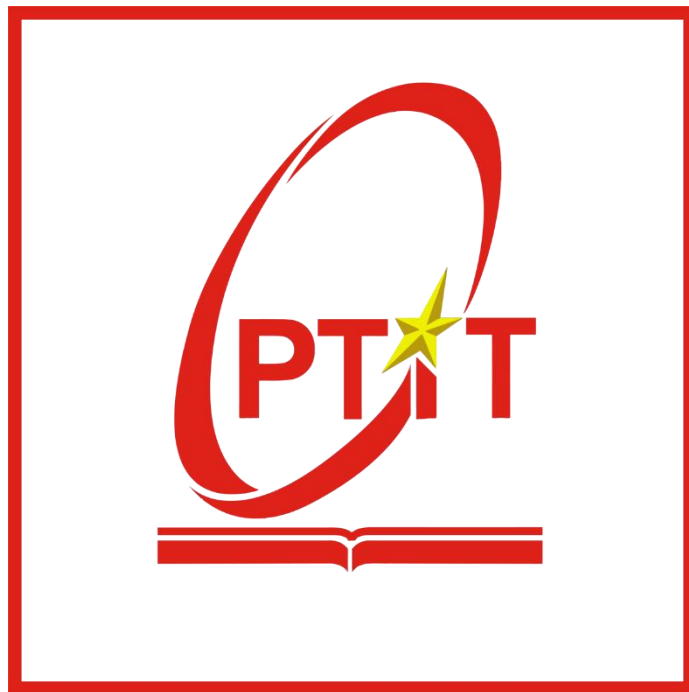


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH



BÁO CÁO GIỮA KỲ

PHÁT TRIỂN HỆ THỐNG THÔNG MINH

Giảng viên hướng dẫn : ThS. NGUYỄN NGỌC DUY

Sinh viên thực hiện	Mã sinh viên	Lớp
----------------------------	---------------------	------------

Nguyễn Phan Nhựt Trường	N20DCCN082	D20CQCN01-N
--------------------------------	-------------------	--------------------

Bùi Tuấn Anh	N20DCCN002	D20CQCN01-N
---------------------	-------------------	--------------------

Tp. Hồ Chí Minh - 09/2023

LỜI MỞ ĐẦU

Những tiến bộ to lớn về công nghệ, đặc biệt là công nghệ thông tin trong những thập niên cuối của thế kỷ 20 đã tạo ra bước ngoặt mới cho sự phát triển kinh tế xã hội toàn cầu. Chính trên nền tảng đó, một phương thức thương mại mới đã xuất hiện và phát triển nhanh chóng, đó là thương mại điện tử. Thương mại điện tử chính là một công cụ hiện đại sử dụng mạng Internet giúp cho các doanh nghiệp có thể thâm nhập vào thị trường thế giới, thu thập thông tin nhanh hơn, nhiều hơn và chính xác hơn. Với thương mại điện tử, các doanh nghiệp cũng có thể đưa các thông tin về sản phẩm của mình đến các đối tượng khách hàng tiềm năng khác nhau ở mọi nơi trên thế giới với chi phí thấp hơn nhiều so với các phương pháp truyền thống.

Trong xu thế phát triển mạnh mẽ của thương mại điện tử trên thế giới, các doanh nghiệp Việt Nam đặc biệt là các doanh nghiệp vừa và nhỏ, bộ phận chiếm tỷ lệ lớn trong tổng số các doanh nghiệp Việt Nam, cũng đã bước đầu nhận thức được ích lợi và tầm quan trọng của việc ứng dụng thương mại điện tử. Tuy nhiên, do còn nhiều hạn chế trong nhận thức của bản thân các doanh nghiệp cũng như các điều kiện cơ sở hạ tầng, việc ứng dụng thương mại điện tử trong các doanh nghiệp vừa và nhỏ ở Việt Nam có thể nói mới ở mức độ sơ khởi. Vì thế, các doanh nghiệp vừa và nhỏ cần xây dựng cho mình một chiến lược kinh doanh phù hợp với điều kiện thực tế hiện nay nhằm tiếp cận nhiều hơn nữa với thương mại điện tử để có thể khai thác tối đa lợi ích mà phương thức kinh doanh này đem lại.

Ngày nay, các cửa hàng trực tuyến xuất hiện càng nhiều. Cửa hàng trực tuyến sẽ đem lại nhiều lợi ích về cho người sử dụng. Đối với doanh nghiệp, tiết kiệm được không gian trưng bày sản phẩm, tiết kiệm nguồn nhân lực, tiết kiệm chi phí mặt bằng...Đối với khách hàng, tiết kiệm được thời gian đi lại, lựa chọn được những sản phẩm ưa thích...Đây cũng là lý do để nhóm em chọn đề tài: **“Xây dựng hệ thống website Bán giày nam kết hợp ứng dụng thông minh để xuất sản phẩm”** làm đề tài môn Phát triển hệ thống thông minh. Để hoàn thành đề tài, em muốn bày tỏ lòng biết ơn sâu sắc đến Thầy Nguyễn Ngọc Duy - người đã trực tiếp hướng dẫn và hỗ trợ em trong suốt quá trình làm. Thầy đã cung cấp cho chúng em những ý kiến quý báu và chỉ dẫn chính xác để giúp nhóm hoàn thiện tốt đề tài.

Mục Lục

CHƯƠNG 1: TỔNG QUAN.....	1
1 Mục tiêu đề tài	1
2. Yêu cầu của đề tài	1
2.1 Yêu cầu chung	1
2.2 Xây dựng website thương mại điện tử	1
3. Các chức năng chính	1
4. Đối tượng và phạm vi nguyên cứu	2
5. Giải pháp thực hiện	2
6. Lý thuyết	3
6.1. Ngôn ngữ C#	3
6.2. Entity Framework với mô hình MVC:	4
6.3. HTML, CSS, JavaScript:	8
6.4. Công cụ và môi trường: SQL Server và Visual Studio .NET:	12
CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ	16
1. Phân tích	16
1.1 Mô tả nghiệp vụ	16
1.2 Yêu cầu hệ thống	16
1.3 Bảo mật và quyền hạn	18
2. Mô hình UseCase	18
2.1 Danh sách các Use Case	20
2.2 Đặc tả Use Case	21
3 Sơ đồ tuần tự	25
3.1 Đăng nhập	26
3.2 Quy trình thêm sản phẩm vào giỏ hàng	26
3.3 Quy trình đặt mua sản phẩm	27
4. Sơ đồ lớp	27
4.1 Sơ đồ lớp mức căn bản	28
5. Sơ Đồ Diagram	29
6. Sơ Đồ ERD	30
7. Sơ Đồ cơ sở dữ liệu	31

8. Giao diện :	35
CHƯƠNG 3: KẾT LUẬN	38
1 Kết quả đạt được.	38
1.1 Ưu điểm.	38
1.2 Nhược điểm.	38
2. Hướng phát triển của hệ thống (bao gồm phát triển ứng dụng thông minh)	39
3. Phân chia công việc:	40
CHƯƠNG 4: HỆ THỐNG THÔNG MINH	40
Giới thiệu về hệ thống gợi ý (Recommender systems hoặc Recommendation systems)	40
So sánh các phương pháp gợi ý	43
Xây dựng hệ thống gợi ý Item - item	44
Step 1: Nhập các thư viện	45
Step 2: Tải dữ liệu lên	45
Step 3: Phân tích các dữ liệu khám phá	45
Step 4: Biến đổi dữ liệu	47
Step 5: Tìm các đôi giày tương tự bằng cách sử dụng k-Nearest Neighbours	48

CHƯƠNG 1: TỔNG QUAN

1 Mục tiêu đề tài

- Giúp sinh viên chúng em hiểu biết về các phương pháp cũng như cách thức xây dựng một website bán hàng.
- Cụ thể hóa được kiến thức đã học để áp dụng làm một dự án thực tế cụ thể là hệ thống website bán hàng.
- Xây dựng một website mang lại thuận tiện cho người dùng, tiết kiệm thời gian khi có nhu cầu mua một sản phẩm.
- Xây dựng Website giúp cho nhân viên quản lý thông tin một cách trực quan, thuận tiện.
- Ứng dụng kiến thức về hệ thống thông minh để tạo ra hệ thống đề xuất sản phẩm thông minh.

2. Yêu cầu của đề tài

2.1 Yêu cầu chung

- Tìm hiểu về hoạt động kinh doanh của website bán hàng.
- Tìm hiểu các giải pháp xây dựng một website bán hàng như thế nào.
- Xây dựng một website mang tính thương mại hóa.
- Xây dựng một trang web với đầy đủ các tính năng phục vụ cho việc bán hàng trực tuyến
- Xây dựng các chức năng giúp người quản trị dễ dàng quản lý nội dung trang web.
- Xây dựng chức năng thông minh và mang lại trải nghiệm thông minh cho người dùng

2.2 Xây dựng website thương mại điện tử.

- Website thương mại điện tử có một số chức năng như sau:
 - + Đối với khách hàng: Xem danh sách các sản phẩm, Xem thông tin chi tiết một sản phẩm, Tìm kiếm sản phẩm, Đưa sản phẩm vào giỏ hàng, Đặt mua, Thanh toán, Góp ý, Liên hệ, Xem các tin tức, Chức năng thông minh đề xuất sản phẩm...
 - + Đối với quản trị: Quản trị sản phẩm, quản trị nhân viên, Quản trị đơn hàng, Quản trị khách hàng, Quản trị tin tức, góp ý...

3. Các chức năng chính

- Website bán giày nam được xây dựng với tiêu chí phục vụ cho tất cả mọi người, trong lĩnh vực thời trang với mong muốn đáp ứng đầy đủ các mặt hàng, các chức năng của một website bán hàng trực tuyến.
- Website bao gồm các loại người dùng chính: Người dùng chung, Khách hàng thành viên, Người quản trị:

- + Đối với Người dùng chung: Xem danh sách sản phẩm, Xem chi tiết sản phẩm, Đưa sản phẩm vào giỏ hàng, cập nhật giỏ hàng, Tìm kiếm, Đăng Ký, Chức năng thông minh đề xuất sản phẩm...
- + Đối với Khách hàng thành viên: Ngoài những chức năng như Người dùng chung còn có thêm một số chức năng như Đặt mua, Xem danh sách đơn hàng, Quản lý đơn hàng, Thanh Toán, Đăng nhập...
- + Đối với nhân viên: Được xem như là một khách hàng thành viên và còn có chức năng quản lý tùy thuộc vào chức năng như Quản lý sản phẩm, đơn hàng, khách hàng, quản lý thông tin khác. Ngoài ra còn tư vấn khách hàng, giải đáp thắc mắc, phản hồi ý kiến cho khách hàng.

4. Đối tượng và phạm vi nguyên cứu

- Website bán giày nam được xây dựng với tiêu chí phục vụ cho tất cả mọi người trong lĩnh vực thời trang với mong muốn đáp ứng đầy đủ các mặt hàng, các chức năng của một website bán hàng trực tuyến.
- Với khoảng thời gian có hạn nên chúng em đã giới hạn phạm vi nguyên cứu ở một mức độ hợp lý cho mình.
 - + Xây dựng các chức năng cho khách hàng.
 - + Xây dựng các chức năng cho nhân viên quản trị.
 - + Tìm hiểu một vài công nghệ mới ứng dụng trên ASP.NET (Entity Framework với mô hình MVC) và các thuật toán học máy cơ bản để phát triển hệ thống thông minh.

5. Giải pháp thực hiện

- Môi trường phát triển ứng dụng:
 - + Lập trình bằng ngôn ngữ C#
 - + Sử dụng Entity Framework với mô hình MVC
 - + Làm giao diện với Html, Css và Javascript
 - + Lập trình cơ sở dữ liệu (SQL Server)
 - + Công cụ xây dựng cơ sở dữ liệu SQL Server
 - + Công cụ đã dùng để xây dựng ứng dụng: Visual Studio.NET
 - + Sử dụng kỹ thuật collaborative để tạo ra các gợi ý cho người dùng
- Môi trường triển khai ứng dụng:
 - + Hệ điều hành Windows
 - + Để chạy được Website yêu cầu máy phải cài ít nhất một trình duyệt web.
 - + Yêu cầu .NET FRAMEWORK 4.0 trở lên.

6. Lý thuyết

6.1. Ngôn ngữ C#

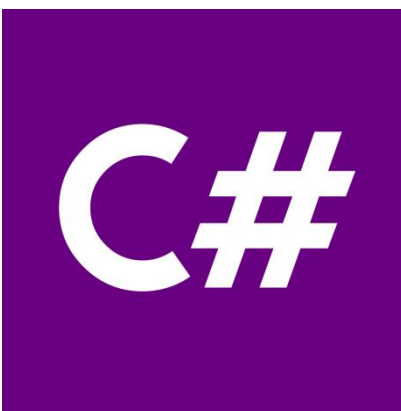
- **C# là gì?**

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.

C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), . . . trở nên rất dễ dàng.



- **Đặc trưng của C#:**

Các đặc điểm để làm cho C# là ngôn ngữ lập trình chuyên nghiệp được sử dụng rộng rãi:

- **C# là ngôn ngữ đơn giản**

Như ta đã biết thì ngôn ngữ C# dựng trên nền tảng C++ và Java nên ngôn ngữ C# khá đơn giản. Nếu chúng ta thân thiện với C và C++ hoặc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nhưng nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn. Một vài trong các sự cải tiến là loại bỏ các dư thừa, hay là thêm vào những cú pháp thay đổi.

- C# là ngôn ngữ hiện đại

Một vài khái niệm khá mới mẻ khá mơ hồ với các bạn vừa mới học lập trình, như xử lý ngoại lệ, những kiểu dữ liệu mở rộng, bảo mật mã nguồn..v.v... Đây là những đặc tính được cho là của một ngôn ngữ hiện đại cần có. Và C# chứa tất cả các đặc tính ta vừa nêu trên. Các bạn sẽ dần tìm hiểu được các đặc tính trên qua các bài học trong series này.

- C# là một ngôn ngữ lập trình thuần hướng đối tượng

Lập trình hướng đối tượng(tiếng Anh: Object-oriented programming, viết tắt: OOP) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (*abstraction*), tính đóng gói (*encapsulation*), tính đa hình (*polymorphism*) và tính kế thừa (*inheritance*). C# hỗ trợ cho chúng ta tất cả những đặc tính trên. Và để hiểu rõ hơn thì chúng ta sẽ có một chương trình bày về phần này.

