**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

---------------------------------------

**Logo

Description automatically generated**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH KỸ THUẬT PHẦN MỀM**

**XÂY DỰNG ỨNG DỤNG WEBSITE THƯ VIỆN ĐIỆN TỬ SỬ DỤNG VUE.JS, ASP.NET VÀ MYSQL**

**GVHD : TS. Trần Tiến Dũng**

**Sinh viên : Phạm Văn Hải**

**Mã số sinh viên : 2019606843**

**Hà Nội – Năm 2023**

MỤC LỤC

[LỜI CẢM ƠN 5](#_Toc154352780)

[Chương 1. Tổng quan về đề tài 6](#_Toc154352781)

[1.1. Giới thiệu về đề tài 6](#_Toc154352782)

[1.2. Lí do chọn đề tài 6](#_Toc154352783)

[1.3. Đối tượng: 7](#_Toc154352784)

[1.4. Phạm vi: 7](#_Toc154352785)

[1.5. Giải pháp 8](#_Toc154352786)

[1.5.1. Visual Studio Code: 9](#_Toc154352787)

[1.5.2. APS.NET: 11](#_Toc154352788)

[1.5.3. Kiến trúc Clean Architecture: 13](#_Toc154352789)

[1.5.4. Hệ quản trị cơ sở dữ liệu MySQL: 16](#_Toc154352790)

[1.5.5. dbForge Studio 20](#_Toc154352791)

[1.5.6. Mock-up 21](#_Toc154352792)

[1.5.7. IBM Rational Rose 22](#_Toc154352793)

[1.5.8. Github 22](#_Toc154352794)

[Chương 2. Phân tích thiết kế 23](#_Toc154352795)

[2.1. Phân tích thiết kế hệ thống website bán hàng tạp hóa 23](#_Toc154352796)

[2.1.1. Biểu đồ ca sử dụng 23](#_Toc154352797)

[2.1.2. Xây dựng cấu trúc cơ sở dữ liệu 23](#_Toc154352798)

[2.1.3. Lựa chọn cơ sở dữ liệu 23](#_Toc154352799)

[2.1.4. Đặc tả use case 24](#_Toc154352800)

[2.1.5. Phân tích use case 30](#_Toc154352801)

[Chương 3. Cài đặt chương trình 48](#_Toc154352802)

[3.1. Khởi chay chương trình 48](#_Toc154352803)

[3.2. Website Thư Viện Điện Tử 48](#_Toc154352804)

[3.2.1. Màn hình đăng nhập 48](#_Toc154352805)

[3.2.2. Màn hình Tổng quan 49](#_Toc154352806)

[3.2.3. Màn hình danh sách sách 49](#_Toc154352807)

[3.2.4. Màn hình Thêm/Sửa sách điện tử 50](#_Toc154352808)

[3.2.5. Màn hình danh sách bạn đọc 50](#_Toc154352809)

[3.2.6. Màn hình cổng bạn đọc 51](#_Toc154352810)

[3.2.7. Màn hình xem chi tiêt sách 52](#_Toc154352811)

[3.2.8. Màn hình đọc sách 52](#_Toc154352812)

[KẾT LUẬN 53](#_Toc154352813)

[TÀI LIỆU THAM KHẢO 54](#_Toc154352814)

# DANH MỤC HÌNH ẢNH

[Bảng 1.1Biểu tượng VSCode 9](file:///F:\do-an-tn\PhamVanHai_2019606843.docx#_Toc154353034)

[Bảng 1.2 Biểu tượng ASP.NET 11](#_Toc154353035)

[Bảng 1.3 Cấu trúc thành phần ASP.NET 12](#_Toc154353036)

[Bảng 1.4 Cấu trúc Clean Architecture 14](#_Toc154353037)

[Bảng 1.5 MySql 16](#_Toc154353038)

[Bảng 1.6 Bảo mật cao của MySql 19](#_Toc154353039)

[Bảng 2.1 Cấu trúc cơ sở dữ liệu 24](#_Toc154353040)

[Bảng 3.1 Màn hình đăng nhập 48](#_Toc154353041)

[Bảng 3.2 Màn hình Tổng quan. 49](#_Toc154353042)

[Bảng 3.3 Màn hình danh sách sách điện tử. 49](#_Toc154353043)

[Bảng 3.4 Màn hình Thêm/Sửa sách điện tử. 50](#_Toc154353044)

[Bảng 3.5 Màn hình danh sách bạn đọc. 50](#_Toc154353045)

[Bảng 3.6 Màn hình cổng bạn đọc. 51](#_Toc154353046)

[Bảng 3.7 Xem chi tiết sách 52](#_Toc154353047)

[Bảng 3.8 Màn hình đọc sách 52](#_Toc154353048)

# LỜI CẢM ƠN

*Là một sinh viên trường đại học Công nghiệp Hà Nội, đồ án tốt nghiệp là minh chứng cho những kiến thức đã có sau bốn năm học tập. Trong quá trình hoàn thành đồ án tốt nghiệp, ngoài sự cố gắng của bản thân em cũng nhận được sự hướng dẫn tận tình của thầy cô cùng các anh chị bạn bè. Qua đây, em xin chân thành cảm ơn khoa Công nghệ thông tin, trường đại học Công nghiệp Hà Nội đã trang bị kiến thức cho em trong suốt quá trình học tập và tạo cơ hội để em được làm đồ án tốt nghiệp. Đặc biệt, em cảm ơn tới thầy TS. Trần Tiến Dũng đã luôn giúp đỡ, hướng dẫn, chỉ bảo tận tình để em hoàn thành tốt đồ án của mình. Em đã cố gắng hoàn thành đồ án tốt nghiệp nhưng vẫn rất mong nhận được sự đóng góp ý của thầy cô và các bạn để đồ án của em được hoàn thiện hơn.*

*Em xin chân thành cảm ơn!*

# Tổng quan về đề tài

Xây dựng website Thư Viện Điện Tử sử dụng VueJs, ASP.NET và MySql

## Giới thiệu về đề tài

Chủ đề của đề tài là Xây dựng một website Thư Viện Điệm Tử được triển khai thông qua VueJs, ASP.NET và MySql. Với mục tiêu áp dụng những kiến thức về lập trình website, nghiên cứu về hệ thống quản lý sách thực hành xây dựng một trang web đọc sách. Qua đó biết được cách ứng dụng các framework xây dựng website vào thực tế, đồng thời rèn luyện kĩ năng phân tích thiết kế hệ thống và xây dựng website. Do sự bùng nổ của internet, mọi người có thể tìm kiếm thông tin ở bất cứ đâu, bất kỳ thời điểm nào, làm thay đổi thói quen đọc sách của phần lớn mọi người, thay vì phải đến thư viện truyền thống để tìm kiếm thông tin, thì giờ đây mọi người có thể đọc những quyển sách yêu thích của mình ở bất kỳ đâu chỉ bằng các thiết bị có kết nối internet.

