

**TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN**



**THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2025-2026**

**XÂY DỰNG
HỆ THỐNG BÁN THỨC ĂN NHANH**

Giảng viên hướng dẫn:

ThS. Phạm Thị Trúc Mai

Sinh viên thực hiện:

Họ tên: Đào Công Duy

MSSV: 110122058

Lớp: DA22TTD

Vĩnh Long , tháng 12 năm 2025

**TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN**



**THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2025-2026**

**XÂY DỰNG
HỆ THỐNG BÁN THỨC ĂN NHANH**

Giảng viên hướng dẫn:

ThS. Phạm Thị Trúc Mai

Sinh viên thực hiện:

Họ Tên: Đào Công Duy

MSSV: 110122058

Mã Lớp: DA22TTD

Vĩnh Long , tháng 12 năm 2025

[illegible]

(Ký tên và ghi rõ họ tên)

This image shows a full page of a handwriting practice worksheet. It consists of multiple rows of horizontal dotted lines spaced evenly down the page, providing a guide for letter height and placement. The background is plain white, and there are no other markings or text present.

(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Trước tiên, em xin bày tỏ lòng biết ơn chân thành đến cô Phạm Thị Trúc Mai, giảng viên Khoa Kỹ thuật và Công nghệ, Trường Đại học Trà Vinh, người đã tận tình hướng dẫn và chỉ bảo em trong suốt quá trình thực hiện đề tài “Xây dựng hệ thống bán thức ăn nhanh online”. Những chia sẻ, góp ý quý báu và sự tận tâm của cô đã giúp em mở rộng kiến thức, phát triển tư duy, cũng như giải quyết hiệu quả các vấn đề phát sinh trong quá trình thực hiện đề tài.

Em cũng xin gửi lời cảm ơn sâu sắc đến quý Thầy Cô Trường Đại học Trà Vinh, đặc biệt là các Thầy Cô thuộc Khoa Kỹ thuật và Công nghệ, những người đã tận tâm giảng dạy, truyền đạt kiến thức và trang bị cho em nền tảng vững chắc trong học tập và nghiên cứu.

Cuối cùng, em nhận thức rằng bài làm của mình vẫn còn nhiều hạn chế và thiếu sót. Đây là lần đầu em tiếp cận với việc xây dựng hệ thống web sử dụng PHP kết hợp với XAMPP và SQL, nên kinh nghiệm của em vẫn còn hạn chế. Vì vậy, em rất mong nhận được những ý kiến đóng góp từ quý Thầy Cô để hoàn thiện bản thân và tích lũy thêm kinh nghiệm quý báu cho các bài báo cáo trong tương lai.

Em xin gửi lời cảm ơn chân thành một lần nữa và hy vọng sẽ tiếp tục nhận được sự hướng dẫn, hỗ trợ từ quý Thầy Cô trong chặng đường học tập phía trước.

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	11
1.1. Đặt vấn đề	11
1.2 .Mục đích nguyên cứu	11
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT	12
2.1.Giới thiệu về ngôn ngữ PHP	12
2.1.1.Tổng quan về ngôn PHP	12
2.1.2.Các Thành Phần của ngôn ngữ PHP	13
2.1.3. Ưu nhược điểm của ngôn ngữ lập trình PHP	14
2.1.4 .Ứng dụng của ngôn ngữ PHP trong lập trình	16
2.1.5. Sự khác biệt giữ PHP với các ngôn ngữ lập trình khác	17
2.2. Gới thiệu về SQL	18
2.2.1.Tổng quan về MySQL	18
2.2.2.Các thành phần của MySQL	19
2.2.3 .Ưu nhược điểm của MySQL	20
2.2.4 Một số tính năng nổi bật của MySQL	22
2.3 .Tổng quan về HTML	23
2.3.1.Tổng quan	23
2.3.2 .Mối quan hệ giữa HTML, CSS và JavaScript	24
2.3.4 Ưu nhược điểm của HTML	25
2.4. Tổng qua về JavaScript	26
2.4.1. JavaScript là gì?	26
2.4.2. Lịch sử phát triển của JavaScript	27
2.4.3. Cách hoạt động và công dụng của JavaScript	28
2.4.4. Ưu nhược điểm của JavaScript	29
2.5. Ứng dụng mô hình MVC trong hệ thống bán thức ăn nhanh online	31
2.5.1 Khái quát mô hình MVC	31
2.5.2 Áp dụng mô hình MVC trong hệ thống website bán thức ăn nhanh	31
2.5.3 Lợi ích của việc áp dụng mô hình MVC	33
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	34
3.1. Mô tả đề tài	34
3.2. Đặc tả yêu cầu hệ thống	35

3.2.1. Yêu cầu chức năng	35
3.2.2. Yêu cầu phi chức năng	36
3.3. Thiết kế dữ liệu	38
3.3.1. Mô hình thực thể kết hợp (ERD)	38
3.3.2. Mô hình vật lý	39
3.3.3. Mô hình dữ liệu	40
3.3.4. Mô tả các bảng	40
3.3.5. Sơ đồ Use case của User (khách hàng)	43
3.3.6. Sơ đồ Use case của quản trị (Admin)	45
3.3.6. Sơ đồ DFD	47
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	52
4.1 Giao diện của người dùng	52
4.1.1. Giao diện đăng nhập	52
4.1.2. Giao diện đăng ký	53
4.1.3. Giao diện Trang chủ	54
4.1.4. Giao diện Món ăn	55
4.1.5. Giao diện Giỏ hàng	56
4.1.6. Giao diện Đơn hàng	57
4.1.7. Giao diện Liên hệ	58
4.2. Giao diện của quản trị viên	59
4.2.1. Giao diện đăng nhập	59
4.2.2. Giao diện Trang chủ	60
4.2.3. Giao diện Món ăn	61
4.2.4. Giao diện Dashboard	62
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	68
5.1. Kết luận	68
5.2. Hướng phát triển	68
DANH MỤC TÀI LIỆU THAM KHẢO	69

DANH MỤC HÌNH ẢNH

Hình 3.1. Mô hình thực thể kết hợp	38
Hình 3.2. Mô hình vật lý	39
Hình 3.3. Sơ đồ Use Case của khách hàng (người dùng)	43
Hình 3.4. Sơ đồ Use Case của quản trị viên	45
Hình 3.5. Sơ đồ DFD mức 0	47
Hình 3.6. Sơ đồ DFD mức 1	48
Hình 3.7. Sơ đồ DFD mức 2	50
Hình 4.1. Giao diện đăng nhập	52
Hình 4.2. Giao diện đăng kí	53
Hình 4.3. Giao diện trang chủ	54
Hình 4.4. Giao diện món ăn	55
Hình 4.5. Giao diện giỏ hàng	56
Hình 4.6. Giao diện đơn hàng	57
Hình 4.7. Giao diện liên hệ	58
Hình 4.8. Giao diện đăng nhập	59
Hình 4.9. Giao diện trang chủ	60
Hình 4.10. Giao diện món ăn	61
Hình 4.11. Giao diện Dashboard	62
Hình 4.12. Giao diện thêm món ăn	63
Hình 4.13. Giao diện đơn hàng	64
Hình 4.14. Giao diện đánh giá	65
Hình 4.15. Giao diện quản lí khách hàng (người dùng)	66
Hình 4.16. Giao diện liên hệ	67

DANH MỤC BẢNG BIỂU

Bảng 3.1. USER - Người dùng	40
Bảng 3.2. PRODUCTS - Sản phẩm	40
Bảng 3.3. ORDERS - Đơn hàng	41
Bảng 3.4. ORDER_ITEMS - Chi tiết đơn hàng	41
Bảng 3.5. ORDER_STATUS_HISTORY - Lịch sử trạng thái đơn	41
Bảng 3.6. PRODUCT_REVIEWS - Đánh giá sản phẩm	42

TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

Đề tài “Xây dựng hệ thống bán thức ăn nhanh” được thực hiện nhằm xây dựng một website hỗ trợ quản lý và đặt món ăn trực tuyến cho các cửa hàng thức ăn nhanh quy mô nhỏ. Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, nhu cầu đặt món online ngày càng phổ biến, việc xây dựng một hệ thống bán hàng riêng giúp cửa hàng chủ động quản lý sản phẩm, đơn hàng và thông tin khách hàng một cách hiệu quả hơn.