- C# là một ngôn ngữ ít từ khóa

C# được mô tả là ngôn ngữ sử dụng giới hạn những từ khóa (gồm khoảng 80 từ khóa và mười mấy kiểu dữ liệu xây dựng sẵn). Nếu bạn nghĩ rằng ngôn ngữ có càng nhiều từ khóa thì sẽ càng mạnh mẽ hơn. Điều này không phải sự thật, lấy ví dụ ngôn ngữ C# làm điển hình nhé. Nếu bạn học sâu về C# bạn sẽ thấy rằng ngôn ngữ này có thể được sử dụng để làm bất cứ nhiệm vụ nào.

Ngoài những đặc điểm trên thì còn một số ưu điểm nổi bật của C#:

- C# có cấu trúc khá gần gũi với các ngôn ngữ lập trình truyền thống, nên cũng khá dễ dàng tiếp cận và học nhanh với C#.
- C# có thể biên dịch trên nhiều nền tảng máy tính khác nhau.
- C# được xây dựng trên nền tảng của C++ và Java nên nó được thừa hưởng những ưu điểm của ngôn ngữ đó.
- C# là một phần của .NET Framework nên được sự chống lưng khá lớn đến từ bộ phận này.
- C# có IDE Visual Studio cùng nhiều plug-in vô cùng mạnh mẽ.

6.2. Entity Framework với mô hình MVC:

● Entity framework là gì?

Entity framework được phát triển từ năm 2008 bởi Microsoft . Hiểu một cách đơn giản, entity framework là một thư viện ORM (Object Relational Mapping) hay một framework làm việc cùng cơ sở dữ liệu do microsoft khuyến nghị giúp phản xạ quan hệ đối tượng qua lại giữa các

object của một chương trình, bảng cơ sở dữ liệu, bản ghi,...bằng cách cung cấp cơ chế tự động hoá các hoạt động liên quan đến CSDL cho các doanh nghiệp mà việc lưu trữ, truy cập dữ liệu và xây dựng ứng dụng đơn giản và dễ dàng.



- **Đặc điểm cơ bản của Entity Framework**

Để có thể hiểu rõ hơn về Entity framework và ứng dụng được Entity framework linh hoạt trong công việc thì việc tìm hiểu thêm về các đặc điểm cơ bản của nó là không thể nào thiếu được.

Dưới đây là ba đặc điểm cơ bản và quan trọng mà một lập trình viên nên biết đến và hiểu rõ khi sử dụng Entity framework.

- Bằng một lớp con của DBContent mà tất cả các cơ sở dữ liệu đều được hiển thị. Bạn hoàn toàn có thể tự động thực hiện hay can thiệp vào những thao tác như thể hiện object mà bạn xây dựng trên mỗi hàng hay thể hiện thuộc tính của object ở mỗi cột.
- Với Entity framework, bạn có thể truy vấn dữ liệu sử dụng LINQ mà không cần đến SQL. Điều này giúp dễ dàng hơn khi thực hiện các truy vấn CRUD từ code C# với các class.
- Khả năng thực hiện các thao tác với cấu trúc dữ liệu nhanh chóng và đơn giản như tạo bảng, thay đổi cấu trúc bảng, tạo cơ sở dữ liệu,...mà không bị mất dữ liệu nhờ công cụ Migration.

- **Các thành phần của Entity Framework**

Việc tìm hiểu thêm về các thành phần của Entity framework cũng giúp bạn hiểu rõ hơn về Entity framework. Các thành phần cơ bản này bao gồm:

- Entity Data Model: là thành phần dùng để lưu trữ các thông tin ánh xạ, nối các lớp model với cơ sở dữ liệu, mang lại sự tách biệt giữa chương trình và CSDL với 3 thành phần chính là Conceptual Model, Mapping và Storage Model.

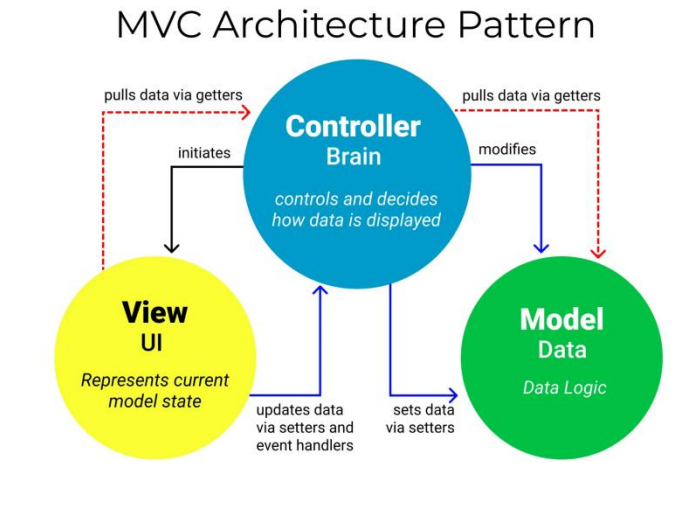
- LINQ to Entities và Entity SQL: Là hai ngôn ngữ truy vấn được sử dụng với mục đích giúp viết các truy vấn tới object model và trả về các thực thể được định nghĩa trong Conceptual Model.
- Object Service: Cụ thể hoá quá trình chuyển đổi và truy xuất dữ liệu. Ngoài ra, nó còn đảm nhận nhiệm vụ quản lý và theo dõi trạng thái thay đổi của Object.
- Entity Client Data Provider: Thành phần này chịu trách nhiệm tương tác với Data provider và chuyển đổi truy vấn LINQ to Entities và Entity SQL để truy vấn SQL.

● **Lợi ích của việc sử dụng Entity Framework là gì?**

Entity framework là công cụ phổ biến được sử dụng bởi hầu hết các doanh nghiệp. Lý do mà Entity framework được yêu thích như vậy chính là những lợi ích to lớn mà nó mang lại:

- Tăng năng suất: Khả năng giảm lượng code và các công việc dư thừa của Entity framework giúp tăng năng suất cho người sử dụng, tiết kiệm thời gian, chất xám và công sức của họ trong quá trình làm việc.
- Đơn giản hoá bảo trì phần mềm: Với lượng code ít khi sử dụng Entity framework, việc bảo trì và kiểm tra phần mềm được đơn giản hơn nhiều đặc biệt là khi bạn phải thường xuyên bảo trì.
- Không cần viết code SQL: Bạn có thể sử dụng LINQ to Entities khi thực hiện các truy vấn dữ liệu để thao tác với các Object được ra đời từ Entity framework.
- Tiết kiệm thời gian viết code: Bạn sẽ tiết kiệm được thời gian viết code khá lớn để thao tác với database bởi Entity framework sẽ tạo ra các classes cho việc truy xuất CSDL.
- Entity framework tự động tạo ra các commands, classes tương ứng cho việc insert, select, update, delete,...dữ liệu từ CSDL quan hệ.
- Làm việc đơn giản với EF: Giúp người dùng có thể làm việc với EF một cách đơn giản và thuận tiện hơn bởi Entity framework được đặt trong .NET Framework và tích hợp trong Visual studio.

- **Mô hình MVC:**



MVC (viết tắt của Model-View-Controller) là một mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính. MVC chia một ứng dụng thành ba phần tương tác được với nhau để tách biệt giữa cách thức mà thông tin được xử lý nội hàm và phần thông tin được trình bày và tiếp nhận từ phía người dùng.

Khi sử dụng đúng cách, mẫu MVC giúp cho người phát triển phần mềm cô lập các nguyên tắc nghiệp vụ và giao diện người dùng một cách rõ ràng hơn. Phần mềm phát triển theo mẫu MVC tạo nhiều thuận lợi cho việc bảo trì vì các nguyên tắc nghề nghiệp và giao diện ít liên quan với nhau.

Trong mẫu *Model-View-Controller*, mô hình (*model*) tượng trưng cho dữ liệu của chương trình phần mềm. Tầm nhìn hay khung nhìn (*view*) bao gồm các thành phần của giao diện người dùng. Bộ kiểm tra hay bộ điều chỉnh (*controller*) quản lý sự trao đổi giữa dữ liệu và các nguyên tắc nghề nghiệp trong các thao tác liên quan đến mô hình.

Model: Đây là phần của ứng dụng chịu trách nhiệm cho dữ liệu và logic dữ liệu. Model thường tương ứng với cơ sở dữ liệu và đối tượng dữ liệu trong ứng dụng. Trong ngữ cảnh EF, Model thường sử dụng để đại diện cho các thực thể cơ sở dữ liệu (*entities*) và chứa các lớp đại diện cho bảng trong cơ sở dữ liệu.

View: Đây là phần giao diện người dùng, nơi người dùng tương tác với ứng dụng. View thường là trang web, giao diện người dùng, hoặc các thành phần giao diện.

Controller: Controller là lớp xử lý logic của ứng dụng. Nó nhận yêu cầu từ người dùng thông qua View, tương tác với Model để truy cập hoặc cập nhật dữ liệu, sau đó cung cấp kết quả cho View hiển thị cho người dùng.

- **Kết hợp Entity Framework với Mô hình MVC:**

Khi kết hợp Entity Framework với Mô hình MVC, bạn có thể tận dụng sức mạnh của cả hai để phát triển ứng dụng web dễ dàng và hiệu quả. Dưới đây là cách thực hiện điều này:

Model: Model trong Mô hình MVC thường tương ứng với các đối tượng dữ liệu được định nghĩa bằng Entity Framework. Bạn có thể sử dụng EF để tạo các thực thể cơ sở dữ liệu (entities) và quản lý các kết nối với cơ sở dữ liệu.

Controller: Controller trong Mô hình MVC có trách nhiệm xử lý logic ứng dụng. Bạn có thể sử dụng Controller để gọi các phương thức của EF để thực hiện truy cập dữ liệu, thực hiện các truy vấn LINQ, và cập nhật dữ liệu trong cơ sở dữ liệu.

View: View trong Mô hình MVC hiển thị dữ liệu cho người dùng. Bạn có thể sử dụng View để hiển thị dữ liệu từ Model và kết hợp nó với giao diện người dùng.

Khi EF được tích hợp vào Mô hình MVC, bạn có thể thực hiện các hoạt động như thêm, sửa đổi, xóa và truy vấn dữ liệu từ cơ sở dữ liệu một cách dễ dàng bằng cách sử dụng các đối tượng Model. Entity Framework giúp giảm thiểu công việc lặp đi lặp lại liên quan đến truy cập cơ sở dữ liệu và tạo mã dễ bảo trì hơn.

6.3. HTML, CSS, JavaScript:



Trong lập trình, HTML, CSS và JavaScript được xem là những thành phần không thể thiếu để hình thành nên một website hoạt động ổn định và bắt mắt như hiện nay.

● Tìm hiểu về HTML

HTML là gì?

HTML - HyperText Markup Language, là loại ngôn ngữ đánh dấu siêu văn bản, thường được sử dụng trong các tài liệu web.

Nhưng xét về bản chất, HTML lại không được xem như một ngôn ngữ lập trình, mà nó giống như một ngôn ngữ xác định đâu là ý nghĩa, mục đích và cấu trúc của một tài liệu.

Do đó, thay vì sử dụng ngôn ngữ lập trình để thực hiện các chức năng, HTML lại dùng các thẻ để xác định các loại nội dung khác nhau, cũng như mục đích phục vụ cho trang web.

Vai trò của HTML:

HTML có nhiệm vụ xây dựng cấu trúc siêu văn bản trên một website, hay khai báo các tập tin kỹ thuật số như hình ảnh, video, nhạc.

Ngoài ra, nó còn được dùng để phân chia các đoạn văn bản, heading, links, blockquotes...

Đồng thời thực hiện dễ dàng các công việc sau:

- Thêm tiêu đề, định dạng văn bản, ngắt dòng điều khiển.
- Tạo danh sách, nhấn mạnh văn bản, tạo ký tự đặc biệt, chèn hình ảnh và tạo liên kết.
- Góp phần xây dựng bảng, điều khiển một số kiểu mẫu.

● Tìm hiểu về CSS:

CSS là gì?

CSS - Cascading Style Sheets, là ngôn ngữ định dạng được sử dụng để mô tả cách trình bày trong các trang web, bao gồm màu sắc, cách bố trí và phông chữ, giúp nội dung thể hiện được phong cách riêng cho tổng thể website.

Nhờ có CSS nên các thành phần trong HTML của website có thể hiện thị bắt mắt trên giao diện của trang.

CSS và HTML hoạt động hoàn toàn độc lập, đồng thời có thể sử dụng trên bất kỳ ngôn ngữ đánh dấu nào được xây dựng dựa trên XML.

CSS hoạt động tuân theo chuẩn chung mà W3C đưa ra.

Trong CSS được phân thành 3 phong cách chính, đó là:

- Style CSS Internal: Là style được tải lên mỗi khi trang web được refresh.
- Style CSS Inline: Ở phong cách này, bạn có thể dễ dàng chỉnh sửa bất kỳ yếu tố nào đó, mà không cần truy cập trực tiếp vào file CSS.
- External Style: Giúp bạn có thể tạo phong cách ở file khác áp dụng CSS vào trang bạn muốn, đồng thời cải thiện tối đa thời gian tải trang. Đặc biệt, bạn có thể ứng dụng phong cách này để tạo phong cách cho nhiều trang cùng thời điểm.

Vai trò của CSS:

CSS sẽ giúp bạn xây dựng nên “bộ mặt” hoàn hảo cho một website, đồng thời cũng mang đến cho nó phần giao diện “ưa nhìn” và trông lôi cuốn hơn trong quá trình giao tiếp.

Ngoài ra, CSS còn giúp webmaster xác định được phong cách và định nghĩa nội dung web.

Thông qua đó, nhà lập trình có thể thực hiện nhanh chóng hơn các công việc sau:

- Tạo style và định dạng cho những nội dung được thể hiện dưới dạng văn bản của HTML.
- Tiết kiệm thời gian và công sức của lập trình viên nhờ khả năng điều khiển định dạng.
- Phân biệt cách hiển thị của trang web và nội dung của trang qua việc căn chỉnh bố cục, màu sắc cũng như font chữ.

Tìm hiểu về JavaScript

JavaScript là gì?

JavaScript hay còn được viết tắt là JS, đây là một trong những ngôn ngữ lập trình phổ biến được nhiều lập trình viên sử dụng, do Brendan Eich ra mắt vào năm 1995.

Nhờ các tính năng nổi bật của mình mà hầu hết các Slideshow, Pop-up quảng cáo, cùng tính năng Autocomplete của Google đều có sự góp sức của JavaScript.

JavaScript được sử dụng nhằm mục đích biến những website tĩnh nhàm chán sang trang thái động cuốn hút, đồng thời có khả năng tạo sự tương tác, cải thiện hiệu suất máy chủ và nâng cao trải nghiệm người dùng.