## Lí do chọn đề tài

Trong vài năm trở lại đây, với sự phát triển của nền công nghiệp 4.0. Đặc biệt là thuật ngữ “Chuyển đổi số” được nhắc đến liên tục trong thời gian gần đây, các nềm tảng truyền thống dần dần được số hóa. Và tất nhiên việc số hóa các thư viện cũng không nằm ngoài cuộc đua đó. Một thư viện điện tử cung cấp một nguồn tài nguyên rất lớn và tiện lợi cho cộng đồng. Người đọc có thể dễ dàng truy cập sách, bài luận, và thông tin khác mọi nơi chỉ cần kết nối internet. Website thư viện điện tử giúp tổ chức và quản lý tài nguyên một cách hiệu quả hơn so với thư viện truyền thống. Bạn có thể dễ dàng cập nhật thông tin, thêm mới sách, và quản lý người dùng. Một trang web thư viện điện tử mở ra cơ hội cho mọi người có thể tiếp cận kiến thức và học tập mà không gặp rào cản về địa lý hay thời gian. Điều này làm tăng khả năng phổ cập giáo dục. Việc xây dựng website thư viện điện tử là cơ hội để bạn áp dụng và tích hợp các công nghệ mới như trí tuệ nhân tạo, học máy, và các công nghệ web tiên tiến để cải thiện trải nghiệm người dùng. Xây dựng một thư viện điện tử giúp bạn xây dựng và duy trì một cơ sở dữ liệu lớn với nhiều thông tin hữu ích. Điều này có thể hỗ trợ nghiên cứu, phân tích xu hướng đọc, và cung cấp thông tin quan trọng cho cộng đồng.

## Đối tượng:

Website Thư Viện Điện Tử hướng đến đối tượng những người yêu sách và muốn mở rộng kiến thức, thư viện điện tử là một cầu nối để họ có thể truy cập vào một thư viện lớn với nhiều thể loại sách khác nhau..

## Phạm vi:

* Phát biểu vấn đề: Với mục tiêu là số hóa các đầu sách để có thể lưu trữ và quản lý một cách lâu dài, giúp độc giả có thể tiếp cận với các đầu sách một cách dễ dàng
* Mục tiêu:
  + Xây dựng một website thư viện điện tử có giao diện thân thiện với người dùng, cung cấp đầy đủ tính năng quản lý sách điện tử, tìm kiếm và đọc sách.
  + Xây dựng website giúp quản lý thư viện có thể dễ dàng up các đầu sách để phục vụ bạn đọc
* Mô tả:
  + Website có giao diện của bạn đọc và giao diện của người quản trị. Bạn đọc có thể tìm kiếm sách, đọc sách. Người quản trị có thể quản lý thông tin về sách, thông kê tình đọc sách.
  + Website có khả năng có giao diện đọc bắt mắt thân thiện với bạn đọc.
* Lợi ích mang lại:
  + Tạo sự tiện dụng, nhanh chóng và thoải mái cho các bạn đọc khi sử dụng.
  + Tiết kiệm thời gian và chi phí khi tìm kiếm sách.
* Những kỹ năng cần thiết:
  + Kỹ năng thuyết trình: phục vụ việc trình bày về project, giúp mọi người hiểu quy trình thực hiện, các nguyên tắc khi xây dựng phần mềm.
  + Kỹ năng tổng hợp kiến thức: do bài tập lớn yêu cầu sử dụng đa dạng các kiến thức không chỉ trong phạm vi môn học mà còn các môn học khác, thậm chí những kỹ thuật hoàn toàn mới, do đó yêu cầu về khả năng tìm kiếm, tổng hợp kiến thức là rất quan trọng.
  + Kỹ năng sử dụng các phần mềm viết mã nguồn, xây dựng giao diện và biên dịch chương trình.

## Giải pháp

Như chúng ta đã biết cách mạng 4.0 đang điễn ra nhanh mạnh trong mọi lĩnh vực, việc này đòi hỏi các ngành nghề đều được áp dụng công nghệ để theo kịp với thời đại. Để theo kịp thời đại đó việc ứng dụng công nghệ phần mềm mã nguồn mở vào xây dựng sách phần mềm được ưu tiên hàng đầu bởi những ưu điểm mà nó mạng lại như: có một cộng đồng lớn các lập trình viên tham gia phát triển, cộng với hàng ngàn người sử dụng và phản hồi, được sử kiểm duyệt của chính đơn vị phát hành nên sẽ có tính bảo mật cao. Tốc độ cập nhật lỗi nhanh và được kiểm chứng bởi cộng đồng…. Website Thư Viện Điện Tử là một nền tảng danh cho trường học vả cho các thư viện, giúp quản lý và lưu trữ các file sách một cách hiệu quả, số hóa các đầu sách. Giúp cho đọc giả có thể dễ dàng tiếp cận với nguồn tri thức.

Công cụ, ngôn ngữ thực hiện đề tài: Vue.js, ASP.NET, MySql, Visual Studio Code, Visual Studio, Mockup, IBM Rational Rose, Github.

### Visual Studio Code:

#### Giới thiệu:

**Visual Studio Code** (VS Code hay VSC) là một trong những trình soạn thảo mã nguồn phổ biến nhất được sử dụng bởi các lập trình viên. Nhanh, nhẹ, hỗ trợ đa nền tảng, nhiều tính năng và là mã nguồn mở chính là những ưu điểm vượt trội khiến VS Code ngày càng được ứng dụng rộng rãi.

Bảng .Biểu tượng VSCode

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn.

#### Một số tính năng của Visual Studio:

* Hỗ trợ nhiều ngôn ngữ lập trình

Visual Studio Code hỗ trợ nhiều ngôn ngữ lập trình như C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, … Vì vậy, nó dễ dàng phát hiện và đưa ra thông báo nếu chương chương trình có lỗi.

* Hỗ trợ đa nền tảng

Các trình viết code thông thường chỉ được sử dụng hoặc cho Windows hoặc Linux hoặc Mac Systems. Nhưng Visual Studio Code có thể hoạt động tốt trên cả ba nền tảng trên.

* Cung cấp kho tiện ích mở rộng

Visual Studio hỗ trợ tải xuống tiện ích mở rộng. Điều này vẫn sẽ không làm giảm hiệu năng của phần mềm, bởi vì phần mở rộng này hoạt động như một chương trình độc lập.

* Kho lưu trữ an toàn

Đi kèm với sự phát triển của lập trình là nhu cầu về lưu trữ an toàn. Với Visual Studio Code có thể dễ dàng kết nối với Git hoặc bất kỳ kho lưu trữ hiện có nào.

* Hỗ trợ web

Visual Studio Code hỗ trợ nhiều ứng dụng web. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website.

* Hỗ trợ viết Code

Một số đoạn code có thể thay đổi chút ít để thuận tiện cho người dùng. Visual Studio Code sẽ đề xuất các tùy chọn thay thế nếu có.

* Hỗ trợ thiết bị đầu cuối

Visual Studio Code có tích hợp thiết bị đầu cuối, giúp không phải chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác.

* Màn hình đa nhiệm

Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục – mặc dù chúng không hề liên quan với nhau.

* Intellisense

Hầu hết các trình viết mã đều có tính năng nhắc mã Intellisense, nhưng ít chương trình nào chuyên nghiệp bằng Visual Studio Code. Nó có thể phát hiện nếu bất kỳ đoạn mã nào không đầy đủ.

* Hỗ trợ Git

Visual Studio Code hỗ trợ kéo hoặc sao chép mã trực tiếp từ GitHub. Mã này sau đó có thể được thay đổi và lưu lại trên phần mềm.

