**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

**LỜI CẢM ƠN**

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

**\*\*\*\*\*\***

Lời đầu tiên, em xin được gửi lời cảm ơn chân thành tới trường Đại học Trà Vinh đã tạo điều kiện học tập thuận lợi cho em cũng như toàn thể lớp DK23TTC10 khoa Công Nghệ Thông Tin trong suốt năm học vừa qua. Đặc biệt cảm ơn các thầy và các cô đã truyền đạt những kiến thức cơ bản cần thiết để em có thể thực hiện tốt phần đồ án này cũng như đủ tự tin cho công việc hiện tại.

Do tính chất công việc đang làm chưa được tiếp xúc sâu hơn về quy trình xây dựng một trang web nên quá trình làm đồ án này đã giúp em có cơ hội được tiếp xúc với quy trình xây dựng lên một trang web hoàn chỉnh trên cơ sở những kiến thức đã được học và sự hướng dẫn của giáo viên phụ trách,cũng như các thông qua các bài học của thầy giảng dạy.

Em xin được gửi lời cảm ơn chân thành đến thầy cô và các bạn trong lớp đã nhiệt tình hỗ trợ em trong quá trình thực hiện đề tài này.Vì thời gian và trình độ của em còn hạn chế nên chắc chắn em không tránh khỏi những thiếu sót. Rất mong được nhận những góp ý của thầy để đề tài được hoàn thiện hơn.

*Bình Dương, ngày 20 tháng 02 năm 2025*

Sinh viên thực hiện

Nguyễn Tuyết Yến Nhi

**MỤC LỤC**

**\*\*\*\*\*\***

**NỘI DUNG**

MỞ ĐẦU 1

[CHƯƠNG 1: NGHIÊN CỨU LÝ THUYẾT 2](#_Toc182133894)

[1.1 Tổng quan về ASP.NET 2](#_Toc182133895)

[1.2 Tổng quan về ASP.NET MVC 2](#_Toc182133896)

[1.2.1 Kiến trúc của ứng dụng web ASP.NET MVC 3](#_Toc182133897)

[1.2.2 Các tính năng của nền tảng ASP.NET MVC 5](#_Toc182133898)

[1.2.3 Lợi ích của ứng dụng web dựa trên mô hình MVC 6](#_Toc182133899)

[1.3 Ngôn ngữ CSS 7](#_Toc182133900)

[1.4 Ngôn ngữ Javascript 8](#_Toc182133901)

[1.5 Ngôn ngữ HTML 8](#_Toc182133902)

[CHƯƠNG 2: THỰC HIỆN HÓA NGHIÊN CỨU 21](#_Toc182133930)

[2.1 Mô tả bài toán 21](#_Toc182133931)

[2.2 Mô tả chức năng 21](#_Toc182133932)

[2.3 Sơ đồ Use Case 22](#_Toc182133933)

[2.3.1 UseCase quản trị (Admin) 22](#_Toc182133934)

[2.3.2 UseCase Nhân viên 23](#_Toc182133935)

[2.3.3 UseCase khách hàng 23](#_Toc182133936)

[2.4 Sơ đồ DFD 24](#_Toc182133937)

[2.4.1 Mô hình luồng dữ liệu mức khung cảnh (DFD mức 0) 24](#_Toc182133938)

[2.4.2 Mô hình luồng dữ liệu mức đỉnh (DFD mức 1) 24](#_Toc182133939)

[2.5 Thiết kế, xây dựng dữ liệu 25](#_Toc182133940)

[2.5.1 Mô hình cơ sở dữ liệu 25](#_Toc182133941)

[2.5.2 Mô tả chi tiết các bảng 25](#_Toc182133942)

[CHƯƠNG 3: KẾT QUẢ THỰC NGHIỆM 29](#_Toc182133954)

[CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 41](#_Toc182133983)

[4.1 Kết Luận 41](#_Toc182133984)

[4.1.1 Ưu điểm 41](#_Toc182133985)

[4.1.2 Nhược điểm (hạn chế) 41](#_Toc182133986)

[4.2 Hướng phát triển 41](#_Toc182133987)

[DANH MỤC TÀI LIỆU THAM KHẢO 43](#_Toc182133988)

**DANH MỤC HÌNH ẢNH – BẢNG BIỂU**

**\*\*\*\*\*\***

[Hình 1. 1. Mô hình MVC 3](#_Toc182136752)

[Hình 1. 2. Các bước xử lý request của Asp.Net MVC 4](#_Toc182136753)

[Hình 1. 3. Cú pháp cơ bản của jQuery 17](#_Toc182136754)

[Hình 1. 4. Các thành phần chính trong SQL Server 18](file:///D:\be%20ai\bao%20cao%20asp\170123111_BaoCao.docx#_Toc182136755)

Hình 2. 1. UseCase Admin 22

Hình 2. 2. UseCase Nhân viên 23

Hình 2. 3. UseCase Khách hàng 23

Hình 2. 4. Sơ đồ DFD mức 0 24

Hình 2. 5. Sơ đồ DFD mức 1 24

Hình 2. 6. Mô hình CSDL 25

Hình 2. 7. Bảng Chi tiết đơn hàng 25

Hình 2. 8. Bảng Danh mục 25

Hình 2. 9. Bảng Đơn hàng 26

Hình 2. 10. Bảng Giỏ hàng 26

Hình 2. 11. Bảng Kích thước 26

Hình 2. 12. Bảng Mã giảm giá 26

Hình 2. 13. Bảng Sản phẩm 27

Hình 2. 14. Bảng Tài khoản 27

Hình 2. 15. Bảng Tin tức 27

Hình 2. 16. Bảng Topping đơn hàng 28

Hình 2. 17. Bảng Topping 28

**TÓM TẮT**

**\*\*\*\*\*\***

1. **Nghiên cứu của đề tài**

Xây dựng một website bán hàng với mặt hàng cụ thể là trà sữa bằng ngôn ngữ sử dụng chính là C#, xây dựng trên công nghệ ASP.Net MVC5 của Microsoft

1. **Mục tiêu của đề tài**

Xây dựng website bán trà sữa.

1. **Cách giải quyết vấn đề**

Phân tích chức năng chung của các trang web bán cùng một mặt hàng để đưa ra phương hướng thiết kế, xây dựng trang web đúng với mục đích nghiên cứu ban đầu

1. **Các kết quả đạt được**

* Xây dựng một trang admin để quản lý, đăng các bài viết liên quan đến sản phầm cũng như quản lý doanh thu, đơn đặt hàng, số lượng thành viên và số lượng sản phẩm tại cửa hàng.
* Xây dựng trang hiển thị danh sách sản phẩm, chi tiết sản phẩm, giá cả và liên kết đến trang đặt hàng
* Xây dựng được tính năng giỏ hàng và thanh toán.
* Xây dựng được cơ sở dữ liệu riêng cho trang web

**MỞ ĐẦU**

**\*\*\*\*\*\***

1. **Lý do chọn đề tài**

Trong những năm gần đây, với sự phát triển mạnh mẽ của công nghệ trong lĩnh vực thiết bị di động và Internet đã kéo theo việc ứng dụng CNTT vào việc phát triển website và app di động cho nhu cầu mua sắm và ăn uống cũng phát triển mạnh mẽ

Sau đại dịch Covid-19, thói quen sinh hoạt và tiêu thụ thực phẩm của mọi người cũng đã có sự biến đổi mạnh mẽ, điển hình là thói quen mua sắm trực tuyến ngày càng phát triển. Đối với đồ uống, người tiêu dùng thích có xu hướng tìm kiếm các loại đồ uống nhẹ như cà phê, trà... vì đây được cho là những loại đồ uống giải tỏa căng thẳng và dễ tiêu thụ. Tại Việt Nam, trà sữa đang đứng thứ 2 về lượng người ưa chuộng sử dụng, nắm bắt được nhu cầu đó nên em đã lựa chọn đề tài **“Xây dựng website bán trà sữa”** và xây dựng nó bằng công nghệ ASP.NET MVC5 của Microsoft, việc xây dựng trang web này không chỉ giúp em cập nhật được thông tin của nhiều loại trà sữa khác nhau mà còn có thể giúp khách hàng không cần tốn quá nhiều thời gian để đến tận nơi mua hàng mà vẫn có thể thưởng thức được chúng thông qua chức năng đặt hàng trên trang web.

1. **Mục đích**

Tìm hiểu về ngôn ngữ, công cụ phục vụ thiết kế và lập trình Web. Xây dựng được trang web bán trà sữa với các sản phẩm đa dạng và giao diện thân thiện với người dùng.