Nhưng để thực hiện tốt những hoạt động trên, JavaScript phải cần đến sự hỗ trợ của HTML và CSS.

Vai trò của JavaScript:

JavaScript còn được biết đến là một ngôn ngữ lập trình cho phép các nhà phát triển web thiết kế ra các trang web có khả năng tương tác.

Nhờ có sự hỗ trợ của JavaScript, mà các công việc trở nên đơn giản và dễ dàng hơn khi thực hiện như:

- Bắt đầu chỉnh sửa với thư viện ảnh, nhằm mục đích tạo nên một bố cục chuyển động.
- Hỗ trợ xây dựng các trò chơi, hoạt hình 2D hoặc 3D, ứng dụng cơ sở dữ liệu toàn diện để hoàn thiện website.
- Tăng cường các hành vi người dùng và kiểm soát mặc định trình duyệt.

● Mối quan hệ giữa HTML, CSS và JavaScript:

Qua các phần giải thích về HTML, CSS và JavaScript, bạn có thể thấy mối quan hệ giữa 3 ngôn ngữ này như sau:

- HTML là khung giao diện chính: HTML xác định cấu trúc và nội dung của trang web, đồng thời, tạo ra cơ sở cho việc thêm giao diện và tương tác.
- CSS là lớp trang trí: CSS được sử dụng để tùy chỉnh giao diện, làm cho trang web trở nên hấp dẫn hơn. Nó định nghĩa màu sắc, kích thước, phông chữ và các thuộc tính khác.
- JavaScript cung cấp tương tác: JavaScript thêm tính năng tương tác, cho phép người dùng tương tác với trang web và thực hiện các hành động động.

6.4. Công cụ và môi trường: SQL Server và Visual Studio .NET:

- **Lập trình cơ sở dữ liệu (SQL Server):**



Giới thiệu về SQL Server:

SQL Server là một hệ quản trị cơ sở dữ liệu phát triển bởi Microsoft.

Nó được sử dụng rộng rãi để lưu trữ, quản lý và truy vấn dữ liệu trong các ứng dụng và hệ thống doanh nghiệp.

Ngôn ngữ truy vấn SQL:

SQL (Structured Query Language) là ngôn ngữ sử dụng để tương tác với cơ sở dữ liệu SQL Server.

SQL được sử dụng để tạo, cập nhật, xóa và truy vấn dữ liệu từ cơ sở dữ liệu.

Các loại câu lệnh SQL:

SQL Server hỗ trợ nhiều loại câu lệnh SQL, bao gồm SELECT (truy vấn dữ liệu), INSERT (chèn dữ liệu), UPDATE (cập nhật dữ liệu), DELETE (xóa dữ liệu), và các câu lệnh quản lý cơ sở dữ liệu khác.

Thao tác với bảng và dữ liệu:

SQL Server cho phép bạn tạo bảng để lưu trữ dữ liệu và thực hiện thao tác CRUD (Create, Read, Update, Delete) trên bảng và dữ liệu.

Thực hiện các chức năng phức tạp:

SQL Server cung cấp khả năng thực hiện các chức năng phức tạp như thủ tục lưu trữ, chức năng (function), gọi ý (trigger), và quyền truy cập để quản lý cơ sở dữ liệu.

- **Công cụ xây dựng cơ sở dữ liệu SQL Server:**

SQL Server Management Studio (SSMS):

SQL Server Management Studio là một công cụ quản lý cơ sở dữ liệu phát triển bởi Microsoft để quản lý và tương tác với cơ sở dữ liệu SQL Server.

SSMS cho phép bạn thiết kế cơ sở dữ liệu, tạo bảng, quyền truy cập, thủ tục lưu trữ, và thực hiện truy vấn SQL.

SQL Server Data Tools (SSDT):

SQL Server Data Tools là một môi trường phát triển tích hợp (IDE) dùng để phát triển và quản lý dự án cơ sở dữ liệu trên SQL Server.

SSDT cung cấp các công cụ để thiết kế, quản lý phiên bản và triển khai cơ sở dữ liệu SQL Server.

Entity Framework (EF):

Entity Framework đã được đề cập trong phần lý thuyết về C#. EF cho phép bạn tương tác với cơ sở dữ liệu SQL Server bằng cách sử dụng đối tượng và LINQ, giúp giảm công việc truy cập cơ sở dữ liệu và tạo mã dễ bảo trì hơn.

Lập trình kịch bản (SQL Scripting):

Bạn có thể sử dụng SQL Server để tạo và quản lý các kịch bản SQL để thực hiện các thay đổi cơ sở dữ liệu, sao lưu và khôi phục dữ liệu, và thực hiện quy trình tự động.

- **Visual Studio .NET:**

Visual Studio .NET (hoặc gọi tắt là Visual Studio) là một môi trường phát triển tích hợp (IDE) do Microsoft phát triển để phát triển ứng dụng Windows, ứng dụng web, ứng dụng di động, ứng dụng điện toán đám mây và nhiều loại ứng dụng khác. Dưới đây là một lý thuyết tổng quan về Visual Studio .NET



Phiên bản và Lịch sử:

Visual Studio .NET ra đời lần đầu vào năm 2002 và đã trải qua nhiều phiên bản và bản cập nhật kể từ đó.

Mỗi phiên bản có sự cải tiến và bổ sung mới, bao gồm nâng cấp giao diện, hiệu suất, và tích hợp với các công nghệ và khung làm việc mới.

Hỗ trợ nhiều ngôn ngữ và nền tảng:

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình như C#, VB.NET, C++, Python, và nhiều ngôn ngữ khác.

Nó cho phép phát triển ứng dụng cho nhiều nền tảng, bao gồm Windows, web, di động, đám mây và IoT (Internet of Things).

Giao diện và Trình biên soạn mã:

Visual Studio có một giao diện người dùng dễ sử dụng với nhiều tính năng thuận tiện như trình biên soạn mã IntelliSense, tự động hoàn thiện mã, kiểm tra lỗi ngữ nghĩa và đánh dấu cú pháp.

Nó hỗ trợ trình chỉnh sửa mã mở rộng và tùy chỉnh giao diện người dùng.

Debug và Kiểm tra mã nguồn:

Visual Studio cung cấp các công cụ mạnh mẽ để gỡ lỗi và kiểm tra mã nguồn bao gồm chế độ gỡ lỗi từng bước, theo dõi biến và lệnh điều khiển, và quản lý phiên gỡ lỗi.

Hỗ trợ Source Control:

Visual Studio tích hợp sâu với các hệ thống quản lý phiên bản như Git và TFS (Team Foundation Server). Điều này cho phép bạn quản lý mã nguồn dễ dàng, theo dõi sự thay đổi và làm việc cùng nhóm.

Xây dựng và Triển khai ứng dụng:

Visual Studio cho phép bạn xây dựng và triển khai ứng dụng một cách dễ dàng thông qua quy trình đóng gói và triển khai tích hợp.

Hỗ trợ Công nghệ và Khung làm việc:

Visual Studio hỗ trợ một loạt các công nghệ và khung làm việc, bao gồm ASP.NET, .NET Core, WPF, Xamarin, Azure, và nhiều công nghệ và khung làm việc khác.

Cộng đồng và Thư viện mở rộng:

Visual Studio có một cộng đồng lớn và tích hợp một thư viện mở rộng mạnh mẽ để giúp bạn mở rộng và tùy chỉnh môi trường phát triển của mình.

Hỗ trợ Công cụ bên ngoài:

Visual Studio có thể tích hợp với nhiều công cụ bên ngoài để mở rộng khả năng của nó, bao gồm công cụ quản lý dự án, công cụ thiết kế giao diện và nhiều công cụ khác.

Visual Studio .NET là một IDE mạnh mẽ và linh hoạt để phát triển ứng dụng trên nhiều nền tảng và ngôn ngữ. Nó là công cụ ưa thích của nhiều lập trình viên và doanh nghiệp trên toàn thế giới.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ

1. Phân tích

1.1 Mô tả nghiệp vụ

1.1.1 Dành cho khách hàng

- Website được xây dựng với mục tiêu là nơi trưng bày, giới thiệu sản phẩm và các dịch vụ hỗ trợ khách hàng như: liên hệ, phản hồi, tư vấn, tìm kiếm...
- Khách hàng bắt đầu duyệt những sản phẩm trên website, các sản phẩm được bố trí theo từng loại đa dạng như sản phẩm mới, sản phẩm bán chạy... Khi đã ưng ý một sản phẩm khách hàng có thể kích vào hình sản phẩm hoặc tên để xem thông tin chi tiết về sản phẩm. Sau đó khách hàng có thể chọn nút **“Mua Ngay”** để đưa sản phẩm vào giỏ hàng của mình, giỏ hàng chỉ đơn giản là danh sách các sản phẩm bao gồm những thông tin như mã sản phẩm, tên sản phẩm, đơn giá, số lượng, thành tiền.
- Các giỏ hàng thường cung cấp các tùy chọn để xóa sạch giỏ hàng, xóa một sản phẩm, tiếp tục mua sản phẩm và cập nhật số lượng.
- Ngay sau khi khách hàng có tất cả các sản phẩm cần mua, khách hàng có thể chọn nút **“Đặt Mua”**. Trang đơn hàng này gồm có thông tin về khách hàng (họ tên, email, địa chỉ...), thông tin về địa chỉ giao hàng (địa chỉ, ngày giao...), phương thức thanh toán (chuyển khoản, thanh toán tiền trực tuyến), sau cùng là thông tin về giỏ hàng (mã sản phẩm, tên sản phẩm, đơn giá, số lượng, thành tiền).
- Sau khi khách hàng đã điền đầy đủ thông tin thì chọn nút **“Đặt Hàng”** để hoàn tất quá trình mua hàng.

1.1.2 Dành cho quản trị

- Website cung cấp cho nhân viên hệ thống giao diện dùng để quản trị các thông tin như sản phẩm, đơn hàng, khách hàng, tin tức, góp ý, thống kê...
- Tùy vào quyền hạn của mỗi nhân viên mà chức năng của mỗi người khác nhau.

1.2 Yêu cầu hệ thống

Yêu cầu chức năng

Gồm có các đối tượng chính sử dụng website là: Người dùng chung, Khách hàng thành viên, Người quản trị.

Chức năng của mỗi đối tượng:

- Người dùng chung:
 - Xem danh sách sản phẩm: Khi người dùng truy cập vào website thì hệ thống sẽ tự động hiển thị một số sản phẩm nổi bật. Khi người dùng chọn vào một danh mục sản phẩm cụ thể nào thì tất cả sản phẩm của danh mục đó sẽ hiển thị lên giao diện

- Xem chi tiết sản phẩm: Sau khi tìm thấy sản phẩm cần, người dùng có thể chọn vào sản phẩm để xem thông tin chi tiết của sản phẩm đó, tại đây khách hàng có thể xem thêm các sản phẩm có liên quan thông qua ứng dụng thông minh đề xuất sản phẩm.
 - Tìm kiếm: Khi khách hàng cần tìm một sản phẩm nào đó thì có thể nhập thông tin sản phẩm vào ô tìm kiếm để thực hiện chức năng tìm kiếm. Nếu sản phẩm đó có tồn tại trong dữ liệu thì sẽ hiện thông tin sản phẩm được tìm thấy.
 - Đăng ký: Khi Người dùng chung muốn mua hàng thì có thể thực hiện chức năng đăng ký để trở thành thành viên của website và thực hiện các chức năng mua hàng.
 - Đưa sản phẩm vào giỏ hàng: Sau khi khách hàng tìm thấy sản phẩm đáp ứng nhu cầu của mình thì có thể chọn đưa sản phẩm vào giỏ hàng.
 - Cập nhật giỏ hàng: Khi người dùng đã lựa chọn một sản phẩm nhưng tìm được một sản phẩm hợp lý hơn thì có thể xóa sản phẩm đã có trong giỏ hàng vào chọn lại sản phẩm mới. Ngoài ra, khi cần mua sản phẩm đó với một số lượng thì người dùng cũng có thể cập nhật trong giỏ hàng.
- Khách hàng thành viên: Ngoài các chức năng của Người dùng chung còn có một số chức năng sau:
- Đặt mua: Sau khi xem thông tin sản phẩm đúng với nhu cầu thì người dùng có thể chọn đặt mua để đưa sản phẩm vào giỏ hàng.
 - Xem danh sách sản phẩm: Khách hàng có thể xem lại các sản phẩm mà mình từng đặt mua trên website.
 - Quản lý đơn hàng: Sau khi đặt mua mà chưa thực hiện thanh toán thì khách hàng có thể cập nhật lại đơn hàng.
 - Thanh toán: Khi đã quyết định chắc chắn mua sản phẩm, người dùng có thể chọn chức năng thanh toán để thực hiện xác nhận đơn hàng cho mình. Muốn thực hiện thanh toán thì yêu cầu người dùng phải đăng nhập vào hệ thống. Trong quá trình thực hiện chức năng thanh toán thì người dùng có thể thấy được danh sách sản phẩm mà mình đặt mua trong giỏ hàng. Người dùng cần điền một số thông tin như người nhận (người nhận và người đặt mua là hai người khác nhau), chọn hình thức giao hàng (giao đến tận nơi, chuyển qua các dịch vụ, khách hàng đến công ty để nhận hàng), chọn hình thức thanh toán (thanh toán chuyển khoản, trực tiếp khi nhận hàng, thanh toán qua các dịch vụ thanh toán trực tuyến như ngân lượng, bảo kim). Sau khi thực hiện xong các bước thì người dùng xác nhận đơn hàng.
- Người quản trị: Có tất cả các chức năng của khách hàng thành viên và có thêm chức năng quản lý tùy vào quyền hạn của mỗi nhân viên:

- + Quản lý sản phẩm, đơn hàng, khách hàng, một số thông tin khác: Để thực hiện chức năng quản lý thì yêu cầu nhân viên phải đăng nhập hệ thống. Nhân viên chọn mục mình cần thực hiện quản lý như sản phẩm. Trong quản lý sản phẩm có xem, thêm, sửa, xóa sản phẩm. Chọn thao tác cần thực hiện, điền đầy đủ thông tin trong quá trình thực hiện rồi xác nhận thao tác.
- + Tư vấn, giải đáp thắc mắc, phản hồi ý kiến: Để thực hiện chức năng quản lý thì yêu cầu nhân viên phải đăng nhập hệ thống. Nếu có người sử dụng cần tư vấn, thắc mắc, ý kiến đóng góp thì nhân viên sẽ làm việc thông qua các dịch vụ như yahoo, skype và cũng thể giải đáp trực tiếp trên website.

