TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ**

**CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM**

**Website Quản Lý Quán Karaoke Bằng C#**

*Người hướng dẫn*: **VŨ ĐÌNH HỒNG**

*Người thực hiện*: **PHẠM VŨ QUỐC CƯỜNG – 517H0040**

**TRẦN PHẠM ANH TUẤN–517H0093**

**PHẠM CÔNG–517H0039**

**TÔ QUỐC THẮNG–517H0082**

Lớp : 17050210-17050211

Khoá  **: 21**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ**

**CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM**

**Website Quản Lý Quán Karaoke Bằng C#**

*Người hướng dẫn*: **VŨ ĐÌNH HỒNG**

*Người thực hiện*: **PHẠM VŨ QUỐC CƯỜNG – 517H0040**

**TRẦN PHẠM ANH TUẤN–517H0093**

**PHẠM CÔNG–517H0039**

**TÔ QUỐC THẮNG–517H0082**

Lớp : 17050210-17050211

Khoá  **: 21**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành nhất đến quý thầy cô TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG, những người đã dìu dắt em tận tình, đã truyền đạt cho em những kiến thức và bài học quý báu trong suốt thời gian em theo học tại trường.

Em xin trân trọng gửi lời cảm ơn đến tất cả các thầy cô trong khoa Công Nghệ Thông Tin, đặc biệt là thầy Vũ Đình Hồng đã tận tình hướng dẫn và hỗ trợ chúng em trong suốt quá trình hoàn thành đề tài này. Với sự chỉ bảo tận tình của thầy, em đã có những định hướng tốt trong việc triển khai và thực hiện các yêu cầu của đề tài. Tuy nhiên, với vốn kiến thức còn hạn chế nên sai sót là điều không thể tránh khỏi trong suốt quá trình làm bài. Chúng em rất mong nhận được những ý kiến đóng góp, sửa chữa, phê bình từ phía thầy cô. Để có thể cải thiện thêm kiến thức của mình và biến chúng thành hành trang vô cùng đáng quý của chúng em trong tương lai sau này.

Ngoài ra, em cũng xin gửi lời cảm ơn tới tất cả bạn bè, đặc biệt các bạn trong lớp luôn gắn bỏ, cùng học tập và đưa ra quan điểm góp phần xây dựng đề tài này.

Em xin chân thành cảm ơn!

# ĐỒ ÁN ĐƯỢC HOÀN THÀNH

# TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Chúng tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của thầy Vũ Đình Hồng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 30 tháng 11 năm 2020*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Phạm Vũ Quốc Cường*

*Phạm Công*

*Tô Quốc Thắng*

*Trần Phạm Anh Tuấn*

# PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

Tp. Hồ Chí Minh, ngày tháng năm 2020

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

Tp. Hồ Chí Minh, ngày tháng năm 2020

(kí và ghi họ tên)

MỤC LỤC

[LỜI CẢM ƠN ii](#_Toc51884751)

[ĐỒ ÁN ĐƯỢC HOÀN THÀNH iii](#_Toc51884752)

[TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG iii](#_Toc51884753)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iv](#_Toc51884754)

[MỤC LỤC 1](#_Toc51884755)

[DANH MỤC CÁC HÌNH ẢNH VÀ BẢNG BIỂU 3](#_Toc51884756)

[I. Phương thức truy cập CSDL trong Java bằng JDBC 5](#_Toc51884757)

[1. JDBC là gì? 5](#_Toc51884758)

[2. Các thành phần hoặc tính năng của JDBC. 5](#_Toc51884759)

[2.1 DriveManager. 5](#_Toc51884760)

[2.2 Driver. 5](#_Toc51884761)

[2.3 Connection. 8](#_Toc51884762)

[2.4 Statement, Prepared Statement, Callable Statement Interface. 8](#_Toc51884763)

[2.5 ResultSet. 9](#_Toc51884764)

[2.6 SQLException. 10](#_Toc51884765)

[3. Code demo 10](#_Toc51884766)

