

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**



**NGUYỄN THANH CHƯƠNG**

**XÂY DỰNG WEBSITE QUẢN LÝ BÁN CÀ PHÊ**

**ĐỒ ÁN**

**NGÀNH KHOA HỌC MÁY TÍNH**

# **TRƯỜNG ĐẠI HỌC MỞ TP.HCM**

## **KHOA CÔNG NGHỆ THÔNG TIN**



### **ĐỒ ÁN NGÀNH**

### **CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH**

---

## **XÂY DỰNG WEBSITE QUẢN LÝ BÁN CÀ PHÊ**

---

Giảng viên hướng dẫn:

Lê Ngọc Hiếu

Sinh viên thực hiện:

Nguyễn Thanh Chương

Mã số Sinh Viên:

1751012007

Lớp:

DH17TH04

Khóa:

2017 – 2021

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

## **LỜI CẢM ƠN**

---

---

Để có thể hoàn thiện đồ án tốt nghiệp này, cũng như các kết quả nghiên cứu của đồ án này, em xin trân trọng cảm ơn thầy **Ths. Lê Ngọc Hiếu** (Giảng viên hướng dẫn chính thức). Thầy đã tận tình và giúp đỡ cho em trong suốt quá trình nghiên cứu và thực hiện đồ án này.

Ngoài ra, em xin chân thành cảm ơn công lao quý báu của quý thầy cô trong khoa Công Nghệ Thông Tin trường đại học Mở TP.HCM đã dốc hết sức nhiệt tình chỉ dạy truyền đạt tận tình các kiến thức của mình đến với chúng em những năm vừa qua. Với những kiến thức quý báu mà thầy cô đã truyền đạt đó là nền tảng to lớn để cho em có thể thực hiện và nghiên cứu đồ án, không những thế nó còn là hành trang đắt giá giúp em vững bước trên con đường phát triển sự nghiệp sau này.

Và cũng xin cảm ơn sâu sắc sự động viên quan tâm giúp đỡ của gia đình và các bạn sinh viên cùng lớp học tập đã ủng hộ, động viên và giúp đỡ em vượt qua khó khăn trong suốt quá trình thực hiện đồ án.

Suốt quá trình thực hiện đồ án em đã thực hiện đồ án một cách nghiêm túc và cố gắng nhưng do một phần kiến thức hạn chế nên việc thực hiện đồ án không tránh khỏi những thiếu sót. Em mong sẽ nhận được những lời khuyên và ý kiến đóng góp của Thầy/Cô để đồ án này được hoàn chỉnh;

*Trân trọng cảm ơn!*

TP.HCM, ngày 11 tháng 11 năm 2020

*Sinh viên thực hiện đồ án*

**Nguyễn Thanh Chương**

## **LỜI CAM ĐOAN**

---

---

Tôi cam đoan rằng đề án: “Xây dựng website quản lý bán cà phê” là đề tài do tôi thực hiện.

Ngoại trừ những tài liệu tham khảo được trích dẫn trong đề án này, Tôi cam đoan rằng toàn phần hay những phần nhỏ của đề án này chưa từng được công bố hoặc được sử dụng để nhận bằng cấp ở những nơi khác.

Không có sản phẩm/nghiên cứu nào của người khác được sử dụng trong đề án này mà không được trích dẫn theo đúng quy định.

Đề án này chưa bao giờ được nộp để nhận bất kỳ bằng cấp nào tại các trường đại học hoặc cơ sở đào tạo khác.

TP.HCM, ngày 11 tháng 11 năm 2020

*Sinh viên thực hiện đề án*

*Nguyễn Thanh Chương*

SVTH: Nguyễn Thanh Chương

GVHD:Lê Ngọc Hiếu

## NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

[illegible]

Kết luận:-----

TP.HCM, ngày -- tháng -- năm 20--

GIÁO VIÊN HƯỚNG DẪN

(ký tên)

-----

**NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN**

TP.HCM, ngày -- tháng -- năm 20--

NGƯỜI NHẬN XÉT

(ký tên)

-----

SVTH: Nguyễn Thanh Chương

GVHD:Lê Ngọc Hiếu

## NHẬN XÉT CHUNG VỀ QUÁ TRÌNH LÀM VIỆC CỦA SINH VIÊN

[illegible]

## MỤC LỤC

-----

DANH MỤC HÌNH.....	1
DANH MỤC TỪ VIẾT TẮT.....	3
LỜI MỞ ĐẦU.....	4
Chương 1. Tổng quan về đồ án.....	5
1.1. Thương mại điện tử là gì?.....	5
1.2. Kinh doanh quán cà phê.....	5
1.3. Lý do chọn đề tài.....	7
1.4. Mục tiêu nghiên cứu.....	7
1.5. Đối tượng nghiên cứu.....	7
1.6. Phương tiện nghiên cứu.....	7
1.7. Phạm vi đề tài.....	8
Chương 2. Những cơ sở lý thuyết của đề tài.....	8
2.1. Khái niệm cơ sở dữ liệu và hệ quản trị cơ sở dữ liệu.....	8
2.1.1. Cơ sở dữ liệu là gì?.....	8
2.1.2. Hệ quản trị cơ sở dữ liệu là gì?.....	9
2.2. Kiến trúc ado .net entity framework.....	10
2.2.1. Các tính năng chính ADO.NET Entity framework.....	11
2.3. ASP.NET CORE.....	12
2.3.1. Tổng quan về ASP.NET Core.....	12
2.4. RESTful API.....	12
2.4.1. RESTful API là gì?.....	12
2.4.2. RESTful api hoạt động như thế nào?.....	13
2.4.3. JSON.....	13
2.5. REACJS.....	14
2.5.1. Giới thiệu tổng quan về ReactJs.....	14
2.5.2. Single Page Application.....	15



2.6. VISUAL STUDIO CODE.....	16
2.6.1. Tổng quan về Visual Studio Code.....	16
2.6.2. Một số điểm nổi bật.....	16
Chương 3. Mô tả hệ thống.....	18
3.1. Thiết kế và xây dựng cơ sở dữ liệu.....	18
3.1.1. Các chức năng và yêu cầu đặt ra.....	18
3.1.2. Các thực thể.....	18
3.1.3. Các mối kết hợp.....	23
3.1.4. Các mô hình cơ sở dữ liệu.....	25
Chương 4. Giao diện phần mềm.....	28
4.1. Giao diện người dùng chung.....	28
4.1.1. Giao diện trang chủ.....	28
4.1.2. Giao diện đăng nhập.....	29
4.1.3. Giao diện đăng ký.....	30
4.1.4. Giao diện Menu đặt món.....	32
4.1.5. Giao diện xác nhận đặt hàng.....	35
4.1.6. Giao diện trang thông tin khách hàng.....	36
4.2. Giao diện phân hạng admin.....	37
Chương 5. Kết luận và hướng phát triển.....	41
5.1. Kết quả đạt được.....	41
5.1.1. Về kiến thức và học tập.....	41
5.1.2. Về website.....	41
5.2. Hạn chế của đề tài.....	41
5.3. Hướng phát triển.....	42
DANH SÁCH TÀI LIỆU THAM KHẢO.....	43

**DANH MỤC HÌNH**

Hình 1.1 Thống kê thị hiếu sử dụng cà phê.....	6
Hình 1.2 Thống kê địa điểm uống cà phê phổ biến.....	6
Hình 2.1 Cấu trúc hoạt động của ADO .NET Entity Framework.....	10
Hình 2.2 Cấu trúc RESTful API.....	13
Hình 2.3 Cấu trúc Json format cơ bản.....	14
Hình 2.4 ReactJS.....	14
Hình 2.5 Kiến trúc Single Page Application.....	16
Hình 2.6 Trình gỡ lỗi Visual Studio Code.....	17
Hình 2.7 Terminal Visual Studio Code.....	17
Hình 3.1 Bảng thực thể người dùng.....	19
Hình 3.2 Bảng thực thể sản phẩm.....	20
Hình 3.3 Bảng thực thể danh mục sản phẩm.....	20
Hình 3.4 Bảng thực thể đơn hàng.....	21
Hình 3.5 Bảng thực thể chi tiết đơn hàng.....	21
Hình 3.6 Bảng thực thể vai trò.....	22
Hình 3.7 Bảng thực thể giảm giá.....	22
Hình 3.8 Mối kết hợp User - Order.....	23
Hình 3.9 Mối kết hợp Category - Product.....	23
Hình 3.10 Mối kết hợp User - Role.....	24
Hình 3.11 Mối kết hợp Order - Discount.....	24
Hình 3.12 Mô hình CDM website quản lý và đặt cà phê.....	25
Hình 3.13 Mô hình PDM của website quản lý và đặt cà phê.....	26
Hình 3.14 Mô hình OOM của website quản lý và đặt cà phê.....	27
Hình 4.1 Giao diện trang chủ.....	28
Hình 4.2 Giao diện đăng nhập.....	29
Hình 4.3 Giao diện đăng nhập thất bại.....	30
Hình 4.4 Giao diện đăng kí thành công.....	31

Hình 4.5 Ví dụ về hàm băm MD5.....	31
Hình 4.6 Giao diện menu đặt món.....	32
Hình 4.7 Giao diện hiển thị sản phẩm theo danh mục.....	33
Hình 4.8 Giao diện khi người dùng thêm sản phẩm vào đơn hàng.....	33
Hình 4.9 Giao diện giỏ hàng.....	34
Hình 4.10 Giao diện xác nhận đặt hàng.....	35
Hình 4.11 Hệ thống xác nhận đặt hàng.....	35
Hình 4.12 Giao diện thông tin khách hàng.....	36
Hình 4.13 Giao diện xem các đơn hàng đã đặt .....	37
Hình 4.14 Giao diện chi tiết đơn hàng đã đặt.....	37
Hình 4.15 Giao diện trang admin.....	38
Hình 4.16 Giao diện quản lý tài khoản.....	38
Hình 4.17 Giao diện quản lý danh mục sản phẩm.....	39
Hình 4.18 Giao diện quản lý sản phẩm.....	39
Hình 4.19 Giao diện quản lý đơn hàng.....	40
Hình 4.20 Giao diện quản lý chi tiết đơn hàng.....	40

**DANH MỤC TỪ VIẾT TẮT**

<b>Từ viết tắt</b>	<b>Giải thích</b>
API	Application Programming Interface
IDE	Integrated Development Environment
MVC	Model-View-Controller
DBMS	Database management system
EF	Entity Framework
API	Application Programming Interface

## LỜI MỞ ĐẦU

Với cuộc sống hiện đại và lối sống nhộn nhịp bận rộn trong thời đại 4.0 hiện nay. Việc mà chúng ta có thể bỏ thời gian ra để lê la các hàng quán cà phê là điều không dễ dàng gì. Để có thể thưởng thức một ly cà phê thơm ngon một thứ thức uống giúp chúng ta tràn đầy năng lượng để tiếp tục công việc phải tốn khá nhiều thời gian để đi ra hàng quán và mua một ly cà phê mang về. Điều đó càng khó khăn hơn trong thời buổi dịch Covid-19 hiện nay người dùng e ngại đến những nơi đông người như quán cà phê. Việc này dẫn đến doanh thu của các cửa hàng cà phê cổ điển giảm doanh thu trầm trọng. Do đó việc đổi mới thay đổi cách thức kinh doanh là điều phải thực hiện sớm để có thể chuyển mình tận dụng cơ hội trong lúc khó khăn như hiện nay. Vì thế website quản lý và bán cà phê trực tuyến được hình thành. Với mô hình kinh doanh trực tuyến đáp ứng đủ các nhu cầu cũng như khắc phục được các khó khăn mà loại hình kinh doanh cổ điển gặp phải. Không những website giúp khách hàng có thể đặt được một ly cà phê và có thưởng thức tại nhà mà website còn cung cấp cho chủ cửa hàng một giao diện trực quan để có thể quản lý cửa hàng hiệu quả hơn.

Website được xây dựng trên kiến trúc MVC mô hình ba lớp, sử dụng công cụ phát triển phần mềm Microsoft Visual Studio 2019 sử dụng nền tảng công nghệ ADO.NET CORE và Entity Framework để tương tác với cơ sở dữ liệu. Đồng thời sử dụng công cụ quản trị hệ cơ sở dữ liệu Microsoft SQL Sever để thiết kế và xây dựng cơ sở dữ liệu.

## **Chương 1. Tổng quan về đề án**

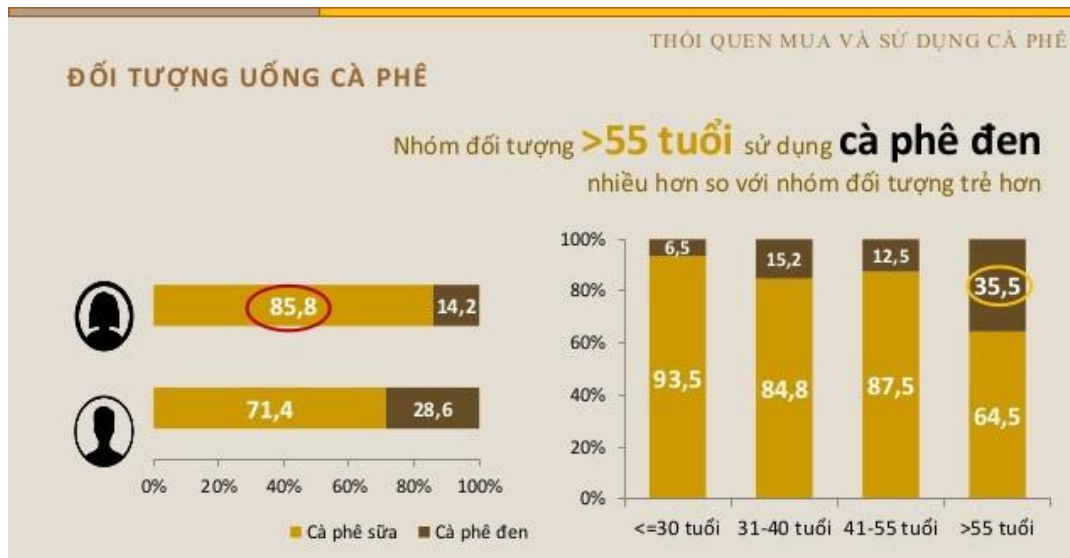
### **1.1. Thương mại điện tử là gì?**

E-commerce ( thương mại điện tử ) là hoạt động mà việc mua bán sản phẩm trên dịch vụ Internet. Thương mại điện tử hoạt động dựa trên các công nghệ như thương mại di động, chuyển tiền điện tử, quản lý chuỗi cung ứng, quảng cáo online, xử lý giao dịch trực tuyến và hệ thống thu thập thông tin dữ liệu tự động. Thương mại điện tử được thúc đẩy bởi những tiến bộ công nghệ của ngành công nghiệp bán dẫn, và là lĩnh vực lớn nhất của ngành công nghiệp điện tử.