Đề tài được triển khai dựa trên việc khảo sát nhu cầu thực tế của cửa hàng, phân tích yêu cầu hệ thống và áp dụng các kiến thức về lập trình web. Hệ thống được xây dựng bằng ngôn ngữ PHP kết hợp với cơ sở dữ liệu MySQL, triển khai trên môi trường XAMPP và sử dụng HTML, CSS để thiết kế giao diện người dùng.

Kết quả đạt được là xây dựng thành công một hệ thống bán thức ăn nhanh online với các chức năng cơ bản như xem danh sách món ăn, quản lý giỏ hàng, đặt hàng và trang quản trị. Hệ thống hoạt động ổn định, giao diện đơn giản, dễ sử dụng, đáp ứng được mục tiêu đề ra và có thể tiếp tục mở rộng trong tương lai.

MỞ ĐẦU

1. Lý do chọn đề tài

Trong thời đại công nghệ số hiện nay, việc ứng dụng công nghệ thông tin vào hoạt động kinh doanh ngày càng trở nên phổ biến, đặc biệt trong lĩnh vực kinh doanh thức ăn nhanh. Nhu cầu đặt món ăn trực tuyến của khách hàng ngày càng tăng nhờ sự tiện lợi, nhanh chóng và dễ tiếp cận trên các thiết bị thông minh.

Tuy nhiên, nhiều cửa hàng thức ăn nhanh quy mô nhỏ vẫn chưa có hệ thống bán hàng trực tuyến riêng, chủ yếu quản lý thủ công hoặc phụ thuộc vào các nền tảng trung gian, gây khó khăn trong việc quản lý sản phẩm, đơn hàng và thông tin khách hàng. Vì vậy, việc xây dựng một hệ thống bán thức ăn nhanh online đơn giản, dễ sử dụng và chi phí thấp là rất cần thiết. Xuất phát từ thực tế đó, đề tài “*Xây dựng hệ thống bán thức ăn nhanh*” được lựa chọn để nghiên cứu và thực hiện.

2. Mục đích nghiên cứu

Xây dựng một website bán thức ăn nhanh online hoạt động ổn định trên nền tảng PHP và MySQL, đáp ứng nhu cầu đặt món trực tuyến của khách hàng.

Hỗ trợ người dùng xem món ăn, đặt hàng và quản lý giỏ hàng một cách thuận tiện, góp phần nâng cao trải nghiệm người dùng.

Xây dựng trang quản trị giúp cửa hàng dễ dàng quản lý sản phẩm, đơn hàng và cập nhật thông tin một cách hiệu quả.

Rèn luyện kỹ năng phân tích, thiết kế và lập trình một hệ thống web hoàn chỉnh, đồng thời vận dụng kiến thức đã học vào thực tế.

3. Nghiên cứu đối tượng

Hệ thống bán thức ăn nhanh online phục vụ cho hoạt động kinh doanh của cửa hàng thức ăn nhanh quy mô nhỏ.

Người dùng là khách hàng truy cập website để xem thực đơn, đặt món và theo dõi đơn hàng.

Quản trị viên là người thực hiện quản lý hệ thống, bao gồm quản lý sản phẩm, đơn hàng và thông tin liên quan.

Các công nghệ sử dụng trong quá trình nghiên cứu và xây dựng hệ thống gồm PHP, MySQL, XAMPP, HTML5 và CSS.

4. Phạm vi nghiên cứu

Đề tài tập trung xây dựng hệ thống bán thức ăn nhanh với các chức năng cơ bản như xem danh sách món ăn, quản lý giỏ hàng và đặt hàng trực tuyến.

Hệ thống được triển khai và chạy trên môi trường localhost (XAMPP), phù hợp cho mục đích học tập và nghiên cứu.

Các chức năng nâng cao như thanh toán trực tuyến, bảo mật chuyên sâu và tích hợp các dịch vụ bên thứ ba chưa được triển khai trong phạm vi đề tài.

CHƯƠNG 1: TỔNG QUAN

1.1. Đặt vấn đề

Trong thời đại công nghệ số hiện nay, việc ứng dụng công nghệ thông tin vào hoạt động kinh doanh ngày càng trở nên phổ biến và cần thiết. Đặc biệt trong lĩnh vực kinh doanh thức ăn nhanh, nhu cầu đặt món trực tuyến của khách hàng ngày càng tăng cao nhờ sự tiện lợi, nhanh chóng và dễ sử dụng trên các thiết bị di động.

Tuy nhiên, nhiều cửa hàng thức ăn nhanh nhỏ lẻ vẫn chưa có hệ thống bán hàng trực tuyến riêng mà chủ yếu quản lý thủ công hoặc phụ thuộc vào các nền tảng trung gian. Điều này gây khó khăn trong việc quản lý món ăn, đơn hàng, khách hàng cũng như theo dõi doanh thu, đồng thời làm giảm khả năng kiểm soát và hiệu quả kinh doanh của cửa hàng.

Xuất phát từ nhu cầu thực tế đó, việc xây dựng một website bán thức ăn nhanh online giúp khách hàng dễ dàng xem thực đơn, đặt món và theo dõi đơn hàng, đồng thời hỗ trợ quản trị viên trong việc quản lý hệ thống một cách hiệu quả là vô cùng cần thiết. Vì vậy, đề tài “Xây dựng hệ thống bán thức ăn nhanh” được lựa chọn nhằm đáp ứng nhu cầu trên và áp dụng kiến thức đã học vào thực tế.

1.2 .Mục đích nguyên cứu

Mục đích của đề tài là xây dựng một hệ thống website bán thức ăn nhanh online hoạt động ổn định, có giao diện thân thiện và dễ sử dụng cho người dùng. Hệ thống cho phép khách hàng xem danh sách món ăn, quản lý giỏ hàng, đặt món và theo dõi trạng thái đơn hàng một cách thuận tiện.

Bên cạnh đó, đề tài hướng đến việc xây dựng trang quản trị dành cho cửa hàng nhằm quản lý món ăn, đơn hàng, khách hàng và theo dõi doanh thu tổng hợp. Hệ thống được phát triển bằng PHP kết hợp với MySQL và triển khai trên môi trường XAMPP, phù hợp với mô hình cửa hàng vừa và nhỏ.

Thông qua đề tài, sinh viên có cơ hội vận dụng kiến thức đã học để xây dựng một hệ thống web hoàn chỉnh, đồng thời rèn luyện kỹ năng phân tích yêu cầu, thiết kế cơ sở dữ liệu và lập trình hệ thống theo đúng quy trình phát triển phần mềm.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1. Giới thiệu về ngôn ngữ PHP

2.1.1. Tổng quan về ngôn ngữ PHP

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình kịch bản chạy phía máy chủ (server-side scripting language), được sử dụng rộng rãi trong việc phát triển các ứng dụng web động. Ngôn ngữ này được giới thiệu lần đầu vào năm 1995 bởi Rasmus Lerdorf và đến nay đã trở thành một trong những ngôn ngữ phổ biến nhất trong lĩnh vực lập trình web nhờ tính đơn giản, linh hoạt và khả năng ứng dụng cao.