[3.1 Chuẩn bị. 10](#_Toc51884767)

[3.2 Thực hiện. 10](#_Toc51884768)

[II. Phương thức truy cập CSDL trong java bằng Active JDBC 20](#_Toc51884769)

[1. ActiveJDBC là gì? 20](#_Toc51884770)

[2. Các thành phần của ActiveJDBC. 20](#_Toc51884771)

[2.1 Tạo bảng 20](#_Toc51884772)

[2.2 Tạo model 20](#_Toc51884773)

[2.3 Kết nối với Database 21](#_Toc51884774)

[2.4 Thêm dữ liệu 21](#_Toc51884775)

[2.5 Các lệnh truy vấn dữ liệu 22](#_Toc51884776)

[2.6 Cập nhật/ thay đổi dữ liệu 22](#_Toc51884777)

[2.7 Xóa dữ liệu 22](#_Toc51884778)

[2.8 Phân trang 23](#_Toc51884779)

[III. Phương thức truy cập CSLD trong Java bằng JPA 24](#_Toc51884780)

[1. JPA là gì ? 24](#_Toc51884781)

[2. Kiến trúc JPA 24](#_Toc51884782)

[2.1 Entity 25](#_Toc51884783)

[2.2 Entity Manager 25](#_Toc51884784)

[2.3 Entity Manager Factory 26](#_Toc51884785)

[2.4 Persistence 26](#_Toc51884786)

[2.5 Entity Transaction 26](#_Toc51884787)

[2.6 Query 26](#_Toc51884788)

[3. Các tính năng của JPA 27](#_Toc51884789)

[4. JPQL là gì? 28](#_Toc51884790)

[4.1. Các thành phần hoặc tính năng của JPQL. 28](#_Toc51884791)

[5. Criteria API là gì? 29](#_Toc51884792)

[5.1. Các thành phần hoặc tính năng của Criteria API. 30](#_Toc51884793)

[6. Ưu điểm của JPA 30](#_Toc51884794)

[7. Một số ORM Framework hỗ trọ JPA 31](#_Toc51884795)

[8. Code demo 31](#_Toc51884796)

[8.1 Sử dụng annotation để mapping với Database 31](#_Toc51884797)

[8.2 Các cách tạo class truy suất dữ liệu 33](#_Toc51884798)

[IV. Phương thức truy cập CSDL trong java bằng 37](#_Toc51884799)

[1. Hibernate Framework là gì? 37](#_Toc51884800)

[2. Kiến trúc của Hibernate 37](#_Toc51884801)

[3. Hibernate caching 38](#_Toc51884802)

[4. HQL (Hibernate Query Language) 39](#_Toc51884803)

[4.1 Ưu điểm của HQL 39](#_Toc51884804)

[4.2 Các phương thức thông dụng của HQL 39](#_Toc51884805)

[5. Hibernate annotation 39](#_Toc51884806)

[6. Hibernate transaction management 40](#_Toc51884807)

[7. Các ưu điểm của Hibernate. 41](#_Toc51884808)

[7.1 Mã nguồn mở và nhẹ. 41](#_Toc51884809)

[7.2 Hiệu suất nhanh. 41](#_Toc51884810)

[7.3 Truy vấn độc lập với cơ sở dữ liệu. 41](#_Toc51884811)

[7.4 Tự động tạo ra các bảng của cơ sở dữ liệu. 41](#_Toc51884812)

[7.5 Đơn giản hóa các lệnh hợp bảng (Join, InnerJoin). 41](#_Toc51884813)

[7.6 Cung cấp các thống kê truy vấn và trạng thái cơ sở dữ liệu. 41](#_Toc51884814)

[7.7 Hỗ trợ nhiều Database 41](#_Toc51884815)

[7.8 Hỗ trợ nhiều công nghệ 42](#_Toc51884816)

[8. Code demo 42](#_Toc51884817)

[8.1 Mapping Hibernate sử dụng XML 42](#_Toc51884818)

[8.2 Mapping Hibernate sử dụng annotation 45](#_Toc51884819)