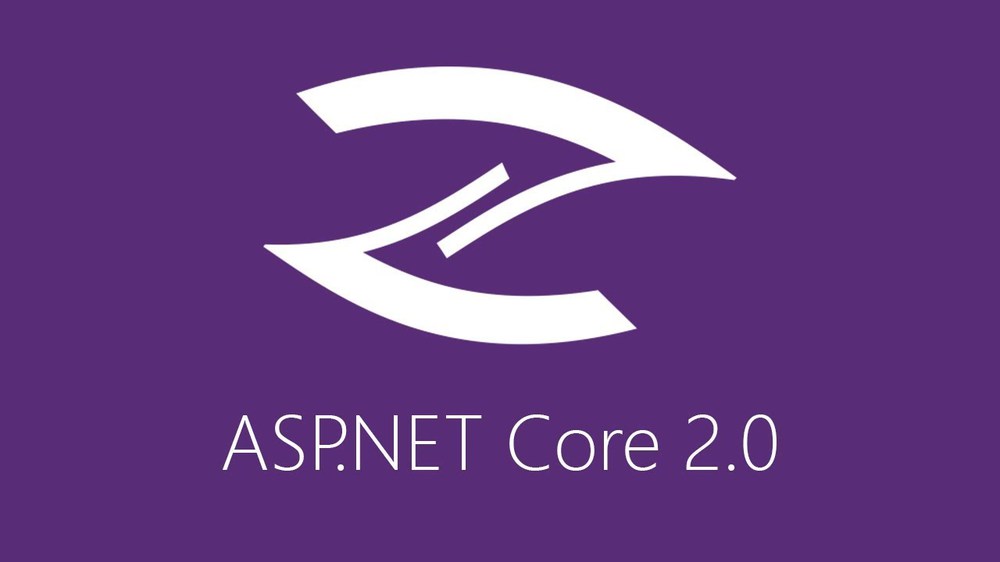
* *Bình luận*

Việc để lại nhận xét giúp người dùng dễ dàng nhớ công việc cần hoàn thành.

### **APS.NET:**

#### **Giới thiệu:**

**ASP.NET**là nền tảng ứng dụng web do Microsoft phát triển cho phép người dùng có thể thiết kế các trang web động, ứng dụng web. Về bản chất, ASP.NET là tập con của **.NET Framework**, sở hữu mã nguồn mở và kế thừa những tính năng, ưu điểm của ASP cổ điển.

[](https://bizflyportal.mediacdn.vn/bizflyportal/images/asp16176979457662.jpg)

Bảng . Biểu tượng ASP.NET

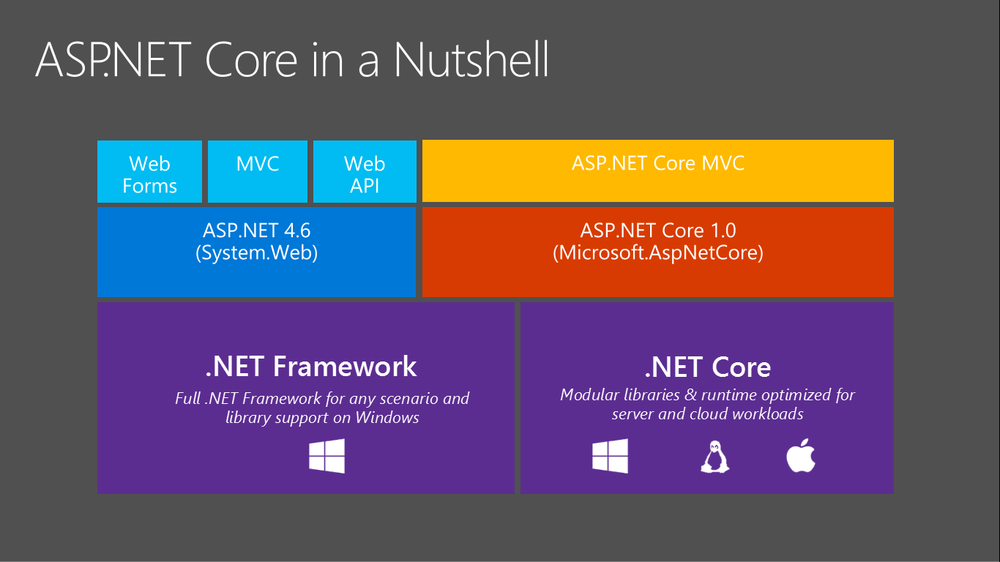
#### Asp.net là gì?

**Nền tảng ASP.NET** được xây dựng dựa trên CLR, cho phép nhà phát triển tạo lập, thực thi các mã bằng mọi ngôn ngữ .NET. Nền tảng này được tạo ra nhằm làm việc với HTTP, được tích hợp sẵn HTML, CSS và JavaScript, giúp hỗ trợ lập trình viên tạo các website động và ứng dụng web.

Không giống .NET,**ASP.NET** chỉ hỗ trợ thiết kế, tạo lập các dịch vụ web hoặc ứng dụng web. Đây cũng là nguyên nhân khiến nó được coi như một tập con của .NET Framework.

#### Cấu trúc thành phần của Asp.net :

Theo kết quả nghiên cứu mà các chuyên gia công bố, cấu trúc thành phần của **ASP.NET** bao gồm các yếu tố: Ngôn ngữ, thư viện và thời gian chạy CLR.

[](https://bizflyportal.mediacdn.vn/bizflyportal/images/asp16176981085235.png)

Bảng . Cấu trúc thành phần ASP.NET

Cấu trúc thành phần của Asp.net

* **Ngôn ngữ**: Là tập con của .NET Framework, ASP.NET sở hữu rất nhiều ngôn ngữ lập trình như C#, VB.net, PHP, JavaScript,... Trong đó C# và VB.net được sử dụng nhiều nhất trong phát triển ứng dụng web.
* **Thư viện:**.ASP. NET Framework kế thừa các lớp thư viện từ .NET. Thư viện này được dùng rất phổ biến trong phát triển ứng dụng web.
* T**hời gian chạy CLR:** Đây là một trong các cơ sở hạ tầng của phần đông các kiểu ngôn ngữ lập trình phổ thông. Trong các tình huống bình thường, CLR thực hiện các thao tác để xử lý các trường hợp đặc biệt và thu gom rác.

#### Đặc điểm cơ bản của Asp.net là gì?

Đặc điểm cơ bản của Asp.net: Trạng thái code rời, khả năng quản lý các trạng thái và bộ nhớ cache.

* **Trạng thái code rời:** ASP.Net tách rời các mã code và cả thiết kế để đảm bảo khả năng duy trì ứng dụng. Các file trong nền tảng này được tách rời thành nhiều tệp, giúp nhà phát triển có khả năng giữ cho ASPX.NET dễ dàng, đơn giản.
* **Khả năng quản lý các trạng thái**: ASP.Net có sẵn các phương tiện quản lý, kiểm soát trạng thái. Đặc trưng này giúp các ứng dụng, dịch vụ web ghi nhớ trạng thái nên thường được sử dụng trong thiết kế tính năng giỏ hàng trong các trang thương mại điện tử.
* **Bộ nhớ Cache:** ASP.Net thực hiện lưu trữ bộ nhớ Cache nhằm cải thiện hiệu suất làm việc của ứng dụng. Điều này hỗ trợ lưu trữ thông tin, hỗ trợ truy xuất, phản hồi trong thời gian ngắn, từ đó tăng hiệu suất làm việc của ứng dụng.

