. Độc giả có thể tìm kiếm và khám phá các tác phẩm mới, tương tác với tác giả và cộng đồng độc giả khác, và thưởng thức những câu chuyện thú vị.

### **2.2. Phân tích yêu cầu hệ thống.**

#### **2.2.1. Khái quát chung về hệ thống.**

Trang web viết truyện chữ được tạo ra nhằm mục đích tạo ra môi trường để cho mọi người có sáng tạo những câu truyện mà mình nghĩ ra và có thể thưởng thức những truyện hay từ các tác giả ngoài ra còn có thể đưa ra đánh giá góp ý để người sáng tác có thể tiếp nhận ý kiến từ độc giả để rút được kinh nghiệm từ đó có thể trau chuốt cho các tác phẩm tiếp theo

Trang web viết truyện chữ được thiết kế để trở thành một công cụ giúp mọi người đọc truyện và có thể sáng tác. Để người dùng có thể viết truyện, các thành viên cần phải đăng ký bằng email cá nhân và điền các thông tin cần thiết nhất nhằm tạo tính xác thực.

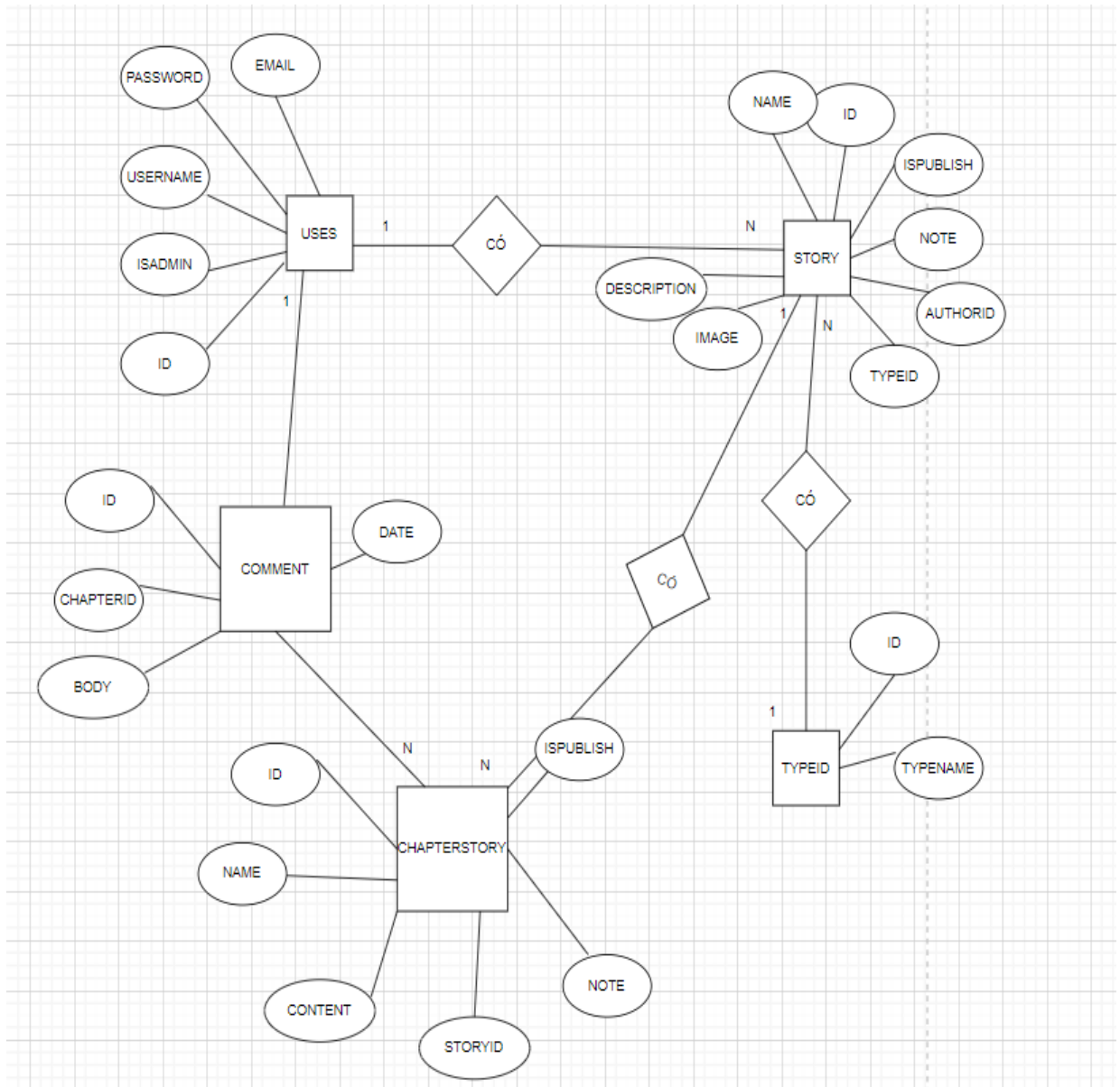
Với hệ thống này, người dùng có thể tạo ra một truyện mới với các thành phần bao gồm tên truyện mô tả truyện thể loại truyện sau đó sẽ lưu ở trạng thái chưa công khai. Để đạt được trạng thái công khai thì người sáng tác cần phải đạt được những yêu cầu tốt sau:

- Tên truyện không bị trùng với tên truyện đã có của tác giả khác.
- Tên truyện và phần mô tả không được chứa các từ ngữ nhạy cảm.

- Nội dung của các chương truyện phải phải đạt được số từ nhất định, không được giống với nội dung chương truyện của truyện nào đó đã tồn tại trong trang web và đặt biệt nội dung không được chứa các từ ngữ nhạy cảm.

Bên admin kiểm duyệt có quyền xem nội dung của truyện trước khi công khai và có quyền công khai cũng như hủy công khai truyện. Ngoài ra admin có quyền chủ động gửi email đánh giá tiêu chuẩn kiểm duyệt truyện đến người sáng tác sau khi nội dung đã được học máy kiểm duyệt.

### Mô hình ERD

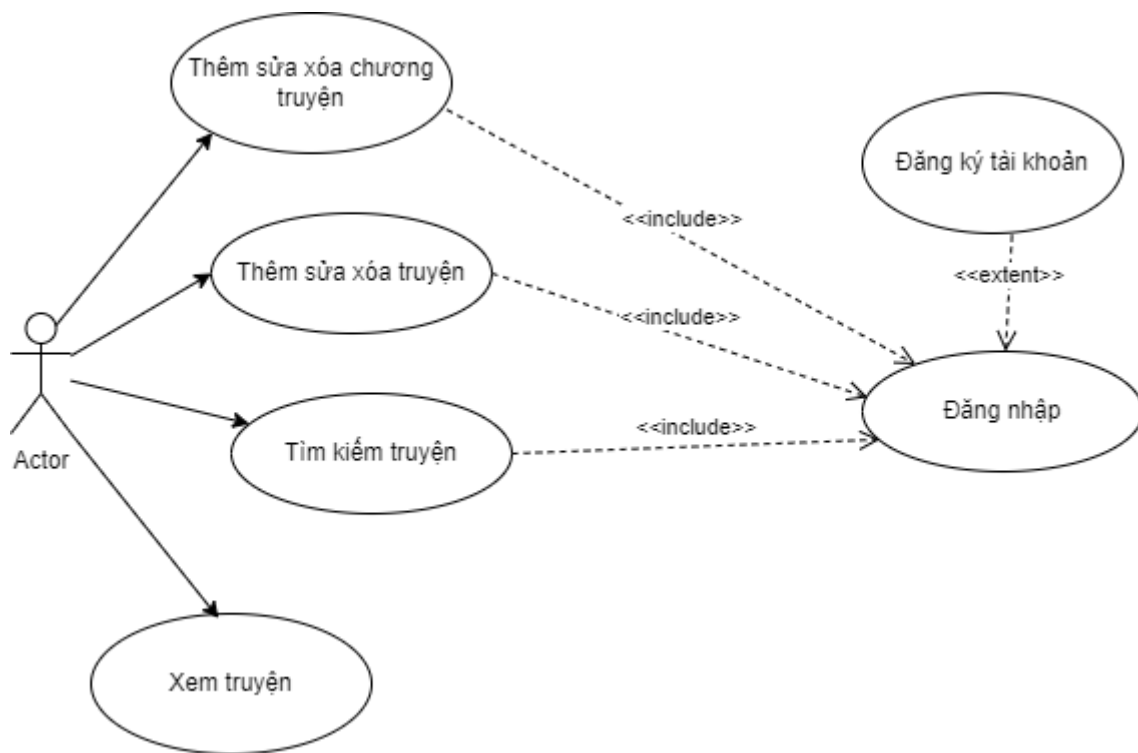


Hình 4 Mô hình ERD

## 2.3. Các Use case của hệ thống.

### 2.3.1. Biểu đồ Use case

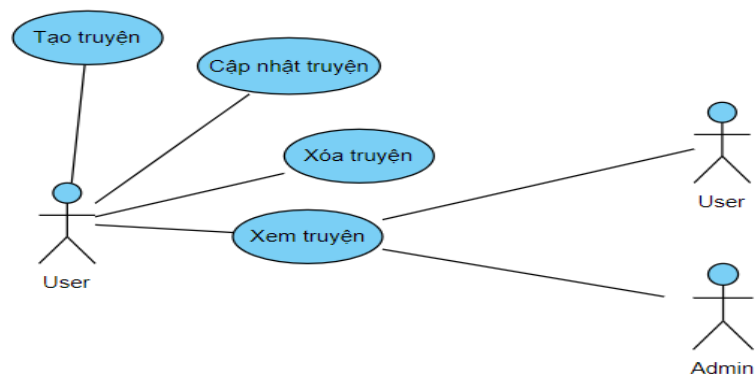
#### Use case tổng quát.