Thương mại điện tử, có thể được thực hiện trên máy tính, máy tính bảng hoặc điện thoại thông minh có thể được coi như một phiên bản kỹ thuật số của mua sắm theo danh mục đặt hàng qua thư. Gần như mọi sản phẩm và dịch vụ có thể tưởng tượng đều có sẵn thông qua các sàn giao dịch thương mại điện tử, bao gồm sách, âm nhạc, vé máy bay và các dịch vụ tài chính như đầu tư chứng khoán và ngân hàng trực tuyến. Hiện nay, nó được coi là một công nghệ rất đột phá.

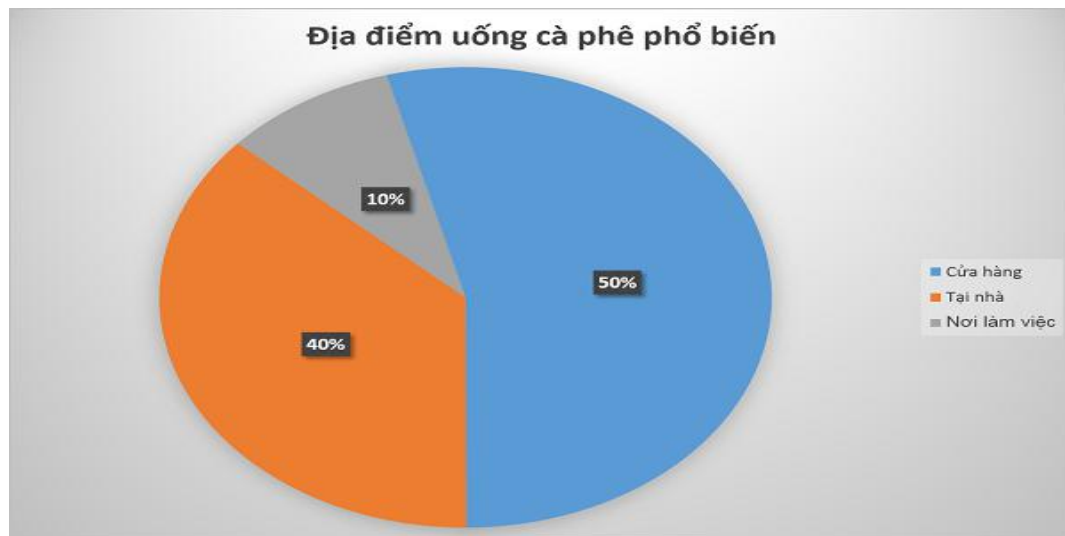
### **1.2. Kinh doanh quán cà phê**

Hiện nay việc tìm một quán cà phê để có thể đáp ứng nhu cầu giải khát cũng có thể để trò chuyện với bạn bè, làm việc, học nhóm là không khó. Nhu cầu của xã hội hoạt động không ngừng nghỉ như hiện nay với cà phê ngày càng tăng. Thị hiếu của khách hàng theo đó mà thay đổi nhanh chóng. Để đáp ứng nhu cầu đó các quán cà phê cũng theo đó mọc lên như nấm sau mưa với muôn hình vạn trạng.



Hình 1.1 Thống kê thị hiếu sử dụng cà phê

Tuy nhiên giới với nhịp sống hiện nay việc đi ra quán và thưởng thức cà phê là một việc vô cùng xa xỉ. Các năm về trước khó mà có thể tưởng tượng được việc bạn có thể ở nơi làm việc hoặc tại nhà và đặt một ly cà phê để thưởng thức. Tuy nhiên với sự phát triển của công nghệ việc đó trở thành một điều đơn giản. Chỉ cần một vài thao tác đơn giản trên website và chờ ít phút bạn có thể có ngay một ly cà phê tại nơi mà không cần đi đâu xa. Việc này dần dần trở thành một thói quen đối với mọi người vì tính tiện dụng của nó.



Hình 1.2 Thống kê địa điểm uống cà phê phổ biến

### 1.3. Lý do chọn đề tài

Với nhu cầu như trên việc mở một quán cà phê có áp dụng thương mại điện tử là một điều tất yếu nếu muốn sống sót trên thị trường. Đó là lí do lớn nhất để em chọn đề tài này với mục đích giúp các chủ cửa hàng giải quyết những vấn đề về quản lý cũng như kinh doanh.

### 1.4. Mục tiêu nghiên cứu

Website hỗ trợ trong việc quản lý các công việc như quản lý các đơn hàng, quản lý thông tin khách hàng, quản lý nhân viên, quản lý các sản phẩm, cũng như giúp người dùng có thể thao tác dễ dàng tiết kiệm thời gian, dễ sử dụng và quản lý.

- Xây dựng một website có tính tiện dụng, giao diện thân thiện với người dùng.
- Giúp chủ cửa hàng quản lý thuận tiện, tối ưu và tiết kiệm thời gian hơn.
- Thực hành được các kỹ năng đã học vào đề tài thực tiễn, tăng khả năng xử lý tư duy thông qua các vấn đề gặp phải trong bài.

### 1.5. Đối tượng nghiên cứu

- Đối tượng nghiên cứu: Website quản lý và đặt cà phê
- Khách thể nghiên cứu:
  - + Những chủ sở hữu các quán cà phê
  - + Quy trình nghiệp vụ bán cà phê
  - + Nghiệp vụ quản lý cửa hàng

### 1.6. Phương tiện nghiên cứu

- Nghiên cứu cơ sở lý thuyết về thiết kế cơ sở dữ liệu, mô hình ba lớp MVC, Entity Framework, ReactJs.
- Lập trình hướng đối tượng trong C#.
- Sử dụng các IDE phổ biến để lập trình giao diện và triển khai website.
- Sử dụng hệ quản trị cơ sở dữ liệu SQL Sever để quản lý và thiết kế cơ sở dữ liệu



## 1.7. Phạm vi đề tài

Phạm vi đề tài gói gọn trong các nghiệp vụ quản lý cửa hàng. Sử dụng ASP.NetCore, API Controller ở phía backend và ReactJs ở phía giao diện tương tác với người dùng (Frontend)

# Chương 2. Những cơ sở lý thuyết của đề tài

## 2.1. Khái niệm cơ sở dữ liệu và hệ quản trị cơ sở dữ liệu

### 2.1.1. Cơ sở dữ liệu là gì?

Cơ sở dữ liệu là một tập hợp có tổ chức của các thông tin có cấu trúc hoặc dữ liệu, thường được lưu trữ dưới dạng điện tử trong hệ thống máy tính. Cơ sở dữ liệu thường được kiểm soát bởi hệ quản trị cơ sở dữ liệu (DBMS). Cùng với nhau, dữ liệu và DBMS, và cùng với các ứng dụng được liên kết với chúng, được gọi là một hệ thống cơ sở dữ liệu, thường được rút gọn và gọi là cơ sở dữ liệu.

Dữ liệu trong các loại cơ sở dữ liệu phổ biến nhất đang hoạt động ngày nay thường được mô hình hóa thành các hàng và cột trong một loạt bảng để giúp xử lý và truy vấn dữ liệu hiệu quả. Dữ liệu sau đó có thể dễ dàng truy cập, quản lý, sửa đổi, cập nhật, kiểm soát và tổ chức. Hầu hết các cơ sở dữ liệu sử dụng ngôn ngữ truy vấn có cấu trúc (SQL) để viết và truy vấn dữ liệu.

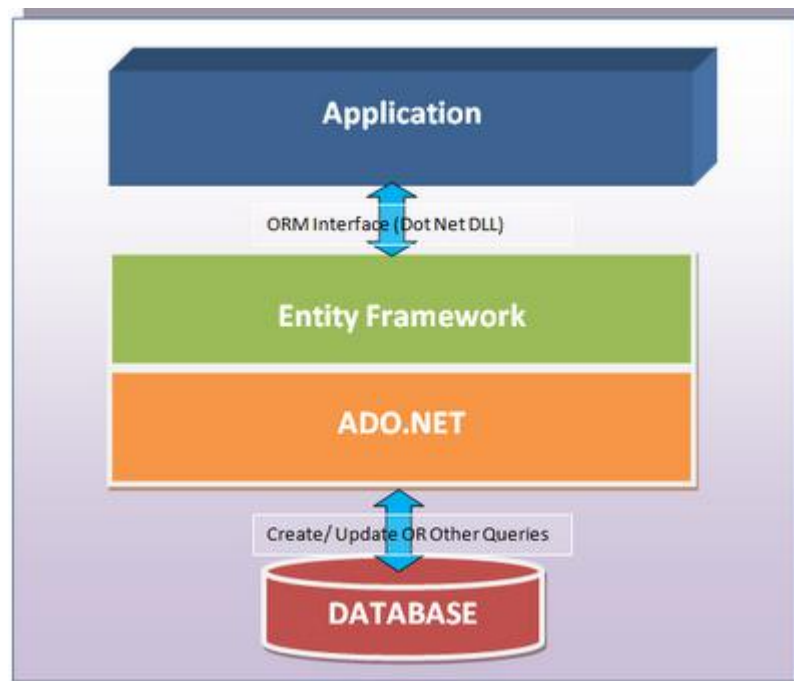
SQL là ngôn ngữ lập trình được gần như tất cả các cơ sở dữ liệu quan hệ sử dụng để truy vấn, thao tác và định nghĩa dữ liệu cũng như cung cấp quyền kiểm soát truy cập. SQL được phát triển lần đầu tiên tại IBM vào những năm 1970 với Oracle là nhà đầu tư chính, dẫn đến việc triển khai tiêu chuẩn SQL ANSI, SQL đã thúc đẩy nhiều phần mở rộng từ các công ty như IBM, Oracle và Microsoft. Mặc dù SQL vẫn được sử dụng rộng rãi ngày nay, nhưng các ngôn ngữ lập trình mới cũng đang bắt đầu xuất hiện.

### 2.1.2. Hệ quản trị cơ sở dữ liệu là gì?

Hệ quản trị cơ sở dữ liệu (DBMS) là một phần mềm để lưu trữ và truy xuất dữ liệu của người dùng. Nó bao gồm một nhóm các chương trình thao tác với cơ sở dữ liệu. DBMS chấp nhận yêu cầu dữ liệu từ một ứng dụng và hướng dẫn hệ điều hành cung cấp dữ liệu cụ thể. Trong các hệ thống lớn, DBMS giúp người dùng và phần mềm của bên thứ ba khác lưu trữ và truy xuất dữ liệu. DBMS cho phép người dùng tạo cơ sở dữ liệu của riêng họ theo yêu cầu của họ. Thuật ngữ “DBMS” bao gồm người dùng cơ sở dữ liệu và các chương trình ứng dụng khác. Nó cung cấp một giao diện giữa dữ liệu và ứng dụng phần mềm.

SQL Server Management Studio (SSMS) là một môi trường tích hợp để quản lý bất kỳ cơ sở hạ tầng SQL nào, từ SQL Server đến Azure SQL Database. SSMS cung cấp các công cụ để cấu hình, giám sát và quản trị các phiên bản của SQL Server và cơ sở dữ liệu. Sử dụng SSMS để triển khai, giám sát và nâng cấp các thành phần tầng dữ liệu được các ứng dụng của bạn sử dụng cũng như xây dựng các truy vấn và tập lệnh. Sử dụng SSMS để truy vấn, thiết kế và quản lý cơ sở dữ liệu và kho dữ liệu của bạn, cho dù chúng ở đâu - trên máy tính cục bộ của bạn hoặc trên cloud

## 2.2. Kiến trúc ado .net entity framework



Hình 2.1 Cấu trúc hoạt động của ADO .NET Entity Framework

Các nhà phát triển dành quá nhiều thời gian để lo lắng về cơ sở dữ liệu của họ, các bảng của nó và các mối quan hệ của chúng, tên và tham số của các thủ tục và views, cũng như lược đồ dữ liệu mà chúng trả về. Entity Framework của Microsoft thay đổi cuộc chơi dành cho các nhà phát triển .NET để chúng ta không còn phải lo lắng về các chi tiết của dữ liệu khi chúng ta viết các ứng dụng của mình. Chúng ta có thể tập trung vào nhiệm vụ viết các ứng dụng của mình hơn là truy cập dữ liệu.

ADO.NET Entity Framework là một nền tảng truy cập dữ liệu mới của Microsoft để viết các ứng dụng .NET. Nó được phát hành vào tháng 7 năm 2008 như một phần của Visual Studio 2008 Service Pack 1 và .NET 3.5 Service Pack 1, hai năm sau khi Microsoft công bố nó tại Hội nghị TechEd 2006 của mình.

Mặc dù các tính năng truy cập dữ liệu hiện có vẫn còn trong ADO.NET, nhưng framework mới này là một phần của chiến lược truy cập dữ liệu cốt lõi của Microsoft trong tương lai.