#### Vì sao nên sử dụng Asp.net?

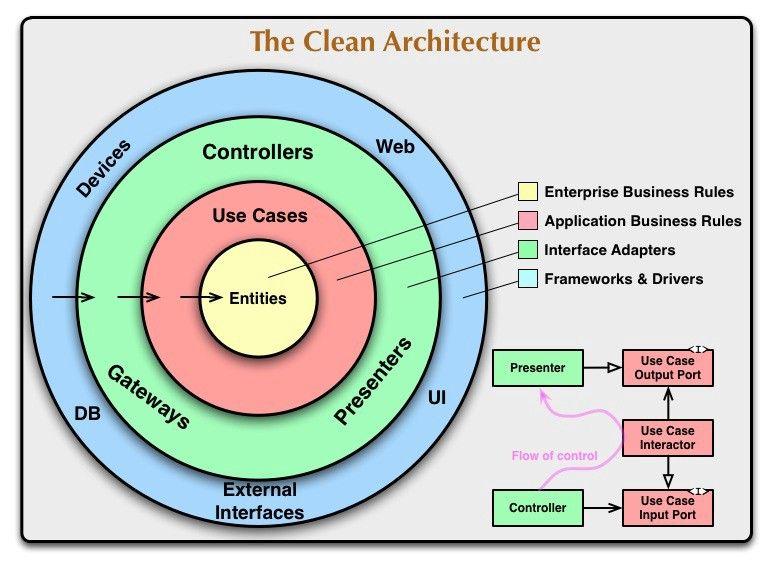
Có khá nhiều lý do khiến nền tảng này được sử dụng trong thiết kế ứng dụng hoặc phát triển website.

* **Tốc độ cao**: ASP.NET cho phép phát triển ứng dụng nhanh chóng và hiệu quả hơn hẳn so với việc sử dụng PHP. Ứng dụng thiết kế bằng nền tảng này được biên dịch và thực thi dưới sự hỗ trợ của kho thư viện lớn.
* **Chi phí thiết kế thấp**: Lập trình viên có thể viết ASP.NET trên các trình soạn thảo đơn giản miễn phí như Visual Studio. Điều này giúp tiết kiệm khá nhiều chi phí bởi bạn không cần mua các phần mềm xây dựng ứng dụng đắt tiền.
* **Được hỗ trợ nhiều khung, mã để tái sử dụng:** ASP.NET được viết bởi khá nhiều loại ngôn ngữ lập trình như C#, VB.net. Bởi vậy mà ứng dụng sở hữu khá nhiều khung, mã code sẵn. Bạn chỉ cần cân nhắc sử dụng mã phù hợp là ổn.
* **Không bị ‘đào thải’:** ASP.NET tuy là mã nguồn mở, cung cấp miễn phí song nó vẫn đang được phát triển không ngừng. Điều này đảm bảo các ứng dụng web của bạn sẽ không trở nên ‘lạc hậu’ sau vài năm sử dụng. Bạn hoàn toàn có thể nâng cấp nó.

### Kiến trúc Clean Architecture:

#### Giới thiệu:

**Clean Architecture** được xây dựng dựa trên tư tưởng "độc lập" kết hợp với các nguyên lý thiết kế hướng đối tượng(đại diện tiêu biểu là Dependency Inversion). Độc lập ở đây nghĩa là việc project không bị phụ thuộc vào framework và các công cụ sử dụng trong quá trình kiểm thử.



Bảng . Cấu trúc Clean Architecture

Kiến trúc của Clean Architecture chia thành 4 layer với một quy tắc phụ thuộc. Các layer bên trong không nên biết bất kỳ điều gì về các layer bên ngoài. Điều này có nghĩa là nó có quan hệ phụ thuộc nên "hướng" vào bên trong. Nhìn vào hình vẽ minh họa sau đây:

* **Entities**: là khái niệm dùng để mô tả các Business Logic. Đây là layer quan trọng nhất, là nơi bạn thực hiện giải quyết các vấn đề - mục đích khi xây dựng app. Layer này không chứa bất kỳ một framework nào, bạn có thể chạy nó mà không cần emulator. Nó giúp bạn dễ dàng test, maintain và develop phần business logic.
* **Use case** : chứa các rule liên quan trực tiếp tới ứng dụng cục bộ (application-specific business rules).
* **Interface Adapter** : tập hợp các adapter phục vụ quá trình tương tác với các công nghệ.
* **Framework and Drivers** : chứa các công cụ về cơ sở dữ liệu và các framework, thông thường bạn sẽ không phải lập trình nhiều ở tầng này. Tuy nhiên cần chắc chắn về mức ưu tiên sử dụng các công cụ này trong project.

Thông thường thì một ứng dụng có thể có tùy ý số lượng các layer. Thường thì một ứng dụng Android sẽ có 3 layer:

* **Outer**: Implementation layer: là nơi mà tất cả mọi thứ của framwork xảy ra, điều này bao gồm tất cả các công cụ Android như là tạo các activity, các fragment, gửi intent, networking và databases.
* **Middle**: Interface adapter layer: là hoạt động như một kết nối giữa business logic và framework specific code.
* **Inner**: Business logic layer: tương tự như trên.

Đối với mỗi layer ở trên core layer đều có trách nhiệm chuyển đổi các model thành các model layer thấp hơn trước khi các layer thấp hơn có thể sử dụng đến chúng. Tại sao việc chuyển đổi model là cần thiết? Ví dụ, các business logic model có thể không thích hợp cho việc hiển thị chúng đối với người dùng cuối, bạn có thể sẽ phải kết hợp nhiều nhiều business logic model cùng một lúc. Vì vậy, nên tạo một lớp ViewModel để có thể dễ dàng hiển thị UI. Sau đó, hãy sử dụng một lớp converter ở outer layer để chuyển đổi các business model sao cho thích hợp với ViewModel.

#### Lợi ích của Clean Architecture:

* Mạch lạc - dễ xem (bản gốc ghi screaming với dụng ý là chỉ cần nhìn cấu trúc package cũng có thể hiểu được mục đích và cơ chế hoạt động của ứng dụng)
* Linh hoạt - thể hiện ở khả năng độc lập, không phụ thuộc vào framework, database, application server.
* Dễ kiểm thử - testable

#### Hạn chế của Clean Architecture

Bên cạnh những lợi ích trên thì Clean Architecture còn những hạn chế sau:

* Không thể sử dụng framework theo cách mỳ ăn liền- do luật dependency inversion.
* Khó áp dụng
* Indirect - quá nhiều interface?
* Cồng kềnh - thể hiện ở việc có quá nhiều class so với các project cùng mục tiêu (tuy nhiên các class được thêm vào đều có chủ ý và đáp ứng đúng quy định khi triển khai kiến trúc)

Nếu suy luận từ những hạn chế của Clean Architecture, rút ra được một số điểm cần chú ý khi áp dụng Clean Architecture:

* Cần duy trì và nâng cấp hệ thống kể cả khi người làm ra nó đã rời khỏi công ty?
* Cần duy trì hệ thống ổn định, không phụ thuộc vào sự sinh-tử của framework?

### Hệ quản trị cơ sở dữ liệu MySQL:

#### Giới thiệu:

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở phổ biến hiện nay.