1. **Đối tượng nghiên cứu**

Ngôn ngữ lập trình C# và ASP.Net, mô hình ASP.Net MVC5, hệ quản trị CSDL SQL Server, các công nghệ hỗ trợ như JavaScript, CSS3, HTML5, các kiến thức về xây dựng website quản lý, website thương mại…

1. **Phạm vi nghiên cứu**

Tìm hiểu công nghệ lập trình web bằng ASP.Net MVC, xây dựng hệ quản trị CSDL SQL Server cho trang web, xây dựng website bán sản phẩm và quản lý

# 

# CHƯƠNG 1: NGHIÊN CỨU LÝ THUYẾT

\*\*\*\*\*\*

## Tổng quan về ASP.NET

ASP.Net là một dạng framework, nói một cách dễ hiểu chính là một bộ khung ứng dụng web. ASP .Net được hoàn thiện bởi Microsoft vào 2012 và là một giải pháp dùng để hỗ trợ quá trình sản xuất các loại website và nội dung động. Nguồn gốc của ASP net được bắt nguồn từ ASP – Active Server Pages. Đây là một giải pháp đến từ Microsoft dùng để sản xuất nội dung cho các trang web. Mỗi khi nói về ASP, ngành công nghệ thông tin cho rằng nó đã có mặt từ những năm 1990. Đi cùng với ASP chính là NT server và các hệ điều hành máy tính để bàn Window 95, 98. Đây chính là khởi nguồn cho ASP net sau này.

Sự ra đời của ASP.Net cho phép các nhà phát triển công nghệ, lập trình viên dễ dàng hơn trong quá trình xây dựng các trang web, ứng dụng hay các dịch vụ có sử dụng nội dung động. ASP.Net đã trải qua nhiều thế hệ phát triển với 2 phiên bản chủ yếu:

Thế hệ 1.0 là sự kết hợp của .NET framework và một phiên bản kế thừa của công nghệ ASP được phát triển bởi Microsoft

Thế hệ tiếp theo – ASP.NET có khả năng hỗ trợ nhiều ngôn ngữ lập trình được biên dịch, điển hình như C++, C#, Visual Basic và Perl.

## Tổng quan về ASP.NET MVC

Mô hình MVC (Model–View–Controller). ASP.NET MVC là 1 phần mềm mã mở, tách rời với thành phần độc quyền ASP.NET Web Forms. Nền tảng MVC bao gồm các thành phần dưới đây:

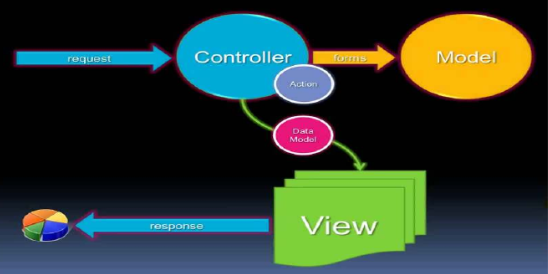
**Model**: Các đối tượng Model là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng Model lấy và lưu trạng thái của model trong CSDL.

**View**: là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, View được tạo dựa vào thông tin dữ liệu model.

**Controller**: là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng.

Một ứng dụng web được phát triển theo mô hình MVC, có nguyên lý hoạt động xử lý yêu cầu của người dùng như sau:

* Khi có yêu cầu phát sinh từ người dùng (phía Client), yêu cầu này sẽ được gởi đến phía Server, tại đây Controller sẽ tiếp nhận yêu cầu để xử lý.
* Bên trong Controller chứa nhiều phương thức (action), mỗi action tương ứng với 1 yêu cầu cụ thể từ người dùng. Controller sẽ lựa chọn action phù hợp với yêu cầu để xử lý. Trong quá trình xử lý, Controller tương tác với Model để có được dữ liệu mà người dùng mong muốn.
* Sau khi Controller làm việc với Model để có được dữ liệu theo yêu cầu, Controller sẽ gởi về cho View thành phần Data Model, và nhiệm vụ của View sẽ chuyển Data Model nhận được thành dữ liệu ứng dụng và gởi trả về phía Client để hiển thị kết quả yêu cầu.

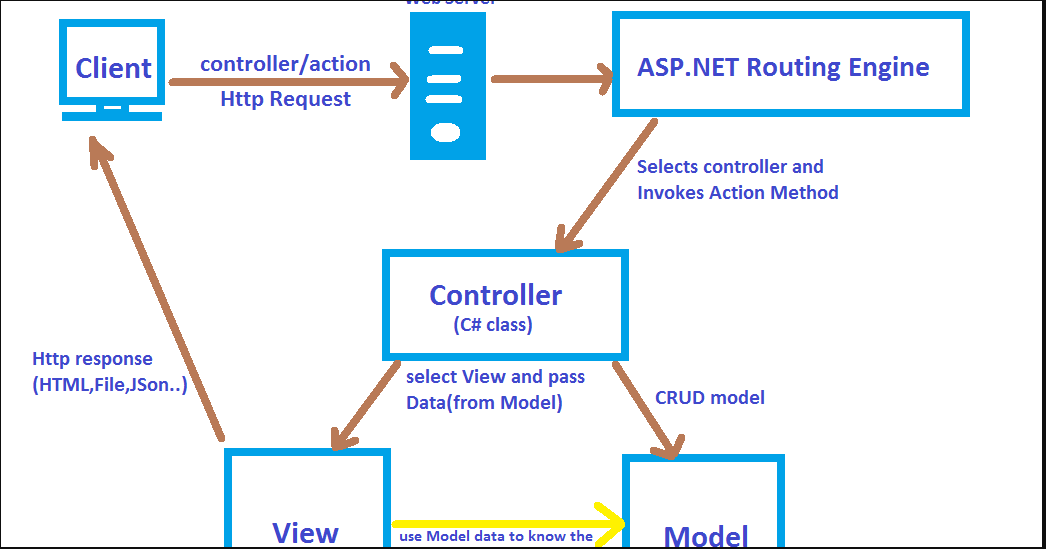


Hình 1. 1. Mô hình MVC

### Kiến trúc của ứng dụng web ASP.NET MVC

Kiến trúc của ứng dụng web ASP.NET MVC bao gồm các thành phần chính sau:

* **MVC Framework**: Đây là một kiến trúc phần mềm được sử dụng để xây dựng ứng dụng web. Nó phân chia ứng dụng thành 3 thành phần: Model, View và Controller.
* **Route engine**: Route engine là thành phần xử lý định tuyến trong ASP.NET MVC. Nó giúp định tuyến các yêu cầu HTTP đến các controller tương ứng để xử lý.
* **Route configuration**: Route configuration là tập hợp các quy tắc định tuyến được cấu hình trong ứng dụng. Nó xác định các đường dẫn URL mà ứng dụng có thể xử lý.
* **Controller**: Controller là thành phần xử lý yêu cầu HTTP. Nó nhận yêu cầu từ Route engine, sau đó xử lý và trả về một Model hoặc một View.
* **Model**: Model là thành phần lưu trữ dữ liệu của ứng dụng. Nó có thể là một lớp đối tượng, một tệp XML hoặc một cơ sở dữ liệu.
* **View engine**: View engine là thành phần hiển thị dữ liệu trên trình duyệt. Nó đọc các tệp View và dữ liệu Model được truyền từ Controller để tạo nội dung HTML cho trang web.
* **View**: View là thành phần hiển thị dữ liệu được trả về từ Controller. Nó có thể là một tệp HTML hoặc một tệp Razor.



Hình 1. 2. Các bước xử lý request của Asp.Net MVC

* **Routing**: ASP.NET MVC sử dụng Route engine để phân tích định tuyến yêu cầu (request) dựa trên các Route configuration (cấu hình đường dẫn). Các đường dẫn sẽ được so khớp với các Route configuration để xác định Controller và Action tương ứng.
* **Controller**: Sau khi xác định được Controller tương ứng, hệ thống sẽ tạo mới một instance của Controller đó và gọi tới Action tương ứng để xử lý yêu cầu.
* **Model Binding**: Khi Controller được gọi, ASP.NET MVC sử dụng Model Binding để lấy dữ liệu được gửi lên bởi client (thông qua request) và đưa vào các Model object để sử dụng cho việc xử lý.
* **Action Execution**: Sau khi các Model object đã được tạo và điền đầy đủ dữ liệu, hệ thống sẽ gọi đến Action tương ứng để thực thi các xử lý kết quả.
* **Result Execution**: Sau khi Action được thực thi, hệ thống sẽ trả về một Result object để hiển thị kết quả trên View. Result object có thể là các loại kết quả khác nhau như View, Redirect, File, Content, JSON, ...
* **View Rendering**: Khi Result là kiểu View, ASP.NET MVC sẽ sử dụng View Engine để tìm và render View tương ứng với Action. View Engine sẽ tìm kiếm các file View trong thư mục Views theo cấu trúc tương ứng với tên Controller và tên Action.
  + **Response**: Khi View đã được render thành HTML, hệ thống sẽ trả về kết quả Response cho client. Kết quả này có thể bao gồm HTML, file, JSON, hoặc bất kỳ loại dữ liệu nào khác tùy thuộc vào loại Result được trả về.