### 2.2.1. Các tính năng chính ADO.NET Entity framework

- Cross-platform: EF Core là một khuôn khổ đa nền tảng có thể chạy trên Windows, Linux và Mac.
- Modelling: EF (Entity Framework) tạo EDM (Entity Data Model) dựa trên các thực thể POCO (Plain Old CLR Object) với các thuộc tính get / set của các kiểu dữ liệu khác nhau. Nó sử dụng mô hình này khi truy vấn hoặc lưu dữ liệu thực thể vào cơ sở dữ liệu bên dưới.
- Querying: EF cho phép chúng ta sử dụng các truy vấn LINQ (C # / VB.NET) để lấy dữ liệu từ cơ sở dữ liệu bên dưới. Database Provider sẽ dịch các truy vấn LINQ này sang ngôn ngữ truy vấn dành riêng cho cơ sở dữ liệu (ví dụ: SQL cho cơ sở dữ liệu quan hệ). EF cũng cho phép chúng ta thực thi các truy vấn SQL trực tiếp đến cơ sở dữ liệu.
- Saving: EF thực hiện các lệnh INSERT, UPDATE và DELETE đối với cơ sở dữ liệu dựa trên các thay đổi xảy ra với các thực thể của ta khi ta gọi phương thức SaveChanges(). EF cũng cung cấp phương thức bất đồng bộ SaveChangesAsync() .
- Concurrency: EF sử dụng Optimistic Concurrency để bảo vệ các thay đổi do người dùng khác thực hiện ghi đè từ khi dữ liệu được tìm nạp từ cơ sở dữ liệu.
- Transactions: EF cung cấp trình quản lý tự động các phiên hoạt động khi truy vấn hoặc lưu dữ liệu. Nó cũng cung cấp các tùy chọn để tùy chỉnh quản lý phiên hoạt động.
- Migrations: EF cung cấp một tập hợp các lệnh có thể được thực thi trên NuGet Package Manager Console hoặc The Command Line Interface để tạo hoặc quản cơ sở dữ liệu bên dưới.

## **2.3. ASP.NET CORE**

### **2.3.1. Tổng quan về ASP.NET Core**

ASP.NET Core là phiên bản mới của ASP.NET web framework với mục đích chủ yếu là để triển khai trên nền tảng .NET Core.

ASP.NET Core miễn phí và trở thành một mã nguồn mở và đa nền tảng để xây dựng các ứng dụng dựa trên đám mây, chẳng hạn như ứng dụng web, ứng dụng IoT và mobile backends. Nó được thiết kế để chạy trên đám mây hoặc on-premise.

Giống như .NET Core, nó được kiến trúc theo mô-đun với tài nguyên tối thiểu và sau đó các tính năng nâng cao khác có thể được thêm vào bằng NuGet Package Manager Console theo yêu cầu của ứng dụng. Điều này dẫn đến hiệu suất cao, yêu cầu ít bộ nhớ hơn, chi phí triển khai ít hơn và dễ bảo trì.

## **2.4. RESTful API**

### **2.4.1. RESTful API là gì?**

API (Application Programming Interface) Nó là một tập hợp các quy tắc cho phép các chương trình tương tác với nhau. Lập trình viên tạo API trên máy chủ và cho phép người dùng tương tác với nó.

REST định nghĩa API. Nó là viết tắt của “Chuyển đổi cấu trúc dữ liệu”. Đó là một tập hợp các quy tắc mà các lập trình viên tuân theo khi họ tạo API của mình. Một trong những quy tắc này quy định rằng bạn sẽ có thể lấy một phần dữ liệu (được gọi là tài nguyên) khi bạn liên kết đến một URL cụ thể.

Với mỗi URL được gọi là request và dữ liệu được trả về gọi là response.

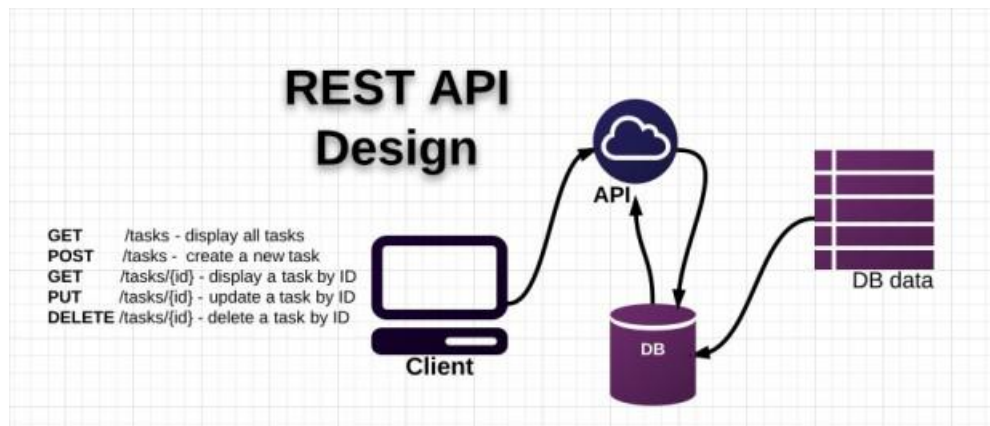
API RESTful là một kiểu kiến trúc cho phép API sử dụng các HTTP request để truy cập và sử dụng dữ liệu. Dữ liệu đó có thể được sử dụng bằng cách dùng các phương thức HTTP GET, PUT, POST và DELETE, để có thể đọc, cập nhật, tạo và xóa các thao tác liên quan đến tài nguyên.

### 2.4.2. RESTful api hoạt động như thế nào?

API RESTful sử dụng các lệnh để lấy tài nguyên. API RESTful sử dụng các phương thức HTTP hiện có được xác định bởi giao thức RFC 2616, chẳng hạn như:

- GET để lấy một tài nguyên.
- PUT để thay đổi trạng thái hoặc cập nhật tài nguyên, có thể là một đối tượng hoặc tệp.
- POST để tạo tài nguyên đó
- DELETE để loại bỏ nó.

Các phương thức trên được gọi là CRUD( Create - Read - Update - Delete ) tạo, đọc , sửa, xóa.



Hình 2.2 Cấu trúc RESTful API

### 2.4.3. JSON

JSON (JavaScript Object Notation) là một kiểu định dạng dữ liệu. Để mang lại sự tiện dụng trong việc viết và đọc. JSON là một định dạng văn bản hoàn toàn độc lập với ngôn ngữ nhưng sử dụng các quy ước quen thuộc với các lập trình viên thuộc họ ngôn ngữ C, bao gồm C, C ++, C #, Java, JavaScript, Perl, Python và nhiều thứ khác. Các thuộc tính này làm cho JSON trở thành một ngôn ngữ trao đổi dữ liệu lý tưởng.

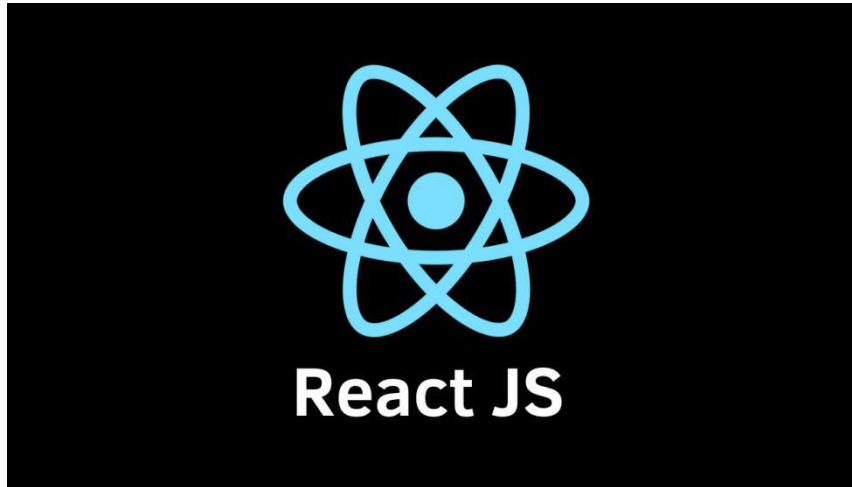
Cấu trúc của JSON có hai phần đó là key và value. Thường được bao bọc bằng cặp dấu {}. Các phần key và value bắt buộc phải bọc bằng cặp dấu nháy đôi.

```
{
  "status": "0, Good, the operation completed successfully.",
  "info": "Position Loading wagon, mm",
  "calvints": 1499990404234,
  "sourcets": "2017-07-14 00:00:04.064000",
  "value": "-10.0036686659",
  "id": "ns=2;s=/Nck/MachineAxis/aaIm [3]",
  "tag": "aaIm_Z",
  "serverts": "2017-07-14 00:00:04.124000",
  "type": "Double",
  "origin": "GU-1"
}
```

Hình 2.3 Cấu trúc Json format cơ bản

## 2.5. REACJS

### 2.5.1. Giới thiệu tổng quan về ReactJs



Hình 2.4 ReactJS

React là một trong những thư viện JavaScript phổ biến nhất trên thế giới, nhưng nó không thú vị vì nó phổ biến. Nó phổ biến vì nó thú vị.

React.js xây dựng giao diện người dùng cho các ứng dụng web single page bằng cách chia giao diện người dùng thành các phần tử có thể kết hợp. React.js là một thư viện giao diện người dùng, ban đầu được phát triển như một thư viện nội bộ cho Facebook, và sau đó nó có nguồn mở vào năm 2013. Nó nổi tiếng và chủ yếu được sử dụng vì nhiều lý do.

Một trong những lý do chính là nó có thể gom nhóm các phần tử HTML và tái sử dụng nhiều lần.

### 2.5.2. Single Page Application

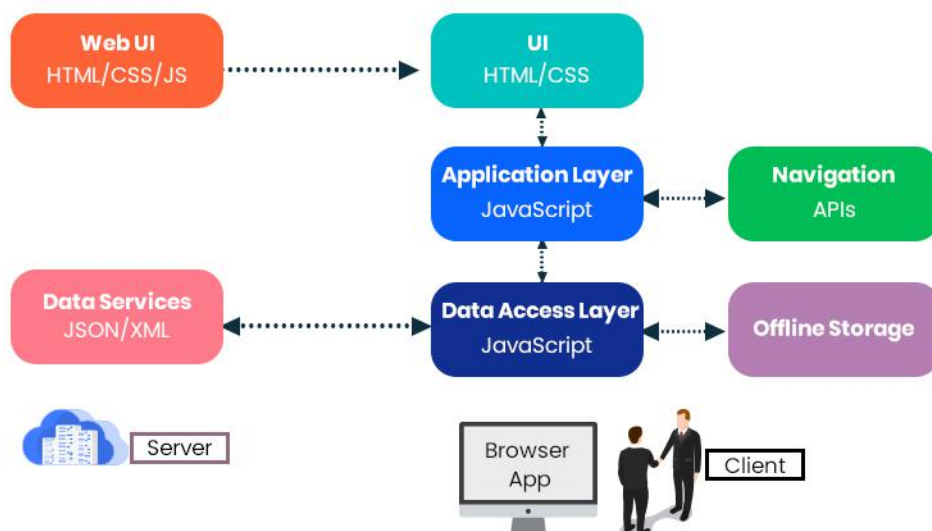
Dã sử một trong những người bạn của bạn đã đăng một bức ảnh lên Facebook. Bây giờ bạn đi và thích hình ảnh và sau đó bạn cũng bắt đầu xem các nhận xét. Bây giờ, trong khi duyệt qua các nhận xét, bạn thấy rằng số lượt thích đã tăng lên 100 vì bạn đã thích bức ảnh, ngay cả khi không tải lại trang. Sự thay đổi số lượng kỳ diệu này là do Reactjs

React sẽ cập nhật và hiển thị thành phần phù hợp một cách hiệu quả khi dữ liệu của bạn thay đổi.

Đó chính là cách mà Single Page Application hoạt động. Single page application là ứng dụng không cần tải lại trang trong quá trình sử dụng và hoạt động trong trình duyệt. Hãy nghĩ về các ứng dụng ta sử dụng hàng ngày: Facebook, Google Maps, Gmail, Twitter, Google Drive hoặc thậm chí GitHub. Tất cả những điều này là những ví dụ về một SPA.

SPA mang lại trải nghiệm tốt hơn cho người dùng, tăng tốc độ duyệt web giảm thiểu các tài nguyên phát sinh. SPA sẽ tải lên toàn bộ trang web trong lần đầu sau đó dựa trên request của người dùng SPA sẽ tải lên các các phần tương ứng ngay trên trang đó.

## Architecture of SPA





## Hình 2.5 Kiến trúc Single Page Application

### 2.6. VISUAL STUDIO CODE

#### 2.6.1. Tổng quan về Visual Studio Code

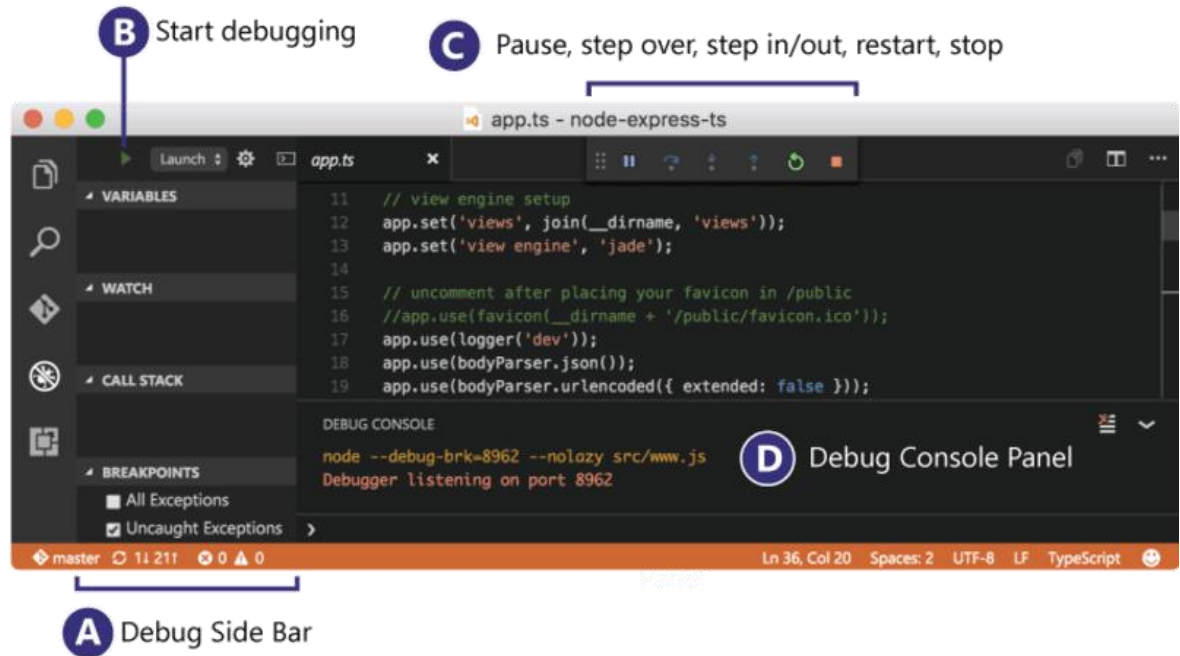
Visual Studio Code là một trình soạn thảo mã rất mạnh mẽ và dễ sử dụng. Nó đi kèm với sự hỗ trợ rộng rãi mới mọi ngôn ngữ lập trình, có khả năng tùy biến cao với nhiều phần mở rộng khác nhau và nó miễn phí. Nó thân thiện và dễ sử dụng đối với cả người lập trình viên mới và những người lâu năm.