**Hệ quản trị cơ sở dữ liệu MySQL**được hiểu như là chương trình dùng để quản lý hệ thống cơ sở dữ liệu, trong đó, cơ sở dữ liệu là một hệ thống lưu trữ thông tin được sắp xếp rõ ràng, phân lớp ngăn nắp. Nó giúp bạn có thể truy cập dữ liệu một cách thuận lợi và nhanh chóng nhất. Vì hỗ trợ đa số các ngôn ngữ lập trình nên MySQL chính là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất trên thế giới. Hiện MySQL đang được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng.



Bảng . MySql

MySQL là cơ sở dữ liệu tốc độ cao, ổn định hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Đặc biệt, **hệ quản trị cơ sở dữ liệu MySQL**hoàn toàn miễn phí nên người dùng có thể thoải mái tải về từ trang chủ. Nó có rất nhiều những phiên bản cho các hệ điều hành khác nhau. MySQL được sử dụng cho việc bỗ trợ PHP, Perl và nhiều ngôn ngữ khác. Là nơi lưu trữ những thông tin trên các trang web viết bằng framework PHP hay Perl…

#### Đặc điểm của hệ quản trị cơ sở dữ liệu MySQL:

* **Hệ quản trị cơ sở dữ liệu MySQL**chính là một trong những phần mềm quản trị CSDL dạng server based, hệ gần giống với SQL server of Microsoft.
* MySQL là phần mềm quản lý dữ liệu thông qua CSDL. Và mỗi một CSDL đều có bảng quan hệ chứa dữ liệu riêng biệt.
* MySQL có cơ chế quản lý sử dụng riêng giúp cho mỗi người sử dụng đều có thể quản lý cùng lúc một hay nhiều CSDL khác nhau. Và mỗi người dùng đều có 1 username và password để truy nhập và truy xuất đến CSDL. Khi truy vấn đến CSDL của MySQL, phải cung cấp tài khoản và mật khẩu có quyền sử dụng cơ sở dữ liệu đó.
* MySQL tương thích tốt với môi trường PHP, giúp hệ thống hoạt động mạnh mẽ. Bạn có thể viết hoặc chỉnh sửa code MySQL trên các IDE dành cho PHP

#### Những lợi ích khi sử dụng hệ quản trị cơ sở dữ liệu MySQL

Ra mắt người dùng hoàn toàn miễn phí,**hệ quản trị cơ sở dữ liệu MySQL**mang đến rất nhiều những lợi ích thiết thực, được ứng dụng trong nhiều dự án của các công ty công nghệ, một trong số đó có thể kể đến những công ty chuyên lập trình như MonaMedia, Misa,…

* **Hiệu năng sử dụng cao**

Hầu hết, **hệ quản trị cơ sở dữ liệu MySQL** đang được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Và họ đánh giá rất cao ở hiệu năng sử dụng của MySQL. Với kiến trúc storage-engine, MySQL đặc trưng cho các ứng dụng chuyên biệt, đặc biệt là đối với những trang web có dung lượng lớn, phục vụ hàng triệu bạn đọc. Hoặc đối với những hệ thống xử lý giao dịch tốc độ cao thì MySQL đều cùng có thể đáp ứng được những khả năng xử lý khắt khe của mọi hệ thống. Đặc biệt, với những tiện ích tải tốc độ cao, cơ chế xử lý nâng cao cùng bộ nhớ cache. MySQL đưa ra tất cả những tính năng cần có, đây là giải pháp hoàn hảo nhất ngay cả đối với những hệ thống doanh nghiệp khó tính nhất hiện nay.

* **MySQL hỗ trợ giao dịch mạnh mẽ**

**Hệ quản trị cơ sở dữ liệu MySQL**không những mang lại hiệu năng sử dụng cao. Mà nó còn đưa ra một trong số những engine giao dịch cơ sở dữ liệu tốt nhất trên thị trường hiện nay. Tính năng này bao gồm: Khóa mức dòng không hạn chế; hỗ trợ giao dịch ACID hoàn thiện; khả năng giao dịch được phân loại và hỗ trợ giao dịch đa dạng mà người đọc không cản trở cho người viết và ngược lại. Với MySQL, dữ liệu sẽ được đảm bảo trong suốt quá trình server có hiệu lực. Các mức giao dịch độc lập sẽ được chuyên môn hóa, nếu phát hiện có lỗi khóa chết ngay tức thì.

* **Tốc độ rất nhanh**

Tốc độ xử lý dữ liệu của mySQL rất nhanh.

Đánh giá chung của các nhà phát triển, tất cả họ đều cho rằng **hệ quản trị cơ sở dữ liệu MySQL**là cơ sở dữ liệu nhanh nhất. Đây là nơi để cho các website có thể trao đổi thường xuyên các dữ liệu bởi nó có engine xử lý tốc độ cao. Khả năng chèn dữ liệu cực nhanh và hỗ trợ mạnh mẽ các chức năng chuyên dụng cho trang web. Các tính năng này cũng được sử dụng cho môi trường lưu trữ dữ liệu mà hệ quản trị này tăng cường đến hàng terabyte cho các server đơn. Ngoài ra còn có những tính năng khác như: chỉ số băm, bảng nhớ chính, bảng lưu trữ và cây B được gói lại để giúp giảm các yêu cầu lưu trữ tối đa đến 80%. Với tốc độ nhanh, thật không thể phủ nhận **hệ quản trị cơ sở dữ liệu MySQL**là sự lựa chọn tốt nhất cho cả ứng dụng web cũng như các ứng dụng của doanh nghiệp ngày nay.

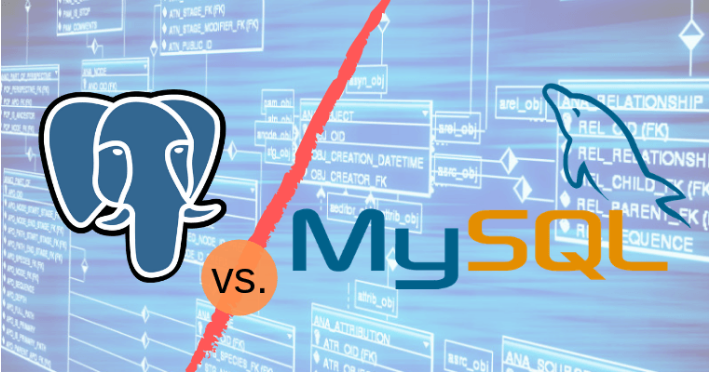
* **Dễ dàng sử dụng**

MySQL ngoài được biết đến với tốc độ khá cao, ổn định thì nó thực sự là một hệ thống cơ sở dữ liệu rất đơn giản, rất dễ sử dụng. Ít phức tạp khi cài đặt và quản trị hơn các hệ thống lớn. Đặc biệt nó có thể hoạt động trên tất cả các hệ điều hành.

* **Hỗ trợ ngôn ngữ truy vấn**

**MySQL hệ quản trị cơ sở dữ liệu**là ngôn ngữ của sự lựa chọn cho tất cả các hệ thống cơ sở dữ liệu hiện đại. Người dùng hoàn toàn có thể truy cập MySQL bằng cách sử dụng các ứng dụng mà hỗ trợ ODBC (một giao thức giao tiếp cơ sở dữ liệu được phát triển bởi Microsoft). Nhiều client có thể truy cập đến server trong cùng một thời gian. Đặc biệt các client có thể sử dụng nhiều cơ sở dữ liệu một cách đồng thời. Bạn cũng có thể truy cập MySQL tương tác với khi sử dụng một vài giao diện để đưa vào các truy vấn và xem kết quả.