### Các tính năng của nền tảng ASP.NET MVC

Tách bạch các tác vụ của ứng dụng (logic nhập liệu, business logic, và logic giao diện), dễ dàng kiểm thử và mặc định áp dụng hướng phát triển TDD. Tất cả các tính năng chính của mô hình MVC được cài đặt dựa trên interface và được kiểm thử bằng cách sử dụng các đối tượng mocks, mock object là các đối tượng mô phỏng các tính năng của những đối tượng thực sự trong ứng dụng. Có thể sử dụng bất kỳ nền tảng unit-testing nào tương thích với nền tảng .NET.

MVC là một nền tảng khả mở rộng (extensible) & khả nhúng (pluggable). Các thành phần của ASP.NET MVC được thiết kể để chúng có thể được thay thế một cách dễ dàng hoặc dễ dàng tùy chỉnh. Bạn có thể nhúng thêm view engine, cơ chế định tuyến cho URL, cách kết xuất tham số của action-method và các thành phần khác. ASP.NET MVC cũng hỗ trợ việc sử dụng Dependency Injection (DI) và Inversion of Control (IoC). DI cho phép bạn gắn các đối tượng vào một lớp cho lớp đó sử dụng thay vì buộc lớp đó phải tự mình khởi tạo các đối tượng. IoC quy định rằng, nếu một đối tượng yêu cầu một đối tượng khác, đối tượng đầu sẽ lấy đối tượng thứ hai từ một nguồn bên ngoài, ví dụ như từ tập tin cấu hình. Và nhờ vậy, việc sử dụng DI và IoC sẽ giúp kiểm thử dễ dàng hơn.

ASP.NET MVC có thành phần ánh xạ URL mạnh mẽ cho phép xây dựng những ứng dụng có các địa chỉ URL xúc tích và dễ tìm kiếm. Các địa chỉ URL không cần phải có phần mở rộng của tên tập tin và được thiết kế để hỗ trợ các mẫu định dạng tên phù hợp với việc tối ưu hóa tìm kiếm (URL) và phù hợp với lập địa chỉ theo kiểu REST.

Hỗ trợ sử dụng đặc tả (các thẻ) của các trang ASP.NET(.aspx), điều khiển người dùng (.ascx) và trang master page (.marter). Có thể sử dụng các tính năng có sẵn của ASP.NET như là sử dụng lồng các trang master page, sử dụng in-line expression (<%= %>), sử dụng server controls, mẫu, data-binding, địa phương hóa (localization) và hơn thế nữa.

Hỗ trợ các tính năng có sẵn của ASP.NET như cơ chế xác thực người dùng, quản lý thành viên, quyền, output caching và data caching, seession và profile, quản lý tình trạng ứng dụng, hệ thống cấu hình…

ASP.NET MVC 3 còn bổ sung một view engine mới là Razor View Engine cho phép thiết lập các view nhanh chóng, dễ dàng và tốn ít công sức hơn so với việc sử dụng Web Forms view engine. ASP.NET MVC 4 hỗ trợ tốt hơn cho các thiết bị di động bổ sung ASP.NET Web API. ASP.NET MVC 5 hỗ trợ chứng thực qua các API khác (facebook,google+..), Bootstrap được thêm vào hổ trợ phần thiết kế giao diện.

### Lợi ích của ứng dụng web dựa trên mô hình MVC

* Dễ dàng quản lý sự phức tạp của ứng dụng bằng cách chia ứng dụng thành ba thành phần model, view, controller
* Nó không sử dụng view state hoặc server-based form. Điều này tốt cho những lập trình viên muốn quản lý hết các khía cạnh của một ứng dụng.
* Nó sử dụng mẫu Front Controller, mẫu này giúp quản lý các requests (yêu cầu) chỉ thông qua một Controller. Nhờ đó bạn có thể thiết kế một hạ tầng quản lý định tuyến. Để có nhiều thông tin hơn, bạn nên xem phần Front Controller trên web site MSDN
* Hỗ trợ tốt hơn cho mô hình phát triển ứng dụng hướng kiểm thử (TDD)
* Nó hỗ trợ tốt cho các ứng dụng được xây dựng bởi những đội có nhiều lập trình viên và thiết kế mà vẫn quản lý được tính năng của ứng dụng

## Ngôn ngữ CSS

CSS (Cascading Style Sheets) là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML,…

CSS cung cấp nhiều thuộc tính trình bày dành cho các đối tượng với sự sáng tạo trong việc kết hợp các thuộc tính giúp mang lại hiệu quả cao.

CSS đã được hỗ trợ bởi tất cả các trình duyệt và hiển thị “như nhau” trên mọi hệ điều hành.

CSS đưa ra phương thức áp dụng từ một file CSS ở ngoài. Có hiệu quả đồng bộ khi tạo một website có hàng trăm trang hay khi muốn thay đổi một thuộc tính trình bày nào đó.

CSS được cập nhật liên tục mang lại các trình bày phức tạp và tinh vi hơn.

## Ngôn ngữ Javascript

Với HTML chỉ biểu diễn thông tin chứ chưa phải là các trang Web động có khả năng đáp ứng các sự kiện từ phía người dùng.

Hãng Netscape đã đưa ra ngôn ngữ script có tên là LiveScript để thực hiện chức năng này. Sau đó đổi tên thành JavaScript để tận dụng tính đại chúng của ngôn ngữ lập trình Java.

JavaScript là ngôn ngữ dưới dạng script có thể gắn với các file HTML. Được trình duyệt diễn dịch, trình duyệt đọc JavaScript dưới dạng mã nguồn.

JavaScript là ngôn ngữ dựa trên đối tượng, nghĩa là bao gồm nhiều kiểu đối tượng, ví dụ đối tượng Math với tất cả các chức năng toán học. Nhưng JavaScript không là ngôn ngữ hướng đối tượng như C++/Java.

## Ngôn ngữ HTML

HTML (Hypertext Markup Language-Ngôn ngữ đánh dấu siêu văn bản) là ngôn ngữ đánh dấu chuẩn dùng soạn thảo các tài liệu World Wide Web, chỉ rõ một trang Web được hiển thị như thế nào trong trình duyệt.

Ngôn ngữ HTML dùng các tag hoặc các đoạn mã lệnh để chỉ cho các trình duyệt (Web browsers) cách hiển thị các thành phần của trang như text và graphics.

HTML là ngôn ngữ xác định cấu trúc của thông tin.

HTML sử dụng một loạt các thẻ và thuộc tính và được dùng để hiện thị văn bản và các thông tin khác, cung cấp siêu liên kết tới các tài liệu khác

# CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

\*\*\*\*\*\*

## Mô tả bài toán

Mục đích của đề tài nhằm xây dựng hệ thống phần mềm hoạt trên môi trường web để quản lý việc bán hàng được thuận tiện, giúp cho người quản lý sẽ dễ dàng trong việc quản lý sản phẩm, quản lý user của nhân viên hoặc khách hàng, có thể kiểm soát được đơn hàng của khách và thống kê doanh thu của cửa hàng. Còn khách hàng mua hàng trên web sẽ được thoải mải hơn vì không tốn thời gian để đến cửa hàng, mà vẫn mua được sản phẩm mà mình muốn.

Trước tiên, để thực hiện các chức năng quản lý cửa hàng, người quản lý cần phải đăng nhập tài khoản và mật khẩu vào trang admin thì mới thực hiện được các chức năng đó. Trong đó đã phân quyền các nhân viên để thực hiện các chức năng khác nhau, chỉ có người quản lý chính mới được đầy đủ quyền kiểm soát.

## Mô tả chức năng

* *Chức năng hệ thống đối với người quản lý (admin)*
* Thêm/xóa/sửa thông tin các sản phẩm mới hoặc có sẵn tại cửa hàng.
* Cập nhật giá cho sản phẩm.
* Quản lý bài viết về các sản phẩm nổi bật hoặc chương trình khuyến mãi, tin tức liên quan.
* Quản lý đơn hàng
* Quản lý, phân quyền user.
* Quản lý hóa đơn, doanh thu
* *Chức năng hệ thống đối với nhân viên*
* Đăng nhập hệ thống
* Cập nhật thông tin sản phẩm.
* Kiểm tra, xử lý đơn hàng
* Phản hồi liên hệ của khách hàng
* *Chức năng hệ thống đối với khách hàng*
* Đăng ký/đăng nhập tài khoản.
* Tìm kiếm sản phẩm theo giá, loại sản phẩm..
* Chọn sản phẩm và xem chi tiết sản phẩm
* Xem tin tức hoặc chương trình khuyến mãi liên quan.
* Đặt hàng, cập nhật giỏ hàng (Thêm, xóa, cập nhật, xóa tất cả trong giỏ hàng).
* Chọn phương thức thanh toán.
* Liên hệ với cửa hàng thông qua trang liên hệ