Hình 5 Use case tổng quát

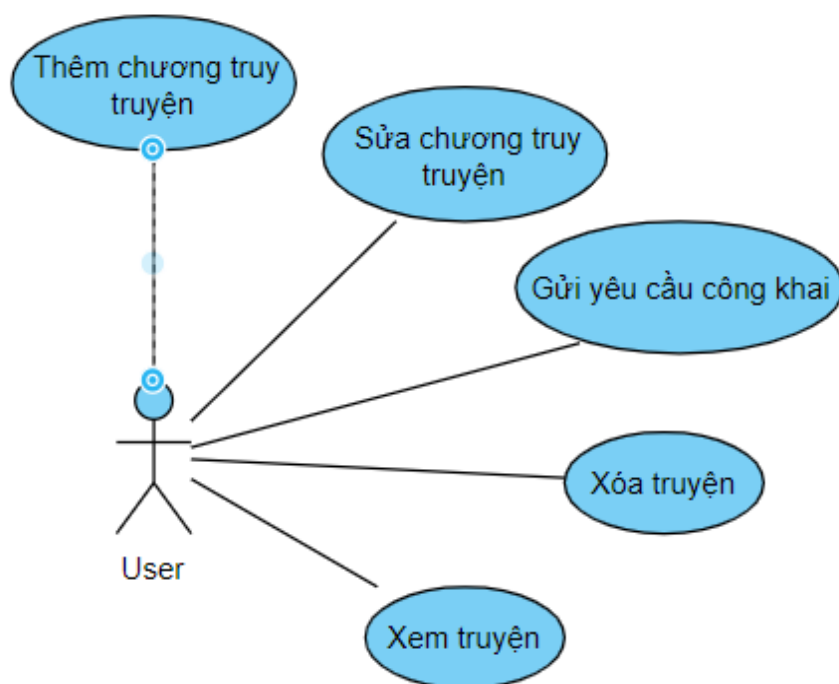


### Use case quản lý truyện.



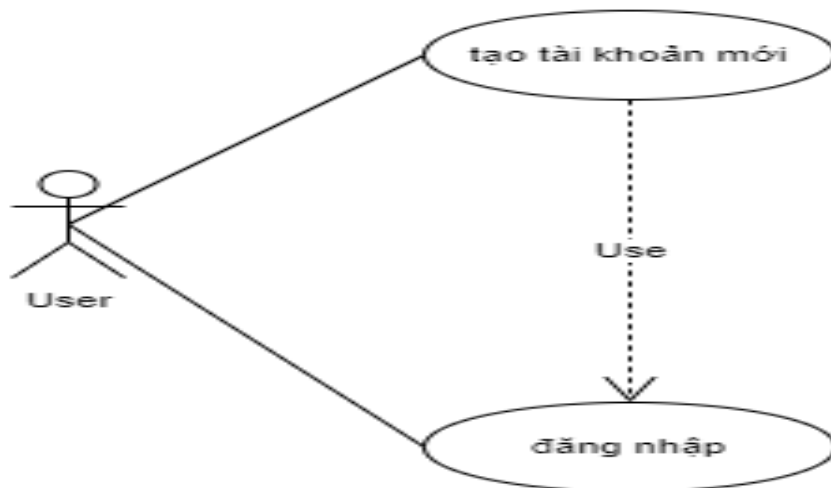
Hình 6 Use Case quản lý truyện

### Use case quản lý chương truyện



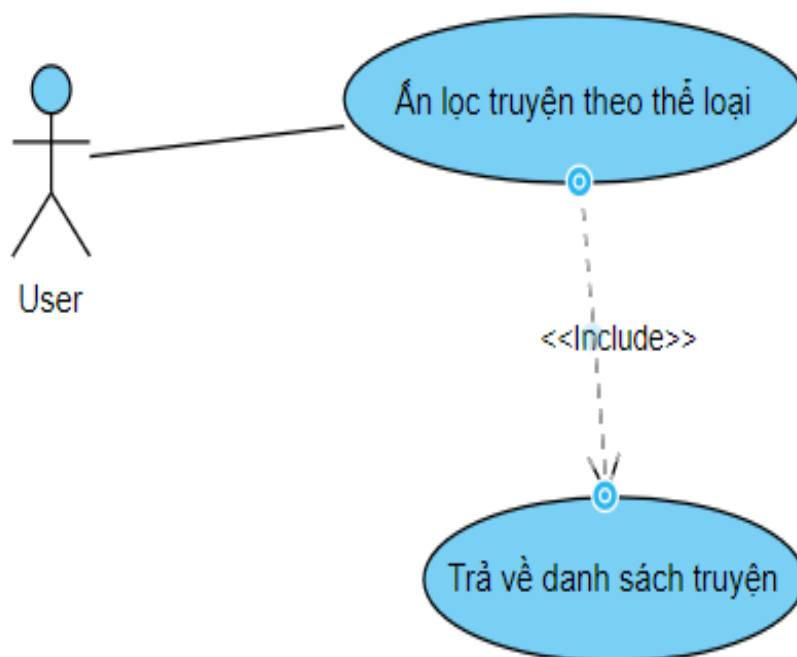
Hình 7 Use Case quản lý chương truyện

### Use case đăng nhập



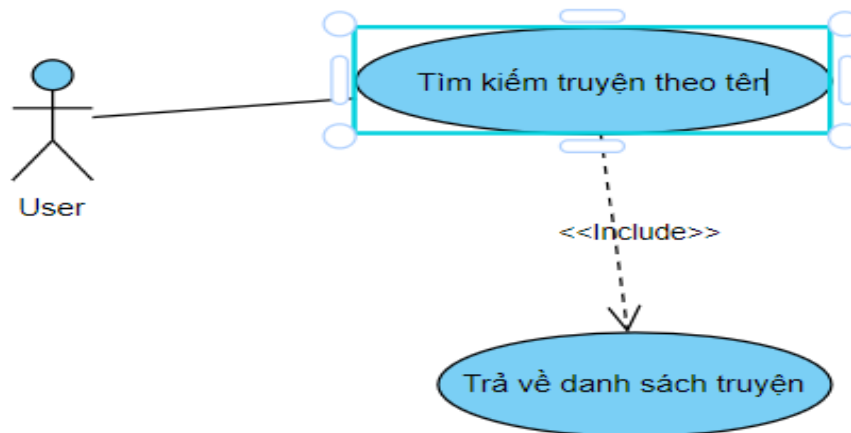
*Hình 8 Use Case đăng nhập*

### Use case lọc truyện theo thể loại.



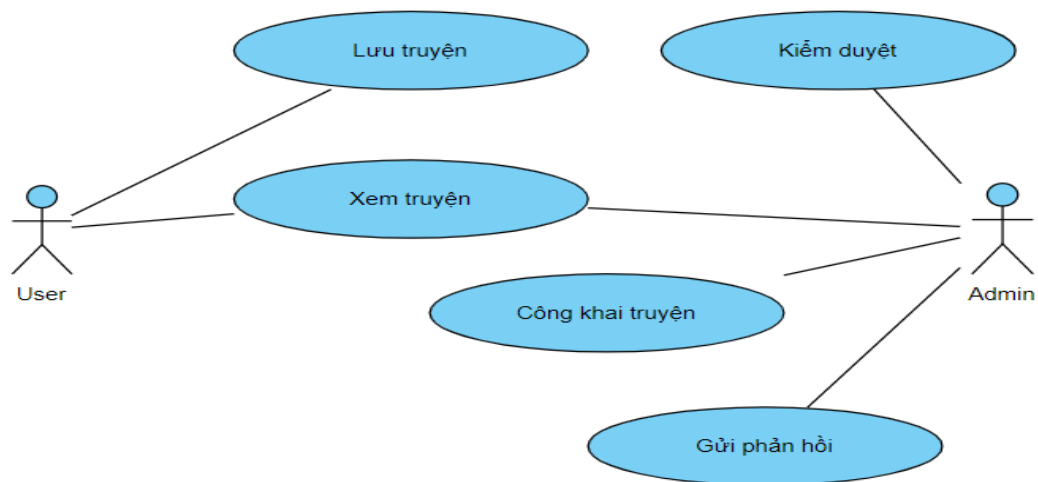
*Hình 9 Use Case lọc truyện theo thể loại*

### Use case tìm kiếm truyện theo tên truyện



*Hình 10 Use Case tìm kiếm truyện theo tên truyện*

### Use Case kiểm duyệt truyện



*Hình 11 Use Case kiểm duyệt truyện*

### 2.3.2. Mô tả các Use case.

#### Use case tổng quát.

Mô tả chi tiết:

- Use case này mô tả toàn bộ hệ thống hoạt động web viết truyện.
- Các bước thực hiện:

- + Hệ thống yêu cầu user cung cấp thông tin đăng nhập gồm tên đăng nhập, email và mật khẩu.
- + User nhập xong thông tin đăng nhập và click nút đăng nhập.
- + Hệ thống check lại thông tin đăng nhập. Nếu đăng nhập thành công hệ thống dựa trên thông tin đăng nhập sẽ đăng nhập vào màn hình chính. Nếu đăng nhập thất bại, hệ thống sẽ hiện thông báo cho người dùng và yêu cầu đăng nhập lại.
- + Sau khi đăng nhập thành công thì người dùng có thể thực hiện tất cả các chức năng được cấp dưới quyền user để viết truyện.

### **Use case quản lý truyện.**

Mô tả chi tiết:

- Đối tượng sử dụng (user): người dùng.
- Use case này mô tả các người dùng tạo, cập nhật, xóa truyện của mình.
- Các bước thực hiện:
  - + Khi user muốn tạo truyện mới thì hệ thống sẽ yêu cầu người dùng nhập các thông tin truyện cần thiết.
  - + User nhập xong thông tin và click nút đăng.
  - + Hệ thống check lại thông tin nếu như có sai sót thì hệ thống sẽ yêu cầu user nhập lại những thông tin và xác thực lại một lần nữa. Nếu các thông tin đó đã đầy đủ thì hệ thống sẽ tiến hành ghi nhận lại thông tin truyện đó cho user
  - + Khi người dùng muốn chỉnh sửa thông tin thì truy cập vào nút viết tại đây sẽ hiển thị thêm nút truyện của tôi, khi người dùng ấn vào nút này thì giao diện sẽ chuyển đến phần quản lý truyện tại đây người dùng có thể thêm sửa xóa truyện cũng như là chương truyện
  - + Truyện đã đăng lên thì cần phải chờ admin kiểm duyệt và gửi phản hồi thông báo vượt qua kiểm duyệt thì sẽ được công khai.

### **Use case quản lý chương truyện**

Mô tả chi tiết:

- Đối tượng sử dụng (user): Người dùng.
- Use case này mô tả chức năng viết truyện.
- Các bước thực hiện:
  - + Sau khi user đã tạo truyện thì sẽ ấn vào nút thêm chương để bắt đầu viết tên chương truyện và nội dung của chương.
  - + Khi đã viết xong thì người dùng phải ấn lưu để nội dung chương được lưu vào kho dữ liệu truyện để chờ admin kiểm duyệt.