* **Tính kết nối và bảo mật cao**



Bảng . Bảo mật cao của MySql

MySQL có tính kết nối và bảo mật cao.

Điều quan trọng nhất của mỗi một doanh nghiệp chính là việc bảo mật dữ liệu tuyệt đối. Và **hệ quản trị cơ sở dữ liệu MySQL**tích hợp các tính năng bảo mật an toàn tuyệt đối. MySQL được nối mạng một cách đầy đủ. Các cơ sở dữ liệu có thể được truy cập từ bất cứ nơi nào trên internet. Bạn có thể chia sẻ dữ liệu của bạn với bất kì ai, bất cứ lúc nào và bất cứ nơi đâu bạn muốn. Nhưng MySQL kiểm soát quyền truy cập nên người không nên nhìn thấy dữ liệu của bạn sẽ không thể nào nhìn được. Với việc xác nhận truy cập cơ sở dữ liệu, MySQL trang bị các kĩ thuật mạnh. Chỉ có những người sử dụng đã được xác nhận mới truy cập được vào cơ sở dữ liệu. Ngoài ra, SSH và SSL cũng được hỗ trợ nhằm đảm bảo kết nối an toàn và bảo mật. Tiện ích backup và recovery cung cấp bởi **MySQL hệ quản trị cơ sở dữ liệu**và các hãng phần mềm thứ 3 cho phép backup logic và vật lý cũng như recovery toàn bộ ngay tại một thời điểm.

* **Tính linh động cao**

**MySQL** có thể hoạt động trên tất cả các hệ điều hành, chạy được với mọi phần cứng từ các máy PC ở nhà cho đến các máy server. Máy chủ **hệ quản trị cơ sở dữ liệu MySQL**đáp ứng nhiều tính năng linh hoạt. Nó có sức chứa để xử lý các ứng dụng được nhúng sâu với 1MB dung lượng để chạy kho dữ liệu khổng lồ lên đến hàng terabytes thông tin. Tính chất mã nguồn mở của MySQL cho phép tùy biến theo ý muốn để thêm các yêu cầu phù hợp cho database server.

* **Mã nguồn mở tự do và hỗ trợ 24/7**

Băn khoăn của rất nhiều doanh nghiệp khi họ gặp khó khăn trong việc giao toàn bộ phần mềm cho một mã nguồn mở. Bởi khó có thể tìm được hỗ trợ hay bảo mật an toàn phục vụ một cách chuyên nghiệp. Nhưng vấn đề khó khăn này sẽ không còn nữa nếu sử dụng **hệ quản trị cơ sở dữ liệu MySQL.**Với MySQL, mọi sự cam kết đều rất rõ ràng, mọi sự cố đều được MySQL bồi thường. Bạn có thể tìm thấy các tài nguyên có sẵn mà hệ quản trị này hỗ trợ. Cộng đồng MySQL rất có trách nhiệm. Họ thường trả lời các câu hỏi trên mailing list chỉ trong vài phút. Nếu lỗi xảy ra, các nhà phát triển sẽ đưa ra cách khắc phục nhanh nhất cho bạn. Và cách khắc phục đó sẽ ngay lập tức có sẵn trên internet.

* **Chi phí sở hữu thấp nhất**

**Hệ quản trị cơ sở dữ liệu MySQL**cung cấp miễn phí cho hầu hết các việc sử dụng trong một tổ chức. Chính vì vậy, sử dụng MySQL cho các dự án, các doanh nghiệp đầu nhận thấy được sự tiết kiệm cho phí rất đáng kể. Người dùng của MySQL cũng không phải mất nhiều thời gian để sửa chữa hoặc vấn đề thời gian chết.

### dbForge Studio

#### Giới thiệu:

DbForge Studio cho MySQL là một môi trường phát triển tích hợp mạnh mẽ (IDE) cho MySQL từ Devart

#### Ưu điểm:

* Xây dựng Database một các dễ dàng. Hỗ trợ thiết kế và vẽ các bảng dữ liệu **sử dụng Diagram**. Có thể vẽ được các bảng và nhìn tổng quan được các releation giữa các bảng.
* Hỗ trợ **Debug SQL**: rất cần thiết khi muốn xem rõ lỗi cụ thể trong đoạn mã truy vấn dài hàng trăm dòng.
* Hỗ trợ **Compare Data**và cấu trúc Database. Rất đắc lực khi bạn làm việc trên nhiều phiên bản của Datatabase. Giúp bạn compare dữ liệu hoặc cấu trúc 2 Database và thực hiện đồng bộ chúng lại.
* **Generate Data** – sinh dữ liệu test. Việc này cực kỳ cần thiết khi bạn có nhu cầu muốn sinh dữ liệu giả và đẹp như dữ liệu thật.
* **Format SQL** – định dạng câu truy vấn đẹp và dễ nhìn. Không phải mất công chỉnh từng câu truy vấn trong chuỗi truy vấn dài ngoằng của mình.
* **Auto-suggestions** (SQL query) – tự động gợi ý và sinh chuỗi câu truy vấn trong quá trình viết câu truy vấn.
* **Import/ Export** dữ liệu, Automation test, và hàng đống các tính năng khác.

#### Nhược điểm:

* **Mỗi hệ quản trị CSDL sẽ có bản.** Cái này không hẳn là nhược điểm vì đôi khi chúng ta sử dụng lại có mong muốn cài 1 phiên bản dành riêng cho nó nhẹ nhàng.
* **Không có bản Free** **cho các tính năng t**ính tới hàng trăm đô la mỹ. Đây không phải là số tiền mà một lập trình viên nào cũng muốn bỏ ra để mua. Và đặc biệt các bạn sinh viên ở VN mình thì đa số đều lắc đầu. Tất nhiên chuyện đắt ở đây cũng có lý, bỏ ra vài triệu để làm việc với nó thì không hề đắt.
* **Chỉ sử dụng trên HĐH Windows**. Với tình hình hiện tại, các ứng dụng phát triển trên nhiều nền tảng, hệ điều hành khác nhau. Nhưng em nó lại chỉ sử dụng được trên Windows.
* Dùng nhiều sẽ thành thói quen. Đôi khi đang quen phong cách hỗ trợ của nó đến khi không có lại thấy khó chịu. Điển hình là đang quen với auto suggestion của nó đến lúc viết truy vấn thuần thấy hơi thốn.

### Mock-up

#### Giới thiệu:

* + Mock-up tạo một khuôn mẫu về giao diện cho đối tượng hoặc thiết bị được tạo ra dựa trên một thiết kế cụ thể theo tỉ lệ hoặc kích thước đầy đủ.
  + Mock-up được xây dựng để truyền đạt ý tưởng chung trên các sách thực tế, nó được sử dụng chủ yếu bởi các nhà thiết kế để có được thông tin phản hồi từ người dùng.
  + Mock-up cung cấp phương pháp hữu hiệu tiết kiệm nhiều thời gian và tiền bạc trong việc thử nghiệm thiết kế.

#### Ứng dụng:

* + Mock-up giúp hỗ thiết kế giao diện của website để tiện cho việc thiết kế giao diện.
  + Mô phỏng hoạt động cụ thể của từng chức năng trong trên ứng dụng trên thực tế.