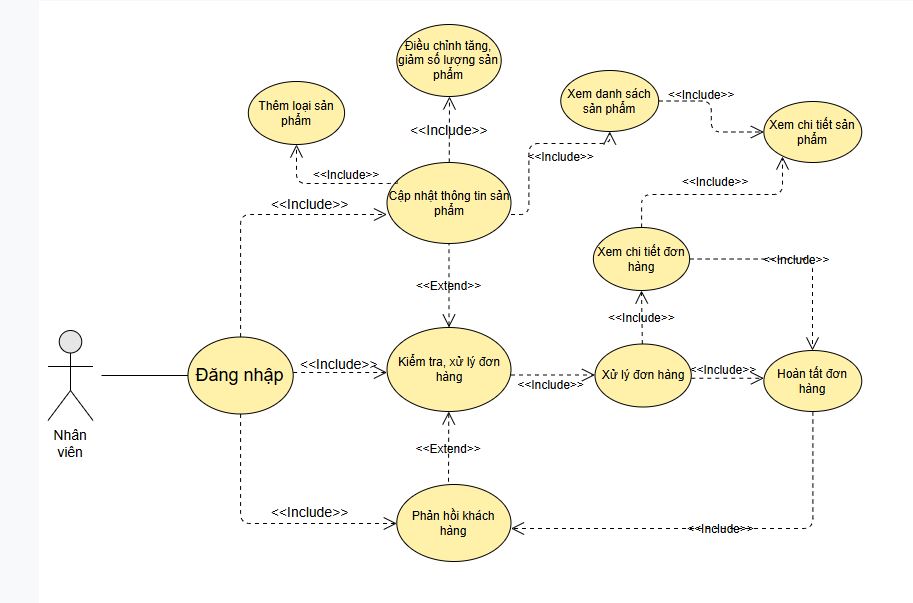
## Sơ đồ Use Case

### UseCase quản trị (Admin)



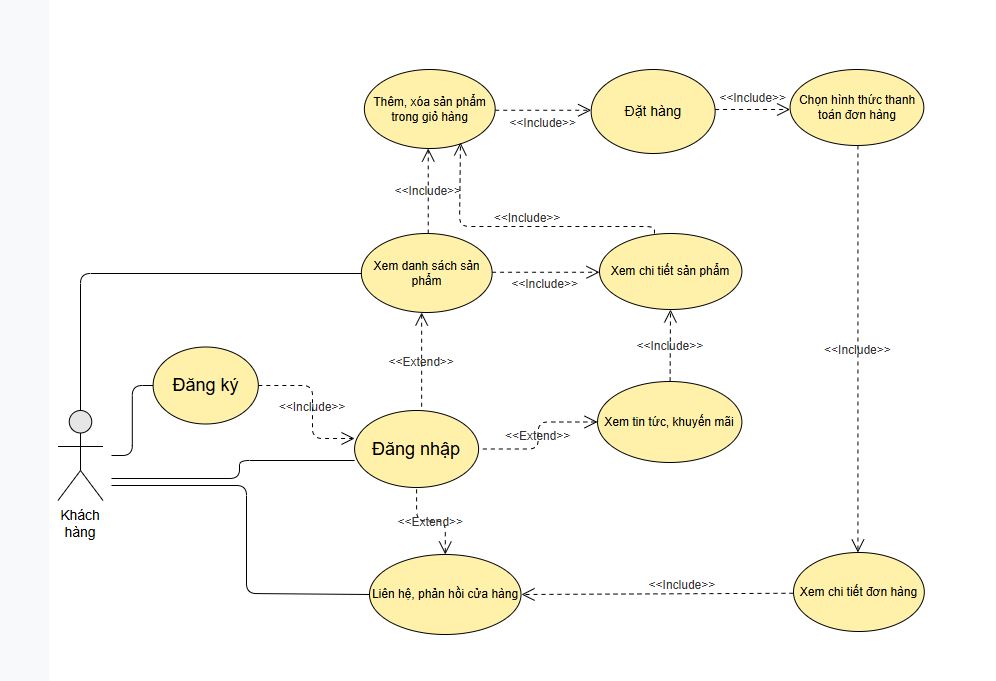
Hình 2. . UseCase Admin

### UseCase Nhân viên



Hình 2. 2. UseCase Nhân viên

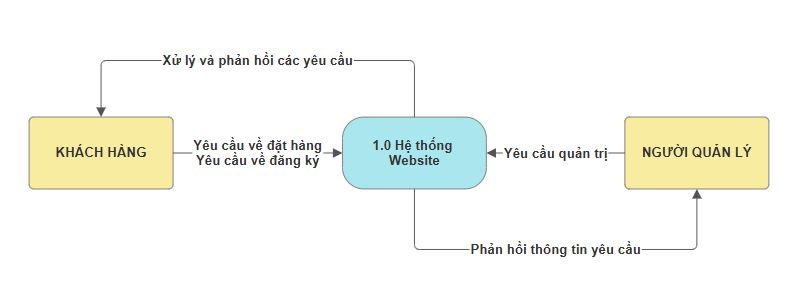
### UseCase khách hàng



Hình 2. 3. UseCase Khách hàng

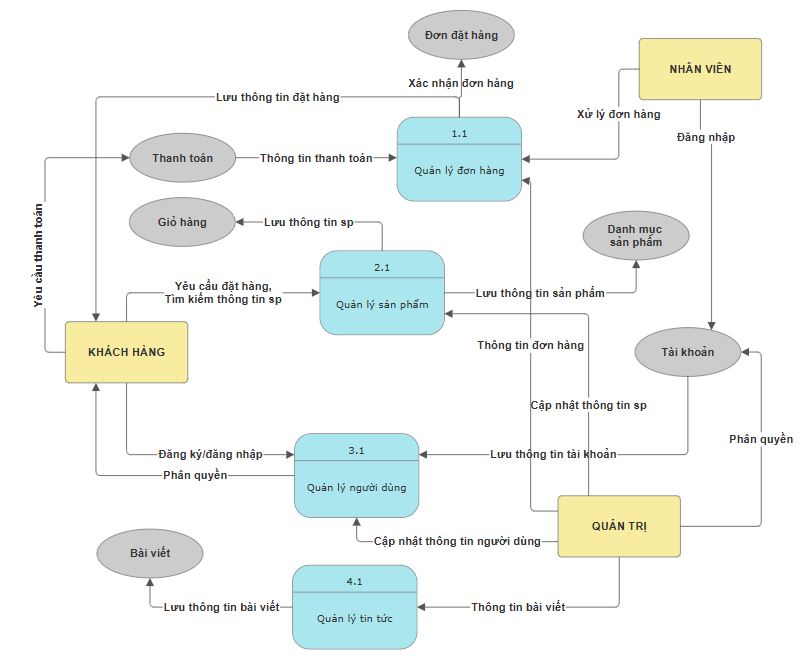
## Sơ đồ DFD

### Mô hình luồng dữ liệu mức khung cảnh (DFD mức 0)



Hình 2. 4. Sơ đồ DFD mức 0

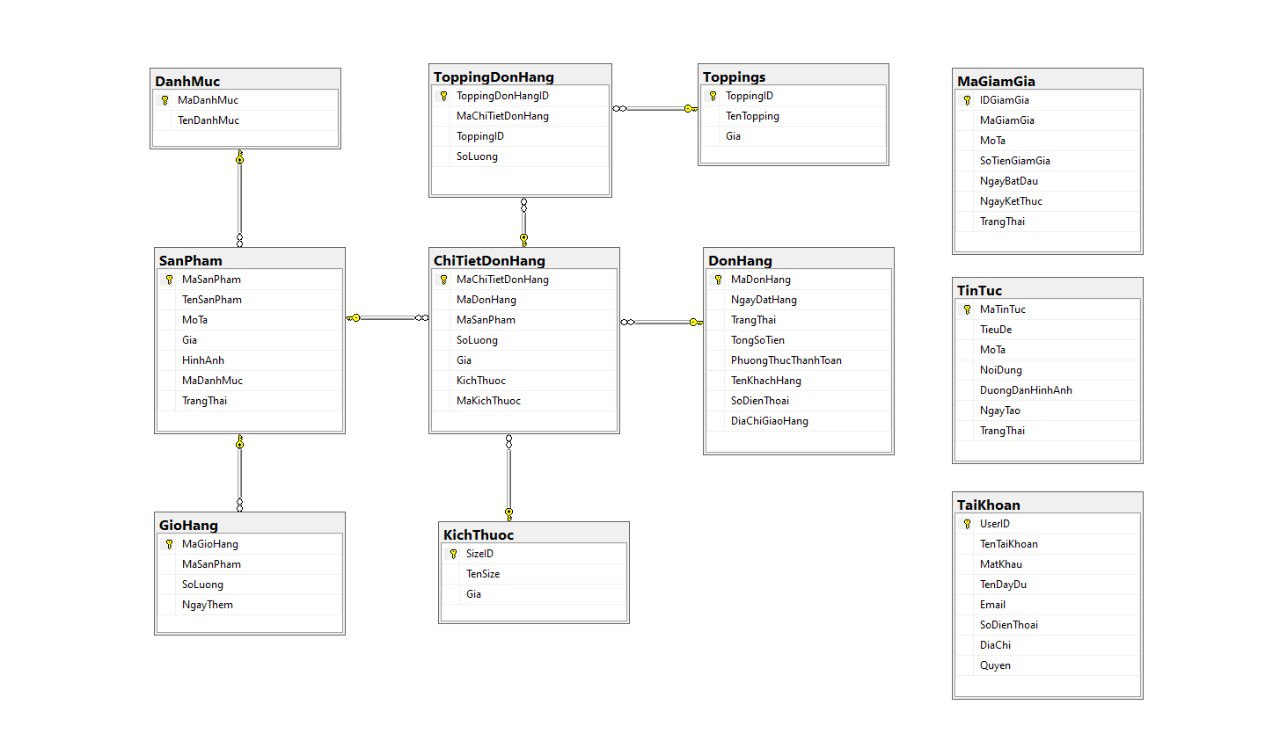
### Mô hình luồng dữ liệu mức đỉnh (DFD mức 1)