### **Use case đăng nhập**

Mô tả chi tiết:

- Đối tượng sử dụng (user): Người dùng.
- Use case này mô tả chức năng đăng nhập.
- Các bước thực hiện:
  - + Hệ thống yêu cầu actor cung cấp thông tin đăng nhập gồm tên đăng nhập và mật khẩu.
  - + Actor nhập xong thông tin đăng nhập và click nút đăng nhập.
  - + Hệ thống check lại thông tin đăng nhập và thông báo thành công/thất bại cho actor. Nếu đăng nhập thất bại, hệ thống sẽ hiện thông báo cho người dùng và yêu cầu đăng nhập lại.
  - + Khi user chưa có tài khoản và muốn đăng ký mới thì user sẽ nhấn vào nút chuyển sang form đăng ký tài khoản mới. Khi đó hệ thống yêu cầu actor nhập các thông tin cần thiết và tiến hành xác minh xem có trùng với user trước đó hay không, nếu mọi thứ đều đã được kiểm tra xong thì hệ thống sẽ tiến hành đăng ký tài khoản mới và trực tiếp đăng nhập cho người dùng.

### **Use case lọc truyện theo thể loại.**

Mô tả chi tiết:

- Đối tượng sử dụng (user): Người dùng.
- Use case này mô tả chức năng tìm kiếm truyện theo thể loại.
- Các bước thực hiện:

- + Khi người dùng ấn vào nút thể loại thì sẽ hiển thị một danh sách thể loại truyện và khi người dùng ấn vào một thể loại thì dao diện sẽ trả về một danh sách thể loại truyện đó.

### **Use case tìm kiếm truyện theo tên truyện**

Mô tả chi tiết:

- Đối tượng sử dụng (user) bao gồm: Người dùng.
- Use case này mô tả chức năng tìm truyện theo tên truyện.
- Các bước thực hiện:
  - + Khi người dùng muốn tìm kiếm truyện thì sẽ nhập tên truyện và ấn tìm kiếm. Trang web sẽ trả về kết quả là một danh sách truyện trong tên có chứa các từ mà user nhập.

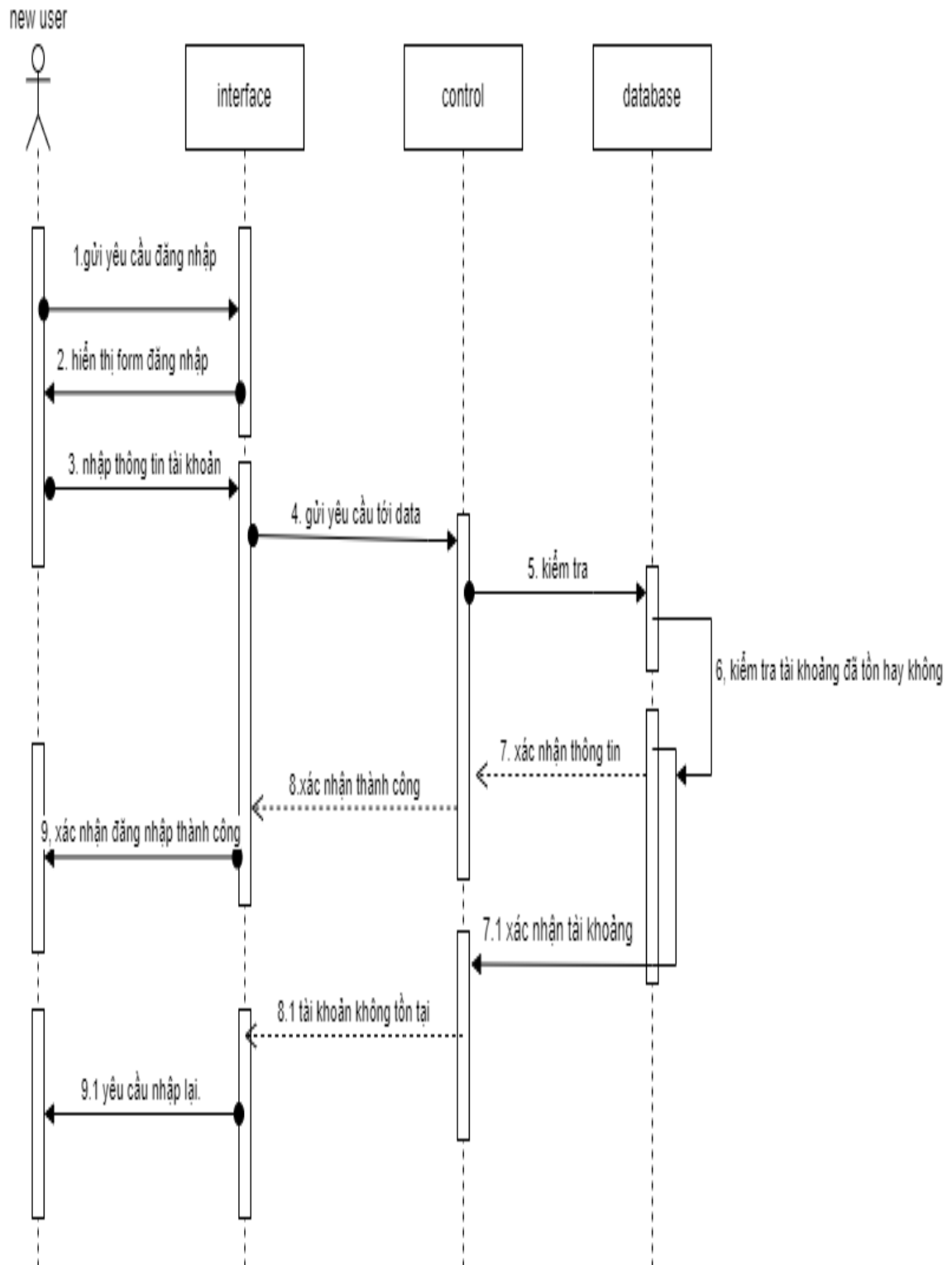
### **Use case kiểm duyệt truyện**

Mô tả chi tiết:

- Đối tượng sử dụng (admin) bao gồm: Admin kiểm duyệt.
- Use case này mô tả chức năng kiểm duyệt truyện.
- Các bước thực hiện:
  - + Khi admin đã đăng nhập vào trang admin kiểm duyệt thì có thể xem nội dung truyện trong danh sách truyện có thể thấy thông tin tác giả.
  - + Admin có thể xem phần kết quả kiểm duyệt tự động từ đó có thể đưa ra quyết định cho phép công khai truyện hay không, nếu truyện hợp lệ thì admin ấn vào nút công khai.
  - + Khi truyện vi phạm kiểm duyệt thì admin sẽ không công khai và gửi phản hồi kết quả kiểm duyệt về user bằng email.