[V. SO SÁNH GIỮA CÁC CÁCH TRUY CẬP CƠ SỞ DỮ LIỆU 54](#_Toc51884820)

[VI. REFERENCE 56](#_Toc51884821)

# DANH MỤC CÁC HÌNH ẢNH VÀ BẢNG BIỂU

**DANH MỤC HÌNH**

[Hình 1 1: Java JDBC Logo 6](#_Toc51884915)

[Hình 1 2: JDBC-ODBC birdge driver 7](#_Toc51884916)

[Hình 1 3: Native-API Driver 7](#_Toc51884917)

[Hình 1 4: Network Protocol driver 8](#_Toc51884918)

[Hình 1 5: Thin driver 8](#_Toc51884919)

[Hình 2 1: Tạo Table 21](#_Toc51884926)

[Hình 2 2: Tạo Model 22](#_Toc51884927)

[Hình 2 3: Kết nối Database (DB) 22](#_Toc51884928)

[Hình 2 4: Cập nhật/ thay đổi dữ liệu 23](#_Toc51884929)

[Hình 2 5: Xóa 1 dữ liệu 23](#_Toc51884930)

[Hình 2 6: Xóa tất cả dữ liệu 24](#_Toc51884931)

[Hình 3 1: Kiến trúc JPA 26](#_Toc51884937)

[Hình 3 2: Mối quan hệ giữa các thành phần 28](#_Toc51884938)

[Hình 3 3: Criteria Queries 30](#_Toc51884939)

[Hình 4 1: Hibernate Logo 38](#_Toc51884944)

[Hình 4 2: Kiến trúc Hibernate 39](#_Toc51884945)

[Hình 4 3: Hibernate Caching 40](#_Toc51884946)

[Hình 4 4: Hibernate transaction management 41](#_Toc51884947)

**DANH MỤC BẢNG BIỂU**

[Bảng 1: So sáng giữa JDBC và JPA 1 55](#_Toc51884857)

[Bảng 2: So sánh giữa JPA và Hibernate 2 56](#_Toc51884870)

# I. GIỚI THIỆU ĐỀ TÀI

## I.1. Lời nói đầu về đề tài:

Ngày nay công nghệ thông tin vô cùng phát triển thì mọi người đều sử dụng máy vi tính để làm việc. Công nghệ thông tin cũng được áp dụng rất nhiều vào các lĩnh vực mà điển hình là lĩnh vực quản lý kinh doanh.

Trong một xã hội hiện đại, năng động ngày nay, giải trí đóng vai trò rất quan trọng. Nó giúp con người giải tỏa những căn thẳng trong công việc, học tập… Karaoke là một trong những loại hình giải trí được đông đảo người dân lựa chọn. Như chúng ta đã biết tại các quán karaoke, việc quản lý kinh doanh thì rất quan trọng nhưng lại còn thủ công, chỉ xây dựng và lưu lại trên sổ sách gây mất thời gian, khó khăn trong việc thống nhất, theo dõi và quản lý cho người dùng. Từ thực tế đó, việc xây dựng được phần mềm quản lý kinh doanh cho quán karaoke là rất cần thiết.

Vì vậy em đã “Xây dựng Website quản lý quán karaoke” với mục đích nghiên cứu, tìm hiểu về và xây dựng website để có thể đáp ứng được nhu cầu quản lý sổ sách cho người dùng, giúp cho mọi người có thể tiết kiệm công sức, thời gian và quản lý dễ dàng, tiện lợi hơn.