PHP hoạt động theo mô hình xử lý phía máy chủ, nghĩa là toàn bộ mã nguồn PHP sẽ được thực thi trên máy chủ web. Sau khi xử lý dữ liệu và các yêu cầu từ phía người dùng, PHP sẽ sinh ra mã HTML và gửi về trình duyệt để hiển thị. Cách thức hoạt động này giúp tăng tính bảo mật cho hệ thống, đồng thời cho phép xây dựng các website có khả năng tương tác cao với người dùng.

Một trong những ưu điểm nổi bật của PHP là khả năng kết nối và làm việc hiệu quả với các hệ quản trị cơ sở dữ liệu, đặc biệt là MySQL. Sự kết hợp giữa PHP và MySQL cho phép xây dựng các hệ thống quản lý dữ liệu mạnh mẽ như website thương mại điện tử, hệ thống bán hàng trực tuyến, quản lý người dùng, giỏ hàng và đơn hàng. Ngoài ra, PHP còn hỗ trợ nhiều hệ quản trị cơ sở dữ liệu khác như PostgreSQL, SQLite, Oracle,...

PHP có cú pháp đơn giản, dễ học, gần gũi với các ngôn ngữ lập trình phổ biến như C và Java, nên phù hợp với người mới bắt đầu tiếp cận lập trình web. Bên cạnh đó, PHP có thể nhúng trực tiếp vào mã HTML, giúp việc xây dựng giao diện và xử lý dữ liệu trở nên thuận tiện hơn.

Ngoài tính đơn giản, PHP còn có khả năng mở rộng tốt nhờ hệ sinh thái thư viện và framework phong phú như Laravel, CodeIgniter, Symfony,... Các framework này hỗ trợ tổ chức mã nguồn theo mô hình MVC (Model – View – Controller), giúp tăng tính bảo trì, mở rộng và bảo mật cho hệ thống.

PHP dễ dàng triển khai trên nhiều hệ điều hành khác nhau như Windows, Linux và macOS. Việc sử dụng môi trường XAMPP giúp người phát triển nhanh chóng cài đặt máy chủ web, PHP và MySQL trên máy cá nhân để phát triển và kiểm thử ứng dụng. Điều này đặc biệt phù hợp với các đề tài học tập và nghiên cứu.

Hiện nay, PHP vẫn được sử dụng rộng rãi trong thực tế. Nhiều hệ thống lớn như WordPress, Wikipedia và các phiên bản đầu của Facebook đều được xây dựng bằng PHP. Điều này cho thấy PHP vẫn giữ vai trò quan trọng trong lĩnh vực phát triển web và là lựa chọn phù hợp cho các dự án vừa và nhỏ.

Trong đề tài “Xây dựng hệ thống bán thức ăn nhanh”, PHP được sử dụng làm ngôn ngữ lập trình chính cho phần backend, đảm nhiệm các chức năng xử lý dữ liệu, quản lý người dùng, đơn hàng và tương tác với cơ sở dữ liệu, góp phần xây dựng một hệ thống web hoàn chỉnh và hiệu quả.

2.1.2. Các Thành Phần của ngôn ngữ PHP

Ngôn ngữ PHP bao gồm nhiều thành phần khác nhau, phối hợp chặt chẽ với nhau nhằm tạo nên một môi trường lập trình web mạnh mẽ và linh hoạt. Mỗi thành phần đảm nhận một vai trò riêng trong quá trình xử lý và thực thi mã nguồn PHP. Các thành phần chính của ngôn ngữ PHP bao gồm:

PHP Core: PHP Core là phần lõi của ngôn ngữ PHP, cung cấp các chức năng cơ bản như xử lý biến, toán tử, vòng lặp, cấu trúc điều kiện, quản lý bộ nhớ và các tính năng cốt lõi khác. Đây là nền tảng bắt buộc để mọi chương trình PHP có thể hoạt động.

PHP Parser: PHP Parser có nhiệm vụ đọc và phân tích cú pháp của mã nguồn PHP, kiểm tra tính hợp lệ của mã lệnh và chuyển đổi mã nguồn thành các cấu trúc trung gian để chuẩn bị cho quá trình thực thi. Mọi đoạn mã PHP đều phải được phân tích cú pháp trước khi chạy.

Zend Engine: Zend Engine được xem là “trái tim” của PHP, đóng vai trò là bộ máy thực thi mã nguồn. Thành phần này chịu trách nhiệm biên dịch mã PHP sang dạng bytecode, tối ưu hóa và thực thi các lệnh, đảm bảo hiệu suất và tính ổn định của hệ thống.

PHP Extensions: PHP cung cấp hệ thống mở rộng (extensions) phong phú cho phép bổ sung nhiều chức năng nâng cao như xử lý hình ảnh, mã hóa dữ liệu, thao tác với chuỗi, làm việc với XML và tương tác với cơ sở dữ liệu. Nhờ các extension, lập trình viên có thể mở rộng khả năng của PHP mà không cần xây dựng lại từ đầu.

Web Server Integration: PHP có khả năng tích hợp chặt chẽ với các máy chủ web phổ biến như Apache, Nginx và IIS thông qua các module hoặc giao thức FastCGI. Sự tích hợp này cho phép PHP tiếp nhận và xử lý các yêu cầu HTTP từ trình duyệt, sau đó tạo ra nội dung web động trả về cho người dùng.

Database Support: PHP hỗ trợ nhiều hệ quản trị cơ sở dữ liệu khác nhau như MySQL, PostgreSQL, SQLite, Oracle... Thông qua các extension cơ sở dữ liệu, PHP có thể thực hiện các thao tác kết nối, truy vấn, cập nhật và quản lý dữ liệu một cách hiệu quả.

Standard PHP Library (SPL): Standard PHP Library (SPL) cung cấp tập hợp các lớp và interface chuẩn nhằm hỗ trợ các cấu trúc dữ liệu, iterator và truy xuất dữ liệu. SPL giúp lập trình viên xây dựng chương trình hiệu quả hơn, giảm thiểu việc phải tự viết lại các chức năng cơ bản.

PHP Configuration (php.ini): php.ini là tập tin cấu hình chính của PHP, cho phép thiết lập các thông số quan trọng như giới hạn bộ nhớ, thời gian thực thi, cấu hình upload file, bật hoặc tắt extension, thiết lập báo lỗi và quản lý session. Việc cấu hình hợp lý giúp tối ưu hiệu suất và nâng cao tính bảo mật cho ứng dụng.

PHP Community và Ecosystem: PHP sở hữu cộng đồng phát triển rộng lớn với nhiều tài liệu, diễn đàn, framework phổ biến như Laravel, Symfony, CodeIgniter cùng hệ sinh thái thư viện phong phú thông qua Composer. Cộng đồng và hệ sinh thái này giúp lập trình viên học hỏi, chia sẻ kinh nghiệm và đẩy nhanh quá trình phát triển ứng dụng.

2.1.3. Ưu nhược điểm của ngôn ngữ lập trình PHP

Ngôn ngữ lập trình PHP là một trong những ngôn ngữ phổ biến trong phát triển web hiện nay. Bên cạnh những ưu điểm nổi bật, PHP cũng tồn tại một số hạn chế nhất định.

Ưu điểm của PHP:

Mã nguồn mở và miễn phí: PHP là ngôn ngữ lập trình mã nguồn mở, không yêu cầu chi phí bản quyền, giúp tiết kiệm chi phí phát triển và triển khai dự án. Người dùng có thể tải về và sử dụng PHP hoàn toàn miễn phí mà không cần trả phí cấp phép.

Dễ học và dễ sử dụng: PHP có cú pháp đơn giản, dễ tiếp cận đối với người mới bắt đầu. Những lập trình viên đã có kiến thức về các ngôn ngữ như C hoặc Java có thể nhanh chóng làm quen và viết được chương trình PHP trong thời gian ngắn.

Tích hợp mạnh với cơ sở dữ liệu: PHP hỗ trợ kết nối với nhiều hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, SQLite..., giúp việc xây dựng và xử lý dữ liệu trong các ứng dụng web trở nên linh hoạt và hiệu quả.