1.3 Bảo mật và quyền hạn

- Đảm bảo tính toàn vẹn dữ liệu phải định kỳ lưu giữ thông tin đề phòng khi có sự cố còn có thể khắc phục một cách dễ dàng.
- Quyền hạn mỗi User được bảo vệ chặt chẽ. Chỉ có Admin mới có quyền tạo mới các User và cập nhật dữ liệu.

2. Mô hình UseCase

Khái niệm UseCase (UC) được Jacobson đề xuất vào năm 1994 khi làm việc cho hãng Ericsson. UC mô tả ai đó sử dụng hệ thống như thế nào, mô tả tương tác giữa người sử dụng với hệ thống phần mềm để thực hiện các thao tác giải quyết công việc cụ thể nào đó. UC không cho biết hệ thống làm việc bên trong như thế nào. UC không phải là thiết kế, cũng không phải là kế hoạch cài đặt, UC là một phần của vấn đề cần giải quyết. Tiến trình của hệ thống được chia nhỏ thành các UC để có thể nhận ra từng bộ phận của UC một cách rõ ràng và để nhiều người có thể cùng xử lý.

UC là nền tảng của phân tích hệ thống. Việc tìm ra đầy đủ các UC đảm bảo rằng hệ thống sẽ xây dựng đáp ứng mọi nhu cầu của người sử dụng. Mỗi UC là tập hành động. Mỗi hành động là cái gì đó mà hệ thống làm, UC là hạt nhân được hệ thống thực hiện hoàn toàn hay không được hiện phần nào.

Bảng 2: Danh sách các Actor

Actor	Chức Năng
Người dùng chung	Xem danh sách sản phẩm Xem thông tin chi tiết sản phẩm Đăng ký Tìm kiếm sản phẩm Đưa sản phẩm vào giỏ hàng Xem tin tức

Khách hàng thành viên	<p>Có các chức năng như Người dùng chung và còn có thêm một số chức năng khác như:</p> <p>Đặt Mua</p> <p>Xem danh sách đơn hàng</p> <p>Quản lý đơn hàng</p> <p>Thanh toán</p> <p>Đăng nhập</p>
Người quản trị	<p>Có tất cả chức năng của Khách hàng thành viên và thêm một số chức năng khác và yêu cầu Đăng Nhập mới được sử dụng:</p> <p>Xem danh sách đơn hàng</p> <p>Cập nhật đơn hàng</p> <p>Cập nhật sản phẩm</p> <p>Cập nhật đơn giá</p> <p>Ghi nhận đơn hàng</p> <p>...</p> <p>Các chức năng còn tùy thuộc vào quyền của nhân viên. Mỗi người có chức năng nhất định</p>
Hệ thống thanh toán (Actor phần mềm)	Tác động đến hệ thống thanh toán trực tuyến

		người quản trị
7	Đăng nhập	Khách hàng thành viên, người quản trị
8	Đặt mua	Khách hàng thành viên, người quản trị
9	Thanh toán	Khách hàng thành viên, người quản trị
10	Xem danh sách đơn hàng	Khách hàng thành viên, người quản trị, Người dùng chung
11	Quản lý đơn hàng	Khách hàng thành viên, người quản trị
12	Quản lý sản phẩm	Người quản trị
13	Quản lý khách hàng	Người quản trị
14	Quản lý một số thông tin khác	Người quản trị
15	Thống kê, báo cáo	Người quản trị

2.2 Đặc tả Use Case

2.2.1 Use Case “Đăng ký”

Tóm tắt: Người dùng chung sử dụng usecase “Đăng ký” để tạo tài khoản cho mình trên website.

Actor: Người dùng chung

Các dòng sự kiện chính

B1. Trên giao diện màn hình chính, Người dùng chung chọn **Đăng ký**.

B2. Hệ thống sẽ hiển thị giao diện đăng ký và Người dùng chung nhập thông tin vào giao diện để lưu vào cơ sở dữ liệu.

B3. Kết thúc Usecase.

Các dòng sự kiện khác: Nếu người dùng không muốn tạo tài khoản thì chọn **Hủy**

Các yêu cầu đặc biệt: Không có

Trạng thái hệ thống khi bắt đầu thực hiện Use-case: Trước khi bắt đầu thực hiện Use-case không cần điều kiện gì.

Trạng thái hệ thống sau khi thực hiện Use-case: Sau khi thực hiện Use-case hệ thống sẽ thông báo tài khoản tạo thành công hay chưa

Phác thảo giao diện:

Sign Up

Sign In

User Name

User Name

Email

Email Address

Name

Name

Mobile

Mobile

Address

Address

Password

Password

Password again

Password again

Confirm

 or

Sign Up with Facebook

3.1.4 Use Case “Đưa sản phẩm vào giỏ hàng”

Tóm tắt: Người dùng chung sử dụng usecase “Đưa sản phẩm vào giỏ hàng” để đặt những sản phẩm mình cần mua vào không gian lưu trữ tạm thời trên web.

Actor: Người dùng chung

Các dòng sự kiện chính

B1. Trên giao diện màn hình chi tiết sản phẩm, Người dùng chung chọn **Đưa vào giỏ hàng**

B2. Hệ thống sẽ lưu trữ thông tin sản phẩm mà Người dùng chung đưa vào giỏ

B3. Kết thúc Usecase.

Các dòng sự kiện khác: Không có

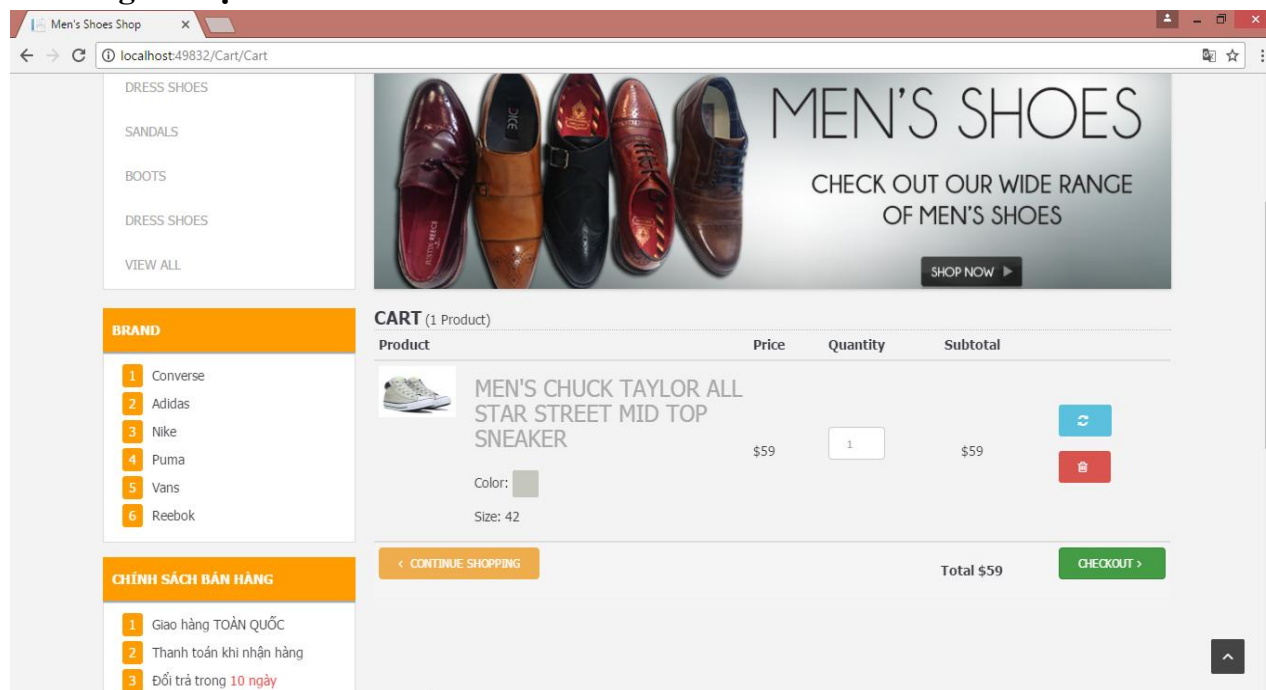
Các yêu cầu đặc biệt: Không có

Trạng thái hệ thống khi bắt đầu thực hiện Use-case: Sau khi khách hàng xem danh sách sản phẩm hoặc thông tin sản phẩm.

Trạng thái hệ thống sau khi thực hiện Use-case: Sau khi thực hiện Use-case hệ thống sẽ xuất thông tin của sản phẩm ra giao diện Giỏ hàng.

Điểm mở rộng: Tại giao diện giỏ hàng sẽ có các chức năng: Xóa giỏ hàng, Tiếp tục mua, Cập nhật số lượng cho sản phẩm đã đặt, Xóa 1 hoặc nhiều sản phẩm

Phác thảo giao diện:



2.2.3 Use Case “Đăng nhập”

Tóm tắt: Khách hàng thành viên sử dụng usecase “Đăng nhập” để tham gia mua hàng trực tuyến.

Actor: Khách hàng thành viên

Các dòng sự kiện chính

B1. Trên giao diện màn hình chính, Khách hàng thành viên chọn **Đăng nhập**.

B2. Hệ thống sẽ hiển thị giao diện đăng nhập và Khách hàng thành viên nhập thông tin vào giao diện để kiểm tra tài khoản đã có hay chưa.

B3. Kết thúc Usecase.

Các dòng sự kiện khác: Nếu Khách hàng thành viên không muốn đăng nhập thì chọn **Thoát**

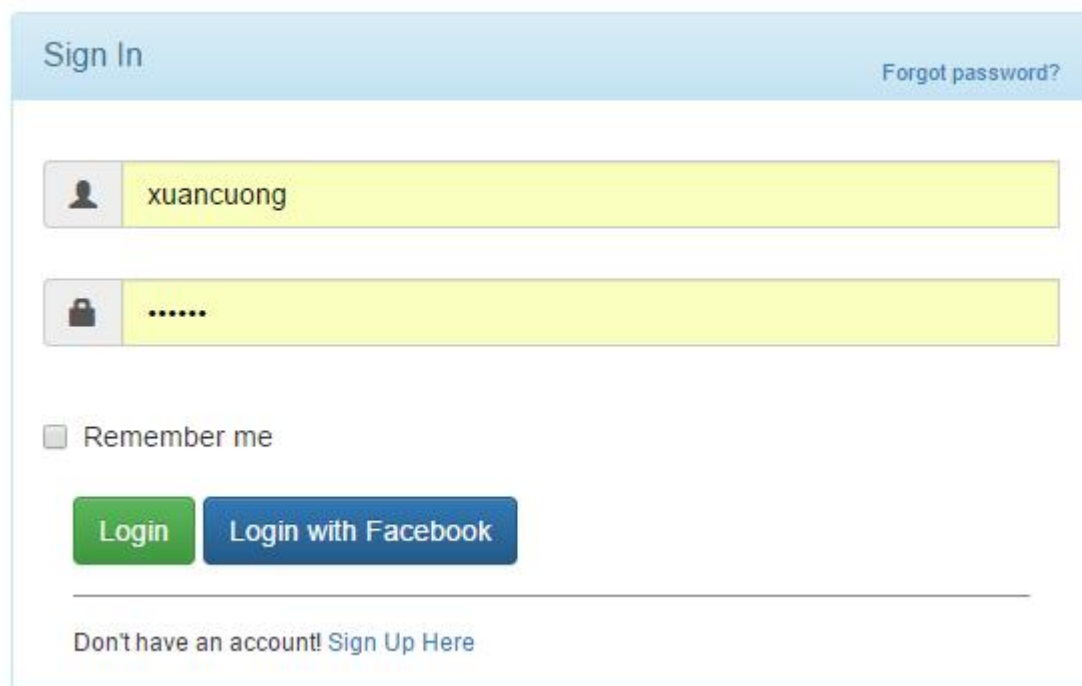
Các yêu cầu đặc biệt: Không có

Trạng thái hệ thống khi bắt đầu thực hiện Use-case: Trước khi bắt đầu thực hiện Use-case yêu cầu phải Đăng ký.

Trạng thái hệ thống sau khi thực hiện Use-case: Sau khi thực hiện Use-case hệ thống sẽ thông báo đăng nhập thành công hay chưa

Điểm mở rộng: Khách hàng thành viên có thể chọn thoát khỏi chế độ đăng nhập bất cứ khi nào (yêu cầu trước đó đã đăng nhập thành công)

Phác thảo giao diện:



The image shows a 'Sign In' form with a light blue header. In the top right corner of the header is a link 'Forgot password?'. Below the header, there are two input fields with yellow backgrounds. The first field has a person icon and the text 'xuancuong'. The second field has a lock icon and six dots. Below these fields is a checkbox labeled 'Remember me'. At the bottom of the form are two buttons: a green 'Login' button and a blue 'Login with Facebook' button. Below the buttons is a horizontal line, and at the very bottom is the text 'Don't have an account! Sign Up Here'.

2.2.4 Use Case “Đặt mua”

Tóm tắt: Khách hàng thành viên sử dụng usecase “Đặt mua” để tham gia mua hàng trực tuyến.

Actor: Khách hàng thành viên

Các dòng sự kiện chính

B1. Trên giao diện giỏ hàng, Khách hàng thành viên chọn **Đặt mua**.

B2. Hệ thống sẽ hiển thị giao diện chứa thông tin khách hàng và danh sách các sản phẩm khách hàng đặt mua. Sau khi nhập đầy đủ thông tin thì khách hàng xác nhận đặt mua.

B3. Kết thúc Usecase.