## I.2. Phân tích phần mềm:

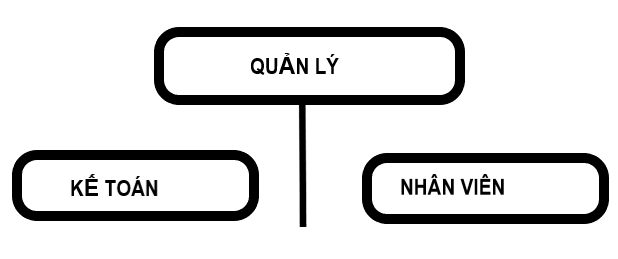
I.2.1 Khảo sát nghiệp vụ

Qua quá trình khảo sát hiện tại đã thu thập được những thông tin nghiệp vụ như sau:

Doanh nghiệp kinh doanh Karaoke có nhiều phòng, mỗi phòng có những thông tin lưu trữ như mã phòng, loại phòng, giá phòng và trạng thái của phòng. Đồng thời sẽ có một trang menu quản lý những thức ăn, nước uống để có thể phục vụ cho khách hàng. Mỗi món sẽ chứa các thông tin như là mã thức ăn, tên thức ăn, giá cả và số lượng hiện đang có trong kho hàng. Mỗi khách hàng sau khi sử dụng xong dịch vụ sẽ được tính tiền và nhận hóa đơn từ nhân viên, thông tin hóa đơn bao gồm: tên nhân viên xử lý hóa đơn, mã hóa đơn, thông tin khách hàng, thời gian sử dụng phòng, thông tin phòng, tiền giờ, tiền dịch vụ, thuế thu nhập VAT.

I.2.2 Mô tả hệ thống:

Doanh nghiệp Karaoke có người quản lý tất cả các hoạt động của doanh nghiệp. Mô hình tổ chức như sau:



Hình 1: Mô hình tổ chức quản lý hoạt động của doanh nghiệp.

* Trong đó:
* Kế toán (Accountant): tham mưu quản lý các hạng mục doanh nghiệp để tra cứu, thanh toán hóa đơn cho khách hàng. Tổng kết doanh thu, đơn hàng và báo cáo tình hình của doanh nghiệp.
* Nhân viên (Staff): Tham mưu quản lý các phòng, dịch vụ của doanh nghiệp. Tiếp nhận thanh toán, xử lý các yêu cầu dịch vụ của khách hàng.
* Quản lý (Admin): Quản lý tất cả các hoạt động của doanh nghiệp, bộ phận thu ngân, nhân viên. Thu nhận tổng kết doanh thu, chi phí của doanh nghiệp. Đề ra kế hoạch kinh doanh, các hoạt động của doanh nghiệp.
* Các hoạt động nghiệp vụ hệ thống:
  + Quản lý thuê phòng:
    - Chi tiết sử dụng dịch vụ.
    - Nhu cầu đổi phòng vì những sự cố bất thường.
    - Thanh toán.
    - In hóa đơn.
  + Quản lý các doanh mục:
    - Doanh mục phòng:
      * Thêm phòng mới.
      * Xóa phòng hiện tại.
    - Doanh mục dịch vụ:
      * Thêm dịch vụ mới.
      * Xóa dịch vụ hiện tại.
      * Sửa thông tin dịch vụ. Các thông tin như giá tiền và số lượng đang có.
    - Doanh mục tài khoản: (Do quản lý thực hiện)
      * Thêm tài khoản mới.
      * Xóa tài khoản hiện tại.
      * Sửa thông tin tài khoản. Các thông tin như tên, số điện thoại, phân quyền tài khoản, mật khẩu tài khoản.
  + Thống kê doanh thu: (Do bộ phận kế toán quản lý)
    - Thống kê theo doanh thu theo tháng.
    - Thổng kê số lượng đơn được đặt theo tháng.
    - Thống kê theo các loại phòng đã được đặt.
    - Báo cáo doanh thu theo năm.
  + Tra cứu thông tin, lọc danh sách thuê phòng. (Theo ngày và tên khách hàng).

## I.3. Xác định yêu cầu:

I.3.1 Yêu cầu chức năng nghiệp vụ:

Hỗ trợ các chức năng quản lý tổng quát của phần mềm: quản lý thuê phòng, quản lý các doanh mục, tài khoản, thống kê, tra cứu. Ứng dụng có giao diện thân thiện dễ sử dụng, có khả năng nâng cấp và bổ sung sữa chữa sau này.