Khả năng chạy đa nền tảng: PHP có thể hoạt động trên nhiều hệ điều hành khác nhau như Windows, Linux và macOS, giúp dễ dàng triển khai ứng dụng trên nhiều môi trường khác nhau.

Hiệu suất xử lý tốt cho ứng dụng web: PHP được tối ưu cho các ứng dụng web, có khả năng xử lý yêu cầu HTTP nhanh và sinh nội dung HTML hiệu quả, góp phần cải thiện tốc độ tải trang.

Cộng đồng lớn và tài liệu phong phú: Với cộng đồng lập trình viên đông đảo trên toàn thế giới, PHP có hệ thống tài liệu, diễn đàn và hướng dẫn phong phú, giúp người học và nhà phát triển dễ dàng tìm kiếm tài nguyên và giải pháp kỹ thuật.

Thư viện và framework mạnh mẽ: PHP được hỗ trợ bởi nhiều thư viện và framework phổ biến như Laravel, Symfony, CodeIgniter, giúp tăng tốc quá trình phát triển ứng dụng và tổ chức mã nguồn theo kiến trúc rõ ràng.

Được sử dụng rộng rãi trong phát triển web: PHP là ngôn ngữ server-side phổ biến và được sử dụng trong nhiều hệ thống lớn như WordPress, phù hợp với nhiều dự án web từ quy mô nhỏ đến lớn.

Khả năng nhúng vào HTML linh hoạt: PHP có thể nhúng trực tiếp vào mã HTML, giúp lập trình viên dễ dàng kết hợp giao diện và xử lý dữ liệu mà không cần công cụ phức tạp.

Nhược điểm của PHP:

Vấn đề bảo mật: Do là ngôn ngữ mã nguồn mở, nếu lập trình không cẩn thận, ứng dụng PHP có thể gặp các lỗ hổng bảo mật như SQL Injection, XSS hoặc tấn công session nếu không áp dụng đúng các biện pháp bảo mật.

Cú pháp và thiết kế chưa nhất quán: PHP tồn tại nhiều hàm với cách đặt tên và tham số không đồng nhất, gây khó khăn cho người mới học trong việc ghi nhớ và sử dụng.

Hiệu suất chưa tối ưu cho ứng dụng lớn: Mặc dù PHP hoạt động tốt với các ứng dụng web vừa và nhỏ, nhưng với các hệ thống lớn, xử lý tải cao, hiệu suất có thể kém hơn so với một số ngôn ngữ server-side khác như Java hoặc Node.js.

Hạn chế trong việc xây dựng ứng dụng ngoài web: PHP chủ yếu phục vụ phát triển web, không phù hợp để xây dựng các ứng dụng desktop hoặc các hệ thống phức tạp ngoài môi trường web.

Quản lý và gỡ lỗi còn phức tạp: PHP thiếu các công cụ gỡ lỗi mạnh mẽ so với một số ngôn ngữ khác, khiến việc bảo trì và xử lý lỗi trong các dự án lớn gặp nhiều khó khăn.

Khả năng xử lý đa luồng hạn chế: PHP không hỗ trợ tốt xử lý đa luồng như các ngôn ngữ như Java, do đó không phải là lựa chọn tối ưu cho các ứng dụng yêu cầu xử lý đồng thời cao.

Không phù hợp cho các lĩnh vực phi web: Đối với các lĩnh vực như trí tuệ nhân tạo, phân tích dữ liệu lớn hay khoa học dữ liệu, PHP không phải là ngôn ngữ được ưu tiên lựa chọn.

2.1.4 .Ứng dụng của ngôn ngữ PHP trong lập trình

Ngôn ngữ PHP được sử dụng rộng rãi trong phát triển các ứng dụng web động và có khả năng triển khai đa dạng trong nhiều lĩnh vực khác nhau của lập trình hiện nay. PHP không chỉ là lựa chọn phổ biến cho các ứng dụng web đơn giản mà còn được ứng dụng trong các hệ thống lớn, phục vụ nhu cầu thực tế và thương mại.

Trước hết, PHP được ứng dụng mạnh mẽ trong phát triển các hệ thống quản lý nội dung (CMS – Content Management Systems). Các nền tảng CMS phổ biến như WordPress, Joomla, Drupal đều được xây dựng bằng PHP, giúp người dùng dễ dàng tạo, quản lý và xuất bản nội dung mà không cần nhiều kiến thức kỹ thuật. Những hệ thống này chiếm một phần lớn trong số các website trên internet hiện nay.

Bên cạnh đó, PHP được sử dụng để xây dựng các hệ thống thương mại điện tử (e-commerce platforms). Nhiều giải pháp thương mại điện tử mạnh mẽ như Magento, WooCommerce, OpenCart được phát triển dựa trên PHP, cung cấp chức năng quản lý sản phẩm, giỏ hàng, thanh toán và đặt hàng cho các cửa hàng trực tuyến.

PHP cũng phù hợp để phát triển ứng dụng web tương tác người dùng, bao gồm diễn đàn, mạng xã hội, hệ thống quản lý quan hệ khách hàng (CRM), hệ thống đặt vé, trang quản lý nội bộ và các ứng dụng mà yêu cầu xử lý dữ liệu phức tạp từ phía người dùng. Khả năng xử lý form, session, cookie và cơ sở dữ liệu khiến PHP trở thành ngôn ngữ lý tưởng cho các ứng dụng như vậy.

Ngoài ra, PHP còn được ứng dụng trong phát triển API và dịch vụ web, cho phép các ứng dụng khác hoặc frontend tương tác với backend thông qua các giao thức như RESTful API. Điều này giúp PHP không chỉ phục vụ website mà còn tương tác được với các ứng dụng di động hay hệ thống SaaS.

Một số ứng dụng khác của PHP bao gồm xử lý tương tác cơ sở dữ liệu, sử dụng các framework để phát triển nhanh, tối ưu và bảo trì mã nguồn, cũng như xây dựng các công cụ quản lý nội bộ cho doanh nghiệp. PHP còn có thể dùng để tạo các script chạy trên dòng lệnh (CLI – Command Line Interface) trong các tác vụ tự động hoặc định kỳ.

Như vậy, nhờ tính linh hoạt, dễ tiếp cận, khả năng xử lý dữ liệu và cộng đồng phát triển mạnh, PHP trở thành một ngôn ngữ phổ biến trong lập trình web hiện đại, đặc biệt phù hợp với các dự án thương mại điện tử, hệ thống nội dung động và các ứng dụng web tương tác.

2.1.5. Sự khác biệt giữa PHP với các ngôn ngữ lập trình khác

Ngôn ngữ PHP có nhiều điểm đặc trưng khiến nó khác biệt rõ rệt so với các ngôn ngữ lập trình khác, đặc biệt trong bối cảnh phát triển web. Dưới đây là những điểm khác biệt chính giữa PHP và một số ngôn ngữ phổ biến hiện nay:

PHP và Python: PHP tập trung chính vào phát triển ứng dụng web động, trong khi Python là ngôn ngữ lập trình tổng quát, được sử dụng rộng rãi trong nhiều lĩnh vực như khoa học dữ liệu, trí tuệ nhân tạo và tự động hóa. Python có cú pháp đơn giản, dễ đọc và hỗ trợ nhiều kiểu lập trình hơn, trong khi PHP tối ưu cho việc sinh nội dung web và gắn kết chặt với HTML.

PHP và JavaScript: PHP là ngôn ngữ chạy phía máy chủ (*server-side*), còn JavaScript ban đầu là ngôn ngữ chạy phía khách (*client-side*). Hiện JavaScript có thể chạy backend qua Node.js, nhưng PHP vẫn giữ thế mạnh về xử lý web truyền thống và hệ thống CMS lớn.