Hình 2. 5. Sơ đồ DFD mức 1

## Thiết kế, xây dựng dữ liệu

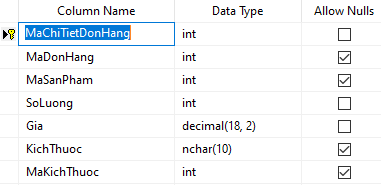
### Mô hình cơ sở dữ liệu



Hình 2. 6. Mô hình CSDL

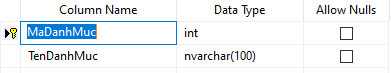
### Mô tả chi tiết các bảng

### Bảng ChiTietDonHang



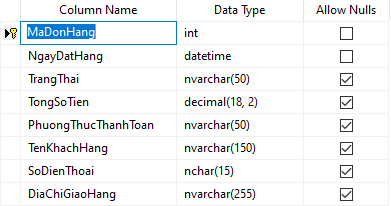
Hình 2. 7. Bảng Chi tiết đơn hàng

### Bảng DanhMuc



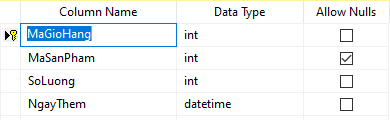
Hình 2. 8. Bảng Danh mục

### Bảng DonHang



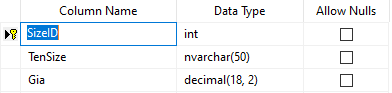
Hình 2. 9. Bảng Đơn hàng

### Bảng GioHang



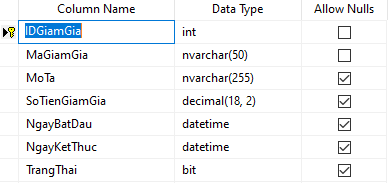
Hình 2. 10. Bảng Giỏ hàng

### Bảng KichThuoc



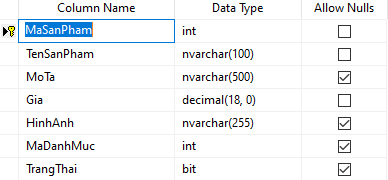
Hình 2. 11. Bảng Kích thước

### Bảng MaGiamGia



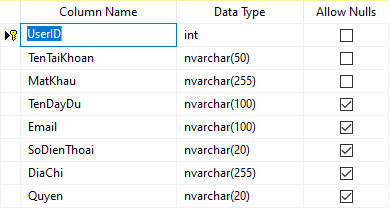
Hình 2. 12. Bảng Mã giảm giá

### Bảng SanPham



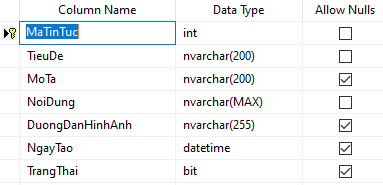
Hình 2. 13. Bảng Sản phẩm

### Bảng TaiKhoan



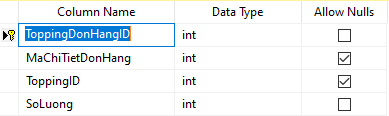
Hình 2. 14. Bảng Tài khoản

### Bảng TinTuc



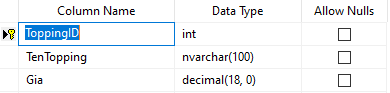
Hình 2. 15. Bảng Tin tức

### Bảng ToppingDonHang



Hình 2. 16. Bảng Topping đơn hàng

### Bảng Toppings

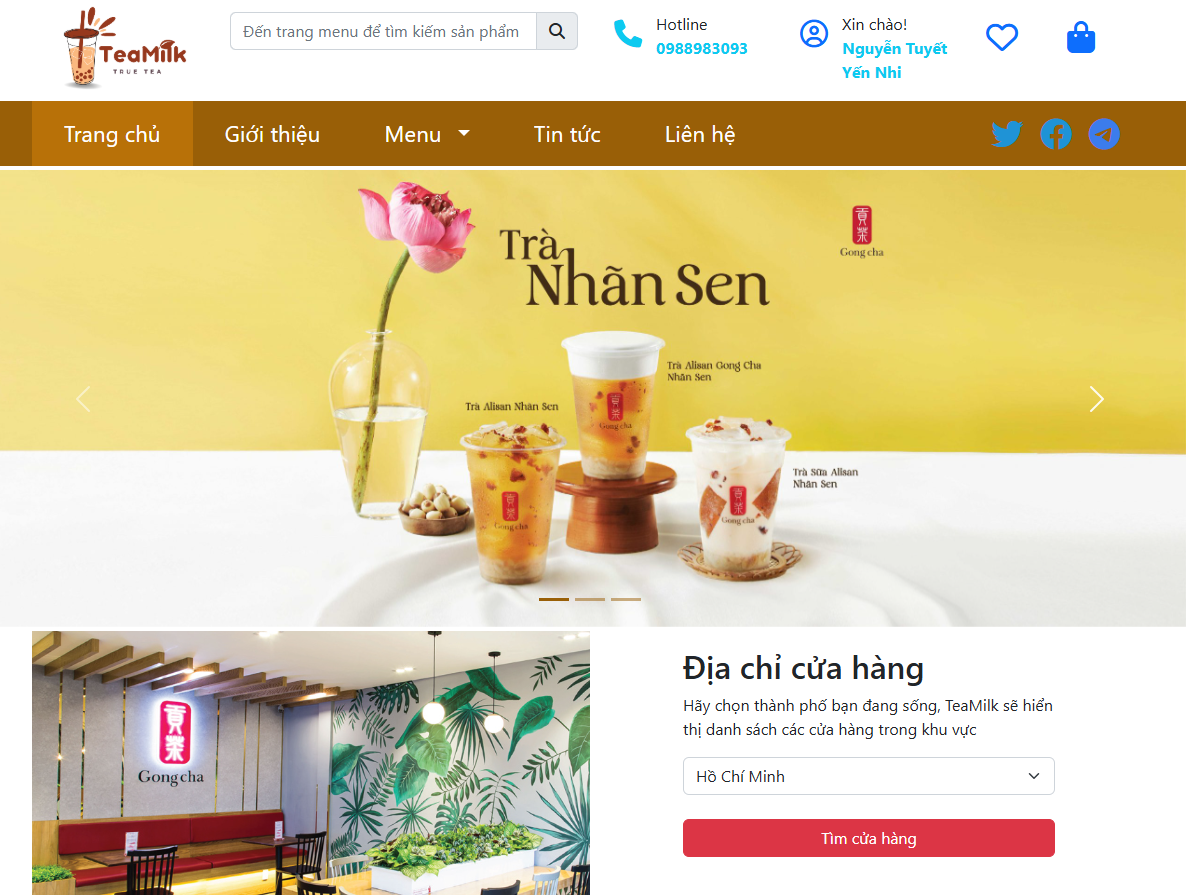


Hình 2. 17. Bảng Topping

# CHƯƠNG 3: KẾT QUẢ THỰC NGHIỆM

\*\*\*\*\*\*

## Giao diện trang chủ người dùng



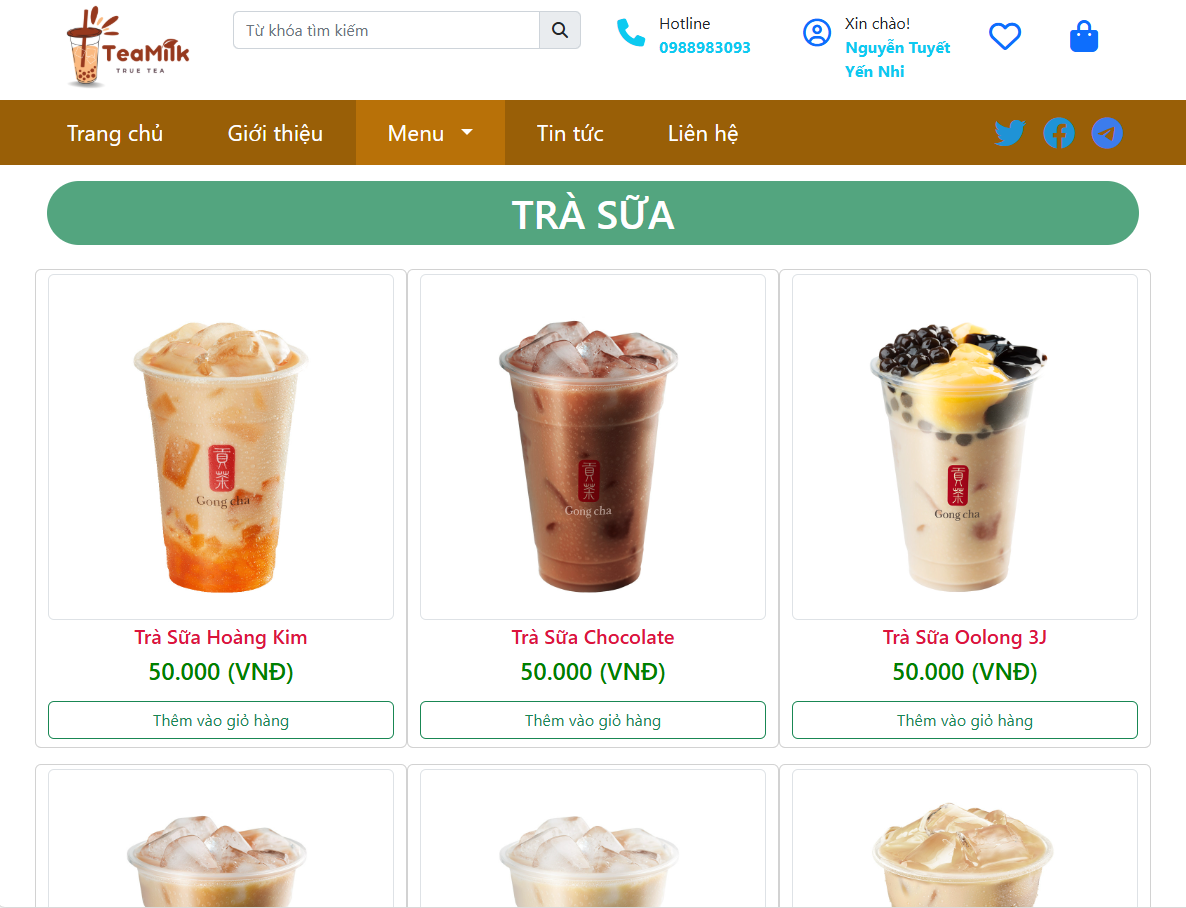
Hình 3. 1. Trang chủ người dùng

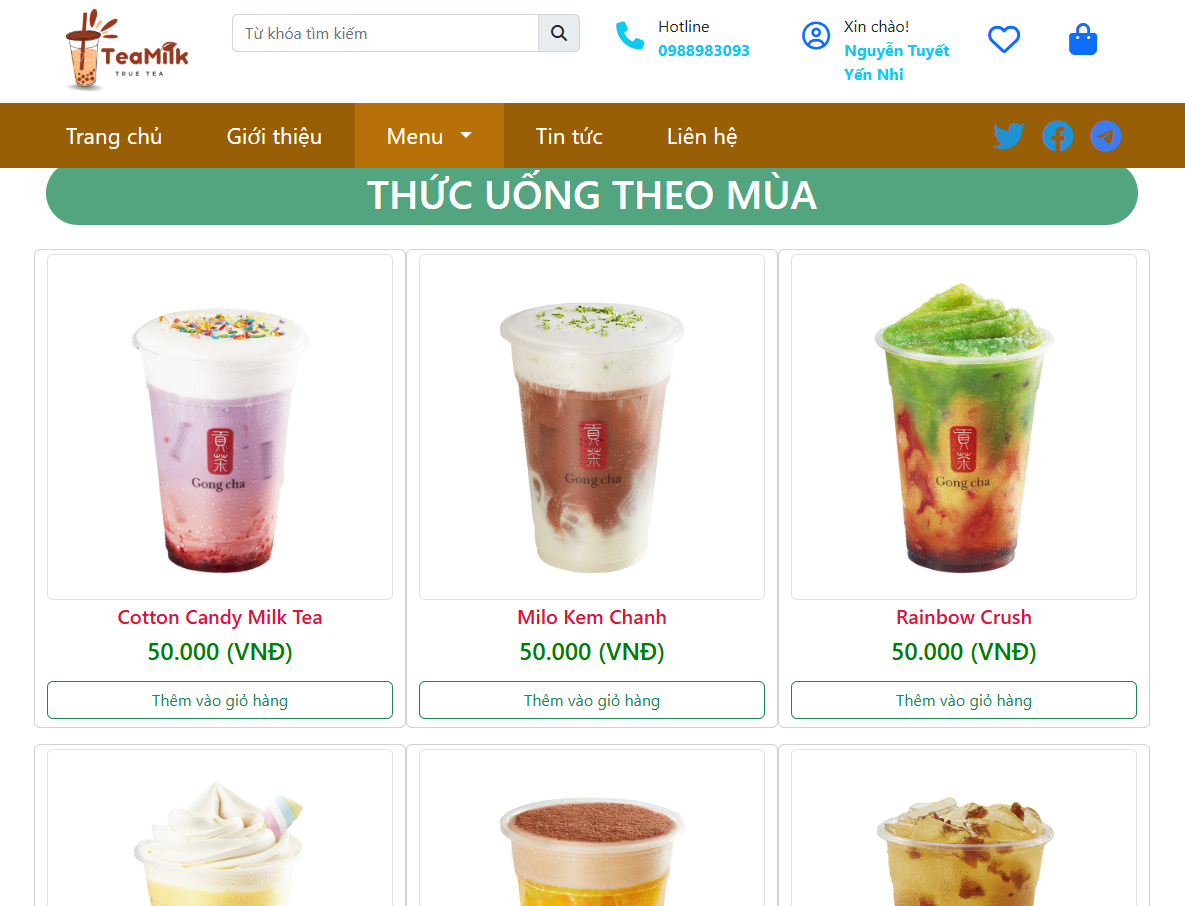
## Giao diện trang giới thiệu



Hình 3. 2. Trang giới thiệu