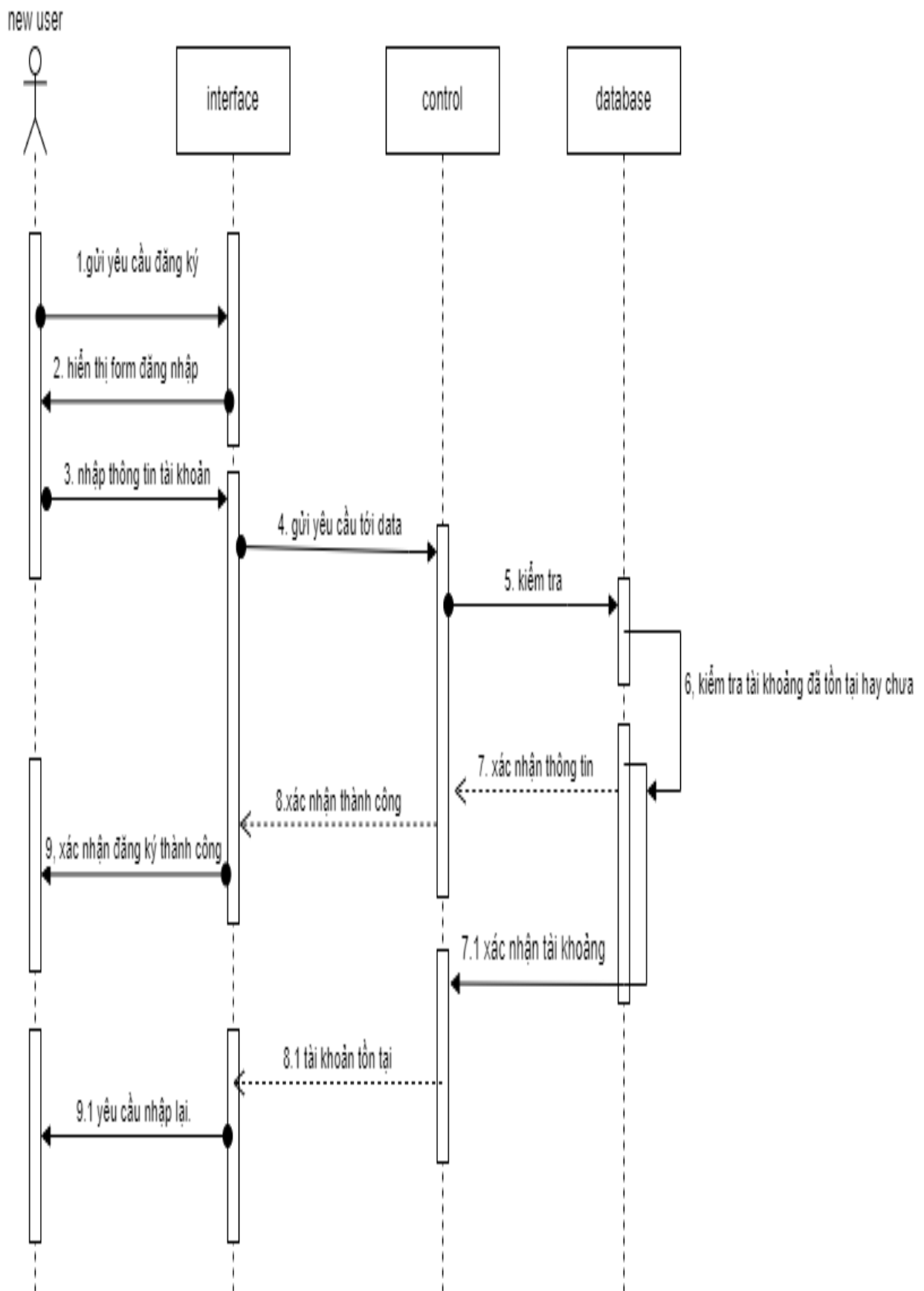
## **2.4. Biểu đồ tuần tự.**

### 2.4.1. Biểu đồ tuần tự chức năng đăng nhập.



Hình 12 Biểu đồ tuần tự chức năng đăng nhập

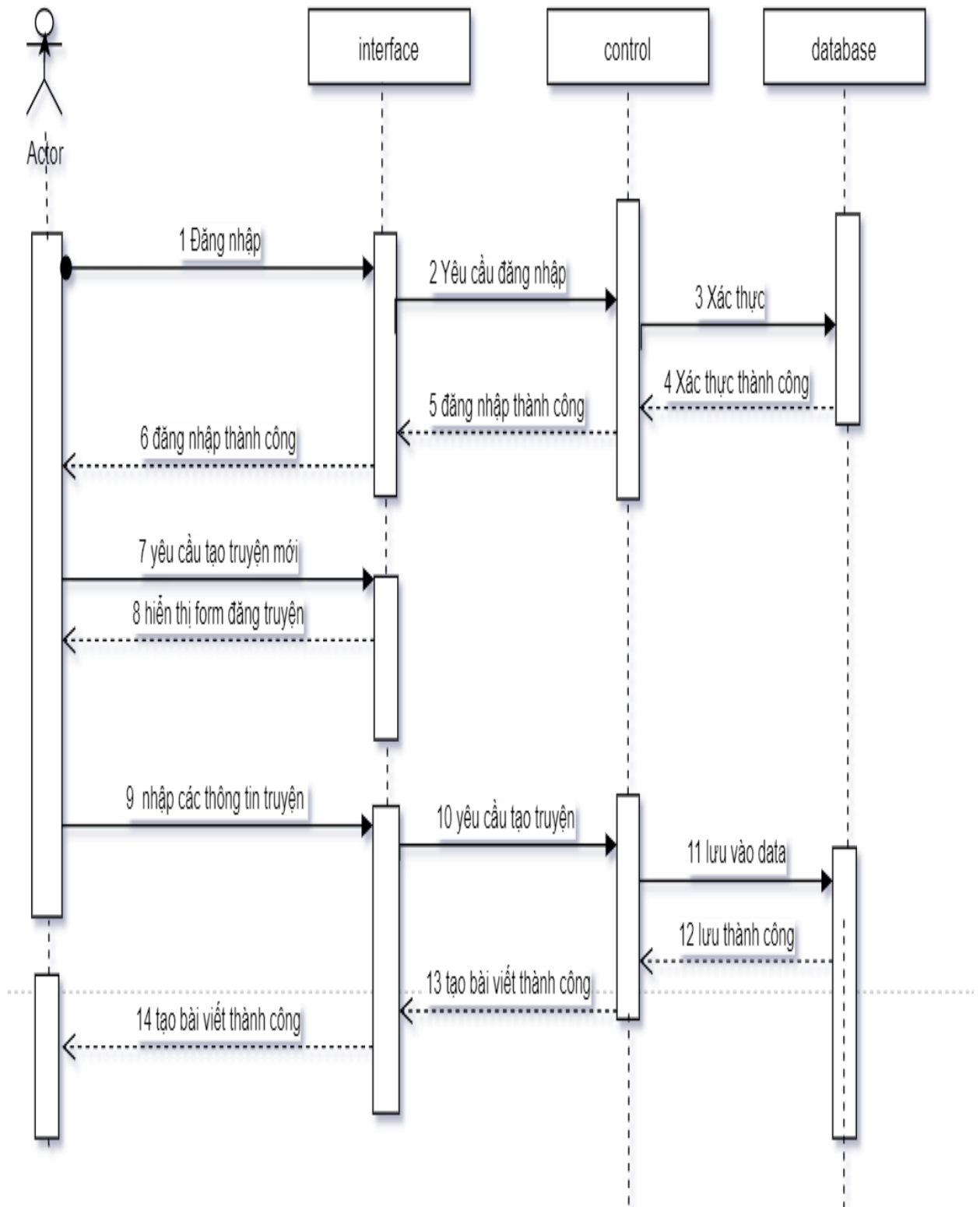
#### 2.4.2. Biểu đồ tuần tự chức năng đăng ký tài khoản mới.



Hình 13 Biểu đồ tuần tự chức năng đăng ký tài khoản mới

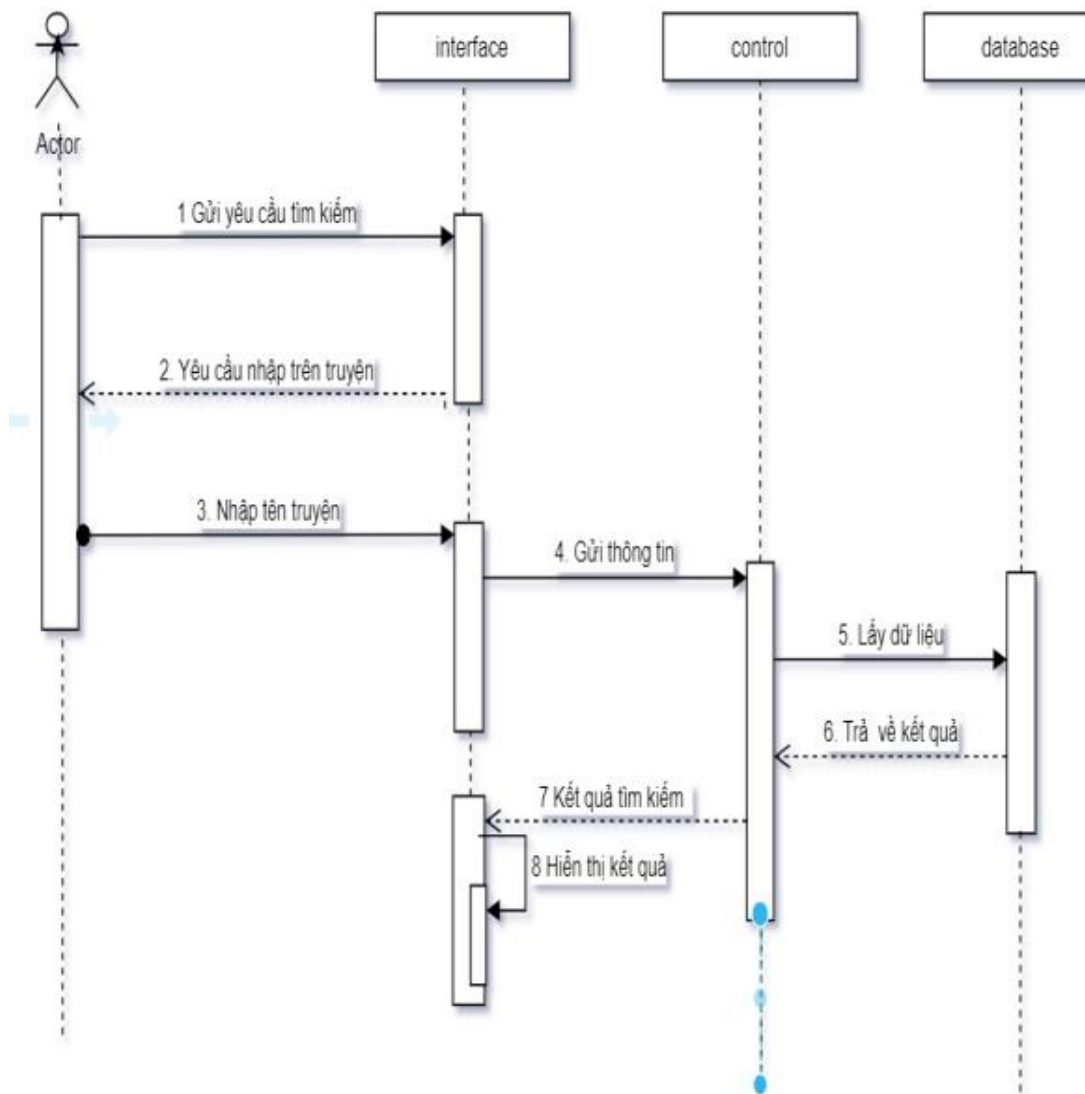


### 2.4.3. Biểu đồ tuần tự chức năng tạo truyện mới.



Hình 14 Biểu đồ tuần tự chức năng tạo truyện

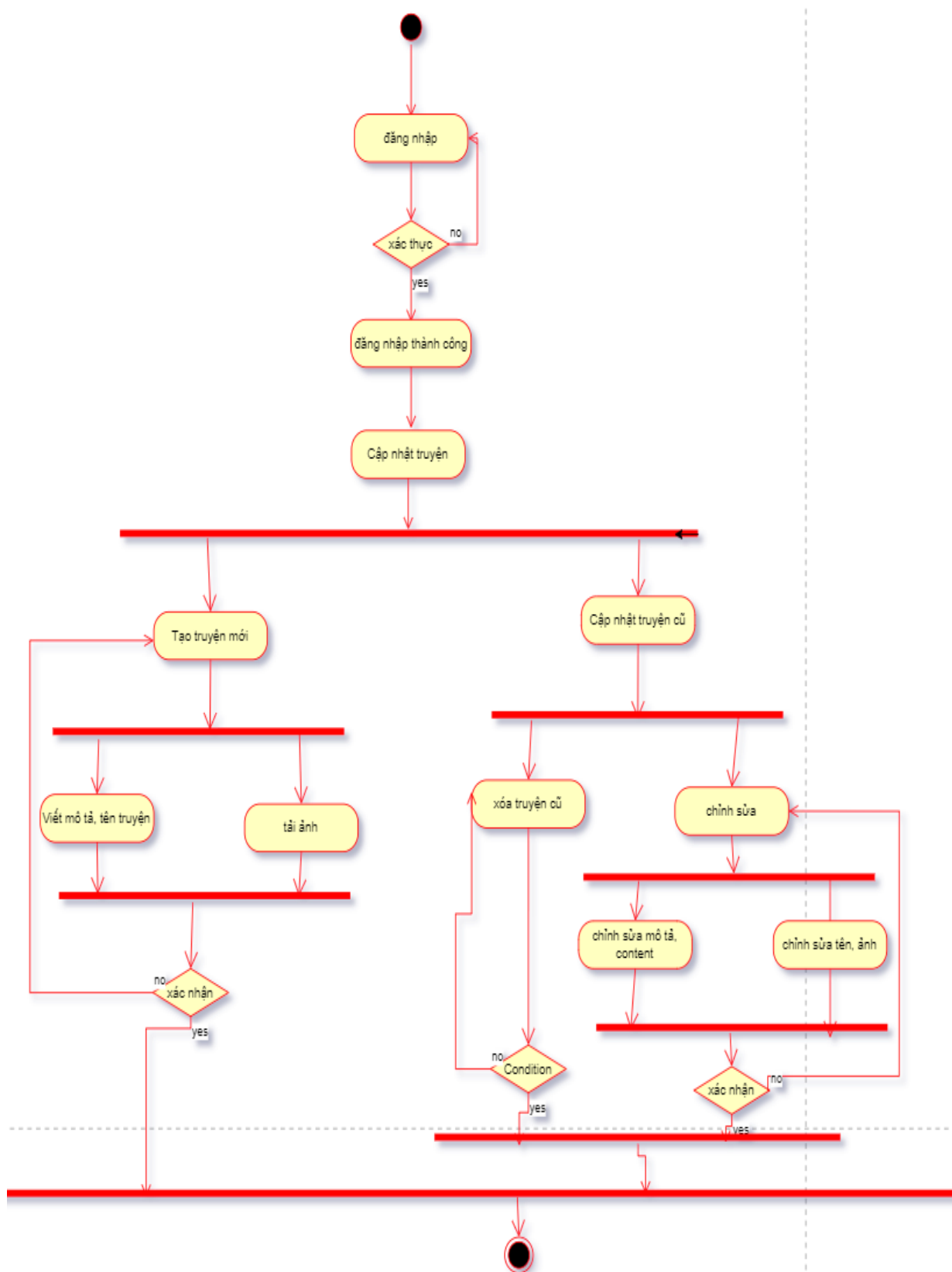
#### 2.4.4. Biểu đồ tuần tự chức năng tìm kiếm theo tên truyện