PHP và Java: Java là ngôn ngữ lập trình mạnh mẽ, hướng đối tượng, thường dùng trong các hệ thống doanh nghiệp lớn với yêu cầu mở rộng cao. PHP nhẹ hơn và phát triển nhanh hơn cho các ứng dụng web vừa và nhỏ, không yêu cầu khai báo kiểu dữ liệu cứng như Java.

PHP và Ruby: Cả PHP và Ruby đều được sử dụng phát triển web, nhưng Ruby nổi bật với cú pháp tinh tế và framework Rails giúp phát triển nhanh. PHP vẫn phổ biến hơn nhờ hỗ trợ rộng rãi, hosting dễ triển khai và cộng đồng lớn.

Cú pháp và cách viết: PHP có cú pháp giống phong cách C, dễ kết hợp với HTML, giúp lập trình viên mới tiếp cận nhanh. Một số ngôn ngữ khác như Python theo đuổi cú pháp đơn giản hơn nữa, còn Java yêu cầu cấu trúc phức tạp, nhiều khai báo hơn.

Mục tiêu sử dụng: PHP được thiết kế chính cho lập trình web động, xử lý request HTTP và sinh nội dung HTML. Các ngôn ngữ khác như Python hay Java không hạn chế vào web mà còn phục vụ khai thác dữ liệu, ứng dụng doanh nghiệp, di động, game...

Quá trình thực thi: PHP được thực thi phía máy chủ mỗi khi có yêu cầu từ trình duyệt, sinh ra HTML trả về client. JavaScript chạy phía client và xử lý tương tác trực tiếp trên trình duyệt; Python và Java có thể được biên dịch hoặc thông dịch tùy môi trường.

Cộng đồng và hệ sinh thái: PHP có cộng đồng lớn với nhiều CMS (như WordPress, Joomla...), framework (Laravel, Symfony...) và tài liệu hỗ trợ. Python cũng có cộng đồng mạnh với nhiều thư viện cho đa mục đích hơn. Java và JavaScript có hệ sinh thái phong phú nhưng phục vụ rộng hơn phạm vi web.

Phạm vi ứng dụng: PHP ưu thế trong xây dựng website, hệ thống thương mại điện tử và CMS. Các ngôn ngữ khác như Python mạnh trong phân tích dữ liệu, Java phù hợp hệ thống lớn, còn JavaScript nổi bật ở tương tác frontend.

2.2. Gợi thiệu về SQL

2.2.1. Tổng quan về MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) được sử dụng rộng rãi trong phát triển ứng dụng web và các hệ thống quản lý dữ liệu. Đây là một phần mềm mã nguồn mở cho phép lưu trữ, truy xuất, cập nhật và quản lý dữ liệu trong một hoặc nhiều bảng có liên hệ với nhau thông qua ngôn ngữ truy vấn có cấu trúc SQL (Structured Query Language). MySQL hỗ trợ môi trường đa người dùng – đa luồng, giúp nhiều người dùng có thể truy cập và xử lý dữ liệu đồng thời mà không ảnh hưởng tới hiệu năng hoạt động chung của hệ thống.

MySQL được thiết kế với kiến trúc client-server, trong đó máy chủ MySQL (MySQL Server) đảm nhiệm việc quản lý và xử lý các truy vấn dữ liệu, còn client (người dùng hoặc ứng dụng) có thể kết nối tới server thông qua các giao thức tiêu chuẩn để thực hiện các thao tác như thêm, sửa, xóa hay truy vấn dữ liệu.

Một trong những ưu điểm lớn của MySQL là khả năng chạy trên nhiều nền tảng hệ điều hành như Windows, Linux, macOS và nhiều hệ UNIX khác, giúp người phát triển dễ dàng triển khai ứng dụng trên môi trường phù hợp. MySQL cũng sử dụng

ngôn ngữ SQL – một ngôn ngữ chuẩn để truy vấn và thao tác dữ liệu nên người dùng có thể dễ dàng viết các câu lệnh truy vấn để xử lý dữ liệu theo nhu cầu.

MySQL hỗ trợ nhiều kiểu dữ liệu khác nhau, bao gồm số nguyên, số thực, chuỗi ký tự, ngày/giờ và dạng dữ liệu mở rộng như BLOB hay ENUM, giúp đáp ứng nhu cầu lưu trữ và xử lý đa dạng của các loại ứng dụng. Ngoài ra, MySQL cũng cung cấp các cơ chế như transaction (giao dịch), khóa hàng, chỉ mục (index) và tính toán vện dữ liệu, giúp đảm bảo hoạt động xử lý dữ liệu được chính xác, nhất quán và an toàn.

Do tính ổn định, hiệu quả và dễ sử dụng, MySQL là lựa chọn phổ biến trong các ứng dụng web, đặc biệt khi kết hợp với ngôn ngữ lập trình như PHP trong mô hình LAMP (Linux, Apache, MySQL và PHP) – một trong những nền tảng phát triển web phổ biến nhất hiện nay. MySQL thường được dùng để xây dựng hệ thống quản lý nội dung (CMS), thương mại điện tử, hệ thống bán hàng trực tuyến, quản lý khách hàng và nhiều ứng dụng khác.

Tóm lại, MySQL là một hệ quản trị cơ sở dữ liệu mạnh mẽ, linh hoạt, có khả năng xử lý lượng lớn dữ liệu và hỗ trợ tốt cho các ứng dụng web hiện đại. Với cộng đồng người dùng rộng lớn và tài liệu phong phú, MySQL giúp việc phát triển, triển khai và quản lý dữ liệu trở nên dễ dàng và hiệu quả hơn.

2.2.2. Các thành phần của MySQL

MySQL được xây dựng theo kiến trúc client – server, bao gồm nhiều thành phần phối hợp chặt chẽ với nhau nhằm quản lý, xử lý và lưu trữ dữ liệu một cách hiệu quả. Mỗi thành phần đảm nhận một vai trò riêng trong quá trình vận hành của hệ quản trị cơ sở dữ liệu MySQL. Các thành phần chính của MySQL bao gồm:

MySQL Server: MySQL Server là thành phần trung tâm của hệ quản trị cơ sở dữ liệu MySQL. Thành phần này chịu trách nhiệm tiếp nhận các yêu cầu truy vấn từ người dùng hoặc ứng dụng, xử lý các câu lệnh SQL và trả về kết quả tương ứng. Ngoài ra, MySQL Server còn quản lý các hoạt động như lưu trữ, truy xuất, cập nhật và bảo mật dữ liệu.

Client Programs: Client Programs là các chương trình cho phép người dùng hoặc ứng dụng kết nối và làm việc với MySQL Server. Một số client phổ biến gồm MySQL Command Line Client, phpMyAdmin và các ứng dụng web được viết bằng

các ngôn ngữ như PHP, Java, Python. Thông qua các client này, người dùng có thể gửi các câu lệnh SQL để thao tác với cơ sở dữ liệu.

SQL Interface: SQL Interface là thành phần tiếp nhận và xử lý các câu lệnh SQL như SELECT, INSERT, UPDATE, DELETE. Thành phần này có nhiệm vụ phân tích cú pháp, kiểm tra tính hợp lệ của câu lệnh và chuyển yêu cầu đến các bộ phận xử lý bên trong MySQL Server.

Query Optimizer: Query Optimizer chịu trách nhiệm tối ưu hóa các câu truy vấn SQL trước khi thực thi. Thành phần này lựa chọn phương án thực thi hiệu quả nhất dựa trên chỉ mục, cấu trúc bảng và dữ liệu hiện có, từ đó giúp tăng tốc độ truy vấn và giảm thời gian xử lý.