### IBM Rational Rose

#### Giới thiệu:

* + IBM Rational Rose là phần mềm dành cho các nhà phát triển xây dựng và thiết kế hệ thống phần mềm, ứng dụng bằng ngôn ngữ mô hình hóa UML
  + Nó cung cấp các công cụ vẽ trực quan, nhanh chóng và tuân thủ các kí pháp được định nghĩa sẵn. Phần mềm được đánh giá là môi trường phát triển extend hoàn hảo cho các nhà phân tích và thiêt kế.

#### Ứng dụng:

* + Xây dựng mô hình use case.
  + Xây dựng các mô hình, biểu đồ phân tích use case như: biểu đồ trình tự, biểu đồ lớp, biểu đồ luồng giao diện, …

### Github

#### Giới thiệu:

* + GitHub là một hệ thống quản lý dự án và phiên bản code thông qua các commit. Các lập trình viên có thể clone lại mã nguồn từ một repository và Github chính là một dịch vụ máy chủ repository công cộng, mỗi người có thể tạo tài khoản trên đó để tạo ra các kho chứa của riêng mình để có thể làm việc.
  + GitHub có hơn 14 triệu người sử dụng với hơn 35 triệu kho mã nguồn, làm cho nó trở thành máy chủ chứa mã nguồn lớn trên thế giới, đã trở thành một yếu tố có sức ảnh hưởng trong cộng đồng phát triển mã nguồn mở.

#### Ứng dụng:

* + Quản lý mã nguồn.
  + Phân chia các nhánh(branch) để phân chia công việc cho từng người để xử lý từng module.

# Phân tích thiết kế

## Phân tích thiết kế hệ thống website bán hàng tạp hóa

### Biểu đồ ca sử dụng



Hình . Biểu đồ ca sử dụng

### Xây dựng cấu trúc cơ sở dữ liệu

### Lựa chọn cơ sở dữ liệu

Lựa chọn cơ sở dữ liệu MySQL cho dự án phát triển nhanh. Ưu điểm của MySQL:

* Miễn phí và mã nguồn mở
* Độ ổn định và độ tin cậy.
* Hiệu suất cao.
* Hỗ trợ đa nền tảng.
* Cộng đồng lớn và hỗ trợ mạnh mẽ.
* Hỗ trợ chuẩn SQL.
* Dễ sử dụng và quản lý.

**Cấu trúc cơ sở dữ liệu**

A computer screen shot of a computer

Description automatically generated

Bảng . Cấu trúc cơ sở dữ liệu

### Đặc tả use case

#### Đăng nhập

* Tên use case: Đăng nhập.
* Mô tả vắn tắt: Use case này cho phép bạn đọc đăng nhập để xác nhận quyền truy cập hệ thống.
* Luồng sự kiện:
  + Use case này bắt đầu khi bạn đọc truy cập màn hình đăng nhập .
  + Hệ thống hiện các ô nhập liệu và yêu cầu người dùng đăng nhập email và mật khẩu.
  + Người dùng nhập tên và mật khẩu sau đó kích vào nút “Đăng nhập”
  + Hệ thống kiểm tra email và mật khẩu đã có trong bảng User. Nếu thỏa mãn điều kiện đăng nhập hệ thống điều hướng bạn đọc đến màn hình chan chủ.
  + Use case kết thúc.

#### Xem danh sách sách đã đọc

* Tên use case: Xem danh sách sách đã đọc.
* Mô tả vắn tắt: Use case này cho phép bạn đọc xem danh sách sách đã đọc.
* Luồng sự kiện:
  + Ca sử dụng này bắt đầu khi bạn đọc truy cập màn hình trang chủ.
  + Hệ thống lấy thông tin lịch sử đọc của bạn đọc trong bảng readhistory và hiển thị các sách đã đọc trong bảng book.
  + Nếu bạn đọc chưa từng đọc sách nào, hệ thống ẩn thông tin các sách đã đọc.
  + Use case kết thúc.

#### Xem danh sách sách

* Tên use case: Xem danh sách sách.
* Mô tả vắn tắt: Use case này cho phép bạn đọc xem danh sách sách của website.
* Luồng sự kiện:
  + Use case này bắt đầu khi bạn đọc truy cập vào màn danh sách sách.
  + Hệ thống lấy danh sách sách có file sách điện tử trong bảng book và hiển thị lên màn hình và được nhóm theo danh mục
  + Người dùng tìm kiếm sách theo nhan đề sách và tên tác giả. Hệ thống lấy thông tin và lọc các sách trong bảng Book theo các thông tin được nhập và hiển thị danh sách các sách sau khi lọc lên màn hình.
  + Use case kết thúc.

#### Xem chi tiết sách

* Tên use case: Xem chi tiết sách.
* Mô tả vắn tắt: Use case này cho phép bạn đọc xem chi tiết sách.
* Luồng sự kiện:
  + Use case này bắt đầu khi bạn đọc nhấn vào nút “Xem chi tiết” tại một sách.
  + Hệ thống lấy thông tin sách gồm: tên sách, mô tả, ảnh bìa, tác giả, nhóm loại sách, lượt đọc từ bảng book và bảng readhistory hiển thị lên màn hình.
  + Use case kết thúc.

#### Đọc sách

* Tên use case: Đọc sách.
* Mô tả vắn tắt: Use case này cho phép bạn đọc thực hiện đọc sách.
* Luồng sự kiện:
  + Ca sử dụng này bắt đầu khi Bạn đọc đăng nhập hệ thống, kích vào nút “Đọc sách” từ một sách hoặc nút “Đọc sách” khi xem chi tiết một sách.
  + Hệ thống lấy thông tin file sách từ bảng book và lịch sử đọc sách của bạn đọc từ bảng readhistory nếu bạn đọc đã từng đọc sách này thì hiển thị thông báo tiếp tục đọc từ trang đã dừng trước đó.
    - Khi ấn vào nút “Không” hoặc icon đóng form xác nhận, hệ thống cho người dùng đọc sách từ đầu.
    - Khi ấn vào nút “Đọc tiếp” hệ thống tự động lật file sách đến trang mà người dùng đã dừng trước đó.
  + Use case kết thúc.

#### Ngừng đọc sách

* Tên use case:Ngừng đọc sách.
* Mô tả vắn tắt: Use case này cho phép bạn đọc ngừng đọc sách.
* Luồng sự kiện:
  + Use case này bắt đầu khi bạn đọc kích vào nút “Thoát” trong màn hình đọc sách. Hệ thống sẽ lấy thông tin trang sách hiện tại đang đọc thực hiện cập nhật vào bảng readhistory và chuyển hướng ra ngoài màn hình danh sách.
  + Use case kết thúc.

#### Quản lý thông tin cá nhân

* Tên use case: Quản lý thông tin cá nhân.
* Mô tả vắn tắt: Use case này cho bạn đọc và người quản trị quản lý thông tin cá nhân của mình.
* Luồng sự kiện:
  + Use case bắt đầu khi người dùng nhấn vào nút “Thông tin cá nhân” trên thanh menu.
  + Hệ thống hiển thị màn hình thông tin cá nhân của người dùng bao gồm: họ tên, địa chỉ, email, số điện thoại, ảnh đại diện từ bảng User.
  + Chỉnh sửa thông tin cá nhân: người dùng thay đổi các trường thông tin trong màn hình thông tin cá nhân và nhấn “Lưu”. Hệ thống hiển thị thông báo xác nhận lưu thông tin cá nhân. Người dùng nhấn nút “Đồng ý” hệ thống cập nhật thông tin cá nhân của người dùng trong bảng User và hiển thị thông báo cập nhật thông tin thành công.
  + Use case kết thúc.