#### 2.6.2. Một số điểm nổi bật

**Đặc điểm chính:** Các tính năng chính làm cho VS Code trở nên độc đáo là thanh bên chứa các tính năng cốt lõi mà bạn sẽ tương tác để viết mã và tái cấu trúc. Mọi thứ khác bạn cần như tiện ích mở rộng thì bạn chỉ cần cài đặt một cách dễ dàng.

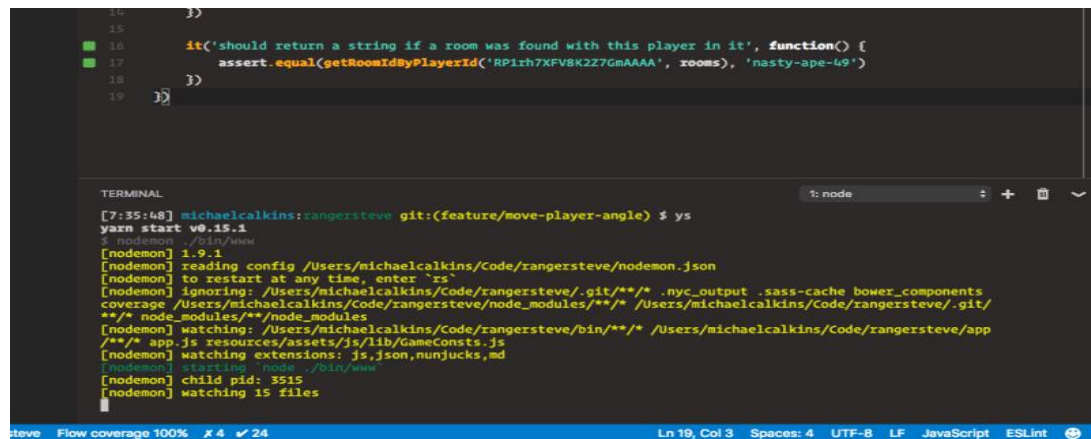
**IntelliSense:** Là công cụ đánh dấu cú pháp và tính năng tự động hoàn thành rất hữu ích cung cấp các cú pháp đúng dựa trên các loại biến, hàm và mô-đun đã khai báo.

**Debugging:** Trình gỡ lỗi tích hợp giúp bạn tăng tốc chỉnh sửa, biên dịch và gỡ lỗi bằng cách thêm các điểm ngắt và trình theo dõi. Theo mặc định, nó hỗ trợ cho NodeJS và có thể gỡ lỗi bất kỳ thứ gì được chuyển sang JavaScript nhưng các runtimes khác như C++ hoặc Python sẽ yêu cầu cài đặt một phần mở rộng.



Hình 2.6 Trình gỡ lỗi Visual Studio Code

Terminal: Khả năng làm việc với một terminal tích hợp đầy đủ tính năng cho phép bạn chạy mọi thứ bạn thường làm từ terminal bạn chọn.



Hình 2.7 Terminal Visual Studio Code

## **Chương 3. Mô tả thiết kế hệ thống của website quản lý và bán cà phê**

### **3.1. Thiết kế và xây dựng cơ sở dữ liệu**

#### **3.1.1. Các chức năng và yêu cầu đặt ra**

- Chức năng đăng nhập và đăng ký tài khoản khách hàng
- Chức năng xem đồ uống và đặt món
- Chức năng quản lý thông tin tài khoản và đơn hàng đã đặt
- Chức năng thanh toán online
- Chức năng thêm xoá sửa tài khoản, sản phẩm, đơn hàng, chi tiết đơn hàng dành cho người quản lý
- Chức năng xuất hoá đơn và thống kê

#### **3.1.2. Các thực thể**

##### ***a. Giới thiệu các thực thể trong hệ thống***

- Thực thể người dùng
- Thực thể danh mục sản phẩm
- Thực thể sản phẩm
- Thực thể đơn hàng
- Thực thể chi tiết đơn hàng
- Thực thể vai trò
- Thực thể khuyến mãi


##### ***b. Chi tiết các thành phần thực thể***

- Thực thể người dùng:

+ Tên thực thể: User

+ Khoá Chính : UserName

- + Khoá Ngoại : RoleId ( Role = 1 tức là người dùng thuộc phân hạng admin  
Role = 2 là phân hạng User )
- + Các thành phần của thực thể gồm có
  - UserName : Tên đăng nhập
  - Password : Mật khẩu đăng nhập
  - FullName: Họ và tên người dùng
  - Phone: Số điện thoại người dùng
  - Location: Địa chỉ người dùng
  - RoleId

User			
	Column Name	Data Type	Allow Nulls
	UserName	varchar(50)	<input type="checkbox"/>
	Password	nchar(50)	<input type="checkbox"/>
	FullName	nvarchar(50)	<input checked="" type="checkbox"/>
	Phone	nchar(50)	<input checked="" type="checkbox"/>
	Location	nvarchar(MAX)	<input checked="" type="checkbox"/>
	RoleId	int	<input type="checkbox"/>
			<input type="checkbox"/>

Hình 3.1 Bảng thực thể người dùng

- Thực thể sản phẩm
- + Tên thực thể: Product
- + Khoá Chính : ProductId
- + Khoá Ngoại : CategoryId
- + Các thành phần của thực thể gồm có
  - ProductId: Mã sản phẩm
  - ProductName: Tên sản phẩm
  - CategoryId: Mã loại sản phẩm
  - UnitPrice: Đơn giá sản phẩm
  - Description: Mô tả sản phẩm

- Hot: Nếu Hot có giá trị bằng 1 thì sản phẩm đang được ưa thích

Product			
	Column Name	Data Type	Allow Nulls
🔑	ProductId	int	<input type="checkbox"/>
	CategoryId	int	<input checked="" type="checkbox"/>
	ProductName	nvarchar(MAX)	<input checked="" type="checkbox"/>
	UnitPrice	bigint	<input checked="" type="checkbox"/>
	Description	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Hot	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3.2 Bảng thực thể sản phẩm

- Thực thể danh mục sản phẩm
- + Tên thực thể: Category
- + Khoá Chính : CategoryId
- + Các thành phần của thực thể gồm có
  - CategoryId: Mã loại sản phẩm
  - CategoryName: Tên loại sản phẩm

Category			
	Column Name	Data Type	Allow Nulls
🔑	CategoryId	int	<input type="checkbox"/>
	CategoryName	nvarchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3.3 Bảng thực thể danh mục sản phẩm

- Thực thể đơn hàng
- + Tên thực thể: Order
- + Khoá Chính : OrderId
- + Khoá Ngoại : UserName, DiscountId
- + Các thành phần của thực thể gồm có

- OrderId: Mã đơn hàng
- UserName: Tên người dùng
- OrderDate: Ngày đặt hàng
- DiscountId: Mã giảm giá
- ShipAddress: Địa chỉ giao hàng

Orders			
	Column Name	Data Type	Allow Nulls
🔑	OrderId	int	<input type="checkbox"/>
	UserName	varchar(50)	<input type="checkbox"/>
	OrderDate	datetime	<input type="checkbox"/>
	DiscountId	int	<input checked="" type="checkbox"/>
	ShipAddress	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

Hình 3.4 Bảng thực thể đơn hàng

- Thực thể chi tiết đơn hàng
- + Tên thực thể: OrderDetail
- + Khoá Chính : OrderId, ProductId
- + Khoá Ngoại : OrderId, ProductId
- + Các thành phần của thực thể gồm có
  - OrderId: Mã đơn hàng
  - ProductId: Mã sản phẩm
  - Quantity: Số lượng

OrderDetails			
	Column Name	Data Type	Allow Nulls
🔑	OrderId	int	<input type="checkbox"/>
🔑	ProductId	int	<input type="checkbox"/>
	Quantity	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3.5 Bảng thực thể chi tiết đơn hàng

- Thực thể vai trò
- + Tên thực thể: Role
- + Khoá Chính : RoleId
- + Các thành phần của thực thể gồm có
  - RoleId: Mã vai trò
  - RoleName: Tên vai trò

Roles			
	Column Name	Data Type	Allow Nulls
🔑	RoleId	int	<input type="checkbox"/>
	RoleName	nchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 3.6 Bảng thực thể vai trò

- Thực thể khuyến mãi
- + Tên thực thể: Discount
- + Khoá Chính : DiscountId
- + Các thành phần của thực thể gồm có
  - DiscountId: Mã giảm giá
  - DiscountName: Tên mã giảm giá
  - DiscountValue: Giá trị giảm giá

Discounts			
	Column Name	Data Type	Allow Nulls
🔑	DiscountId	int	<input type="checkbox"/>
	DiscountName	nvarchar(50)	<input type="checkbox"/>
	DiscountValue	float	<input type="checkbox"/>
			<input type="checkbox"/>

Hình 3.7 Bảng thực thể giảm giá

### 3.1.3. Các mối kết hợp

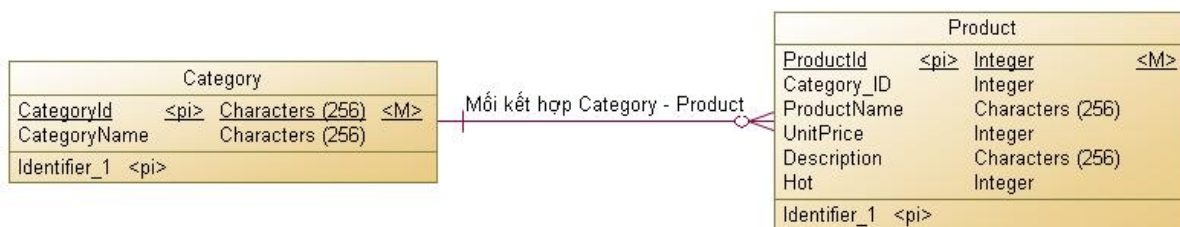
#### a. Mối kết hợp User-Order



Hình 3.8 Mối kết hợp User - Order

Giải thích: Một khách hàng có thể có nhiều đơn hàng, một đơn hàng chỉ thuộc một khách hàng.

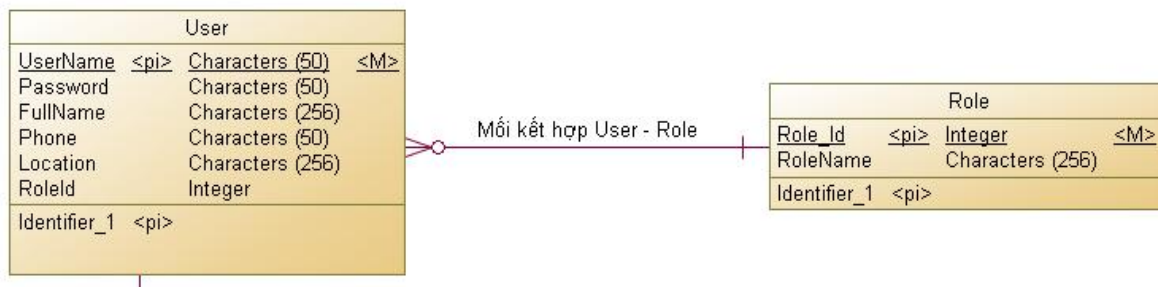
#### b. Mối kết hợp Category - Product



Hình 3.9 Mối kết hợp Category - Product

Giải thích: Một sản phẩm thuộc một loại sản phẩm. Một loại sản phẩm có thể có nhiều sản phẩm

#### c. Mối kết hợp User - Role





Hình 3.10 Mối kết hợp User - Role

Giải thích: Một người dùng chỉ có một vai trò trong hệ thống hoặc là Admin hoặc là User. Một vai trò có thể có nhiều người dùng.