I.3.2 Yêu cầu hệ thống:

1. Phân quyền sử dụng:
   * Người quản trị quản lý (admin): được phép sử dụng tất cả các chức năng công cụ của toàn hệ thống.
   * Bộ phận thu ngân và nhân viên: chỉ được phép sử dụng các chức năng công cụ phù thuộc vào chức vụ của mình.
2. Môi trường:
   * Máy cần cài đặt mạng.
   * Máy in trắng đen hoặc cao hơn và khổ giấy cần dùng là A4 theo yêu cầu của file pdf.

3.3 Yêu cầu phi chức năng:

* Hình thức tra cứu tiện dụng, dễ sử dụng cho cả những người không chuyên về tin học. Hỗ trợ tra cứu nâng cao, có hình ảnh minh họa dễ hiểu, dễ sử dụng, thiết kế đơn giản, sắp xếp hợp lí, dễ dàng tìm thấy các công cụ chức năng khi cần.
* Các màn hình có sự nhất quán chung: Các trang đều được hiển thị theo một form chính.
* Tốc độ tra cứu và sử dụng các tính năng từ trung bình đến nhanh.

# II. Cơ sở lý thuyết

## 1. Hỗ trợ:

* **MSSQL:** Là một nền tảng độc lập được Microsoft phát triển dựa trên RDBMS (hệ thống cơ sở dữ liệu quan hệ) với ngôn ngữ hổ trợ là SQL (ngôn ngữ truy vấn dữ liệu).\*cái này nhiều ý lắm thêm vào
* **HTML (Hypertext Makeup Language):** Giúp tạo và cấu trúc cấu thành phần trong web. Giúp người dùng tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, links, blockquotes…
* **JavaScript:** Là ngôn ngữ lập trình phổ biến. Cũng là một trong 3 ngôn ngữ chỉnh của lập trình web:
  + **HTML:** Giúp bạn thêm nội dung cho các trang web.
  + **CSS:** Định dạng thiết kế, bố cục, phong cách, canh lề trang web.
  + **JavaScript:** Cải thiện hoạt động của trang web.
* **Entity Framework:** Với cách tiếp cận code-first, Entity Framework giúp tạo các đối tượng bảng cơ sở dữ liệu dựa trên model. Tuy nhiên cũng có 1 vài quy ước của code-first cho phép tự động cấu hình 1 model.
  + **Quy ước tên bảng:** khi tạo 1 class mô tả các thực thể là User sẽ được lưu trữ trong database thì Entity Framework sẽ tự động tạo ra 1 bảng có thên Users.
  + **Quy ước khóa chính:** khi tạo 1 thuộc tính tên là UserId trong lớp User của model thì thuộc tính này được nhận làm **Khóa chính**. Hơn nữa nó sẽ thiết lập 1 cột khóa auto-increment để lưu trữ giá trị nếu kiểu khóa là số nguyên.
  + **Quy ước về mối quan hệ:** Entity Framework cung cấp các quy ước khác nhau để nhận biết 1 mỗi quan hệ giữa 2 model dựa vào tên của thuộc tính và kiểu dữ liệu.

## 2. ASP.NET MVC:

* **ASP.Net:** là framework lập trình web được cung cấp bởi Microsoft và được sử dụng để tạo các ứng dụng dựa trên web.
  + - Ngôn ngữ: các loại ngôn ngữ tồn tại cho khung .net. Bao gồm VB.net, C# có thể được sử dụng để phát triển các ứng dụng web.
    - Thư viện: bao gồm các thư viện lớp tiêu chuẩn.
    - Thời gian chạy ngôn ngữ chung (CLR): các chương trình .net đều được chạy trên nền tảng này.(thiếu)
* **MVC:** hỗ trợ với cả HTML, CSS, JavaScript tốt so với web form. Mô hình lập trình gồm 3 phần:
  + - **Model:** chứa các dữ liệu giúp cho việc luân chuyển dữ liệu tốt hơn.
    - **View:** nhận dữ liệu để hiện thị cho người dùng, đồng thời cũng là trang form giúp cho người dùng tương tác.(thiếu search mvc mà xem)
    - **Controller:** điều hướng, nhận request của người dùng từ View sau đó nếu cần xử lý logic thì sẽ thông qua Model rồi sẽ lại trả về cho View hiển thị ra cho người dùng.