Các dòng sự kiện khác: Nếu Khách hàng thành viên không muốn đặt mua thì chọn **Thoát**

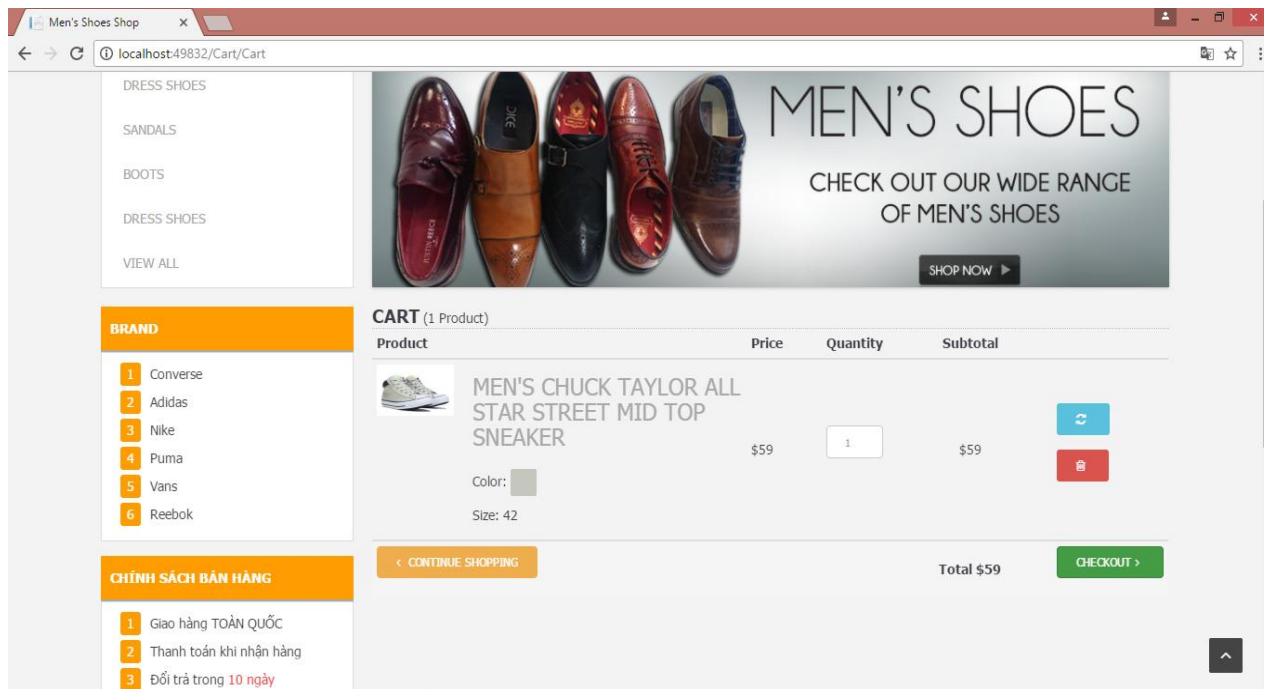
Các yêu cầu đặc biệt: Để thực hiện được usecase này yêu cầu khách hàng thành viên phải thực hiện đăng nhập vào hệ thống

Trạng thái hệ thống khi bắt đầu thực hiện Use-case: Trong giỏ hàng của khách hàng phải có tối thiểu 1 sản phẩm

Trạng thái hệ thống sau khi thực hiện Use-case: Sau khi thực hiện Use-case hệ thống sẽ thông báo đăng nhập thành công hay chưa

Điểm mở rộng: Khách hàng thành viên có thể chọn thoát khỏi chế độ đăng nhập bất cứ khi nào (yêu cầu trước đó đã đăng nhập thành công)

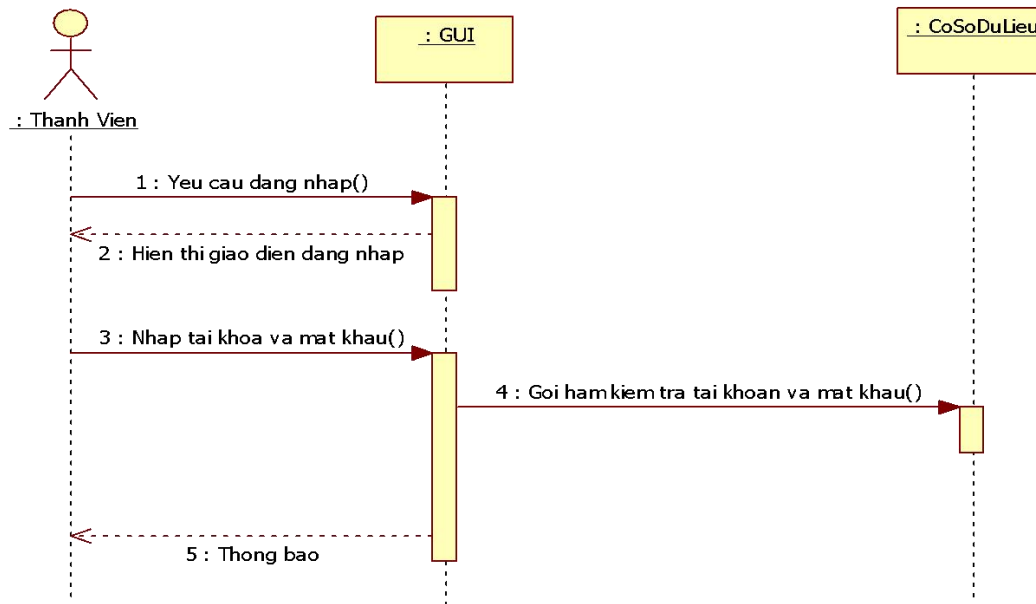
Phác thảo giao diện:



3 Sơ đồ tuần tự

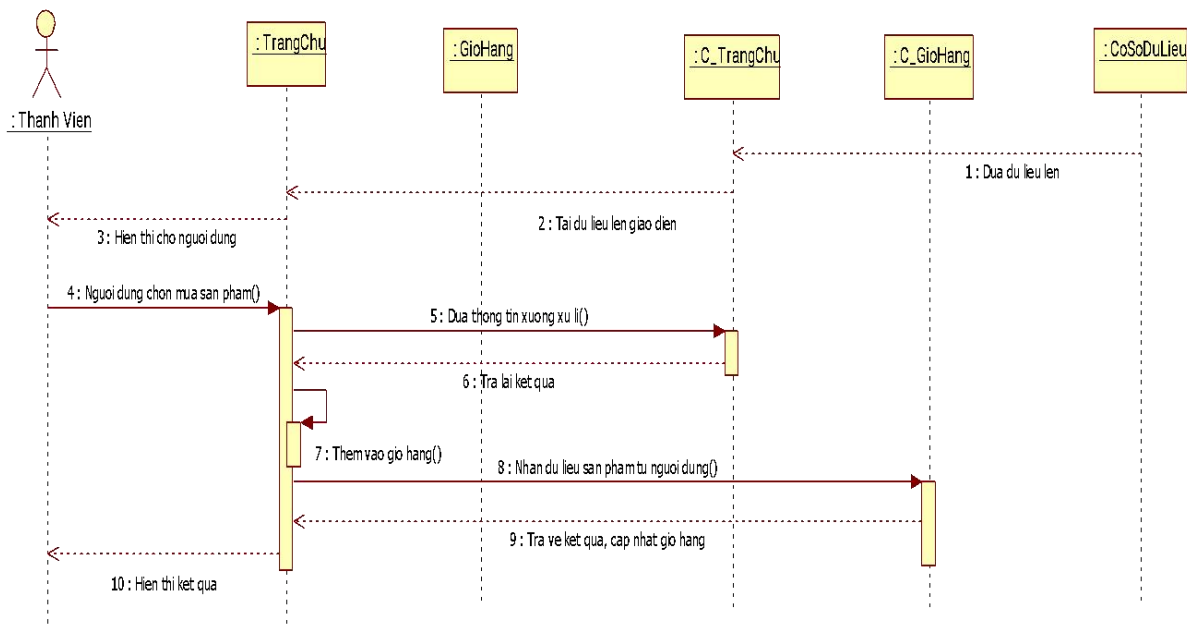
Sơ đồ tuần tự cho ta cái nhìn tổng quát về trình tự xử lý của hệ thống đối với từng chức năng cụ thể. Nó chỉ ra luồng chức năng xuyên qua các UseCase, nó là biểu đồ mô tả tương tác giữa các đối tượng và tập trung vào mô tả trật tự các thông điệp theo thời gian.

3.1 Đăng nhập



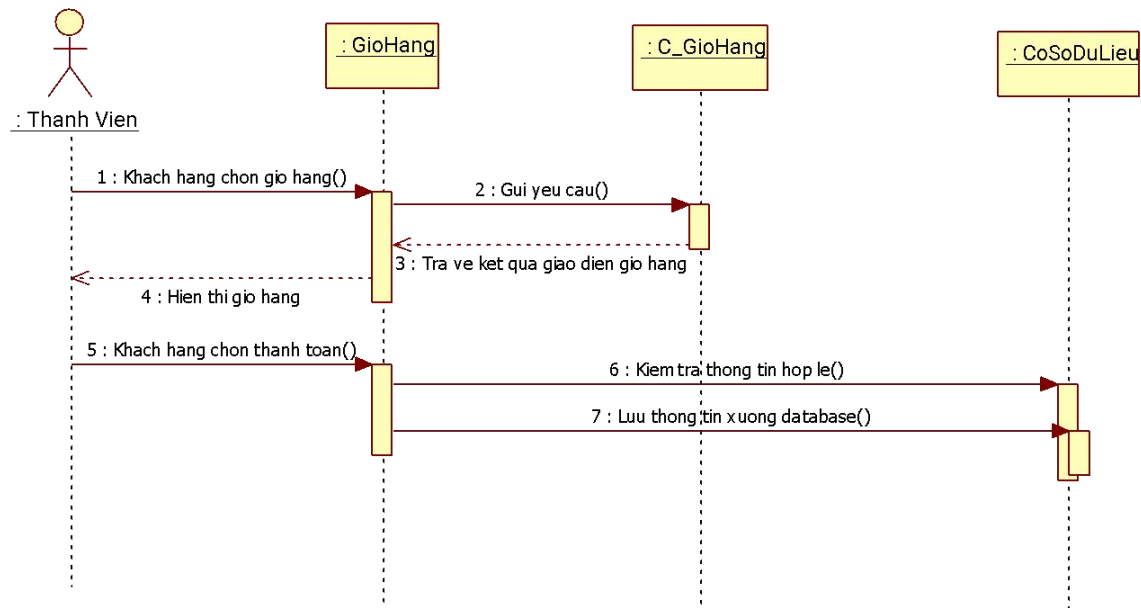
Hình 7: Sơ đồ tuần tự quy trình đăng nhập