**d. Mối kết hợp Order - Discount**

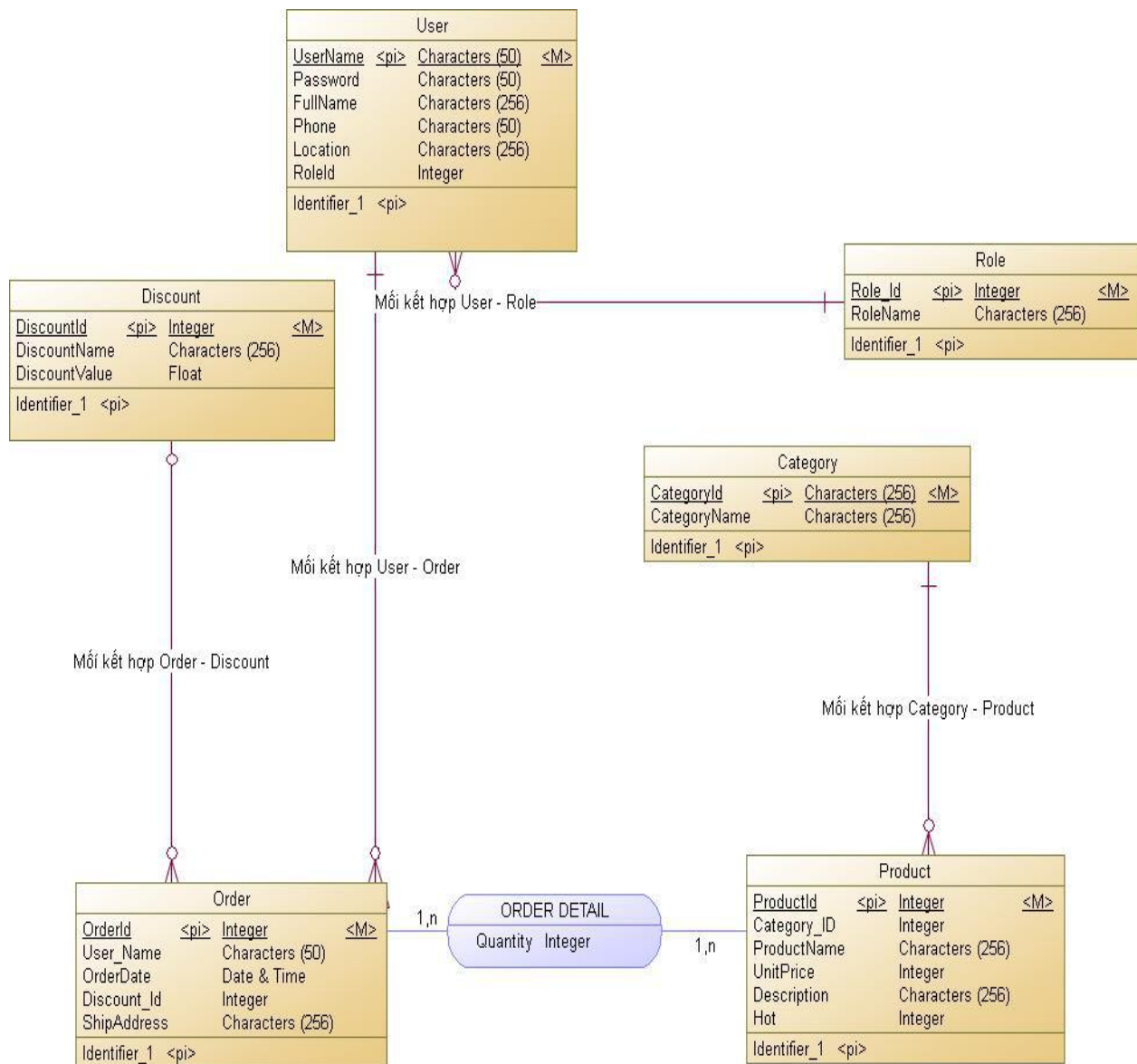


Hình 3.11 Mối kết hợp Order - Discount

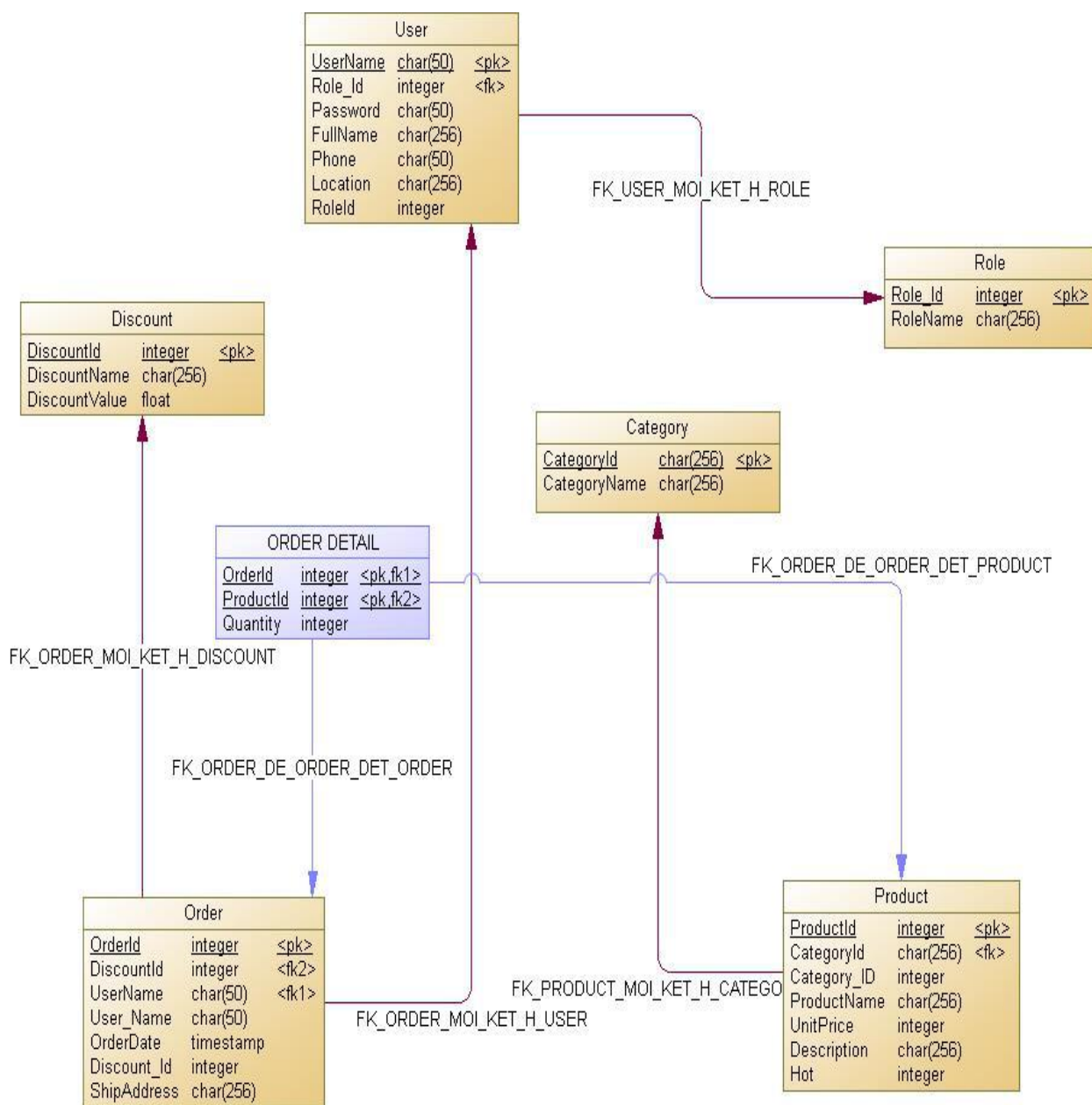
Giải thích: Một người đơn hàng có thể có hoặc không mã giảm giá. Một mã giảm giá có thể áp dụng cho nhiều đơn hàng.