Hình 15 Biểu đồ tuần tự chức năng tìm kiếm theo tên truyện

#### 2.5. Biểu đồ hoạt động.

##### 2.5.1. Biểu đồ hoạt động chức năng cập nhật truyện.



Hình 16 Hoạt động chức năng cập nhật truyện

## 2.6. Cách tổ chức dữ liệu và thiết kế chương trình.

### 2.6.1. Bảng user

Tên thuộc tính	Kiểu dữ liệu	null	Chú thích	Khóa
Id	bigInt	No	Id	Khóa
Username	Varchar	Yes	Tên đăng nhập	
Email	Varchar	No	Email	
Password	Varchar	No	Mật khẩu	

### 2.6.2. Bảng story

Tên thuộc tính	Kiểu dữ liệu	null		Chú thích	Khóa chính
Id	bigInt	No		Id truyện	Khóa chính
Name	Varchar	No		Tên truyện	
Image	Varchar	No		Chứa đường link ảnh	
Description	Varchar	No		Mô tả truyện	
isPublish	Boolean	No		Trạng thái	
Note	LongText	Yes		Kết quả kiểm duyệt	
Type	Integer	No		Thể loại	
Author_id	bigInt	No		Người đăng tác giả	Khóa ngoại

### 2.6.3. Bảng chapterStory

Tên thuộc tính	Kiểu dữ liệu	null	Chú thích	Khóa chính
Id	bigInt	No	Id chương truyện	Khóa chính
Name	Varchar	No	Tên chương truyện	
isPublish	Boolean	No	Trạng thái	
Content	Varchar	No	Nội dung chương	
Note	LongText	Yes	Kết quả kiểm duyệt	
Story_id	bigInt	No	Id truyện	Khóa ngoại

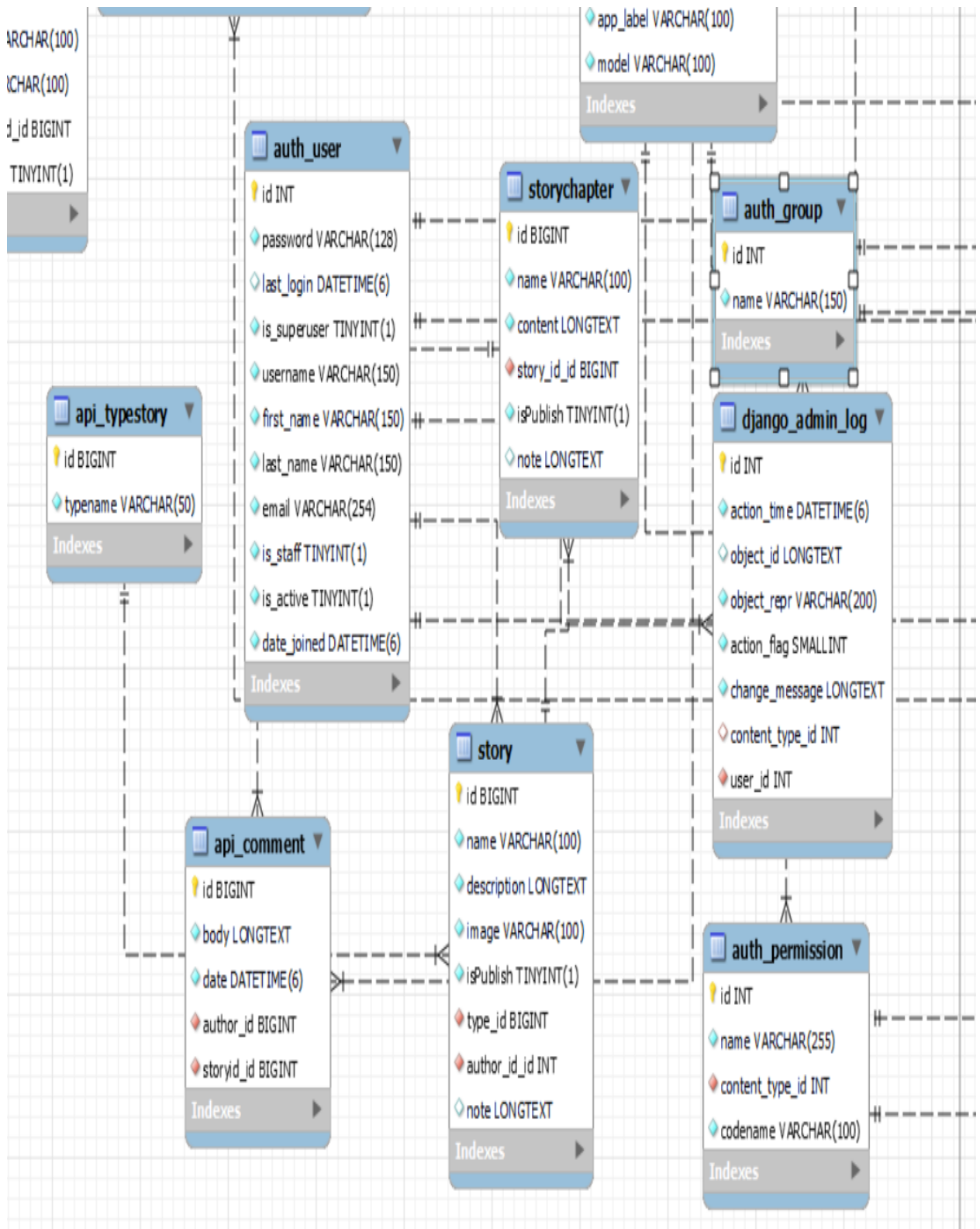
### 2.6.4. Bảng typestory

Tên thuộc tính	Kiểu dữ liệu	null	Chú thích	Khóa chính
Id	bigInt	No	Id bài viết	Khóa chính
Typename	Varchar	No	Thể loại truyện	

### 2.6.5. Bảng comments

Tên thuộc tính	Kiểu dữ liệu	null	Chú thích	Khóa chính
Id	bigInt	No	Id bình luận	Khóa chính
Body	LongText	No	Nội dung	
DateCreate	datetime	No	Thời điểm bình luận	
UserId	bigInt	No	Người bình luận	
StoryId	bigInt	No	Id Truyện	

## 2.7. Mô hình database diagram.



Hình 17 Mô hình database diagram

## CHƯƠNG 3 CÀI ĐẶT CHƯƠNG TRÌNH.

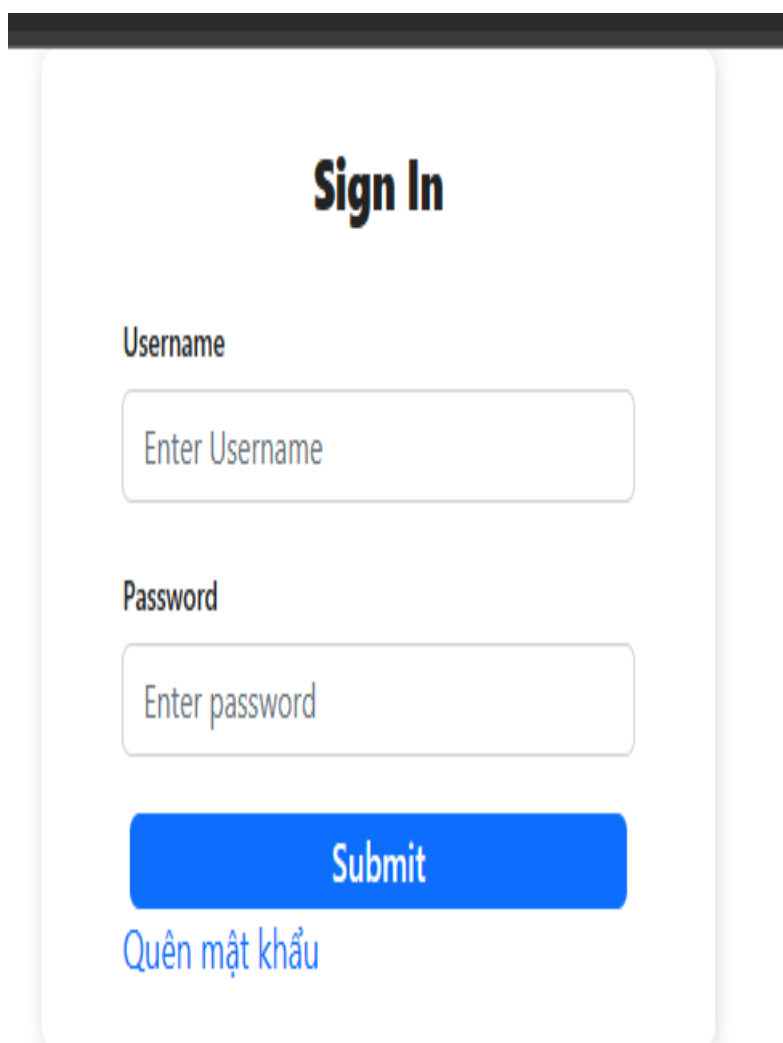
### 3.1. Các chức năng của chương trình.

Với giao diện trực quan, dễ hiểu, ngắn gọn cung cấp giao diện tùy biến cao cho người sử dụng, đặc biệt là người quản trị website, trong giao diện này cung cấp các template có sẵn được viết trên ngôn ngữ html mới nhất kết hợp với CSS tạo ra một giao diện đơn giản, dễ nắm bắt và ổn định.

Các chức năng có thể kể đến như; đăng nhập, đăng ký tài khoản mới; tạo truyện, viết chương truyện, quản lý danh sách truyện và đọc truyện

#### 3.1.1. Giao diện đăng nhập.

Tại trang giao diện đăng nhập, người dùng sẽ phải nhập username và password để có thể truy cập vào được trang viết truyện của mình.