## Giao diện trang sản phẩm



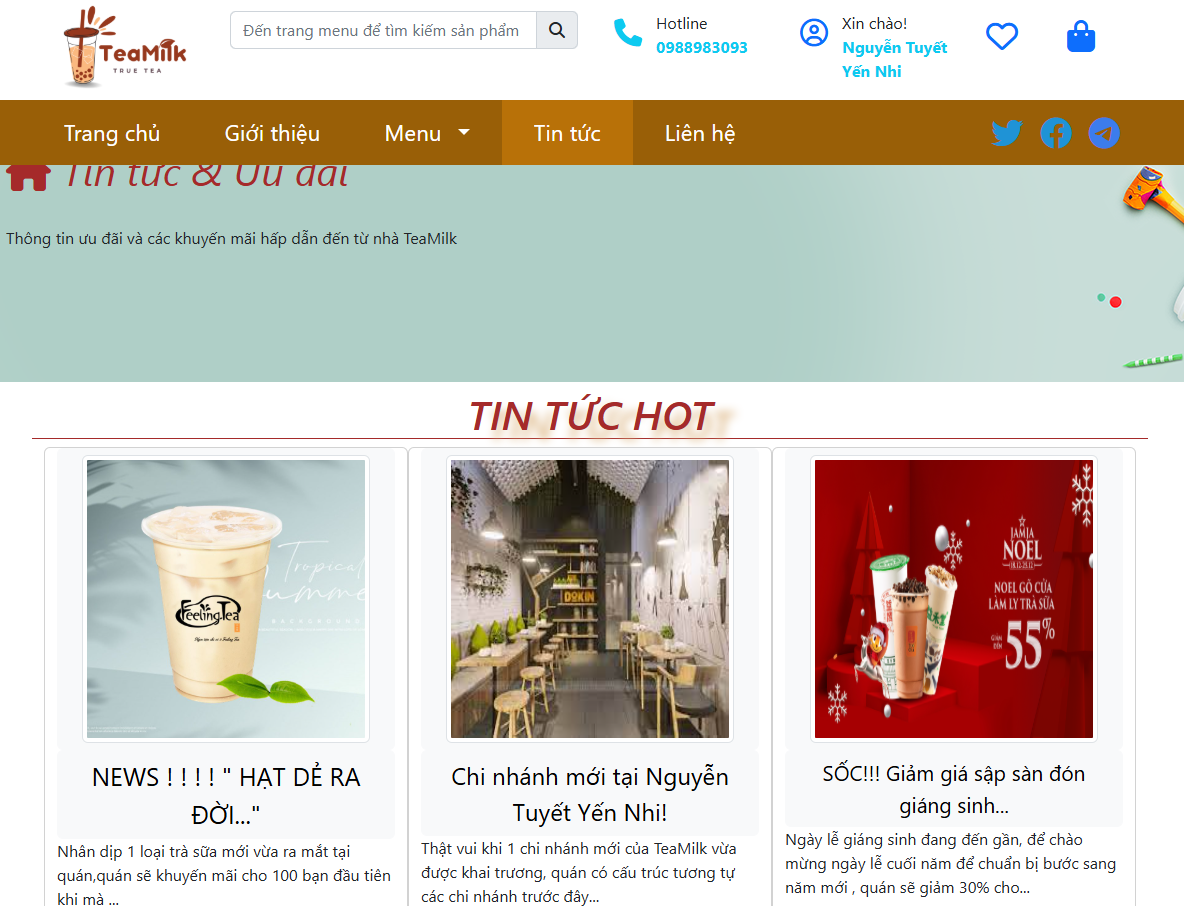


Hình 3. 3. Trang sản phẩm

## Giao diện trang đặt hàng thành công

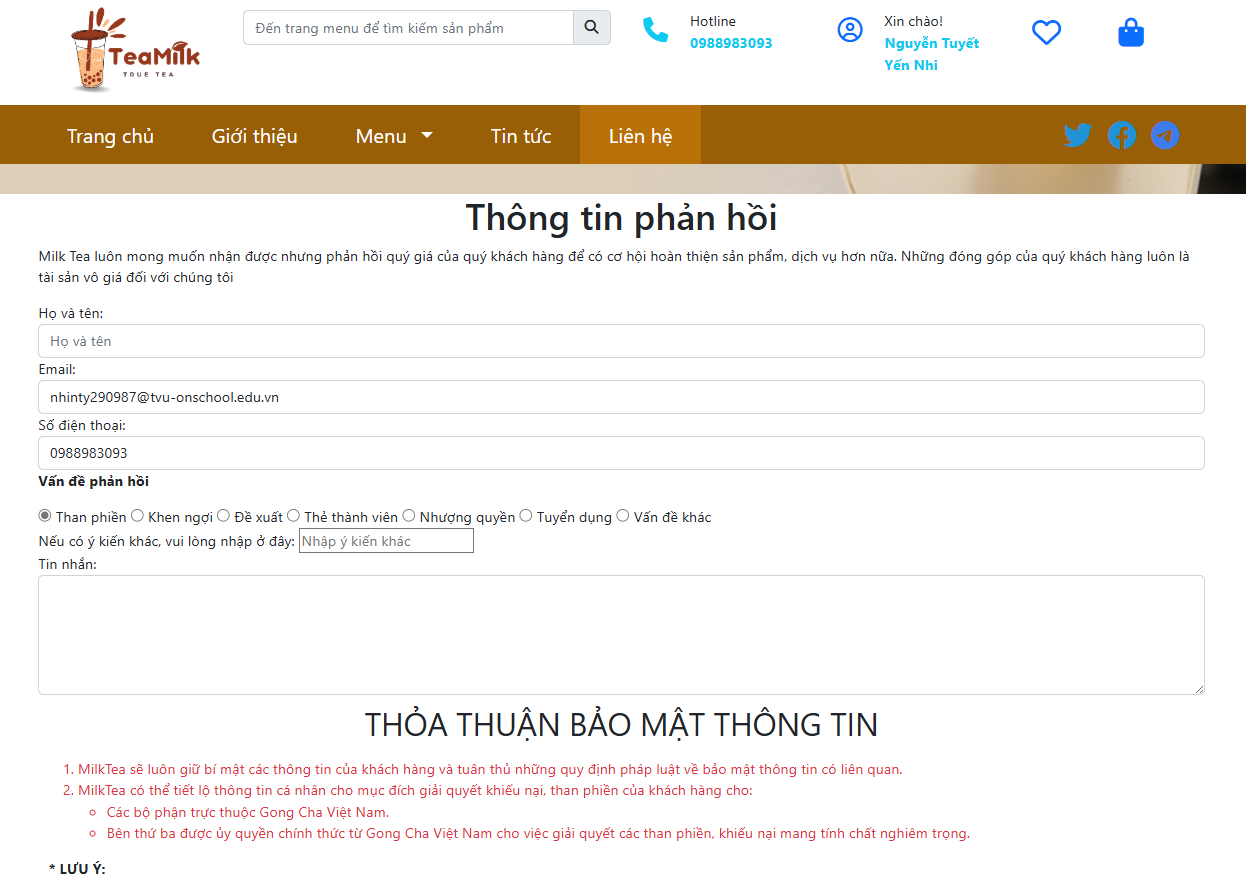


## Giao diện trang tin tức



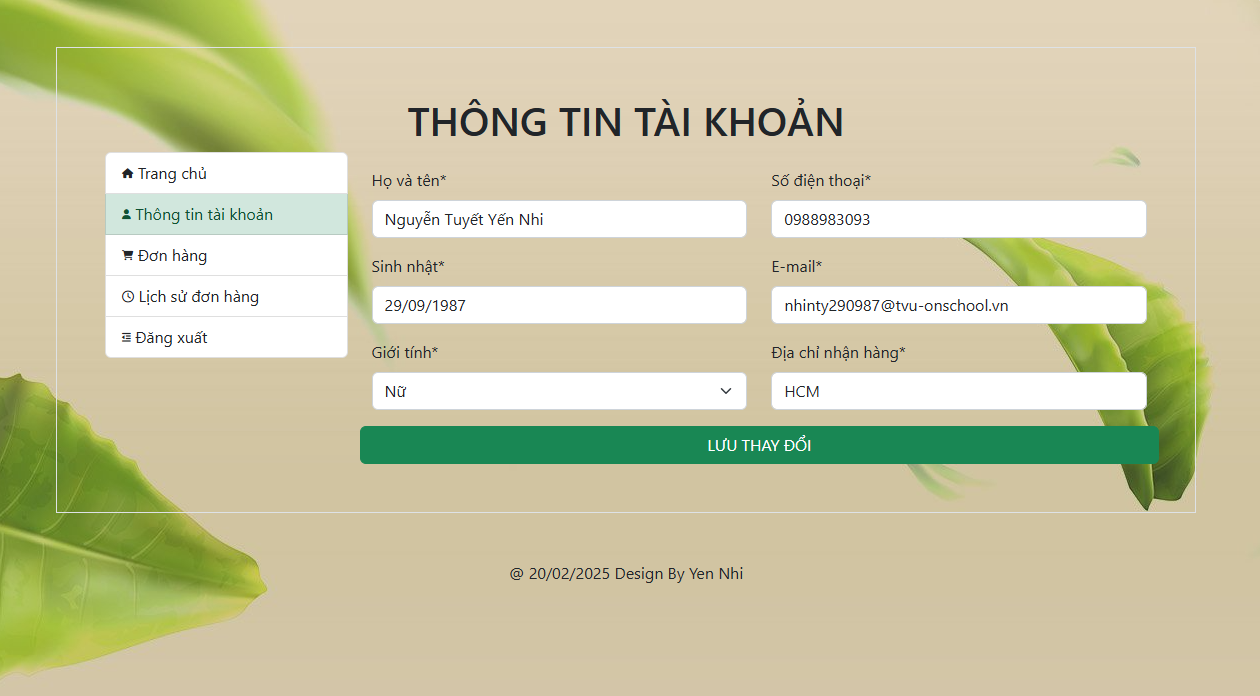
Hình 3. 7. Trang tin tức

## Giao diện trang liên hệ



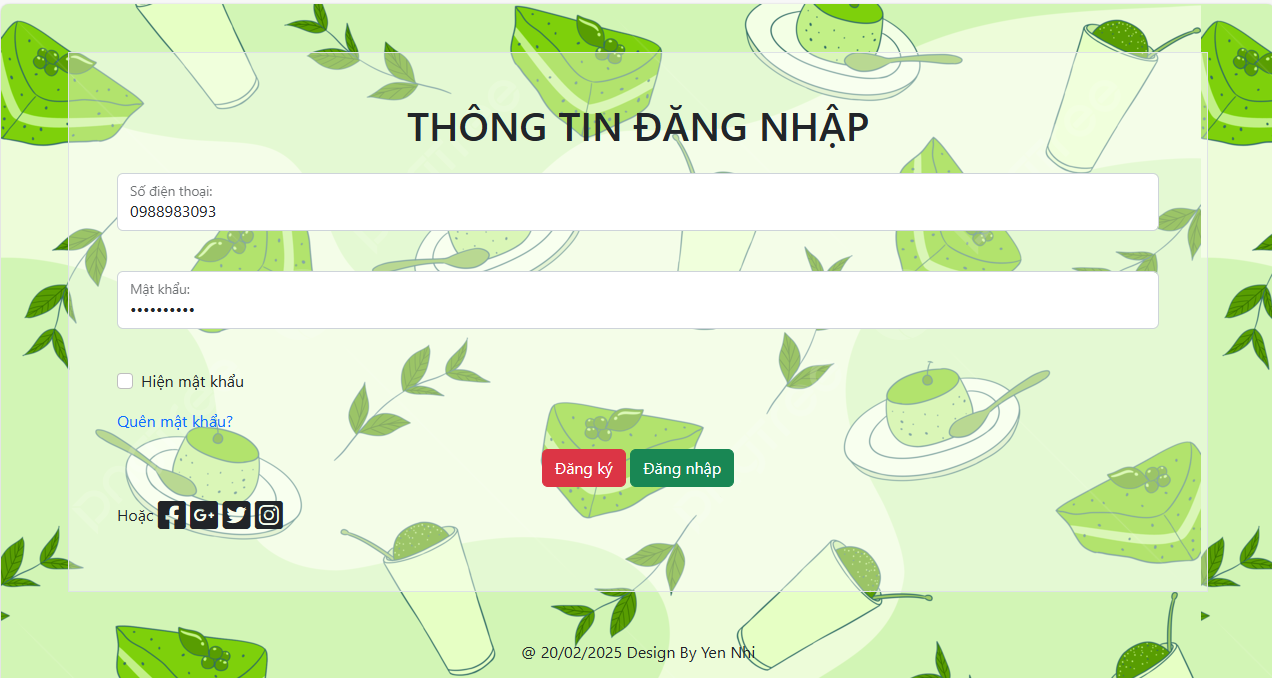
Hình 3. 8. Trang liên hệ

## Giao diện thông tin tài khoản



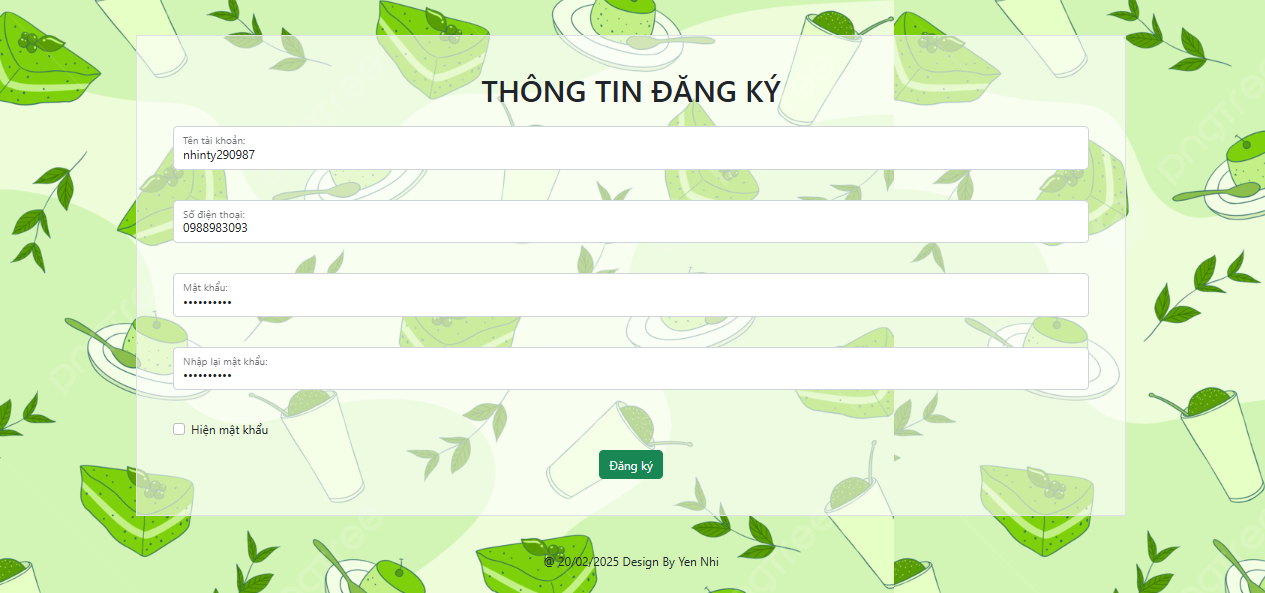
Hình 3. 9. Trang thông tin tài khoản

## Giao diện trang đăng nhập tài khoản



Hình 3. 11. Trang đăng nhập

## Giao diện trang đăng ký tài khoản



Hình 3. 10. Trang đăng ký tài khoản

# CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**\*\*\*\*\*\***

## Kết Luận

### Ưu điểm

Nắm được sự khác biệt giữa công nghệ web ASP.Net. Xây dựng được một trang web thương mại đơn giản trên nền tảng kiến thức đã học.

Tiếp cận và ứng dụng ngôn ngữ lập trình C# vào phát triển web, giúp nắm rõ hơn về sự linh hoạt của ngôn ngữ này trong môi trường web và sự khác biệt trong cấu trúc lập trình giữa ASP .Net Web Forms và ASP.Net MVC. Phát triển kỹ năng lập trình và tiếp cận các công nghệ mang tính nâng cao như các chức năng thanh toán, mở rộng phạm vi ứng dụng của web.

### Nhược điểm (hạn chế)

* Giao diện còn sơ sài, các chức năng còn hạn chế
* Các lỗi trang web chưa tường minh, chưa xây dựng được các trang lỗi cụ thể