### 3.1.4. Các mô hình cơ sở dữ liệu

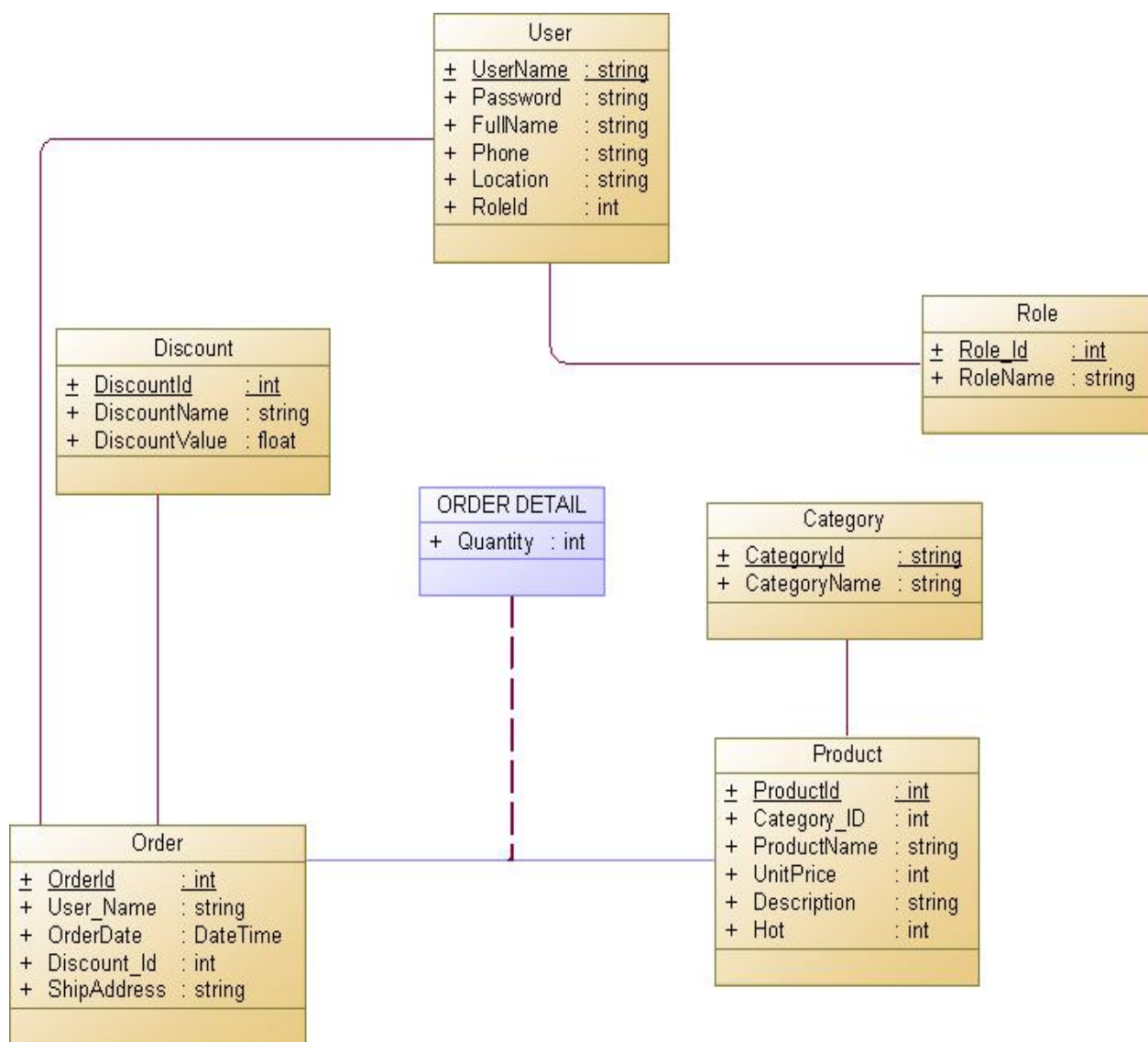
#### a. Mô hình CDM



Hình 3.12 Mô hình CDM website quản lý và đặt cà phê

**b. Mô hình PDM**

Hình 3.13 Mô hình PDM của website quản lý và đặt cà phê

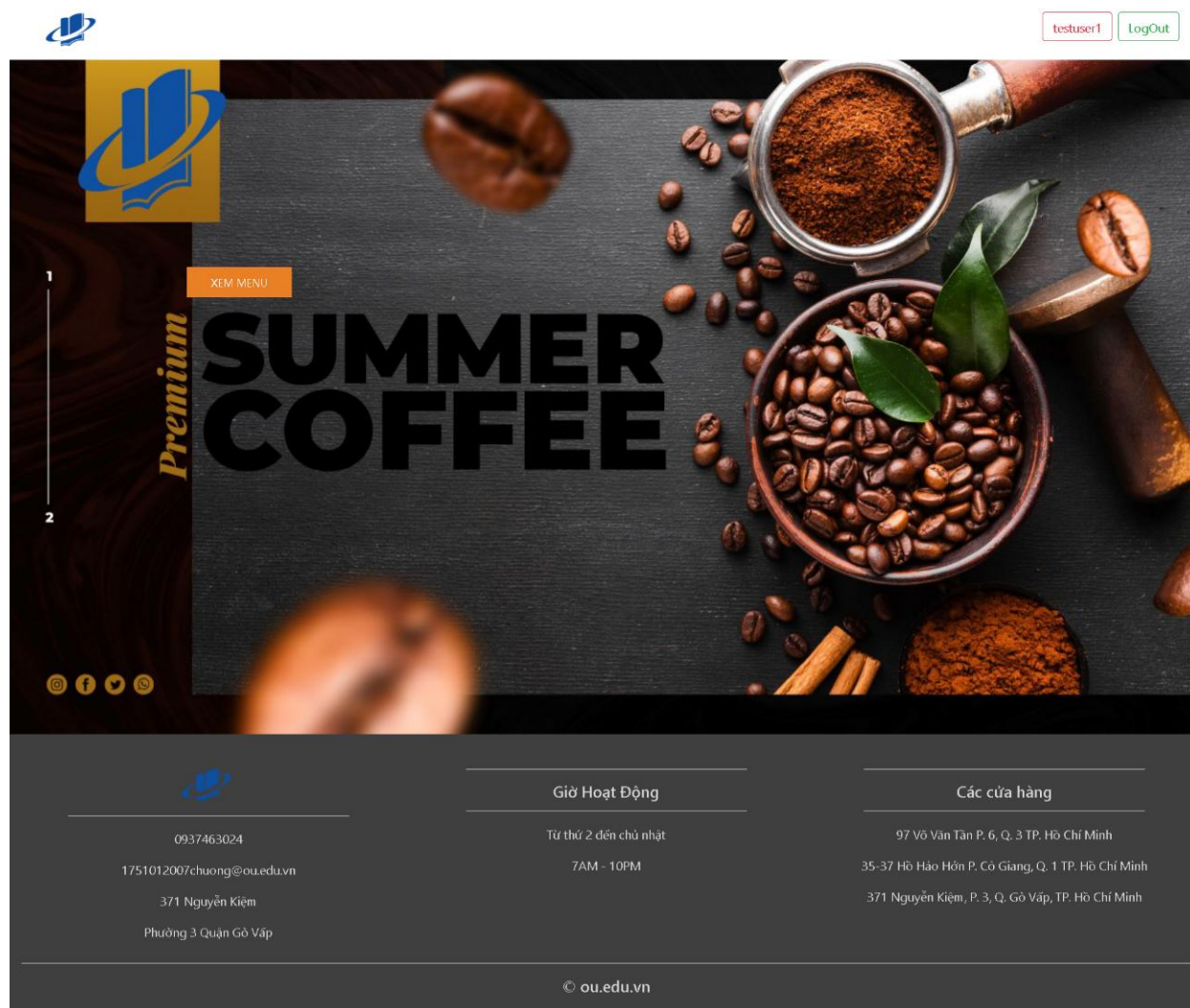
**c. Mô hình OOM**

Hình 3.14 Mô hình OOM của website quản lý và đặt cà phê

## Chương 4. Giao diện phần mềm

### 4.1. Giao diện người dùng chung

#### 4.1.1. Giao diện trang chủ



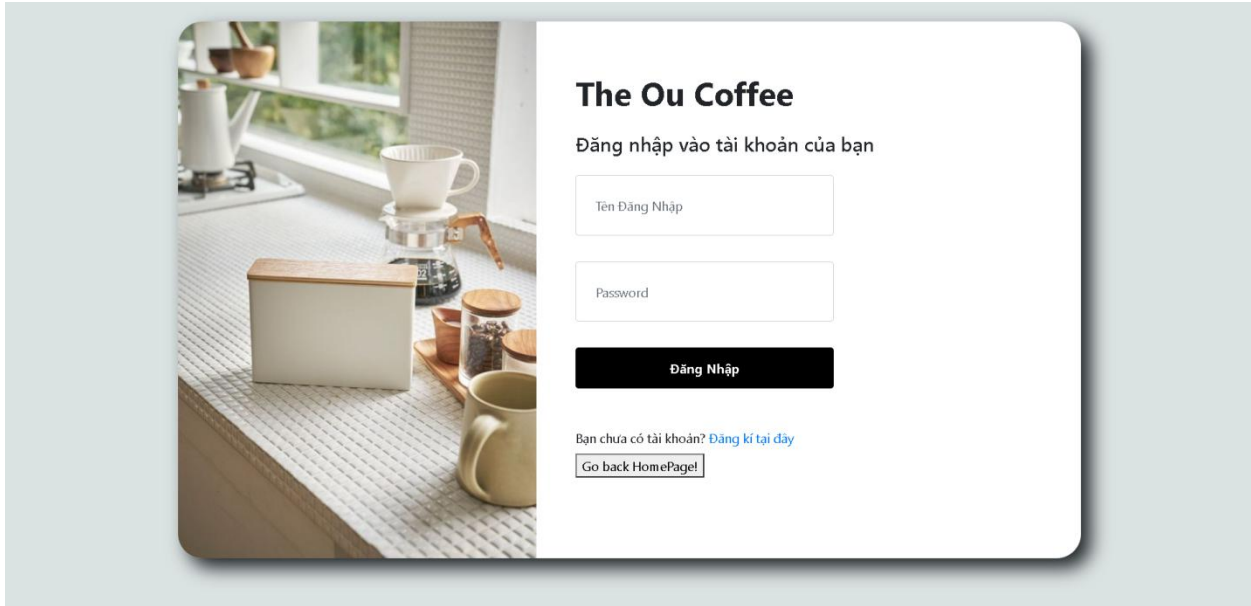
Hình 4.1 Giao diện trang chủ

Giao diện trang chủ gồm 3 phần:

- Phần Header chứa logo và các nút bấm như đăng nhập và đăng ký tài khoản
- Phần Content chứa hình ảnh đại diện cũng như nút bấm xem menu sản phẩm

- Phần Footer chứa các thông tin liên quan về cửa hàng như số điện thoại email, giờ hoạt động, các chi nhánh của cửa hàng.

#### 4.1.2. Giao diện đăng nhập

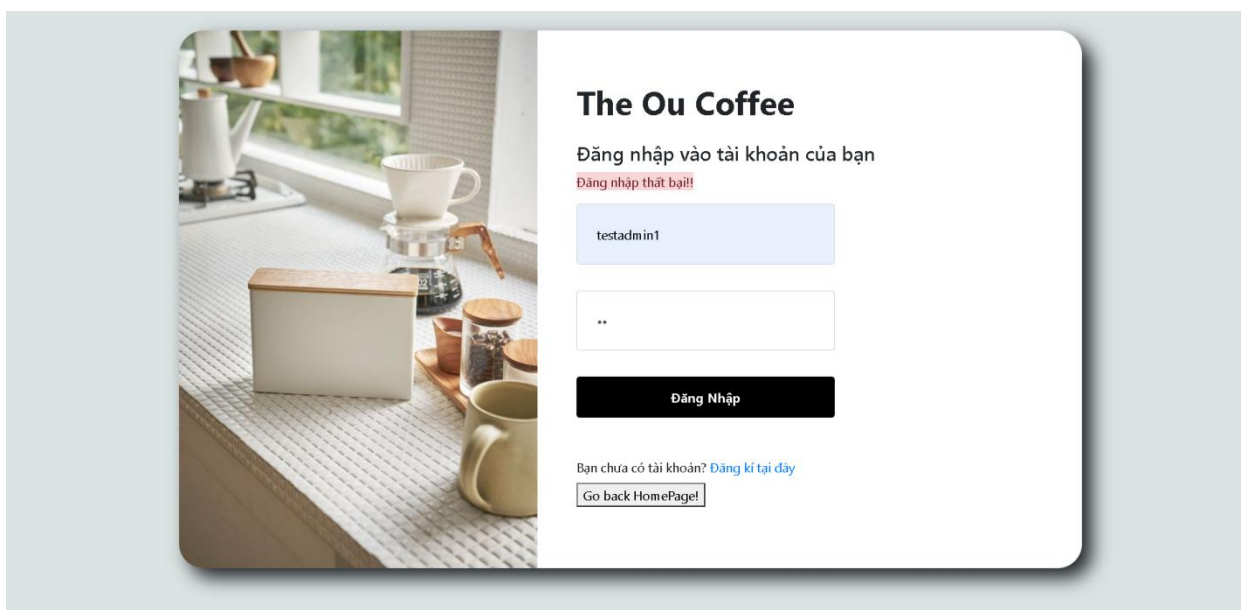


Hình 4.2 Giao diện đăng nhập

Nếu người dùng muốn đặt sản phẩm hoặc chủ cửa hàng muốn tiến vào phần quản lý cửa hàng thì cần phải đăng nhập.

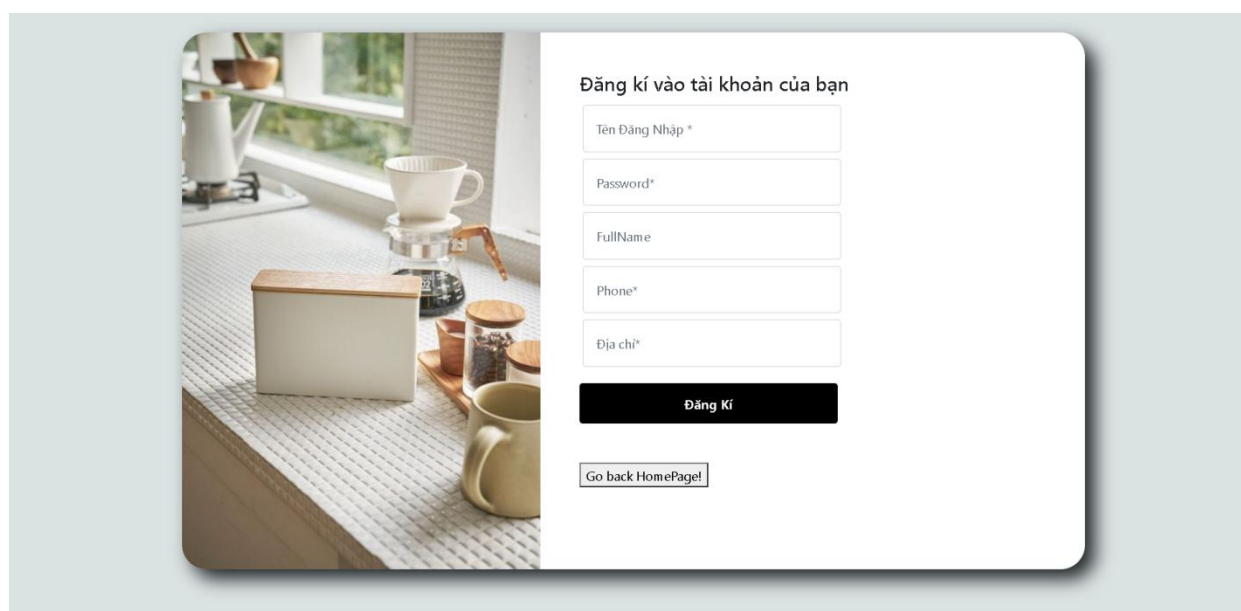
Quy trình đăng nhập được xử lý như sau, khi người dùng click vào nút đăng nhập hệ thống sẽ kiểm tra người dùng đã nhập đủ các trường cần thiết hay chưa và nhắc nhở nếu chưa điền đủ. Sau khi điền đủ tên đăng nhập và mật khẩu hệ thống sẽ gọi API Login sử dụng công nghệ JsonWebToken(JWT) để kiểm tra nếu nhập đúng thông tin tài khoản thì hệ thống phía backend sẽ trả về 1 token lên hệ thống phía client, sau khi đã đăng nhập người dùng sẽ được sử dụng các chức năng mà mã token đó cho phép. Nếu sai 1 trong các trường tên đăng nhập và mật khẩu giao diện sẽ hiển thị “Đăng nhập thất bại” nếu đúng giao diện sẽ được điều hướng đến trang chủ.



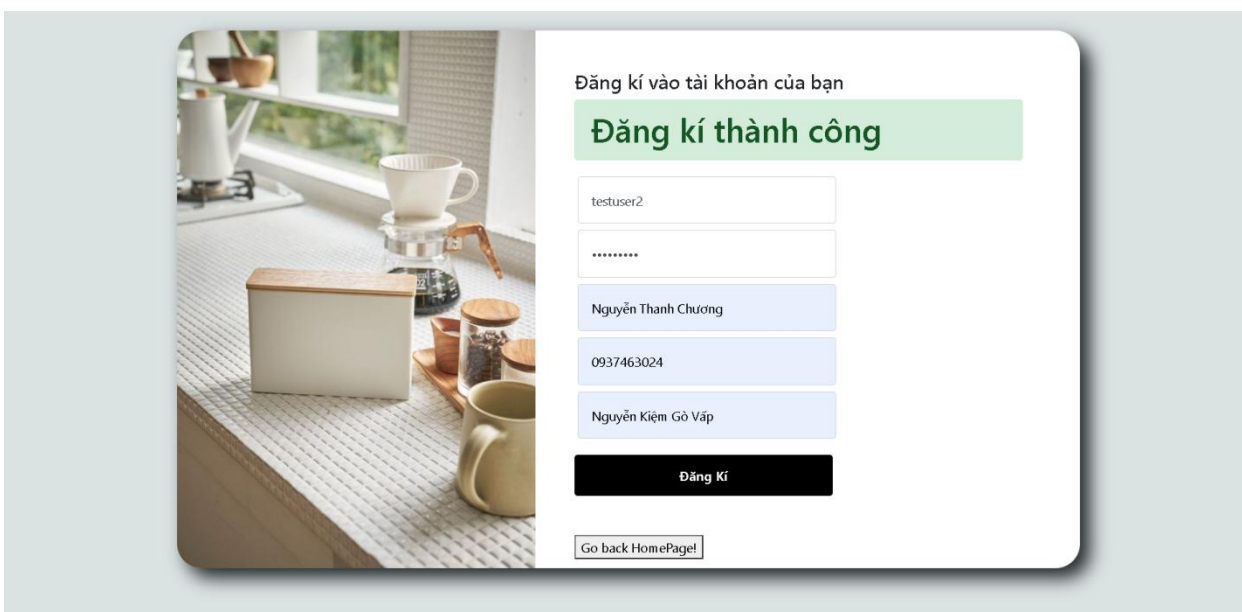


Hình 4.3 Giao diện đăng nhập thất bại

#### 4.1.3. Giao diện đăng ký

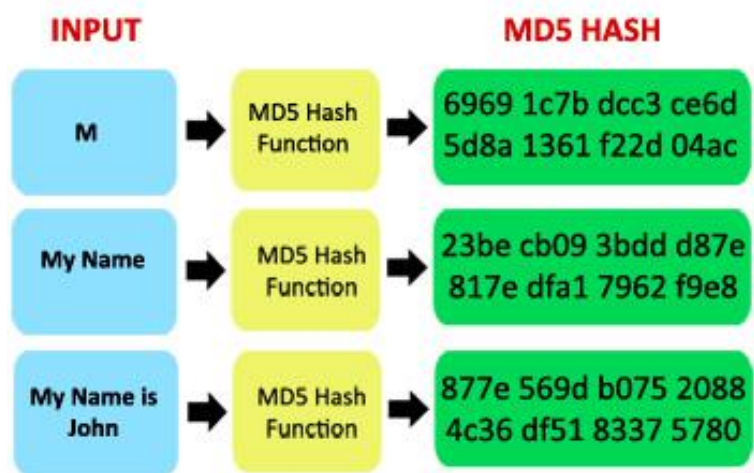


Tương tự như chức năng đăng nhập. Ở giao diện đăng ký người dùng cần nhập đầy đủ các trường bắt buộc như tên đăng nhập và mật khẩu..., và cần tuân thủ theo các quy tắc mà hệ thống đưa ra như tên đăng nhập phải có độ dài lớn hơn 6, số điện thoại phải tuân thủ quy tắc số điện thoại ở Việt Nam.



Hình 4.4 Giao diện đăng kí thành công

Sau khi điền đầy đủ các thông tin cần thiết người dùng click vào nút đăng kí. Hệ thống sẽ kiểm tra xem tài khoản đã được tạo hay chưa nếu trùng tên đăng nhập hệ thống sẽ hiện thị “UserName đã tồn tại!!!”. Nếu đăng kí thành công các thông tin người dùng sẽ được lưu trữ dưới cơ sở dữ liệu và mật khẩu sẽ được hệ thống mã hoá bằng hàm băm MD5 một hàm băm mã theo chuẩn RFC 1321, có giá trị hash dài 128 bit. Điều này sẽ đảm bảo sự an toàn bảo mật cho người dùng.