Storage Engine: Storage Engine là thành phần quyết định cách dữ liệu được lưu trữ, truy xuất và quản lý trong MySQL. MySQL hỗ trợ nhiều storage engine khác nhau, trong đó phổ biến nhất là InnoDB (hỗ trợ giao dịch, khóa hàng và đảm bảo toàn vẹn dữ liệu) và MyISAM (có tốc độ đọc nhanh, phù hợp với dữ liệu ít thay đổi). Nhờ đó, MySQL có thể linh hoạt lựa chọn phương thức lưu trữ phù hợp với từng ứng dụng.

Buffer Manager: Buffer Manager quản lý bộ nhớ đệm của MySQL, cho phép lưu trữ tạm thời các dữ liệu và truy vấn thường xuyên được sử dụng. Việc sử dụng bộ nhớ đệm giúp giảm số lần truy cập vào ổ đĩa, từ đó nâng cao hiệu suất của hệ thống.

Transaction Manager: Transaction Manager chịu trách nhiệm quản lý các giao dịch trong MySQL và đảm bảo các tính chất ACID (Atomicity, Consistency, Isolation, Durability). Thành phần này giúp dữ liệu luôn nhất quán và an toàn trong trường hợp xảy ra lỗi hoặc khi có nhiều truy cập đồng thời.

Security & Authentication: Thành phần bảo mật và xác thực quản lý quyền truy cập của người dùng vào hệ thống cơ sở dữ liệu. MySQL hỗ trợ phân quyền theo người dùng, bảng, cột và cơ sở dữ liệu, giúp đảm bảo dữ liệu chỉ được truy cập bởi các đối tượng được cấp phép.

Data Directory: Data Directory là nơi lưu trữ vật lý các tập tin dữ liệu của MySQL trên máy chủ. Mỗi cơ sở dữ liệu và bảng dữ liệu được lưu dưới dạng các file riêng biệt, giúp MySQL dễ dàng quản lý, sao lưu và phục hồi dữ liệu khi cần thiết.

2.2.3 .Ưu nhược điểm của MySQL

Ưu điểm của MySQL:

Mã nguồn mở và miễn phí: MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở, cho phép người dùng sử dụng hoàn toàn miễn phí mà không cần chi trả chi phí bản quyền. Điều này giúp tiết kiệm đáng kể chi phí trong quá trình phát triển và triển khai các hệ thống phần mềm, đặc biệt phù hợp với sinh viên, doanh nghiệp vừa và nhỏ.

Tương thích đa nền tảng: MySQL có thể hoạt động trên nhiều hệ điều hành khác nhau như Windows, Linux, Unix và macOS. Nhờ đó, hệ thống dễ dàng được triển khai trên nhiều môi trường khác nhau mà không gặp trở ngại về tính tương thích.

Hiệu suất xử lý cao: MySQL có khả năng xử lý truy vấn nhanh và ổn định, đặc biệt hiệu quả đối với các ứng dụng web có tần suất truy vấn đọc dữ liệu lớn. Điều này giúp cải thiện tốc độ phản hồi và nâng cao trải nghiệm người dùng.

Dễ học và dễ sử dụng: MySQL sử dụng cú pháp SQL chuẩn, cấu trúc rõ ràng, dễ tiếp cận đối với người mới bắt đầu học cơ sở dữ liệu. Bên cạnh đó, các công cụ hỗ trợ như MySQL Workbench giúp việc thiết kế, quản lý và truy vấn dữ liệu trở nên trực quan và thuận tiện hơn.

Cộng đồng lớn và tài liệu phong phú: MySQL có cộng đồng người dùng rộng lớn trên toàn thế giới, cùng với hệ thống tài liệu, diễn đàn và hướng dẫn học tập đa dạng. Điều này giúp người học và lập trình viên dễ dàng tìm kiếm tài liệu tham khảo và giải pháp khi gặp vấn đề kỹ thuật.

Hỗ trợ tốt cho phát triển ứng dụng web: MySQL hỗ trợ đầy đủ ngôn ngữ truy vấn SQL và dễ dàng tích hợp với nhiều ngôn ngữ lập trình như PHP, Java, Python. Nhờ đó, MySQL được sử dụng phổ biến trong các hệ thống website và ứng dụng web động.

Độ ổn định và tin cậy cao: MySQL đã được sử dụng trong thời gian dài và được kiểm chứng về độ ổn định, phù hợp với các hệ thống yêu cầu xử lý dữ liệu liên tục và hoạt động lâu dài.

Nhược điểm của MySQL:

Khả năng mở rộng còn hạn chế: Khi xử lý các hệ thống có dữ liệu cực lớn hoặc yêu cầu mở rộng theo chiều ngang (horizontal scaling), MySQL có thể gặp khó khăn và cần nhiều giải pháp bổ sung để đáp ứng nhu cầu.

Thiếu một số tính năng nâng cao: So với một số hệ quản trị cơ sở dữ liệu khác như PostgreSQL hoặc Oracle, MySQL còn hạn chế ở một số tính năng nâng cao như xử lý dữ liệu phức tạp, phân tích chuyên sâu hoặc hỗ trợ một số kiểu dữ liệu hiện đại.

Hiệu năng ghi dữ liệu lớn chưa tối ưu: Trong các hệ thống có khối lượng ghi dữ liệu lớn và liên tục, hiệu suất của MySQL có thể giảm nếu không được cấu hình và tối ưu hợp lý.

Cấu hình phức tạp đối với hệ thống lớn: Việc triển khai các chức năng nâng cao như sao chép dữ liệu (replication), clustering hoặc hệ thống phân tán yêu cầu kiến thức chuyên môn cao và quá trình cấu hình phức tạp.

Hạn chế trong công cụ gỡ lỗi: MySQL chưa có nhiều công cụ hỗ trợ gỡ lỗi và phân tích chuyên sâu như một số hệ quản trị cơ sở dữ liệu thương mại, gây khó khăn trong việc theo dõi và xử lý lỗi ở các hệ thống lớn.

Cấu trúc schema cố định: MySQL sử dụng mô hình bảng với cấu trúc cố định, yêu cầu xác định rõ schema ngay từ đầu, điều này gây khó khăn khi dữ liệu thay đổi thường xuyên hoặc không có cấu trúc rõ ràng.

2.2.4 Một số tính năng nổi bật của MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) phổ biến trên thế giới, được sử dụng rộng rãi trong các ứng dụng web và hệ thống thông tin. MySQL sở hữu nhiều tính năng mạnh mẽ, linh hoạt, đáp ứng tốt nhu cầu từ các website nhỏ đến các hệ thống lớn như thương mại điện tử hay quản lý doanh nghiệp. Các tính năng nổi bật của MySQL bao gồm:

Hỗ trợ đa ngôn ngữ lập trình và đa nền tảng: MySQL tương thích với nhiều ngôn ngữ lập trình phổ biến như PHP, Java, Python, C++, Perl... và có thể hoạt động trên nhiều hệ điều hành như Windows, Linux, macOS. Điều này giúp MySQL dễ dàng tích hợp vào nhiều môi trường và hệ thống khác nhau.

Hỗ trợ SQL chuẩn và truy vấn phong phú: MySQL sử dụng ngôn ngữ truy vấn SQL chuẩn, cho phép thực hiện các thao tác truy vấn dữ liệu như SELECT, INSERT,

UPDATE, DELETE, JOIN, nhóm dữ liệu, sắp xếp và các truy vấn phức tạp khác, đáp ứng tốt nhu cầu xử lý và khai thác dữ liệu.

Đảm bảo tính toàn vẹn dữ liệu và hỗ trợ giao dịch (ACID): MySQL hỗ trợ đầy đủ các đặc tính của giao dịch bao gồm tính nguyên tử (Atomicity), tính nhất quán (Consistency), tính cô lập (Isolation) và tính bền vững (Durability). Nhờ đó, dữ liệu luôn được đảm bảo chính xác và an toàn ngay cả khi xảy ra lỗi hệ thống.