3.2 Quy trình thêm sản phẩm vào giỏ hàng



Hình 8: Sơ đồ tuần tự quy trình thêm sản phẩm vào giỏ hàng

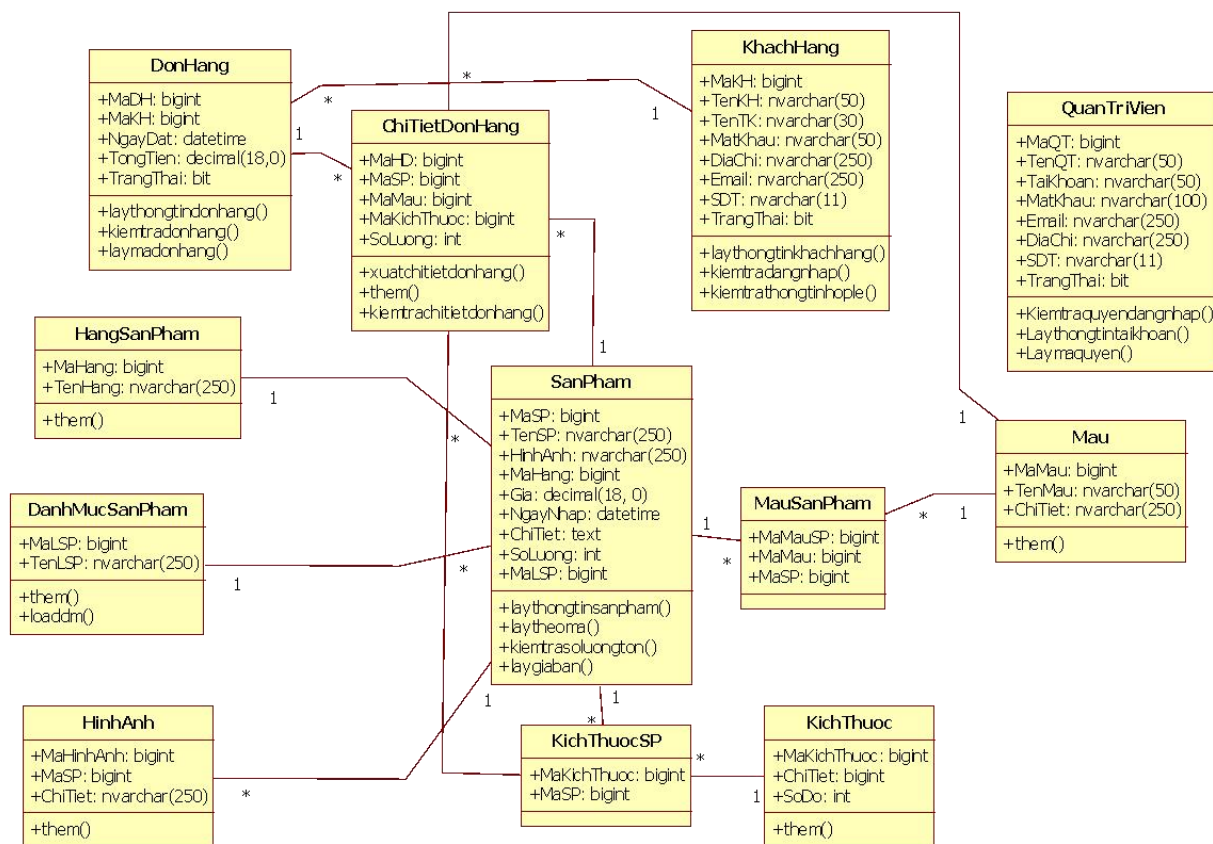
3.3 Quy trình đặt mua sản phẩm



Hình 9: Sơ đồ tuần tự quy trình đặt mua sản phẩm

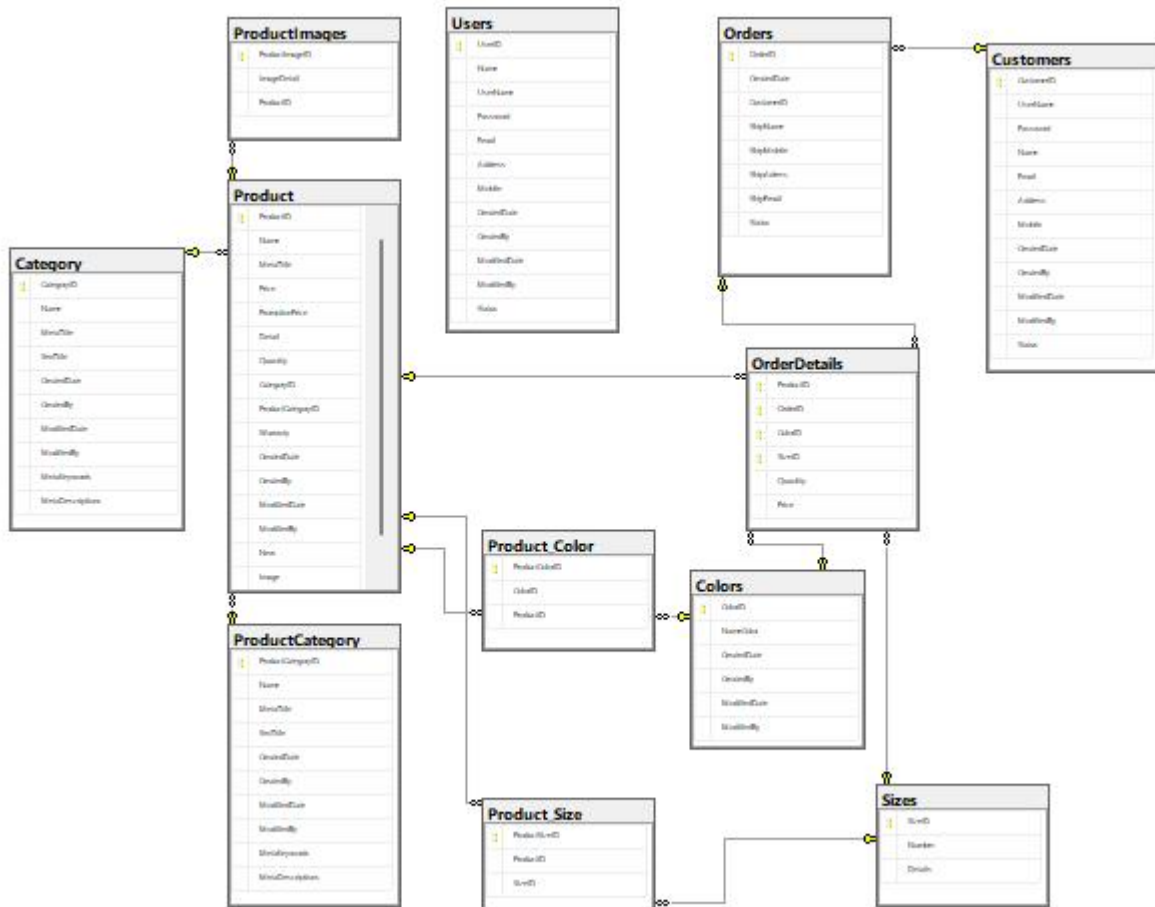
4. Sơ đồ lớp

4.1 Sơ đồ lớp mức căn bản



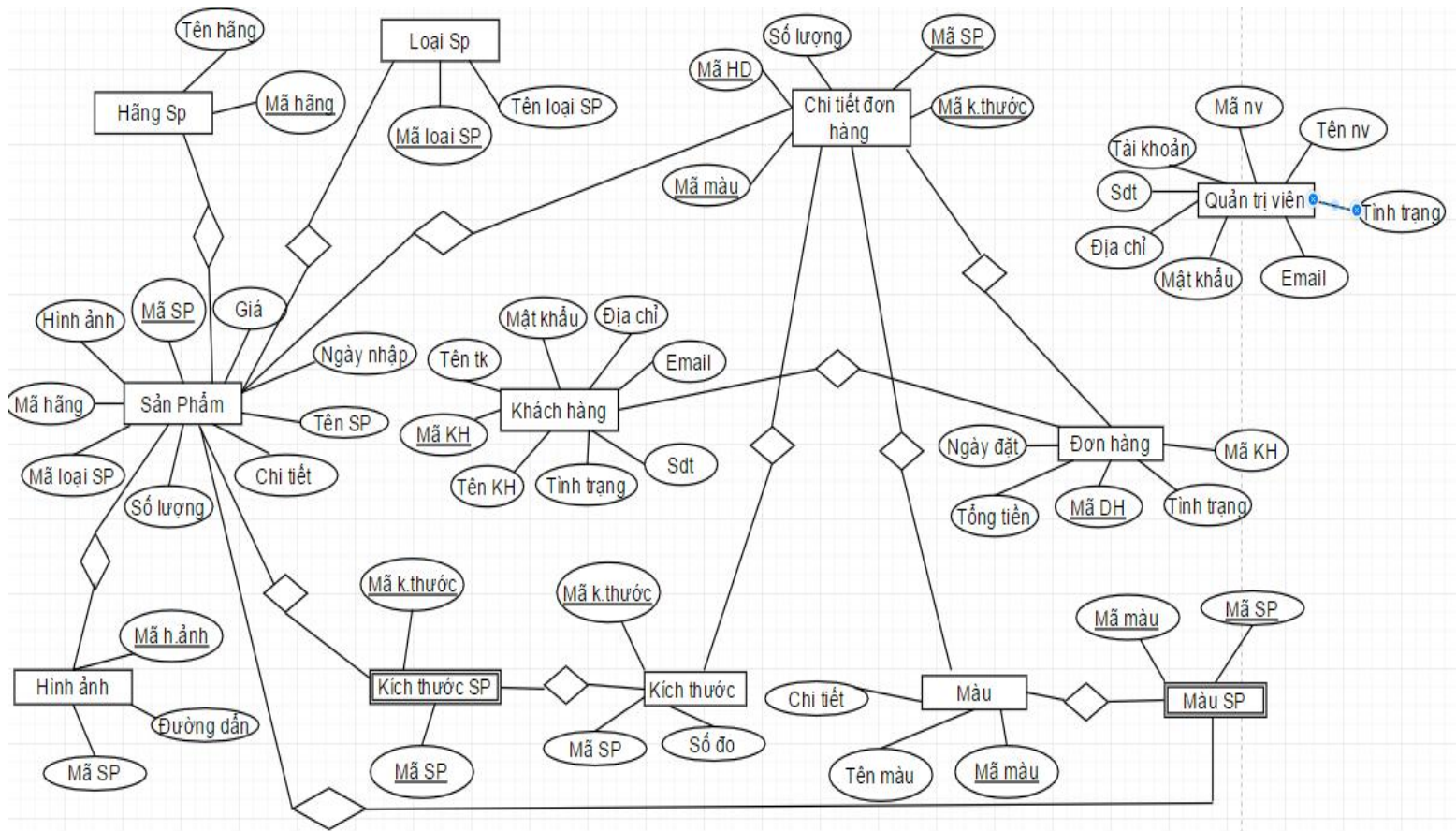
Hình 10: Sơ đồ lớp

5. Sơ Đồ Diagram



Link drive database:<https://drive.google.com/file/d/1BPXx5jPemtnA4EaxgA3ltvey-cOep0AO/view?usp=sharing>

6. Sơ Đồ ERD



7. Sơ Đồ cơ sở dữ liệu

a. Product

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductID</u>	bigint	Mã sản phẩm	Khóa chính
Name	nvarchar(250)	Tên sản phẩm	
Price	decimal(18, 0)	Giá	
Detail	Text	Chi tiết	
Email	nvarchar(MAX)	Email	
Quantity	int	Số lượng còn	
CreatedDate	datetime	Ngày lấy sp	Getdate
Image	nvarchar(250)	Ảnh đại diện	
CategoryID	Bigint	Mã loại sản phẩm	Khóa ngoại
ProductCategoryID	Bigint	Mã hãng sản phẩm	Khóa ngoại

b. ProductImage

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductImageID</u>	bigint	Mã hình ảnh	Khóa chính
TImageDetail	nvarchar(250)	Đường dẫn	
ProductID	Bigint	Mã sản phẩm	

c. Category

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>CategoryID</u>	bigint	Mã loại sản phẩm	Khóa chính
Name	Nvarchar(250)	Tên loại sản phẩm	

d. ProductCategory

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductCategoryID</u>	bigint	Mã hãng sản phẩm	Khóa chính,
Name	nvarchar(250)	Tên mã hãng	

e. Sizes

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>SizeID</u>	bigint	Mã kích thước	Khóa chính,
Number	int	Kích thước	
Detail	Text	Chi tiết	

f. Product_Size

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductID</u>	Bigint	Mã sản phẩm	Khóa chung
<u>SizeID</u>	Bigint	Mã kích thước	Khóa chung

g. Colors

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ColorID</u>	Bigint	Mã màu	Khóa chính,
NameColor	nvarchar(50)	Tên màu	

h. Product_Color

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductID</u>	Bigint	Mã sản phẩm	Khóa chung
<u>ColorID</u>	Bigint	Mã màu	Khóa chung

i. User

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>UserID</u>	Bigint	Mã quản trị viên	Khóa chính
Name	nvarchar(50)	Tên quản trị viên	
UserName	nvarchar(50)	Tên đăng nhập	
Password	nvarchar(50)	Mật khẩu	
Email	nvarchar(50)	Địa chỉ Email	
Address	nvarchar(250)	Địa chỉ nhà	
Mobile	nvarchar(11)	Số điện thoại	
Status	bit	Tình trạng	

j. Order

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>OrderID</u>	bigint	Mã hóa đơn	Khóa chính
CustomerID	bigint	Mã khách hàng	
Status	Int	Tình trạng đơn hàng	
TotalAmount	Decimal(18,0)	Tổng tiền đơn hàng	
ShipName	nvarchar(50)	Tên người nhận	
ShipMobile	nvarchar(11)	Số điện thoại người nhận	
ShipAddress	nvarchar(250)	Địa chỉ người nhận	

ShipEmail	nvarchar(100)	Email người nhận	
-----------	---------------	------------------	--

k. Customer

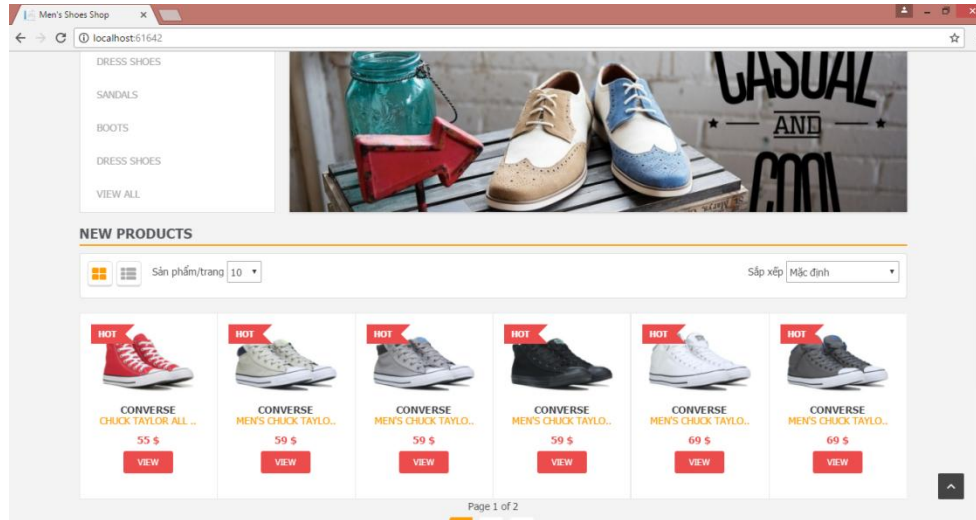
Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>CustomerID</u>	bigint	Mã khách hàng	Khóa chính,
Name	nvarchar(150)	Tên khách hàng	
UserName	nvarchar(50)	Tài khoản	
Password	nvarchar(50)	Mật khẩu	
Address	nvarchar(250)	Địa chỉ khách hàng	
Mobile	nvarchar(11)	Số điện thoại	
Email	nvarchar(100)	Địa chỉ Email	
Status	bit	Tình trạng	

l. OrderDetails

Thuộc tính	Kiểu dữ liệu	Diễn giải	Ràng buộc
<u>ProductID</u>	bigint	Mã sản phẩm	Khóa chung
<u>OrderID</u>	bigint	Mã hóa đơn	Khóa chung
<u>SizeID</u>	bigint	Mã size	Khóa chung
<u>ColorID</u>	bigint	Mã màu	Khóa chung
Quantity	int	Số lượng	

8. Giao diện :

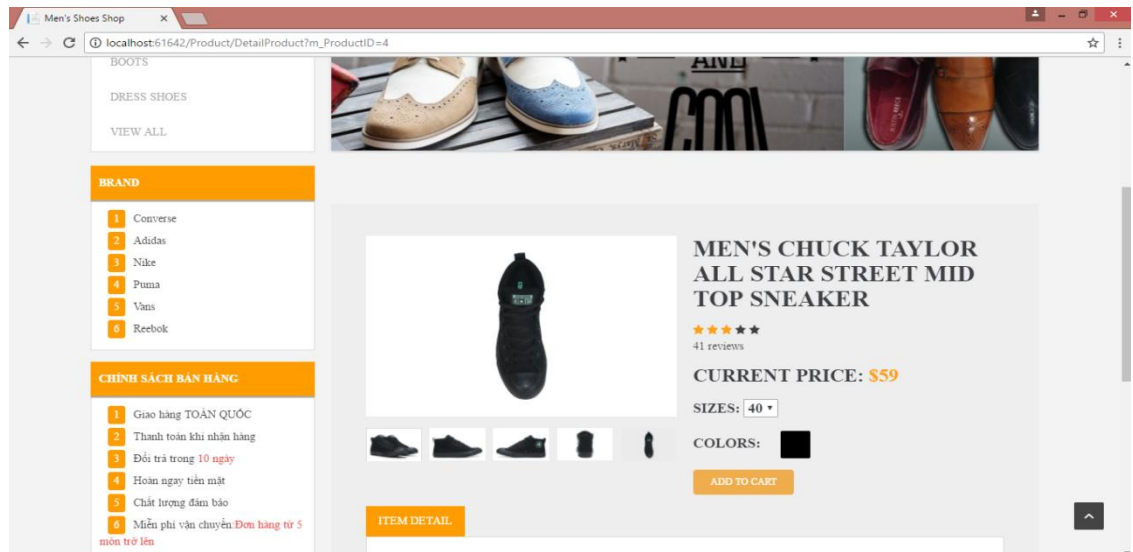
Giao diện chung khi truy cập vào website.