Hình 18 Giao diện đăng nhập

### 3.1.2. Giao diện đăng ký.

Khi người dùng muốn đăng ký tài khoản mới thì cần phải nhập đầy đủ các thông tin cần thiết theo yêu cầu trong form bên dưới để có thể tạo một tài khoản mới cho mình với các thông tin về bản thân. . Khi người dùng nhập sai sẽ hiện lỗi thông báo. Khi đăng ký thành công sẽ vào thẳng trang chủ của phần mềm.

---

## Sign In

Username

Enter Username

Your email

Enter Your email

Password

Enter password

Submit

*Hình 19 Giao diện đăng ký*

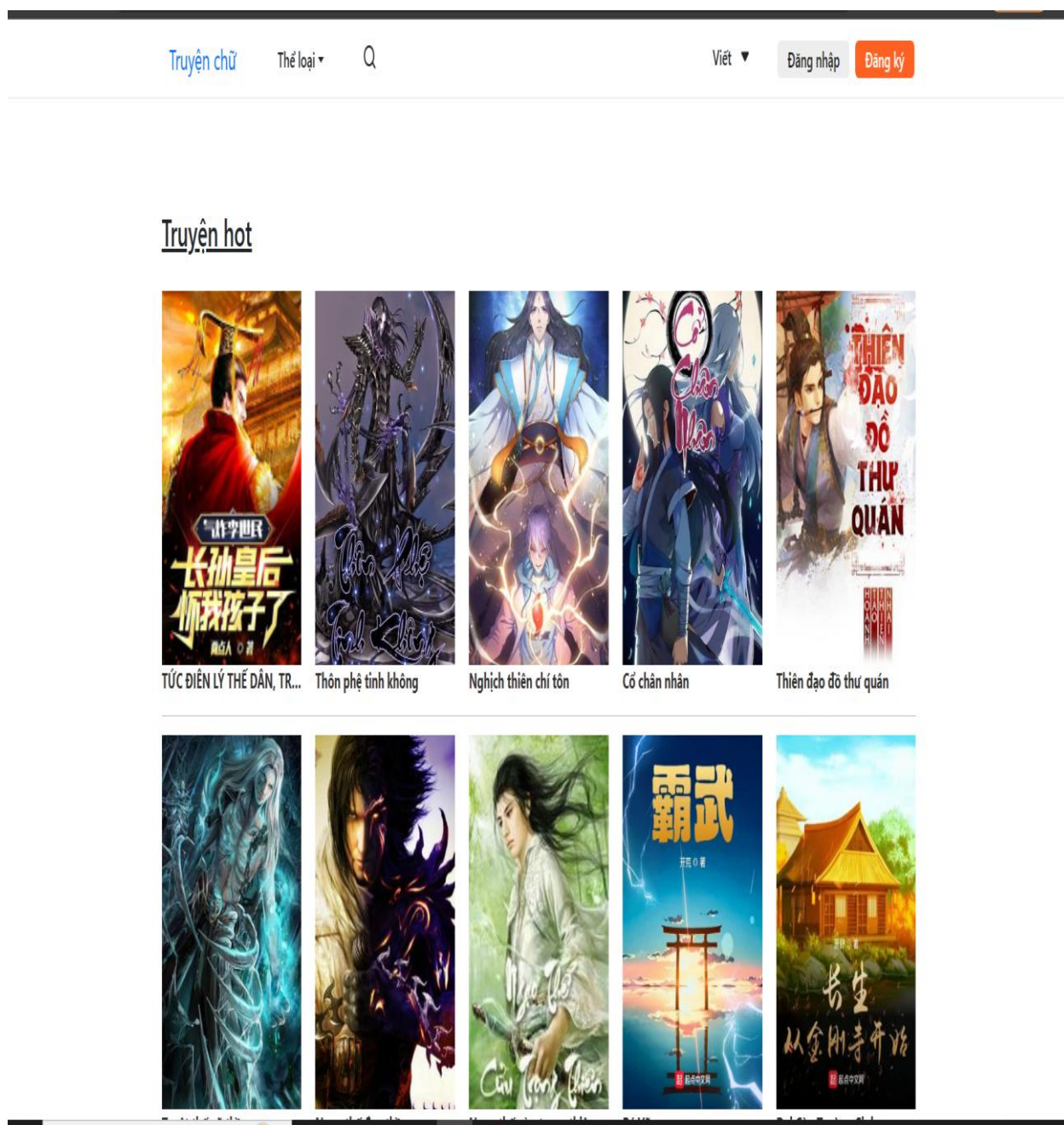


### 3.1.3. Giao diện trang chủ.

Tại giao diện trang chính người dùng có thể thấy được trên cùng là logo của trang viết truyện chữ, có thanh tìm kiếm để người đọc có thể tìm kiếm truyện, bên cạnh đó để đọc truyện thì người dùng không cần nhất định phải đăng nhập hay đăng ký

Tiếp theo theo là hai nút đăng nhập và đăng ký dành cho những ai đã có tài khoản hoặc chưa có tài khoản muốn trở thành những nhà viết truyện thú vị.

Phần phía dưới chính là danh sách các truyện nổi bật được nhiều người đón đọc để cho người đọc có thể tham khảo.



Hình 20 Giao diện trang chủ

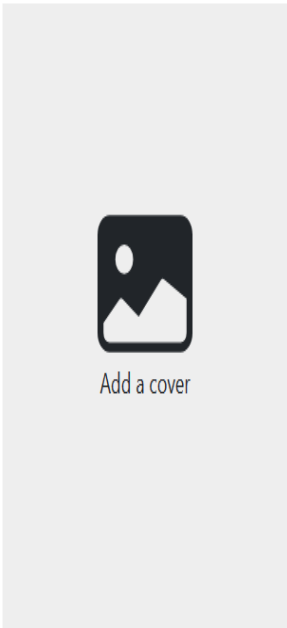
### 3.1.4. Giao diện viết thông tin truyện.

Ở giao diện này yêu cầu người dùng phải điền đầy đủ thông tin bao gồm tên truyện, mô tả truyện, hình ảnh đại diện cho truyện và đặc biệt là chọn thể loại sau khi hoàn tất thì hãy ấn nút tạo truyện để thông tin được lưu vào kho truyện để admin có thể đọc và kiểm duyệt thông tin truyện. Nếu người dùng muốn viết chương truyện thì chỉ cần nhấp vào nút skip để đến trang viết truyện và bắt đầu viết chương đầu tiên.

← Thêm thông tin truyện

Skip

Truyện chữ



Chi tiết truyện

Tên truyện

Mô tả

Thể loại

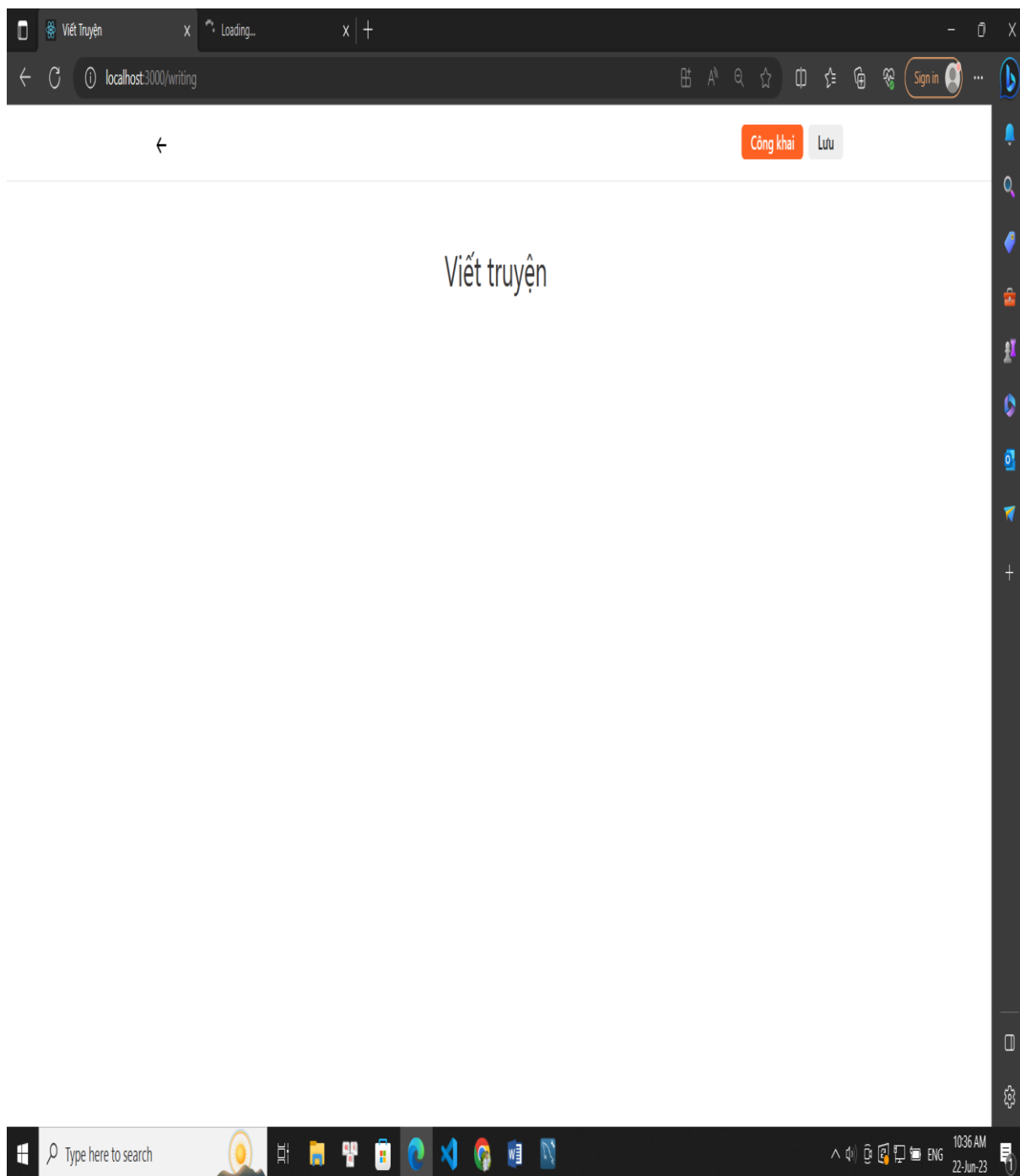
Chọn thể loại

Tạo truyện

Hình 21 Giao diện viết thông tin truyện

### 3.1.5. Giao diện viết truyện.

Đây là trang viết truyện rất đơn giản dễ sử dụng vì ở đây chỉ có viết và lưu truyện. Bắt đầu vào trang thì ta phải viết tên chương truyện vì đây là bắt buộc, mặc dù nội dung thì có thể viết tự do nhưng nếu copy từ truyện của user khác hoặc có chứa các từ nhạy cảm thô tục thì ngay sau khi ấn nút lưu trang admin sẽ tự động kiểm duyệt nội dung.