Quản lý bảo mật và phân quyền hiệu quả: MySQL cung cấp các cơ chế bảo mật như xác thực người dùng, phân quyền chi tiết theo cơ sở dữ liệu, bảng và từng thao tác cụ thể. Điều này giúp kiểm soát quyền truy cập và bảo vệ dữ liệu trước các truy cập trái phép.

Hiệu suất cao và khả năng mở rộng linh hoạt: MySQL có tốc độ xử lý nhanh và khả năng mở rộng tốt, phù hợp với các hệ thống có lượng dữ liệu lớn. Người dùng có thể mở rộng hệ thống thông qua các cơ chế như sao chép dữ liệu (replication), phân vùng bảng và cấu hình cluster để đáp ứng nhu cầu sử dụng ngày càng tăng.

Hỗ trợ sao lưu và phục hồi dữ liệu: MySQL cung cấp nhiều phương pháp sao lưu và khôi phục dữ liệu hiệu quả, giúp đảm bảo an toàn dữ liệu và dễ dàng phục hồi hệ thống khi xảy ra sự cố.

Hỗ trợ nhiều Storage Engine: MySQL cho phép sử dụng nhiều loại storage engine khác nhau như InnoDB, MyISAM,... Mỗi engine có ưu điểm riêng, giúp tối ưu hóa hiệu suất, quản lý chỉ mục và hỗ trợ giao dịch theo nhu cầu của từng ứng dụng.

Công cụ quản lý trực quan và tiện lợi: MySQL Workbench là công cụ quản lý mạnh mẽ, hỗ trợ thiết kế cơ sở dữ liệu, quản trị hệ thống, viết và kiểm thử câu lệnh SQL một cách trực quan, giúp người dùng thao tác dễ dàng và hiệu quả hơn.

2.3 .Tổng quan về HTML

2.3.1.Tổng quan

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu siêu văn bản được sử dụng để tạo và cấu trúc nội dung của trang web trên mạng Internet. Đây là ngôn ngữ cơ bản nhất trong phát triển web, cung cấp cấu trúc cho phần văn bản, hình ảnh, liên kết và các thành phần khác của một trang web, trước khi được bổ sung bởi các ngôn ngữ khác như CSS và JavaScript để tạo giao diện đẹp mắt và chức năng tương tác.

HTML không phải là ngôn ngữ lập trình theo nghĩa truyền thống, mà là ngôn ngữ đánh dấu giúp trình duyệt hiểu và hiển thị nội dung một cách có cấu trúc. HTML giúp định nghĩa các phần tử như tiêu đề, đoạn văn, danh sách, hình ảnh, liên kết, biểu mẫu, bảng và các thành phần khác thông qua các “thẻ” (tags), thường được viết dưới dạng `<tagname>...</tagname>`.

Một tài liệu HTML cơ bản gồm các phần chính như:

- + Phần khai báo DOCTYPE: Xác định kiểu tài liệu HTML.
- + Phần `<html>`: Gói toàn bộ nội dung trang.
- + Phần `<head>`: Chứa thông tin siêu dữ liệu như tiêu đề trang, liên kết CSS, script...
- + Phần `<body>`: Chứa nội dung chính được hiển thị cho người dùng như văn bản, hình ảnh, liên kết, bảng...

HTML được thiết kế từ đầu những năm 1990 và đã trở thành ngôn ngữ chuẩn cho mọi trang web hiện nay. Nó hỗ trợ việc tạo ra các trang web tĩnh và là nền tảng để kết hợp với CSS (Cascading Style Sheets) và JavaScript nhằm tạo ra các trang web động, hấp dẫn và tương tác hơn.

Trong hệ thống web như hệ thống bán thức ăn nhanh online, HTML đóng vai trò xác định cấu trúc giao diện người dùng, giúp trình duyệt hiển thị đúng nội dung như menu thức ăn, giỏ hàng, biểu mẫu đặt hàng... trước khi CSS và JavaScript được áp dụng để trình bày đẹp và xử lý tương tác

2.3.2 .Mối quan hệ giữa HTML, CSS và JavaScript

HTML, CSS và JavaScript là ba công nghệ cơ bản tạo nên một trang web hoàn chỉnh. Mỗi công nghệ đảm nhiệm một vai trò riêng nhưng phối hợp chặt chẽ với nhau để xây dựng giao diện, bố cục và tính năng tương tác cho website.

HTML (HyperText Markup Language) là cấu trúc nền tảng của trang web. Nó xác định nội dung và bố cục của trang, bao gồm tiêu đề, đoạn văn bản, hình ảnh, liên kết, biểu mẫu... mà trình duyệt sẽ hiển thị. HTML được xem như “khung xương” của một trang web.

CSS (Cascading Style Sheets) là ngôn ngữ định kiểu, dùng để điều chỉnh cách trình bày của nội dung HTML. CSS quyết định màu sắc, phông chữ, kích thước, khoảng cách, bố cục và các hiệu ứng visual khác, giúp giao diện trang web trở nên đẹp mắt và hấp dẫn hơn.

JavaScript là ngôn ngữ lập trình phía client giúp trang web trở nên tương tác và động. JavaScript cho phép xử lý sự kiện (như nhấp chuột, di chuột, nhập liệu), thay đổi nội dung mà không cần tải lại trang, và thực hiện các logic phức tạp ngay trong trình duyệt.

Cách ba công nghệ này phối hợp với nhau:

- + Khi trình duyệt tải một trang web, HTML sẽ được đọc trước để dựng lên cấu trúc nội dung cơ bản.

- + Sau đó CSS được áp dụng để trình duyệt biết cách trình bày phần nội dung đó theo các thuộc tính định sẵn.

- + Cuối cùng, JavaScript được thực thi để thêm các chức năng tương tác, như thay đổi nội dung theo hành vi người dùng, hiện/ẩn các phần tử, xử lý form, hay cập nhật nội dung mà không cần tải lại trang.

Có thể hình dung mối quan hệ này như một bộ ba:

- + HTML là “khung xương” – xác định cấu trúc.

- + CSS là “trang trí” – làm đẹp cho phần cấu trúc đó.

- + JavaScript là “tính năng” – làm cho trang web phản hồi và tương tác với người dùng.

Khi kết hợp với nhau, HTML, CSS và JavaScript tạo ra những trang web không chỉ có nội dung đầy đủ mà còn hấp dẫn và linh hoạt trên mọi thiết bị, từ giao diện tĩnh đến các ứng dụng web động hiện đại.

2.3.4 Ưu nhược điểm của HTML

HTML là ngôn ngữ đánh dấu siêu văn bản được sử dụng phổ biến trong việc xây dựng cấu trúc cho các trang web. Mặc dù đóng vai trò nền tảng trong phát triển web, HTML vẫn tồn tại những ưu điểm và hạn chế nhất định.

Ưu điểm :

Dễ học, dễ sử dụng: HTML có cú pháp đơn giản, rõ ràng, phù hợp cho người mới bắt đầu tiếp cận lĩnh vực lập trình web.

Được hỗ trợ rộng rãi: HTML hoạt động tốt trên hầu hết các trình duyệt phổ biến như Chrome, Firefox, Edge... mà không cần cài đặt thêm phần mềm.

Chuẩn mở, miễn phí: HTML là ngôn ngữ mã nguồn mở, không yêu cầu bản quyền, giúp giảm chi phí trong quá trình phát triển website.

Khả năng tương thích cao: Các trang HTML có thể hiển thị trên nhiều thiết bị khác nhau như máy tính, điện thoại, máy tính bảng.

Dễ kết hợp với các công nghệ khác: HTML có thể kết hợp với CSS để thiết kế giao diện và JavaScript, PHP để xây dựng các chức năng động cho website.

Hỗ trợ tốt cho SEO: HTML cung cấp các thẻ ngữ nghĩa như `title`, `meta`, `h1-h6`, giúp công cụ tìm kiếm dễ dàng thu thập và đánh giá nội dung.