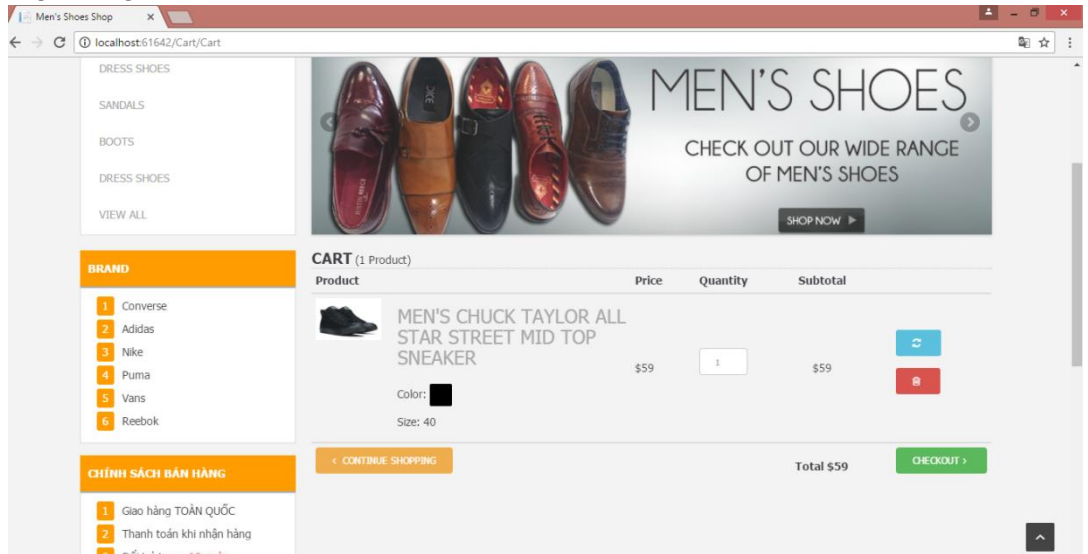
- Giao diện đăng nhập, đăng ký của người dùng.
- Giao



diện chi tiết sản phẩm.



- Giao diện giỏ hàng.



OVERVIEW report overview...

Orders Top Selling Most Viewed New Customers

Customer Name	Date	Amount	Status	
Minh Thảo	5/17/2017 9:12:12 PM	\$ 128	Pending	View
Minh Thảo	5/17/2017 9:12:51 PM	\$ 59	In Process	View
Võ Đại Thắng	5/17/2017 10:41:34 PM	\$ 128	Pending	View
Võ Đại Thắng	5/27/2017 11:12:19 PM	\$ 69	In Process	View
Võ Xuân Cường	6/7/2017 7:20:17 PM	\$ 207	Pending	View
Võ Xuân Thịnh	6/10/2017 2:41:22 AM	\$ 110	Pending	View
Võ Xuân Thịnh	6/10/2017 2:45:40 AM	\$ 55	Pending	View
Võ Xuân Thịnh	6/10/2017 4:54:31 AM	\$ 60	Pending	View

- Giao diện admin.
- Admin thêm sản phẩm

CREATE NEW PRODUCT

Product Name *

Price *

Quantity *

Color & Size: *

Color

Size

39 40 41 42 43 44 45 46 47

CHƯƠNG 3: KẾT LUẬN

1 Kết quả đạt được.

- Hiểu được một số giải pháp xây dựng website thương mại điện tử.
- Xây dựng được một website bán hàng có đầy đủ các chức năng cơ bản đối với khách hàng.
- Hiểu được một số phương pháp thanh toán trực tuyến.
- Ngoài những kết quả đạt được trên thì vẫn còn một số yêu cầu chưa nắm rõ và website vẫn còn một số khuyết điểm cần khắc phục

1.1 Ưu điểm.

- Giới thiệu sản phẩm đến khách hàng. Đối với khách hàng: cho phép thực hiện các chức năng xem hàng hóa.
- Sau khi mua sản phẩm, đặt mua sản phẩm và thanh toán bằng nhiều cách khác nhau.
- Khách hàng có thể thay đổi mật khẩu và mật khẩu thanh toán trong thông tin cá nhân. Khách hàng có thể xem thông tin đơn hàng của mình.
- Hoàn thành hầu hết các chức năng cơ bản của một trang web bán hàng.
- Xây dựng trên mô hình 3 layer dễ dàng quản lý việc thay đổi cấu trúc sau này.

1.2 Nhược điểm.

- Về dữ liệu.
 - + Dữ liệu chưa được đầy đủ.
 - + Dữ liệu trên trang web hiện tại không đúng với thực tế, sử dụng nguồn dữ liệu ảo.
- Về giao diện
 - + Giao diện chưa sinh động
 - + Giao diện cứng, không thể thay đổi template cho giao diện.

2. Hướng phát triển của hệ thống (bao gồm phát triển ứng dụng thông minh)

Với những thiếu sót như phân tích trên, để phần mềm ngày càng hoàn thiện, phục vụ tốt cho nhu cầu của người dùng thì nên có những cập nhật sau đây:

Thiết kế trang web đẹp hơn, tiện dụng hơn, có nhiều chức năng và sinh động hơn. Xây dựng thêm một diễn đàn trên trang web để khách hàng có thể trao đổi bình luận về sản phẩm. Thiết kế thêm vị trí để cập nhật giá cả thị trường. Nâng cao khả năng bảo mật cho chương trình. Ngăn ngừa tối đa các trường hợp đột nhập dữ liệu hệ thống. Chính sửa dữ liệu cho chính xác với thực tế và sẽ nâng cấp dần website để đáp ứng được những nhu cầu của khách hàng trong thời đại mới.

Để phát triển một ứng dụng thông minh đề xuất giày, cần thực hiện các công việc sau:

1. Thu thập dữ liệu: Thu thập dữ liệu về các sản phẩm giày, bao gồm thông tin về màu sắc, kiểu dáng, kích cỡ, giá cả và các đặc tính khác.
2. Phân tích dữ liệu: Sử dụng các công cụ phân tích dữ liệu để phân tích các thông tin thu thập được và tìm ra các mối quan hệ giữa chúng.
3. Xây dựng mô hình: Dựa trên kết quả phân tích, xây dựng một mô hình học máy để đưa ra các đề xuất sản phẩm giày phù hợp với khách hàng.
4. Tích hợp vào trang web: Tích hợp mô hình vào trang web để đưa ra các đề xuất sản phẩm giày cho khách hàng.
5. Kiểm tra và cải tiến: Kiểm tra và cải tiến mô hình để đảm bảo rằng nó hoạt động tốt và đưa ra các đề xuất sản phẩm giày chính xác.

Trong bước số 4 là tích hợp vào trang web cần làm các công việc sau:

1. Xuất mô hình: Đầu tiên, cần xuất mô hình đã được huấn luyện của dưới dạng tệp tin có thể được tải lên trang web. Định dạng phổ biến cho việc xuất mô hình là ONNX (Open Neural Network Exchange) hoặc TensorFlow SavedModel, một vài định dạng khác: PyTorch, Scikit-learn, Keras..
2. Tải lên mô hình: Tiếp theo, cần tải lên tệp tin mô hình đã xuất lên máy chủ của trang web bằng cách sử dụng các công cụ quản lý tệp tin như FTP (File Transfer Protocol) hoặc SSH (Secure Shell)
3. Nhúng mô hình vào trang web: Sau khi tải lên, có thể nhúng mô hình vào trang web bằng cách sử dụng mã HTML và JavaScript. Sử dụng thẻ <script> để tải và chạy mô hình trong trình duyệt của khách hàng.
4. Gọi mô hình từ trang web: Khi mô hình đã được nhúng vào trang web, có thể gọi nó từ các sự kiện hoặc hành động trong trang web. Ví dụ: khi người dùng nhấp vào nút "Đề xuất giày" thì có thể gọi mô hình để đưa ra các đề xuất sản phẩm giày phù hợp.
5. Kiểm tra và cải tiến: Cuối cùng là kiểm tra và cải tiến tích hợp của mô hình trong trang web để đảm bảo rằng nó hoạt động chính xác và đáp ứng yêu cầu của khách hàng.

3. Phân chia công việc:

Nguyễn Phan Nhựt Trường: Tìm hiểu lý thuyết, xây dựng hệ thống, database, code trang web

Bùi Tuấn Anh: xây dựng mô hình học máy

CHƯƠNG 4: HỆ THỐNG THÔNG MINH

Giới thiệu về hệ thống gợi ý (Recommender systems hoặc Recommendation systems)

Tổng quan về hệ thống gợi ý

Hệ thống gợi ý (Recommender systems hoặc Recommendation systems) là một dạng của hệ hỗ trợ ra quyết định, cung cấp giải pháp mang tính cá nhân hóa mà không phải trải qua quá trình tìm kiếm phức tạp. Hệ gợi ý học từ người dùng và gợi ý các sản phẩm tốt nhất trong số các sản phẩm phù hợp.

Hệ thống gợi ý sử dụng các tri thức về sản phẩm, các tri thức của chuyên gia hay tri thức khai phá học được từ hành vi con người dùng để đưa ra các gợi ý về sản phẩm mà họ thích trong hàng ngàn hàng vạn sản phẩm có trong hệ thống. Các website thương mại điện tử, ví dụ như sách, phim, nhạc, báo...sử dụng hệ thống gợi ý để cung cấp các thông tin giúp cho người sử dụng quyết định sẽ lựa chọn sản phẩm nào. Các sản phẩm được gợi ý dựa trên số lượng sản phẩm đó đã được bán, dựa trên các thông tin cá nhân của người sử dụng, dựa trên sự phân tích hành vi mua hàng trước đó của người sử dụng để đưa ra các dự đoán về hành vi mua hàng trong tương lai của chính khách hàng đó. Các dạng gợi ý bao gồm: gợi ý các sản phẩm tới người tiêu dùng, các thông tin sản phẩm mang tính cá nhân hóa, tổng kết các ý kiến cộng đồng, và cung cấp các chia sẻ, các phê bình, đánh giá mang tính cộng đồng liên quan tới yêu cầu, mục đích của người sử dụng đó.

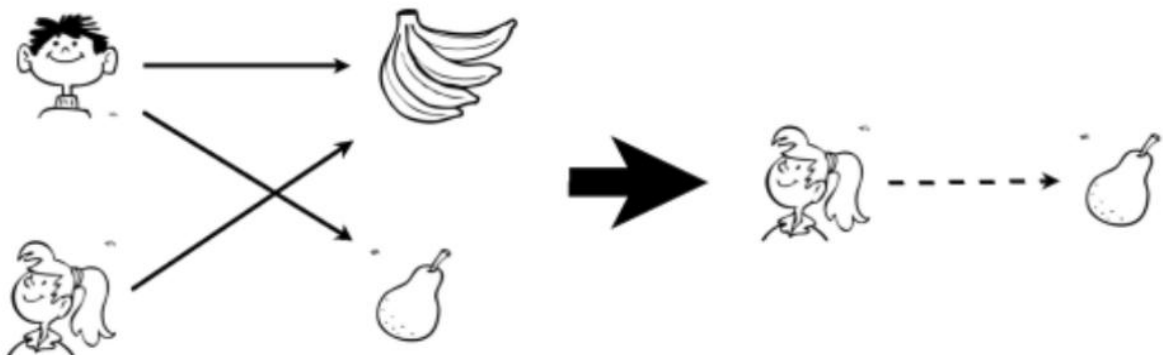
Phương pháp gợi ý	Dữ liệu cơ sở	Dữ liệu đầu ra	Tiến trình xử lý
Dựa theo lọc cộng tác	Các điểm số đánh giá của những người sử dụng trong U đối với các đối tượng trong I.	Các điểm số đánh giá của u cho các đối tượng trong I.	Nhận ra người sử dụng trong U tương tự với u (về sở thích) và sau đó ngoại suy điểm số đánh giá của u cho i.
Dựa theo nội dung	Các đặc điểm của các đối tượng trong I.	Các điểm số đánh giá của u cho các đối tượng trong I.	Tạo ra một mô hình mô tả sở thích của người sử dụng u, sau đó sử dụng để đánh giá mức độ

Phương pháp gợi ý	Dữ liệu cơ sở	Dữ liệu đầu ra	Tiến trình xử lý
			ưa thích của u với i.
Dựa trên cơ sở tri thức	Các đặc điểm của các đối tượng trong I. Các tri thức (hiểu biết) về sự phù hợp giữa các đối tượng với nhu cầu của người sử dụng.	Một sự mô tả nhu cầu và sở thích của người sử dụng u.	Suy luận sự phù hợp giữa I và nhu cầu của u.

1. Hệ thống gợi ý dựa theo lọc cộng tác

- Hệ thống gợi ý dựa theo lọc cộng tác (Collaborative recommendation systems): là phương pháp gợi ý được triển khai rộng rãi nhất và thành công nhất trong thực tế.
- Hệ thống theo lọc cộng tác phân tích và tổng hợp các điểm số đánh giá của các đối tượng, nhận ra sự tương đồng giữa những người sử dụng trên cơ sở các điểm số đánh giá của họ và tạo ra các gợi ý dựa trên sự so sánh này. Hồ sơ (profile) của người sử dụng điển hình trong hệ thống lọc cộng tác bao gồm một vector các đối tượng (item) và các điểm số đánh giá của chúng, với số chiều tăng lên liên tục khi người sử dụng tương tác với hệ thống theo thời gian.
- Một số hệ thống sử dụng phương pháp chiết khấu dựa trên thời gian (time-based discounting) để tính toán cho yếu tố "trượt" đối với sự quan tâm của người sử dụng. Trong một số trường hợp điểm số đánh giá (rating) có thể là nhị phân (thích/không thích) hoặc các giá trị số thực cho thấy mức độ ưu tiên.
- Thế mạnh lớn nhất của kỹ thuật gợi ý theo lọc cộng tác là chúng hoàn toàn độc lập với sự biểu diễn của các đối tượng đang được gợi ý, và do đó có thể làm việc tốt với các đối tượng phức tạp như âm thanh và phim. Schafer, Konstan & Riedl (1999) gọi lọc cộng tác là "tương quan giữa người – với – người" (people-to-people correlation).