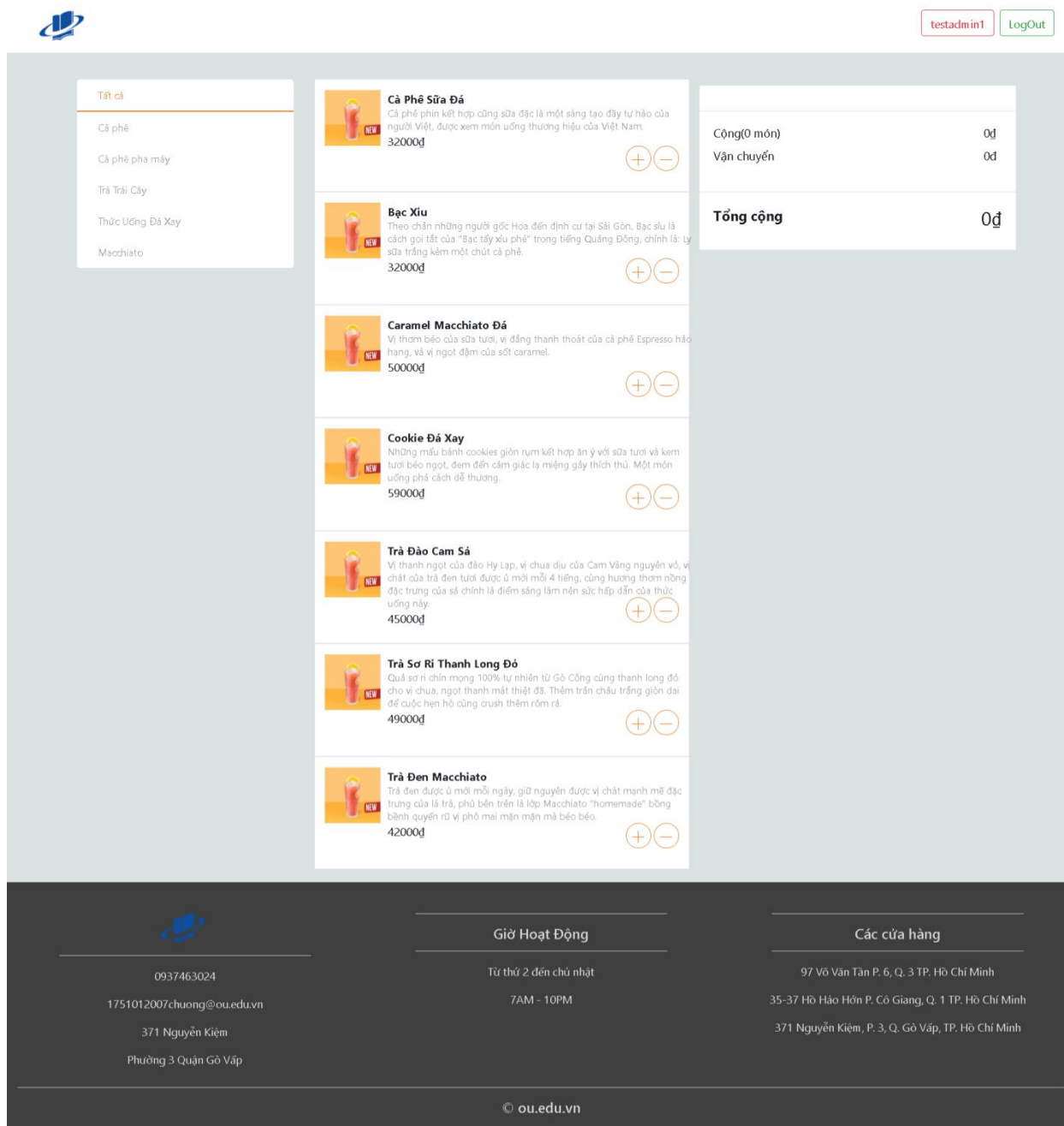


Hình 4.5 Ví dụ về hàm băm MD5



#### 4.1.4. Giao diện Menu đặt món

Sau khi người dùng click chọn nút xem menu ở trang chủ người dùng sẽ được điều hướng đến trang Menu đặt món để lựa chọn các thức uống mà cửa hàng phục vụ.



Hình 4.6 Giao diện menu đặt món

Giao diện menu đặt món gồm 3 phần chính:

- Phần danh mục sản phẩm: giao diện hiển thị các danh mục đồ uống mà cửa hàng phục vụ. Khi người dùng click chọn danh mục mình muốn hệ thống sẽ hiển thị các món thuộc danh mục đó.



Hình 4.7 Giao diện hiển thị sản phẩm theo danh mục

- Phần chi tiết sản phẩm: giao diện hiển thị chi tiết các sản phẩm gồm tên sản phẩm(ProductName), mô tả sản phẩm(Description), đơn giá sản phẩm(UnitPrice) và các nút bấm thêm hoặc xóa các sản phẩm trong đơn hàng của mình.



Hình 4.8 Giao diện khi người dùng thêm sản phẩm vào đơn hàng

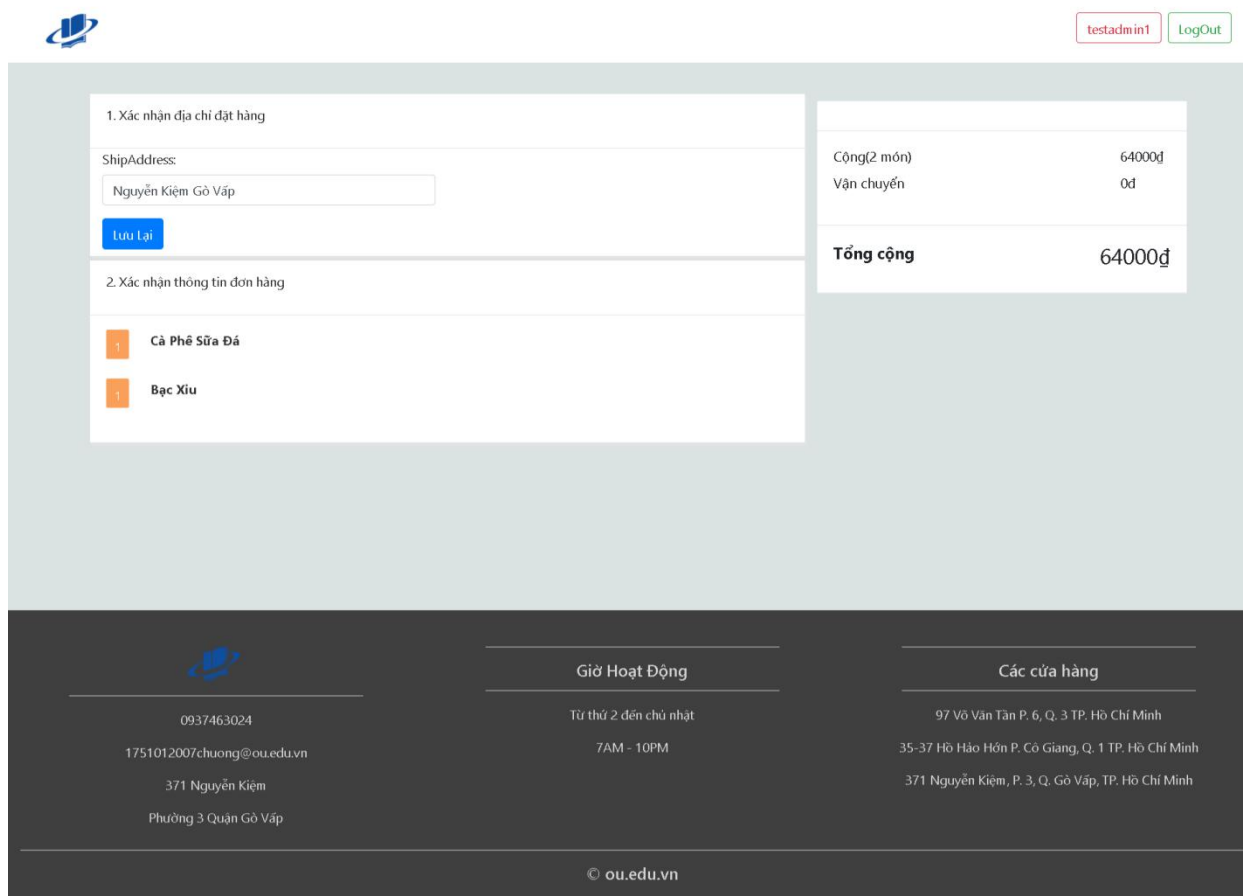
- Phần giỏ hàng: Khi người dùng thêm sản phẩm vào giỏ hàng thì hệ thống sẽ hiển thị tổng giá tiền mà khách hàng phải trả.

Xem giỏ hàng	
Cộng(3 món)	114000đ
Vận chuyển	0đ
<b>Tổng cộng</b>	<b>114000đ</b>

Hình 4.9 Giao diện giỏ hàng

Khi người dùng đã chọn xong các món sản phẩm mà mình muốn đặt sẽ tiếp tục click vào nút xem giỏ hàng để đi đến bước tiếp theo là xác nhận thông tin và đặt hàng. Ở đây người dùng bắt buộc phải đăng nhập để có thể đặt hàng vì hệ thống sẽ sử dụng các thông tin của khách hàng để giao hàng và lập đơn hàng.

### 4.1.5. Giao diện xác nhận đặt hàng



testadmin1 Logout

1. Xác nhận địa chỉ đặt hàng

ShipAddress:

Nguyễn Kiệm Gò Vấp


Lưu Lại

2. Xác nhận thông tin đơn hàng

Cà Phê Sữa Đá

Bạc Xiu

Cộng(2 món)	64000đ
Vận chuyển	0đ
<b>Tổng cộng</b>	<b>64000đ</b>



0937463024

1751012007chuong@ou.edu.vn

371 Nguyễn Kiệm

Phường 3 Quận Gò Vấp

**Giờ Hoạt Động**

Từ thứ 2 đến chủ nhật

7AM - 10PM

**Các cửa hàng**

97 Võ Văn Tần P. 6, Q. 3 TP. Hồ Chí Minh

35-37 Hồ Hảo Hớn P. Cô Giang, Q. 1 TP. Hồ Chí Minh

371 Nguyễn Kiệm, P. 3, Q. Gò Vấp, TP. Hồ Chí Minh

© ou.edu.vn

Hình 4.10 Giao diện xác nhận đặt hàng

Khi tiến vào phần xác nhận thông tin người dùng sẽ được thay đổi thông tin địa chỉ giao hàng và xác nhận lại một lần nữa các món mình muốn đặt. Địa chỉ mặc định sẽ là địa chỉ của người dùng khi tạo tài khoản.

Khi xác nhận và chọn nút lưu đơn hàng hệ thống sẽ một lần nữa xác nhận xem bạn có chắc chắn muốn đặt hàng không.



localhost:3000 says

Bạn có chắc chắn muốn đặt sản phẩm chứ?

OK Cancel

Hình 4.11 Hệ thống xác nhận đặt hàng

Khi người dùng tiền hành đặt hàng đơn hàng sẽ được lưu lại và lưu lên hệ thống để cửa hàng có thể chuẩn bị xác nhận giao hàng cho khách hàng.

Sau đó hệ thống sẽ được điều hướng đến trang profile(thông tin khách hàng)

#### 4.1.6. Giao diện trang thông tin khách hàng

The screenshot displays a web application interface for a customer's profile. At the top, there's a navigation bar with a logo and user controls. The main area features a sidebar with tabs for account and order information. The account tab is active, showing a form with pre-filled details. The footer provides comprehensive contact and location data.

Thông tin tài khoản	Thông tin đơn hàng
Username: testadmin1 Password: FullName: Nguyễn Thanh Chương Phone: 0937463024 Location: Nguyễn Kiệm Gò Vấp Lưu Lại	

Giờ Hoạt Động	Các cửa hàng
Từ thứ 2 đến chủ nhật 7AM - 10PM	97 Võ Văn Tần P. 6, Q. 3 TP. Hồ Chí Minh 35-37 Hồ Hảo Hớn P. Cô Giang, Q. 1 TP. Hồ Chí Minh 371 Nguyễn Kiệm, P. 3, Q. Gò Vấp, TP. Hồ Chí Minh

© ou.edu.vn

Hình 4.12 Giao diện thông tin khách hàng

Tại đây khách hàng có thể xem được các thông tin tài khoản của mình và cũng như thay đổi thông tin để sử dụng trong việc đặt hàng. Ngoài ra khách hàng còn có thể xem được các đơn hàng mình đã đặt cũng như chi tiết đơn hàng đó.