#### Quản lý sách

* Tên use case: Quản lý sách.
* Mô tả vắn tắt: Use case này cho phép người quản trị xem, thêm, sửa, xóa sách trong bảng Book.
* Luồng sự kiện:
  + Use case này bắt đầu khi người quản trị nhấn vào nút “Sách” trên menu chính. Hệ thống sẽ lấy danh sách các sách: Mã sách, Hình ảnh, Tên sách, Danh mục, Giá, Số lượng từ bảng Book và hiển thị lên màn hình.
  + Xem chi tiết sách:
    - Người quản trị chọn một sách và nhấn nút chỉnh sửa. Hệ thông lấy thông tin chi tiết của sách gồm: Mã sách, Hình ảnh đại diện sách, Tên sách, Danh mục, Số lượng, Giá, Danh sách ảnh, Mô tả từ bảng Book và hiển thị lên màn hình.
    - Người quản trị nhấn nút “Thoát” để thoát xem chi tiết sách.
  + Thêm sách:
    - Người quản trị nhấn vào nút “Thêm sách”. Hệ thống hiển thị biểu mẫu yêu cầu nhập các thông tin cần thiết bao gồm: Tên sách, Ảnh sách, Nhóm loại sách, Tác giả, Mô tả.
    - Người quản trị nhập thông tin cho sách cần tạo, mã sách được sinh tự động và nhấn nút “Lưu”. Hệ thống hiển thị thông báo xác nhận lưu thông tin sách. Người quản trị nhấn “Đồng ý”, hệ thống lưu thông tin sách vào bảng Book.
    - Hệ thống hiển thị thông báo “Thêm thành công” và hiển thị lại danh sách các sách.
  + Sửa sách:
    - Người quản trị nhấn nút “Sửa” tương ứng với sách cần sửa thông tin. Hệ thống sẽ lấy thông tin của sách gồm: Tên sách, Ảnh sách, Nhóm loại sách, Tác giả, Mô tả từ bảng Book và hiển thị lên biểu mẫu nhập liệu.
    - Người quản trị sửa các thông tin của sách trong biểu mẫu nhập liệu, sau đó nhấn nút “Lưu”. Hệ thống hiển thị thông báo xác nhận lưu thông tin. Người quản trị nhấn “Đồng ý”, hệ thống sẽ sửa thông tin của sách với thông tin mới được nhập vào trong bảng Book.
    - Hệ thống hiển thị thông báo “Sửa thành công” và hiển thị lại danh sách các sách.
  + Xóa sách:
    - Người quản trị kích vào nút xoá trên 1 dòng sách cần xóa và nhấn nút “Xóa”. Hệ thống hiển thị thông báo xác nhận xóa. Người quản trị nhấn “Đồng ý” hệ thống xóa sách với mã sách tương ứng trong bảng Book.
    - Hệ thống hiển thị thông báo “Xóa thành công” và hiển thị lại danh sách các sách.
  + Use case kết thúc.

#### Xem thống kê

* Tên use case: Xem thống kê.
* Mô tả vắn tắt: Ca sử dụng cho phép người quản trị xem thống kê sách, bạn đọc.
* Luồng sự kiện:
  + Ca sử dụng này bắt đầu khi Người quản trị kích vào “Tổng quan” trên menu chính.
  + Hệ thống lấy số lượng sách trong bảng Book, số lượng bạn đọc trong bảng User, sách được đọc nhiều nhất, bạn đọc đọc nhiều sách nhất, số sách chưa được đính file sách.
  + Use case kết thúc.

### Phân tích use case

#### Đăng ký

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case đăng ký

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case đăng ký

#### Đăng nhập

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case đăng nhập

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case đăng nhập

#### Xem sách đã đọc

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case xem sách đã đọc

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case xem sách gợi ý

#### Xem danh sách sách

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case xem danh sách sách

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case xem danh sách sách

#### Xem chi tiết sách

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case xem chi tiết sách

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case xem chi tiết sách

#### Đọc sách

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case đọc sách

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case đọc sách

#### Quản lý thông tin cá nhân

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case quản lý thông tin cá nhân

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case quản lý thông tin cá nhân

#### Quản lý sách

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case quản lý sách

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case quản lý sách

#### Xem thống kê

* Biểu đồ trình tự



Hình . Biểu đồ trình tự use case quản lý thông tin cá nhân

* Biểu đồ lớp phân tích



Hình . Biểu đồ lớp phân tích use case quản lý thông tin cá nhân

# Cài đặt chương trình

## Khởi chay chương trình

* Khởi chạy IDE Visual Studio code
  + Nhập “npm run dev” vào terminal khởi chạy chương trình
* Khởi chạy IDE Visual Studio
  + Sử dụng phím tắt F5 để khởi chạy Back End

## Website Thư Viện Điện Tử

### Màn hình đăng nhập

A screenshot of a computer

Description automatically generated

Bảng . Màn hình đăng nhập

### Màn hình Tổng quan

A screenshot of a computer

Description automatically generated

Bảng . Màn hình Tổng quan.

### Màn hình danh sách sách

A screenshot of a computer

Description automatically generated

Bảng . Màn hình danh sách sách điện tử.

### Màn hình Thêm/Sửa sách điện tử

A screenshot of a computer

Description automatically generated

Bảng . Màn hình Thêm/Sửa sách điện tử.

### Màn hình danh sách bạn đọc

A screenshot of a computer

Description automatically generated

Bảng . Màn hình danh sách bạn đọc.

### Màn hình cổng bạn đọc

A screenshot of a website

Description automatically generated

Bảng . Màn hình cổng bạn đọc.

### Màn hình xem chi tiêt sách

A screenshot of a book

Description automatically generatedBảng . Xem chi tiết sách

### Màn hình đọc sách

A screenshot of a math test

Description automatically generated

Bảng . Màn hình đọc sách

# KẾT LUẬN

Qua quá trình làm đồ án, mang lại rất nhiều kinh nghiệm, kiến thức bổ ích cũng như có cơ hội áp dụng các kiến thức đã tích lũy được sau những năm tháng học tập tại trường Đại học Công nghiệp Hà Nội, trong đó phải kể đến các kiến thức nổi bật, hệ quản trị cơ sở dữ liệu, thiết kế trang Website để hoàn thành đề tài tốt nghiệp “Xây dựng Website Thư Viện Điện Tử sử dụng Vue.JS, ASP.NET và MySql

Cuối cùng, một lần nữa em xin gửi lời cảm ơn sâu sắc đến quý thầy cô trường Đại học Công nghiệp Hà Nội đã dạy cho em rất nhiều kiến

# TÀI LIỆU THAM KHẢO

[1]. Hoàng Quang Huy (2016), “Giáo trình cơ sở dữ liệu”, H. Giáo dục Việt Nam 2011.

[2].Vũ Thị Dương, Phùng Đức Hòa, Nguyễn Thị Hương Lan (2015), “Giáo trình Phân tích thiết kế hướng đối tượng”, NXB Khoa học và Kỹ thuật.

[3]. https://vuejs.org

[4]. https://dotnet.microsoft.com/en-us/apps/aspnet/apis

[5]. https://www.mysql.com