**2.1 Sử dụng chú thích (Data Annotation)**

Dùng để thêm phần ý nghĩa mở rộng vào dữ liệu thông qua các thẻ thuộc tính. Thuận lợi của tính năng này là giúp chúng ta có thể quản lý dữ liệu được định nghĩa trong một vị trí hoặc không cần phải viết lại nhiều điều luật trên nhiều vị trí khác nhau. Các thuộc tính bao gồm:

* **Validation Attributes (thuộc tính xác thực)**: Dùng để thêm các tập luật xác nhận cho dữ liệu.
* **Display Attributes**: xác định dữ liệu sẽ hiện trị như thế nào ở giao diện.
* **Modelling Attributes**: dùng để đặc tả mục đích sử dụng của lớp thành viên và mối quan hệ giữa các lớp.

**(thêm vô cái này trên mạng dài vãi lzz ra)**

**2.2 Xác nhận dữ liệu (Model Validation)**

Kiểm tra input của User có hợp lệ hay không. Giúp kiểm tra và hiển thị lỗi nếu cần. Việc xác nhận dữ liệu có thể diễn ra ở 2 nơi:

* **Client-side validation:** dữ liệu được kiểm tra ngay ở trên trình duyệt mà không cần phải gửi về server giúp tiết kiệm được băng thông và tăng sự trải nghiệm thay vì phải đợi request đến server và trả về.
* **Server-side validation:** dữ liệu được kiểm tra ở servertrong trường hợp các javascript trên trình duyệt bị disable hoặc là không thể chạy. Ngoài ra còn giúp những dữ liệu mà ở cilent-side không thể xác nhận được thì sẽ được server xác nhận lại.

Ngoài ra ta còn có **ModelState** là nơi lưu chỉ tiết các thuộc tính của các giá trị được cập nhật lên model và các thông tin lỗi trong quá trình validation xuất hiện trong mỗi thuộc tính.

Những validation attributes thường dùng: Required, MinLength, MaxLength, Range, Datatype, RegularExpression, Url, Compare.

???

## 3. Web API:

ASP.NET Web API là framework mở rộng để xây dựng các HTTP services có thể truy cập vào các ứng dụng khác nhau trên nhiểu nền tảng khác nhau. Hội tụ đủ các điều kiện của REST được tối ưu cho các dịch vụ trực tuyến cũng như ứng dụng hiện nay.

Ưu điểm của Web API:

* Dễ viết, tinh chỉnh tốt hơn so với các framework WCF, Web service trước đây.
* Nâng cao trải nghiệm người dùng.
* Thỏa mãn những tiêu chuẩn phổ biến về REST và HTTP.
* Hỗ trợ tốt các tính năng và thành phần của HTTP như: HttpRequestMessage, HttpResponMessage,.. các MVC như: routing, controller, container,…
* Khả năng bảo mật cao, xác nhận 2 chiều khi có Request.
* Dữ liệu trả về có nhiều định dạng khác nhau từ XML đến JSON.

# VI. REFERENCE

1. <https://www.chartjs.org/docs/latest/>
2. <https://www.howkteam.vn/learn/lap-trinh/lap-trinh-c-net-7-5>
3. <https://www.w3schools.com/>
4. <https://weblog.west-wind.com/posts/2007/Sep/24/ASPNET-and-Styles-CSS-Embedding>
5. <https://stackoverflow.com/>
6. <https://www.codemag.com/>
7. <https://dotnet.microsoft.com/apps/aspnet/mvc>
8. <https://viblo.asia/newest>
9. <https://xuanthulab.net/asp-net-core-mvc-tao-ung-dung-mvc-dau-tien-c-csharp.html>