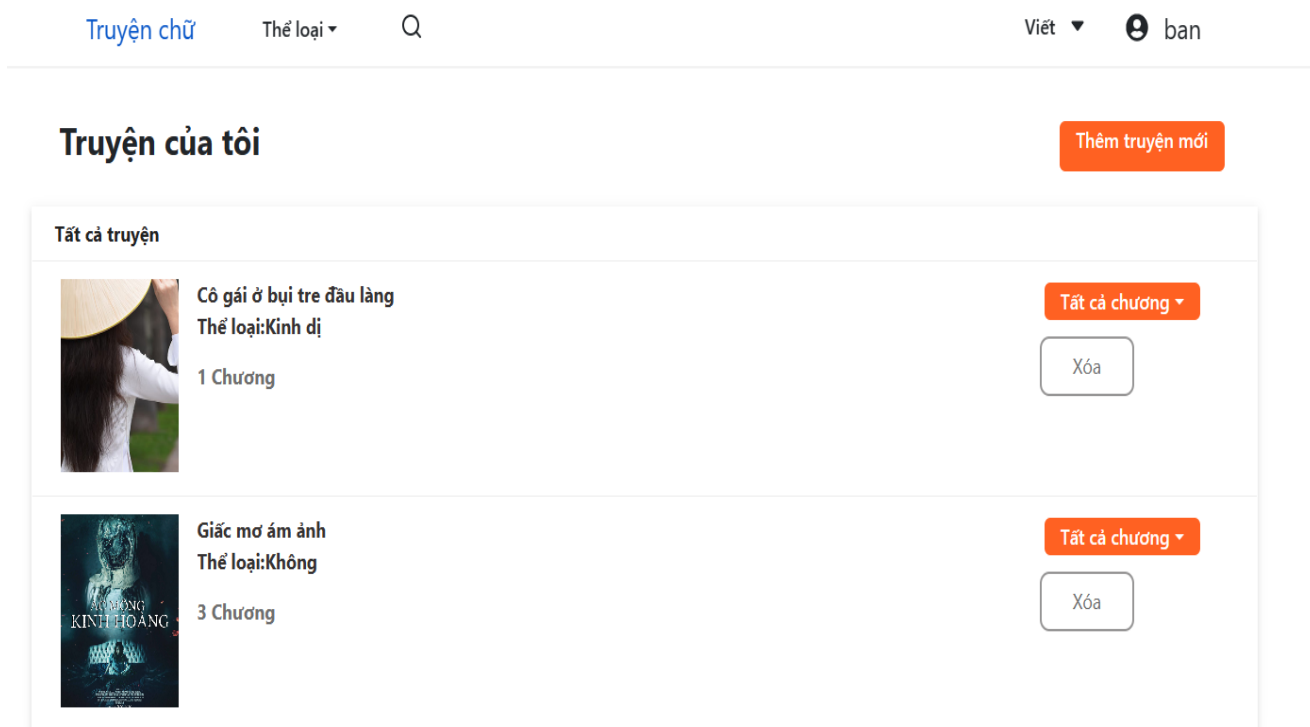


Hình 22 Giao diện viết chương truyện

### 3.1.6. Giao diện quản lý truyện cá nhân.

Giao diện quản lý truyện cá nhân sẽ cho ta thấy tất cả truyện mà bản thân đã tạo, bên trong mỗi truyện có số lượng chương, tên truyện thể loại truyện. Và để có thể cập nhật chương truyện ta chỉ cần chọn vào chương mà ta muốn khi ta ấn vào tất cả chương sau đó web sẽ đưa về trang viết chương truyện với các nội dung chương đã có sẵn.

Phần thêm chương để tạo thêm chương truyện mới và nút xóa để chọn xóa một truyện mà ta muốn.



Hình 23 Giao diện quản lý truyện cá nhân.

### 3.1.7. Giao diện trang đọc thông tin truyện.

Ở trang đọc thông tin truyện đọc giả có thể biết được một số thông tin của tác giả, số lượng chương truyện đã có và có thể hiểu sơ lược về nội dung của truyện qua phần mô tả nằm dưới mục thể loại



## Cô gái ở bụi tre đầu làng

Tác giả: ban

Thể loại: Kinh dị

Số chương: 1

Khi tôi còn nhỏ người dân trong làng thường dặn là không được ra bụi tre trước làng sau 10 giờ tối. Trong một lần qua nhà người bạn chơi vì mãi chơi đùa nên quên mất thời gian, lúc ấy cũng là lúc 10 giờ tối tôi đã phạm sai lầm khi một mình đi qua bụi tre và câu chuyện kinh hoàng đó vẫn ám ảnh tôi tới bây giờ

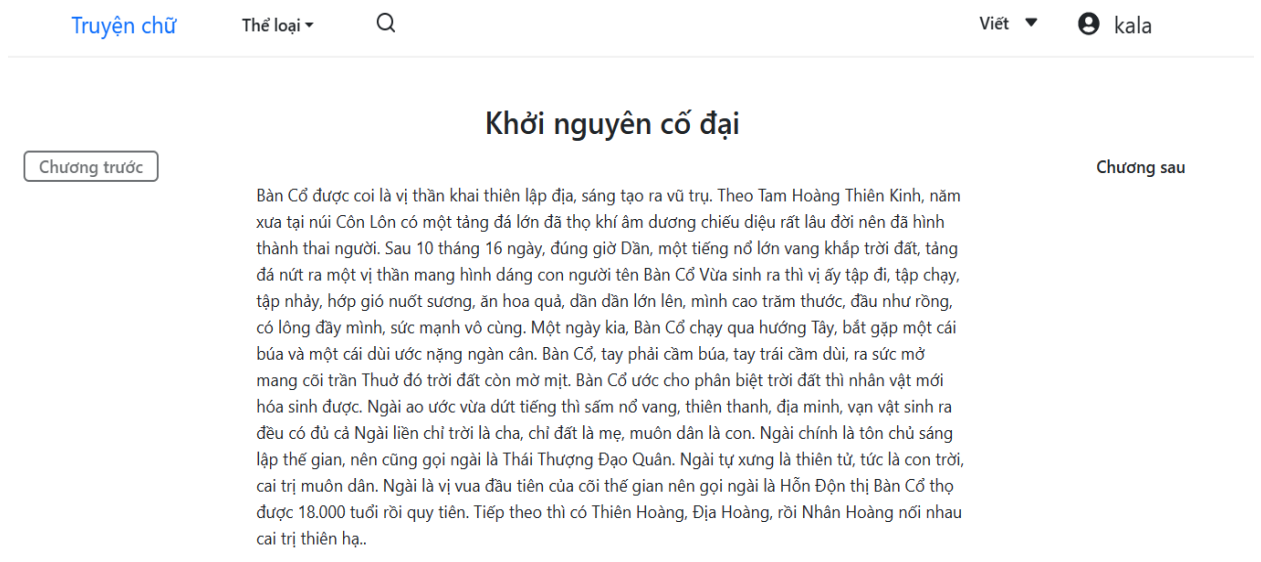
### Danh sách chương

19 Lời kể từ ông nội

Hình 24 Giao diện xem thông tin truyện.

### 3.1.8. Giao diện đọc truyện.

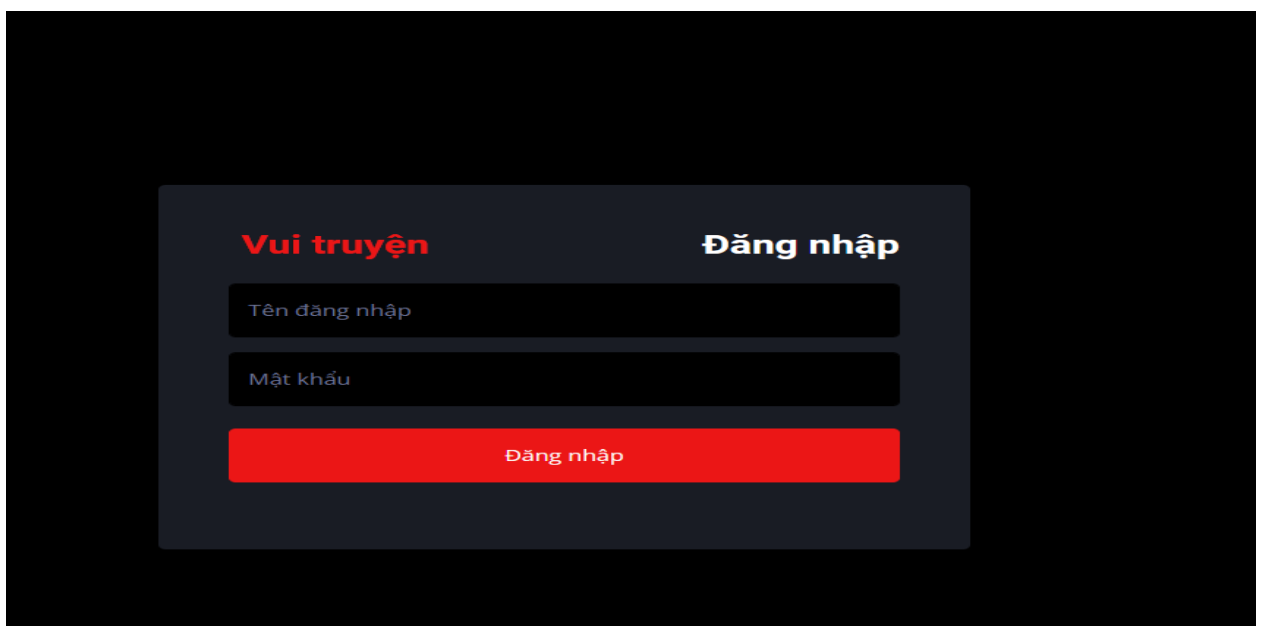
Trong giao diện đọc truyện này chúng ta có thể đọc được nội dung tập truyện, có 2 nút chuyển chương để người đọc dễ dàng chuyển chương khi đọc xong một tập



Hình 25 Giao diện đọc truyện

### 3.1.9. Giao diện trang admin.

Trước khi vào trang quản lý chính của trang admin kiểm duyệt ta cần phải đăng nhập vào trang login của admin như này