* Tuy đã xây dựng được định hướng các chức năng cần thiết nhưng vẫn chưa hoàn thiện kịp thời do thời gian có hạn.

## Hướng phát triển

Cải thiện và tối ưu hóa giao diện trang người dùng, tập trung vào tính thẩm mỹ và thân thiện để mang lại trải nghiệm tốt hơn. Cải thiện khả năng tương thích của trang web để nó có thể hoạt động tốt không chỉ trên máy tính để bàn mà còn trên các thiết bị di động, đảm bảo tính responsive cho mọi kích thước màn hình.

Bổ sung và tích hợp tính năng đăng ký/đăng nhập và đăng nhập nhanh bằng tài khoản mạng xã hội khác như Facebook hoặc Gmail, để tạo sự tiện lợi cho người dùng.

Tích hợp chức năng gửi mail hoặc phản hồi tự động các câu hỏi tương tự trong ngân hàng câu hỏi thay vì cho nhân viên vào phản hồi từng mục tương tự, nếu người mua có yêu cầu gặp nhân viên quản lý thì sẽ chuyển hướng đến cho họ. Điều này giúp người dùng nhanh chóng nhận được phản hồi qua email cá nhân hoặc trực tiếp trên trang web để tránh trường hợp nhân viên chưa phản hồi kịp và bỏ sót các câu hỏi của họ.

Tích hợp thêm các phương thức thanh toán trực tuyến khác, cải thiện quy trình đặt hàng, cập nhật định vị và bản đồ từ lúc nhân viên giao hàng nhận đơn đến khi người mua nhận được hàng, giúp người mua dễ dàng đặt hàng và theo dõi đơn hàng.

Tạo các trang lỗi tùy chỉnh cho các lỗi như 404 (trang không tìm thấy), 500 (lỗi máy chủ).v.v.. để hướng dẫn người dùng quay lại trang chủ hoặc báo lỗi trực tiếp với quản trị viên.