Danh Sách Order			
STT	Order Id	Order Date	Ship Address
1	2	08/11/20 : 18:48:22	Nguyễn Kiệm Gò Vấp
<< < 1 > >>			

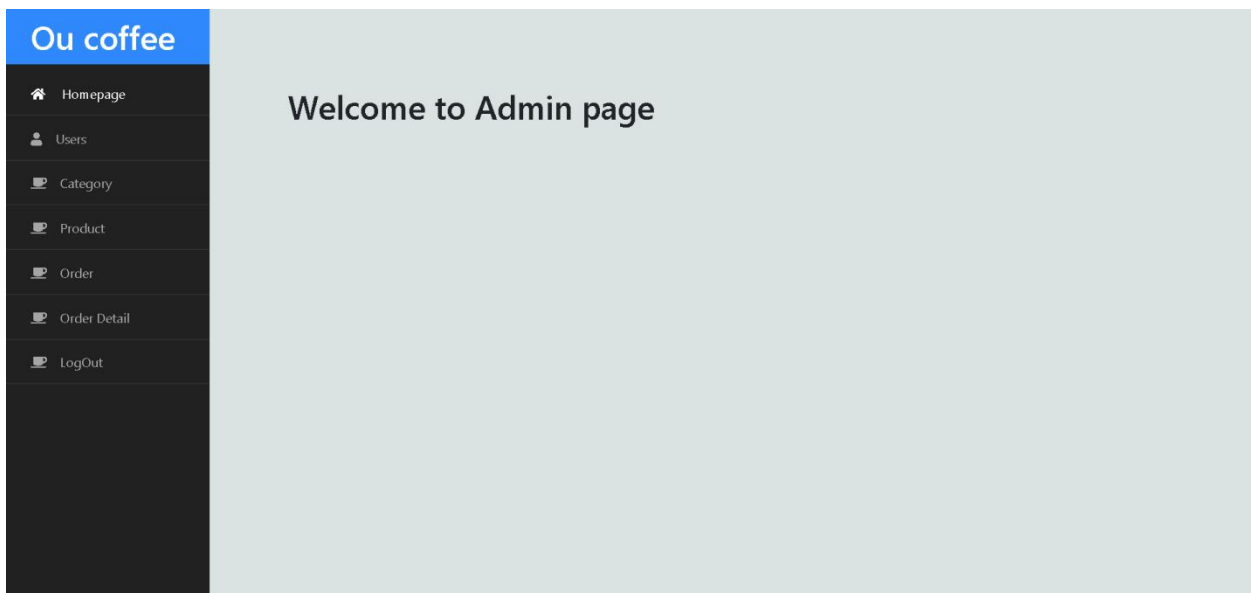
Hình 4.13 Giao diện xem các đơn hàng đã đặt

Danh Sách OrderDetail			
STT	Order Id	Product Id	Quantity
1	2	Cà Phê Sữa Đá	1
2	2	Bạc Xiu	1
<< < 1 > >>			

Hình 4.14 Giao diện chi tiết đơn hàng đã đặt

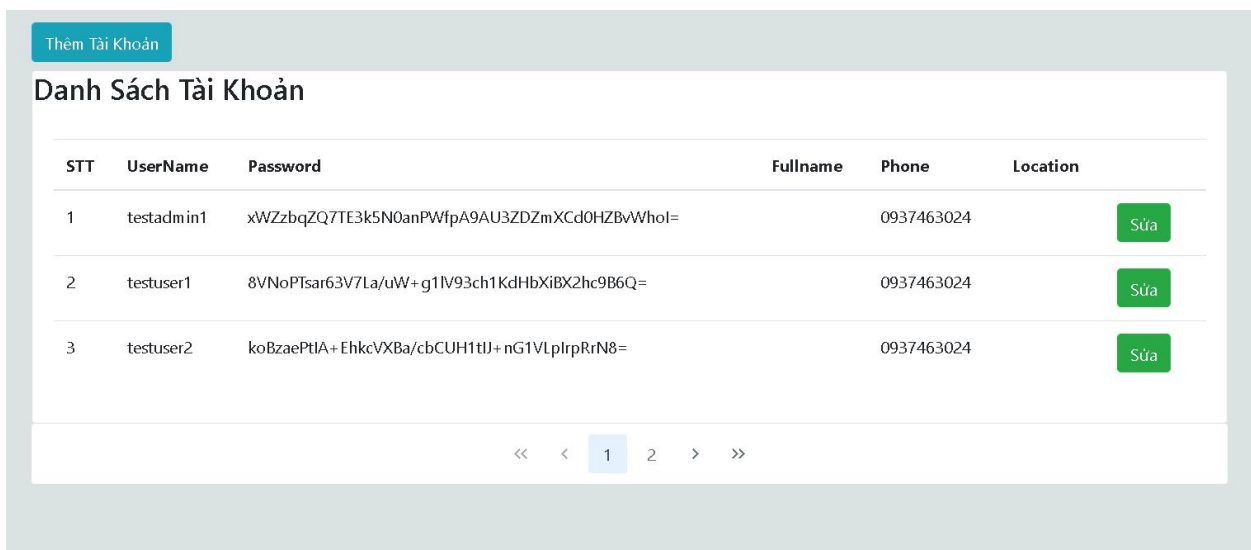
## 4.2. Giao diện phân hạng admin

Khi muốn sử dụng các chức năng của phân hạng admin thì người dùng bắt buộc phải đăng nhập với tài khoản có vai trò admin ( RoleId = 1). Người dùng sẽ đăng nhập bằng giao diện đăng nhập như thông thường và sau đó truy cập đến đường dẫn “/admin” . Nếu người dùng không phải vai trò admin sẽ được điều hướng trở lại trang chủ bảo đảm tính an toàn cho chủ cửa hàng.

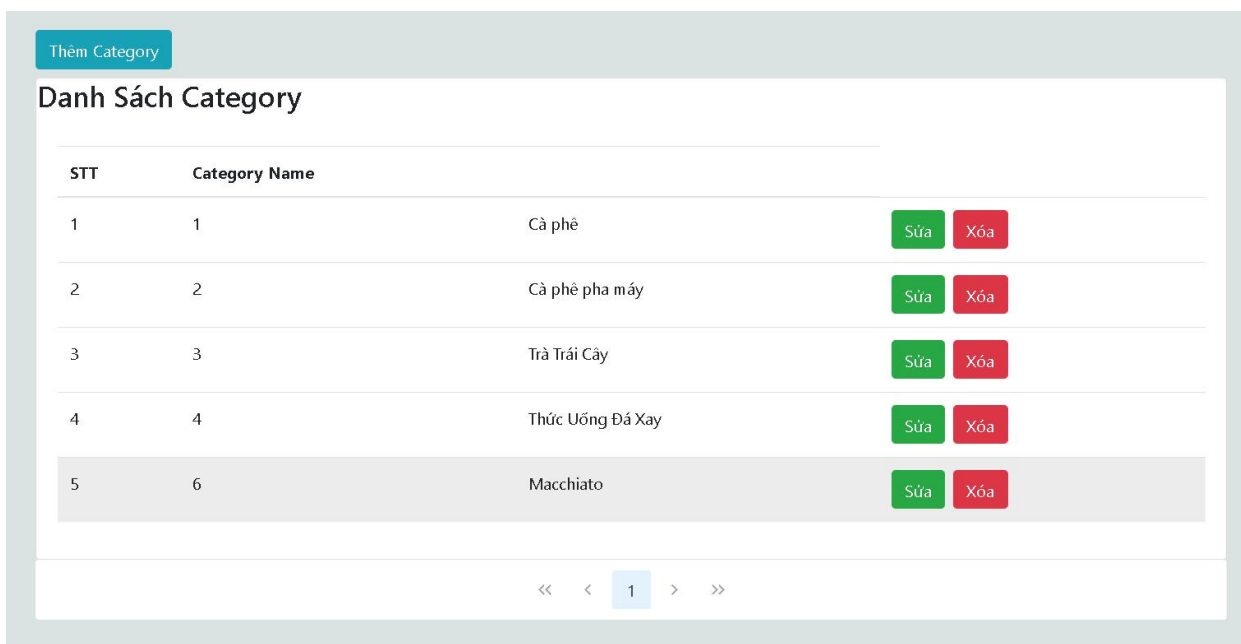


Hình 4.15 Giao diện trang admin

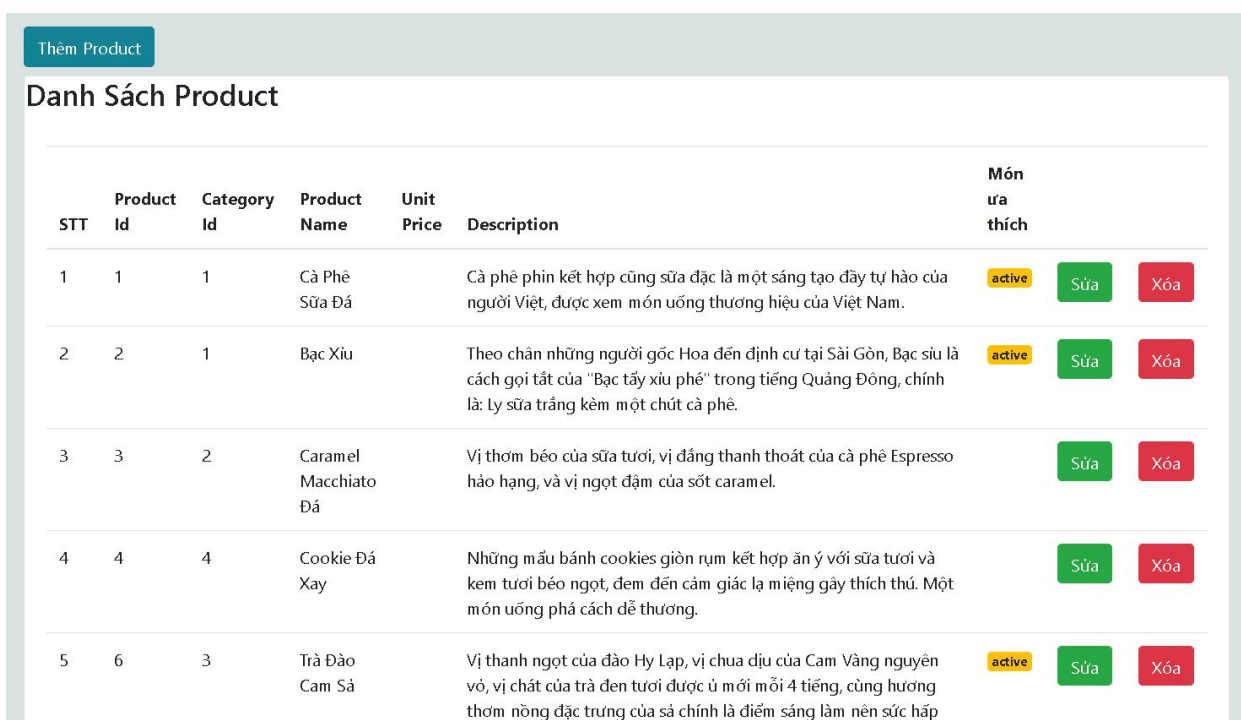
Tại đây người dùng sẽ có thể quản lý được toàn bộ dữ liệu thông tin khách hàng, đơn hàng, chi tiết đơn hàng, loại sản phẩm, sản phẩm... Người dùng có thể thêm xóa sửa các đơn hàng, thêm các sản phẩm mới tùy vào mục đích kinh doanh của mình.



Hình 4.16 Giao diện quản lý tài khoản



Hình 4.17 Giao diện quản lý danh mục sản phẩm



Hình 4.18 Giao diện quản lý sản phẩm



Thêm Order

### Danh Sách Order

STT	Order Id	User Name	Order Date	Discount Id	Ship Address		
1	1	testuser1	07/11/20 : 15:19:57		Gò Vấp	Sửa	Xóa
2	2	testadmin1	08/11/20 : 18:48:22		Nguyễn Kiệm Gò Vấp	Sửa	Xóa

<< < 1 > >>

Hình 4.19 Giao diện quản lý đơn hàng

Thêm OrderDetail

### Danh Sách OrderDetail

STT	Order Id	Product Id	Quantity		
1	1	2	2	Sửa	Xóa
2	1	3	2	Sửa	Xóa
3	2	1	1	Sửa	Xóa
4	2	2	1	Sửa	Xóa

<< < 1 > >>

Hình 4.20 Giao diện quản lý chi tiết đơn hàng

## **Chương 5. Kết luận và hướng phát triển**

### **5.1. Kết quả đạt được**

#### **5.1.1. Về kiến thức và học tập**

Học hỏi thêm về các kiến trúc và lý thuyết cơ bản về cách lập trình và triển khai một trang web hiện nay.

Thiết được một cơ sở dữ liệu đáp ứng được nhu cầu của hệ thống

Tìm hiểu được một số kiến trúc hiện đại như thư viện ReactJs, .NET Core API và cũng có một số kiến thức cơ bản như html và css.

#### **5.1.2. Về website**

Hoàn thiện được đầy đủ các chức năng cơ bản của một website bán hàng, phân quyền được người dùng cơ bản và admin.

Xây dựng được các chức năng cơ bản thêm, sửa ,xóa, cập nhật thông tin nhanh chóng.

Giao diện trực quan dễ sử dụng, dễ dàng phát triển thêm các chức năng trong tương lai.

### **5.2. Hạn chế của đề tài**

Do hạn chế về thời gian và số lượng thành viên nghiên cứu và thực hiện có phần gấp rút nên việc thực hiện và triển khai còn chưa được hoàn thiện đầy đủ.

Kiến thức chuyên môn và kỹ năng thực tế còn hạn chế nên tính chuyên nghiệp code chưa được tối ưu và sạch sẽ, bắt lỗi còn chưa đủ ở một số chỗ.

Website còn một số tính năng chưa thực hiện được như mong muốn như xuất hóa đơn, báo cáo, thống kê. Gây ra một số bất tiện trong việc sử dụng.

### **5.3. Hướng phát triển**

Thiết kế và lập trình lại các tính năng và giao diện trở nên chuyên nghiệp tối ưu hơn cho người dùng.

Tìm hiểu các biện pháp bảo mật thông tin người dùng cho website từ đó đưa vào triển khai thanh toán online qua các cổng thanh toán trực tuyến đang là xu hướng hiện nay

Bổ sung thêm các tính năng: Chăm công, quản lý nhân viên, thanh toán online, sử dụng một số api như google map để hiển thị nơi khách hàng muốn giao đến.

### **DANH SÁCH TÀI LIỆU THAM KHẢO**

- [1] “React Tutorial”, 2020. [Trực tuyến]. Địa chỉ: <https://reactjs.org/tutorial/tutorial.html>. [Truy cập 1/11/2020]
- [2] “React Training/React Router”, 2020. [Trực tuyến]. Địa chỉ: <https://reactrouter.com/web/guides/quick-start>. [Truy cập 1/11/2020]
- [3] “React JWT Authentication (without Redux) example”, 2020. [Trực tuyến]. Địa chỉ: <https://bezkoder.com/react-jwt-auth>. [Truy cập 4/11/2020]
- [4] Lê Ngọc Hiếu, “Lập trình cơ sở dữ liệu”, 2020. [Trực tuyến]. Địa chỉ: <https://github.com/occbuu/LTCSDL-192>. [Truy cập 15/10/2020]
- [5] Lê Ngọc Hiếu, “Lập trình cơ sở dữ liệu”, 2020. [Trực tuyến]. Địa chỉ: <https://www.youtube.com/channel/UCQAHiw9iJIEvu2ToLGdDZrw>. [Truy cập 17/10/2020]
- [6] “SQL Sever”, 2020. [Trực tuyến]. Địa chỉ: <https://quantrimang.com/ms-sql-server-la-gi-145283>. [Truy cập 19/10/2020]
- [7] nhiepuoit “Khóa học lập trình ReacJs”, 2020. [Trực tuyến]. Địa chỉ: <https://www.youtube.com/channel/UCxRgDi15EJsB2ayyP-eg3Tg>. [Truy cập 20/10/2020]