Hình 26 Giao diện trang đăng nhập admin

Tổng số user

Xem tất cả

16

Tổng số truyện

Xem tất cả

4

Recent Salse

Show All

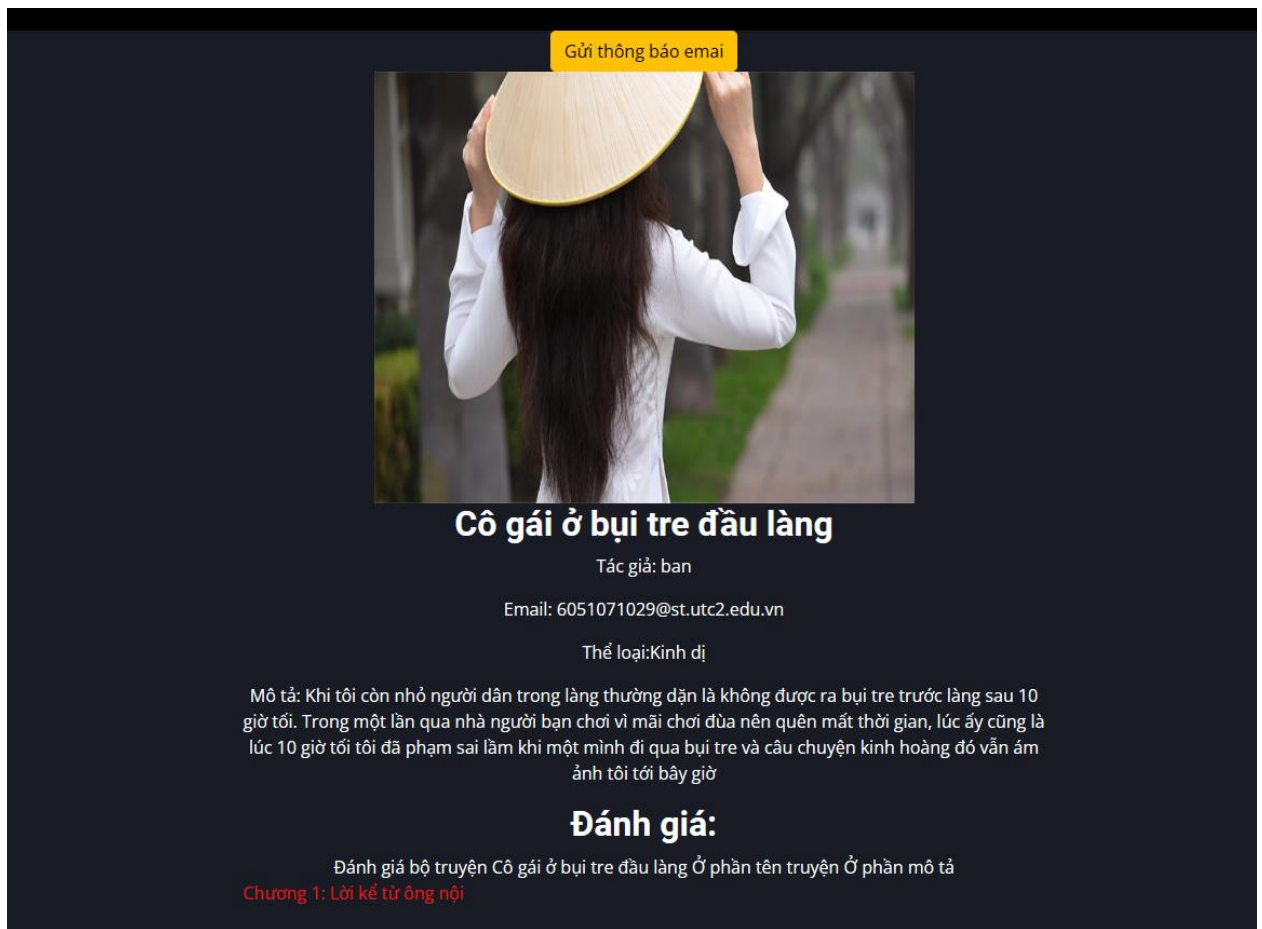
Tên truyện	Thể loại	Tác giả	Số chương	Trạng thái		
Cô gái ở bụi tre đầu làng	Kinh dị	ban	1	Chưa công khai	Chi tiết	Công khai
Giấc mơ ám ảnh	Không	ban	3	Chưa công khai	Chi tiết	Công khai
Phá vỡ chur thiên	Huyền Huyền	kala	6	Chưa công khai	Chi tiết	Công khai
băng	Kiểm Hiệp	hâm	2	Chưa công khai	Chi tiết	Công khai

*Hình 27 Giao diện trang kiểm duyệt của admin*

Trong giao diện này admin có thể xem được tất cả truyện đã có trong kho truyện biết được số lượng truyện chữ cũng như thông tin của truyện. Ngoài ra ở phần trạng thái admin kiểm duyệt có thể ấn nút để chuyển đổi trạng thái công khai và không công khai của truyện nếu trạng thái công khai thì những độc giả có thể đọc được truyện đó

### 3.1.10. Giao diện hiển thị kiểm duyệt.

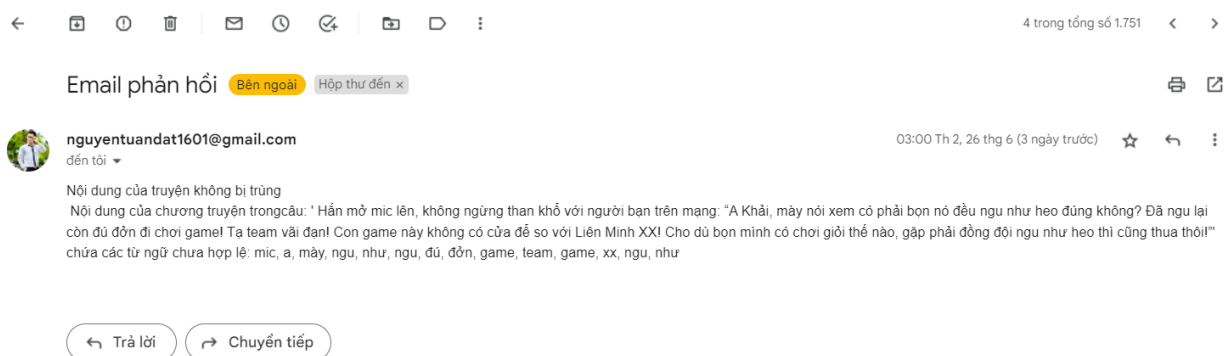
Đây là nơi mà admin có thể xem thông tin truyện thông tin tác giả và nội dung của phần note trả về là kết quả của model kiểm duyệt tất cả nội dung có bên trong. Nút gửi thông báo email sẽ phản hồi kết quả cho người sáng tác biết để từ đó đưa ra phương án chỉnh sửa phù hợp để admin có thể duyệt truyện



Hình 28 Giao diện kiểm duyệt thông tin truyện

### 3.1.11. Gửi mail phản hồi

Admin sau khi xem nội dung đã được kiểm duyệt thì có thể gửi phản hồi về gmail cho tác giả để có thể sớm cập nhật và chỉnh sửa.



Hình 29 Phản hồi mail cho tác giả

## KẾT LUẬN VÀ KIẾN NGHỊ

### Kết quả đạt được

Trong suốt khoảng thời gian nghiên cứu và thực hiện đề tài, em đã cố gắng làm hết khả năng của mình và vì thời gian làm đồ án có hạn nên kết quả đạt được vẫn còn nhiều



hạn chế nhưng em cũng học hỏi được khá nhiều kiến thức mới mẻ cũng như chuyên sâu về thuật toán, lập trình... và những việc em đã đạt được như sau: Trình bày các kiến thức cơ bản về công nghệ, ngôn ngữ, các công cụ hỗ trợ xây dựng trang web; phân tích, thiết kế và giải quyết được các vấn đề trong quá trình xây dựng; xây dựng được trang web viết truyện với những chức năng cơ bản mà một trang web đọc truyện cần có; sử dụng kết hợp được Reactjs, Django, MySQL, ứng dụng thư viện Scikit-learn để tạo được một model kiểm duyệt truyện chữ.

### **Tồn tại**

Trong quá trình làm đồ án, hiển nhiên sẽ có những lỗi chưa khắc phục hoàn toàn được cùng với thời gian có hạn nên sẽ có những chức năng chưa hoàn thiện: giao diện chương trình còn sơ xài; một số chức năng còn chưa thực sự hoàn thiện; số lượng chức năng còn ít; còn thiếu sót như thông báo ; thiếu chức năng thay đổi mật khẩu và reset mật khẩu.

### **Hướng phát triển**

Cần tối ưu hóa đầu vào để cho trang web được hoạt động một cách ổn định và nhanh chóng, cải thiện giao diện sao cho thân thiện với người dùng.

Cần xây dựng thêm chức năng bình luận và cập nhật lượt xem để người đọc có thể trải nghiệm tuyệt vời hơn.

Vì thời gian có hạn cho nên trong quá trình chạy demo sẽ phát sinh một số lỗi vì vậy cần phải chạy thực nghiệm thật nhiều để tìm ra những lỗi đó giúp cho việc trải nghiệm của người dùng được tốt nhất.

## TÀI LIỆU THAM KHẢO

1. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems 2nd Edition*, O'Reilly Media, 2019
2. Chris Albon, *Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning 1st Edition*, Wiley, 2018
3. Jon Duckett, *JavaScript and JQuery: Interactive Front-End Web Development*, John Wiley & Sons,, 2014
4. Sufyan bin Uzayr, *Mastering Django: A Beginner's Guide*, CRC Press, 2022
5. Irv Kalb, *Object-Oriented Python: Master OOP by Building Games and GUIs*, No Starch Press, 2022
6. [https://cfm.ehu.es/ricardo/docs/python/Learning\\_Python.pdf](https://cfm.ehu.es/ricardo/docs/python/Learning_Python.pdf)
7. <https://www.django-rest-framework.org/>
8. <https://dev.mysql.com/doc/>
9. <https://blog.logrocket.com/django-rest-framework-create-api/>
10. <https://medium.com/@cmukesh8688/scikit-learn-count-vectorizers-32b58dee0541>
11. <https://www.w3schools.com/>
12. <https://vietjack.com/javascript/>
13. <https://www.w3schools.com/MySQL/default.asp>
14. [https://warin.ca/ressources/books/2019\\_Book\\_AdvancedGuideToPython3Programm.pdf](https://warin.ca/ressources/books/2019_Book_AdvancedGuideToPython3Programm.pdf)
15. <https://viblo.asia/p/mongodb-la-gi-co-so-du-lieu-phi-quan-he-bJzKmgoPI9N>
16. <https://stackabuse.com/creating-a-rest-api-with-django-rest-framework/>
17. <https://scikit-learn.org/stable/tutorial/index.html>
18. <https://www.youtube.com/watch?v=AfYfvjP1hK8&t=1938s>
19. <https://www.youtube.com/watch?v=rxRYEXBmM88&list=PLJRGQoqpRwdczZl-LndYIQ6ymybtUqT5G>
20. <https://www.youtube.com/watch?v=-JJFQ9bkUbo>