# DANH MỤC TÀI LIỆU THAM KHẢO

\*\*\*\*\*\*

|  |
| --- |
| [MVC là gì: Hiểu rõ mô hình MVC A-Z trong lập trình - ITviec Blog](https://itviec.com/blog/mvc-la-gi/) [Truy cập 19/02/2025] |
| <https://glints.com/vn/blog/asp-net-la-gi/> [Truy cập 19/02/2025] |
| <https://bizfly.vn/techblog/aspnet-la-gi.html> [Truy cập 19/02/2025] |
| <https://topdev.vn/blog/phat-trien-ung-dung-web-ban-chon-asp-net-web-forms-hay-asp-net-mvc/> [Truy cập 19/02/2025] |
| [Giáo trình ASP.net với CSharp - Giáo trình ASP với CSharp Biên tập bởi: Khuyet Danh Giáo trình ASP - Studocu](https://www.studocu.com/vn/document/hoc-vien-cong-nghe-buu-chinh-vien-thong/information-technology/giao-trinh-aspnet-voi-csharp/28655557) [Truy cập 19/02/2025] |
| [(PDF) Pro ASP.NET MVC 5 | tuan sang - Academia.edu](https://www.academia.edu/9734520/Pro_ASP_NET_MVC_5) [Truy cập 19/02/2025] |
| <https://techmaster.vn/posts/33728/aspnet-mvc-la-gi-and-tai-sao-ban-nen-su-dung-no> |
| <https://cmccloud.vn/tin-tuc/sql-server-la-gi> [Truy cập 19/02/2025] |
| <https://sqladvice.com/tong-quan-ve-sql-server/> [Truy cập 19/02/2025] |
| <https://stringee.com/vi/blog/post/boot-strap-la-gi> [Truy cập 19/02/2025] |
| [https://200lab.io/blog/jquery-la- [Truy cập 19/02/2025]gi/?srsltid=AfmBOorqGETPL6cfhJeVA46LjYw7cZE\_GF0lQxOeyqtWOS2xkFbm\_VBQ](https://200lab.io/blog/jquery-la-gi/?srsltid=AfmBOorqGETPL6cfhJeVA46LjYw7cZE_GF0lQxOeyqtWOS2xkFbm_VBQ) [Truy cập 19/02/2025] |