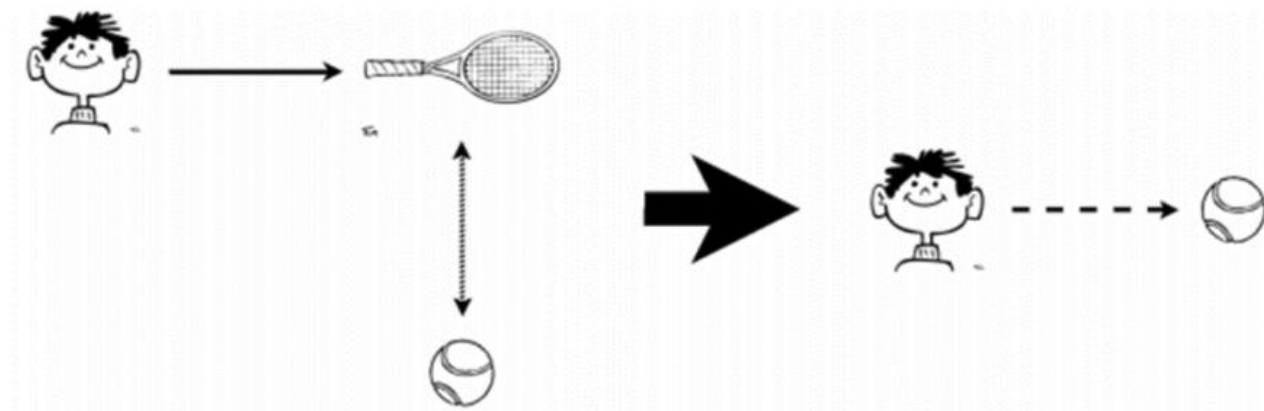
Minh họa:



2. Hệ thống gợi ý dựa theo nội dung

- Hệ thống gợi ý dựa theo nội dung (Content-based recommendation systems): là sự kế thừa và mở rộng của lĩnh vực nghiên cứu lọc thông tin.
- Trong hệ thống thì các đối tượng được biểu diễn bởi các đặc điểm liên quan tới chúng.
- Ví dụ, hệ thống gợi ý văn bản như hệ thống lọc tin NewsWeeder sử dụng những từ của các văn bản như các đặc điểm.
- Một số hệ thống gợi ý dựa trên nội dung học một hồ sơ cá nhân về sở thích của người sử dụng dựa trên các đặc điểm xuất hiện trong chính các đối tượng người sử dụng đã đánh giá (rated). Schafer, Konstan & Riedl gọi gợi ý theo nội dung là "tương quan đối tượng với đối tượng" (item-to-item correlation). Hồ sơ người sử dụng của một hệ thống gợi ý theo nội dung phụ thuộc vào phương pháp học máy được dùng.
- Cây quyết định (Decision trees), mạng neuron (neural nets) và biểu diễn dựa theo vector (vector-based representations) đều có thể được sử dụng để học hồ sơ người dùng. Cũng giống như trong lọc cộng tác, hồ sơ người dùng trong gợi ý dựa theo nội dung là những dữ liệu lâu dài và được cập nhật theo thời gian.

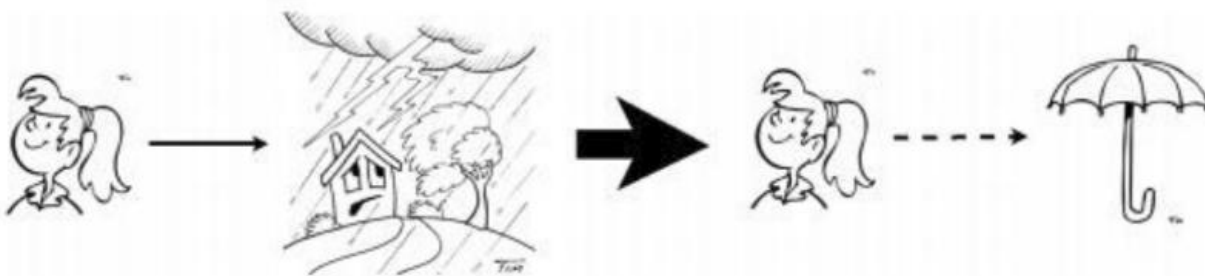
Minh họa:



3. Hệ thống gợi ý dựa trên cơ sở tri thức

- Hệ thống gợi ý dựa trên cơ sở tri thức (Knowledge-based recommenders systems): gợi ý các đối tượng dựa trên các suy luận về nhu cầu và sở thích của người dùng. Theo một nghĩa nào đó, tất cả các kỹ thuật gợi ý có thể mô tả như là làm một số suy luận. Phương pháp tiếp cận dựa trên cơ sở tri thức được phân biệt ở chỗ: chúng có kiến thức làm thế nào một đối tượng cụ thể đáp ứng nhu cầu một người dùng cụ thể, và do đó có thể lập luận về mối quan hệ giữa nhu cầu và các gợi ý cụ thể.

- Sử dụng miền tri thức rõ ràng, có liên quan tới mối quan hệ giữa yêu cầu của người dùng và sản phẩm cụ thể. Ban đầu người ta đưa ra 3 dạng tri thức: tri thức về danh mục (tri thức về sản phẩm được gợi ý), tri thức người sử dụng (tri thức về các yêu cầu của người sử dụng), tri thức về các chức năng (tri thức để ánh xạ các yêu cầu của người sử dụng tới các sản phẩm thoả mãn các yêu cầu đó).
- Phương pháp này không dựa trên tiêu sử người sử dụng nên không gặp phải khó khăn về sản phẩm mới và người dùng mới. Gợi ý trên cơ sở tri thức có khả năng suy diễn, khả năng suy diễn phụ thuộc vào độ phù hợp của yêu cầu người sử dụng với các thuộc tính của sản phẩm.
- Mọi hệ thống dựa trên cơ sở tri thức đều là môi quan hệ thu nhận tri thức. Thực tế, chất lượng của các phương án gợi ý tùy thuộc vào độ chính xác của cơ sở tri thức. Đây cũng là hạn chế lớn nhất của phương pháp này.
- Minh họa:



So sánh các phương pháp gợi ý

1. Hệ thống gợi ý dựa theo lọc cộng tác

- Ưu điểm:
 - Khả năng đa hạng mục
 - Không cần tri thức miền
 - Chất lượng tăng theo thời gian
 - Đủ thông tin phản hồi không tương minh
- Nhược điểm:
 - Vấn đề người dùng mới
 - Vấn đề sản phẩm/đối tượng mới

- Vấn đề "Gray sheep"
- Chất lượng phụ thuộc vào độ lớn dữ liệu lịch sử thao tác của người sử dụng
- Vấn đề về tính bền vững và mềm dẻo

2. Hệ thống gợi ý dựa theo nội dung

- Ưu điểm:
 - Không cần tri thức miền
 - Chất lượng tăng theo thời gian
 - Đủ thông tin phản hồi không tương minh
- Nhược điểm:
 - Vấn đề người dùng mới
 - Chất lượng phụ thuộc vào độ lớn dữ liệu lịch sử thao tác của người sử dụng
 - Vấn đề về tính bền vững và mềm dẻo

3. Hệ thống gợi ý dựa trên cơ sở tri thức

- Ưu điểm:
 - Có thể ánh xạ giữa nhu cầu người dùng và sản phẩm/đối tượng
- Nhược điểm:
 - Cần phải thu thập tri thức.

Xây dựng hệ thống gợi ý Item - item

Nếu bạn từng sử dụng Netflix, bạn sẽ thấy có một phần có tiêu đề là "Bởi vì bạn đã xem bộ phim X", đó là nơi cung cấp các gợi ý cho các bộ phim dựa trên bộ phim gần đây mà bạn đã xem. Đây là một ví dụ điển hình về gợi ý Item-item.

Trong phần này, chúng tôi sẽ tạo ra các gợi ý Item-item bằng cách sử dụng một kỹ thuật gọi là collaborative filtering.

Step 1: Nhập các thư viện

Chúng ta sẽ biểu diễn dữ liệu dưới dạng pandas [DataFrame](#).

DataFrame là gì?

- Một cấu trúc dữ liệu hai chiều của Pandas
- Các cột biểu thị các đặc điểm, các hàng biểu thị các mục
- Tương tự như một bảng tính Excel hoặc bảng SQL

Chúng ta cũng sẽ sử dụng hai gói vẽ đồ thị: matplotlib và seaborn (đây là một gói bọc của matplotlib) để trực quan hóa dữ liệu.

```
In [2]: import numpy as np
import pandas as pd
import sklearn
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
```

Step 2: Tải dữ liệu lên

```
[13]: ratings = pd.read_csv("/kaggle/input/hhhhhhh/ratings.csv")
ratings.head()
```

```
[14]: shoes = pd.read_csv("/kaggle/input/hhhhhhh/shoe.csv")
shoes.head()
```

Step 3: Phân tích các dữ liệu khám phá

Hãy xem có bao nhiêu đánh giá, đôi giày duy nhất và người dùng duy nhất trong tập dữ liệu của chúng ta.

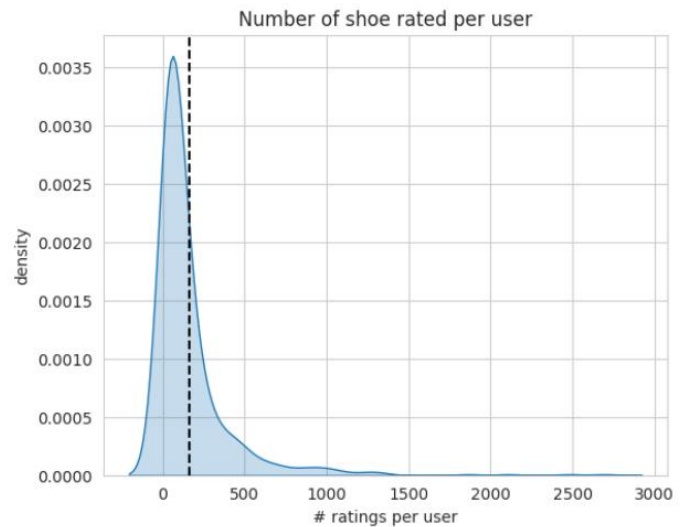
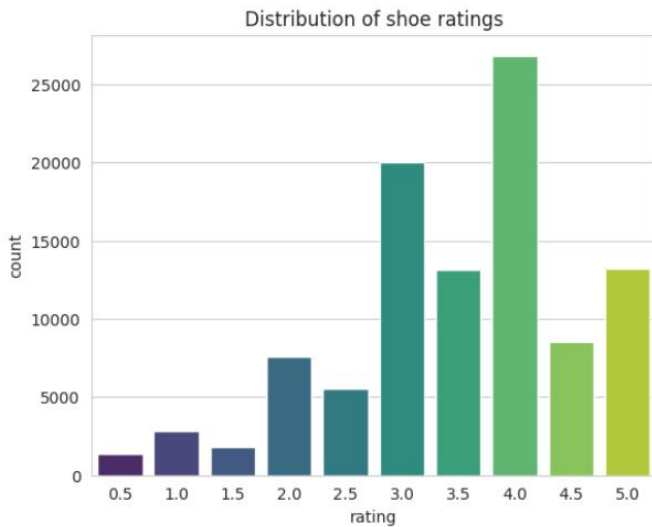
```

n_ratings = len(ratings)
n_shoes = ratings['movieId'].nunique()
n_users = ratings['userId'].nunique()

print(f"Number of ratings: {n_ratings}")
print(f"Number of unique shoeId's: {n_shoes}")
print(f"Number of unique users: {n_users}")
print(f"Average number of ratings per user: {round(n_ratings/n_users, 2)}")
print(f"Average number of ratings per shoe: {round(n_ratings/n_shoes, 2)}")

```

Number of ratings: 100836
 Number of unique shoeId's: 9724
 Number of unique users: 610
 Average number of ratings per user: 165.3
 Average number of ratings per shoe: 10.37



Tìm đôi giày có điểm đánh giá thấp nhất và cao nhất

```

[21]: mean_rating = ratings.groupby('movieId')['rating'].mean()

lowest Rated = mean_rating['rating'].idxmin()
movies.loc[movies['movieId'] == lowest_Rated]

```

```

[21]:
movieId  title  fullPrice
2689    3604  Air Jordan 1 Mid    125

```

```

[22]: highest_Rated = mean_rating['rating'].idxmax()
movies.loc[movies['movieId'] == highest_Rated]

```

```

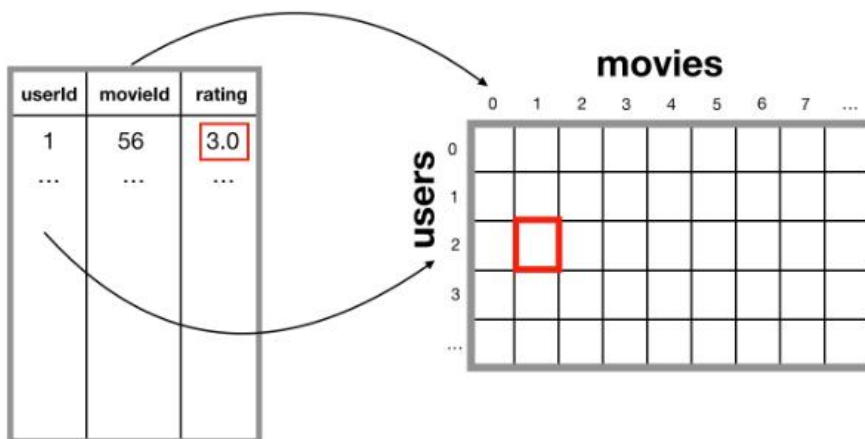
[22]:
movieId  title  fullPrice
48       53  Air Jordan 1 Low    110

```

Step 4: Biến đổi dữ liệu

Chúng ta sẽ sử dụng một kỹ thuật gọi là collaborative để tạo ra các gợi ý cho người dùng. Kỹ thuật này dựa trên giả định về "tính đồng thuận" - những người dùng tương tự thích những thứ tương tự. Collaborative là một loại học không giám sát giúp dự đoán về sở thích của người dùng dựa trên sự học từ sở thích của một tập đông lớn hơn.

Bước đầu tiên của collaborative là biến đổi dữ liệu thành một ma trận người dùng-vật phẩm - còn được gọi là ma trận "tiện ích". Trong ma trận này, các hàng biểu thị người dùng và các cột biểu thị các vật phẩm. Ưu điểm của lọc cộng tác là nó không yêu cầu bất kỳ thông tin nào về người dùng hoặc vật phẩm để tạo ra các gợi ý.



Hàm `create_X()` tạo ra ma trận thưa X với bốn từ điển ánh xạ:

`user_mapper`: ánh xạ từ id người dùng sang chỉ mục người dùng

`movie_mapper`: ánh xạ từ id phim sang chỉ mục giày

`user_inv_mapper`: ánh xạ từ chỉ mục người dùng sang id người dùng

`movie_inv_mapper`: ánh xạ từ chỉ mục phim sang id giày

Chúng ta cần những từ điển này vì chúng ánh xạ hàng và cột tương ứng của ma trận tiện ích với ID người dùng và ID giày tương ứng.

Step 5: Tìm các đôi giày tương tự bằng cách sử dụng k-Nearest Neighbours

Phương pháp này tìm kiếm hàng xóm gần nhất của một bộ phim cụ thể bằng cách xác định các điểm trong tập dữ liệu gần nhất với bộ phim. kNN sử dụng các độ đo khoảng cách như:

Tương đồng Cosine

Khoảng cách Euclidean

Khoảng cách Manhattan

Tương quan Pearson

Mặc dù khó để hình dung, chúng tôi đang làm việc trong không gian M-chiều, trong đó M đại diện cho số lượng bộ phim trong ma trận X của chúng tôi.