Ưu điểm:

Chỉ tạo nội dung tĩnh: HTML không thể tự xử lý logic hay tương tác phức tạp, cần kết hợp với JavaScript hoặc ngôn ngữ phía máy chủ để tạo website động.

Không hỗ trợ xử lý dữ liệu: HTML không có khả năng tính toán, xử lý cơ sở dữ liệu hay kiểm soát luồng chương trình.

Hạn chế về bảo mật: HTML không cung cấp cơ chế bảo mật, việc xử lý dữ liệu người dùng phải phụ thuộc vào các ngôn ngữ backend như PHP.

Khó quản lý khi website lớn:

Khi số lượng trang tăng nhiều, việc quản lý và cập nhật mã HTML thuần sẽ trở nên phức tạp nếu không có công cụ hỗ trợ.

Phụ thuộc vào trình duyệt: Một số thẻ hoặc tính năng mới của HTML có thể không được hỗ trợ đồng đều trên các trình duyệt cũ.

2.4. Tổng qua về JavaScript

2.4.1. JavaScript là gì?

JavaScript là một ngôn ngữ lập trình kịch bản (scripting language) được sử dụng rộng rãi trong phát triển web để tạo ra các trang web động và tương tác với người dùng. Ban đầu JavaScript được thiết kế để chạy phía trình duyệt (client-side) và thực thi trực tiếp trong môi trường web, cho phép thay đổi nội dung trang, xử lý sự kiện, kiểm tra dữ liệu form, tạo hiệu ứng, và nhiều tính năng khác mà HTML và CSS không thể làm được chỉ riêng.

JavaScript là một ngôn ngữ thông dịch (interpreted language) – mã nguồn không cần biên dịch trước khi chạy mà được thực thi trực tiếp bởi trình duyệt hoặc môi trường thực thi. Điều này giúp quá trình phát triển và kiểm thử trở nên nhanh chóng và linh hoạt hơn.

Về bản chất, JavaScript là một ngôn ngữ đa mô hình, hỗ trợ lập trình hướng đối tượng (object-oriented), hàm (functional) và kịch bản (imperative). JavaScript sử dụng

cơ chế kế thừa dựa trên prototype thay vì lớp như nhiều ngôn ngữ khác, giúp mã nguồn linh hoạt và dễ mở rộng.

Mặc dù JavaScript nổi tiếng nhất với vai trò tạo tương tác trên trình duyệt, nhưng ngày nay ngôn ngữ này còn được sử dụng rộng rãi ở nhiều môi trường khác nhau. Ví dụ, với Node.js – một môi trường chạy JavaScript phía server – lập trình viên có thể sử dụng JavaScript để xây dựng server web, API, ứng dụng thời gian thực và nhiều chức năng backend khác, mở rộng vai trò của JavaScript vượt ngoài trình duyệt truyền thống.

Trong phát triển web, JavaScript thường được nhúng trực tiếp vào trang HTML thông qua thẻ `<script>` hoặc liên kết tới file `.js` bên ngoài. Khi trình duyệt tải trang, mã JavaScript được đọc và thực thi ngay trên client, cho phép tương tác với Document Object Model (DOM) – mô hình cấu trúc trang HTML – để thay đổi nội dung trang mà không cần tải lại trang.

JavaScript hiện nay là một trong những ngôn ngữ quan trọng và phổ biến nhất trong phát triển web, được sử dụng trong hầu hết các website và ứng dụng web hiện đại để tạo tính năng động và nâng cao trải nghiệm người dùng.

2.4.2. Lịch sử phát triển của JavaScript

JavaScript là một ngôn ngữ lập trình kịch bản được thiết kế để tạo ra các trang web tương tác và động. Ngôn ngữ này được phát triển lần đầu vào năm 1995 bởi Brendan Eich, khi ông làm việc tại công ty Netscape Communications. Ban đầu JavaScript không có tên như ngày nay; nó được gọi là Mocha, sau đó đổi thành LiveScript và cuối cùng trở thành JavaScript – tên gọi này được chọn nhằm tận dụng phần nào sự nổi tiếng của ngôn ngữ Java vào thời điểm đó, dù hai ngôn ngữ hoàn toàn khác nhau về bản chất và mục đích sử dụng.

Ngay sau khi ra mắt, JavaScript nhanh chóng được tích hợp vào trình duyệt Netscape Navigator, giúp trang web trở nên sống động hơn với các tính năng như xử lý sự kiện người dùng, kiểm tra dữ liệu form và thay đổi phân hiển thị mà không cần tải lại trang. JavaScript đã đánh dấu một bước thay đổi quan trọng trong phát triển web khi lần đầu tiên cho phép các trang web có thể phản hồi hành vi người dùng trực tiếp trên trình duyệt.

Đến năm 1997, JavaScript được chuẩn hóa bởi tổ chức ECMA International với tên gọi ECMAScript thông qua tiêu chuẩn ECMA-262. Việc chuẩn hóa này giúp

JavaScript có một định nghĩa ngôn ngữ thống nhất, làm cơ sở cho các trình duyệt và nền tảng khác nhau triển khai tương thích.

Trong những năm tiếp theo, JavaScript tiếp tục phát triển và được bổ sung nhiều tính năng mới qua các phiên bản chuẩn ECMAScript, chẳng hạn như ES5 (2009) với nhiều cải thiện về tính tương thích, và ES6 (2015) (còn gọi là ES2015) với nhiều tính năng hiện đại như `let`, `const`, arrow functions, class và promise, giúp ngôn ngữ trở nên mạnh mẽ và linh hoạt hơn.

Ngoài ra, sự phát triển của JavaScript không chỉ gói gọn ở việc lập trình phía trình duyệt. Với sự ra đời của các công nghệ như Node.js từ năm 2009, JavaScript đã mở rộng sang môi trường phía máy chủ, cho phép lập trình viên xây dựng server, API và các ứng dụng full-stack chỉ với một ngôn ngữ duy nhất.

Từ một ngôn ngữ đơn giản dùng tạo hiệu ứng trang web, JavaScript đã phát triển trở thành một trong những ngôn ngữ lập trình quan trọng và không thể thiếu của thế giới web hiện đại, được sử dụng trong hầu hết các trang web và ứng dụng web ngày nay.

2.4.3. Cách hoạt động và công dụng của JavaScript

Cách hoạt động của JavaScript:

JavaScript là một ngôn ngữ lập trình kịch bản được thực thi trực tiếp bởi trình duyệt web hoặc các môi trường chạy như Node.js. Không giống như các ngôn ngữ cần biên dịch trước, JavaScript được diễn dịch và thực thi ngay trong quá trình người dùng truy cập trang web.

Khi người dùng tải một trang web, trình duyệt sẽ lần lượt xử lý các thành phần của trang. Trước tiên, trình duyệt phân tích cú pháp mã HTML để xây dựng cấu trúc trang dưới dạng Document Object Model (DOM), sau đó áp dụng các quy tắc định dạng từ CSS. Khi gặp mã JavaScript, trình duyệt sẽ sử dụng JavaScript Engine (ví dụ như V8, SpiderMonkey) để phân tích và thực thi các câu lệnh JavaScript theo thứ tự.

Trong quá trình thực thi, JavaScript có thể tương tác trực tiếp với DOM để thay đổi nội dung, cấu trúc và giao diện của trang web mà không cần tải lại toàn bộ trang. Ngoài ra, JavaScript còn xử lý các sự kiện như nhấp chuột, nhập dữ liệu, cuộn trang, từ đó tạo ra các hành vi tương tác linh hoạt cho người dùng.

Bên cạnh việc chạy trên trình duyệt, JavaScript còn có thể hoạt động ở phía máy chủ thông qua môi trường Node.js, cho phép xây dựng các ứng dụng server-side, xử lý dữ liệu và kết nối cơ sở dữ liệu.

Công dụng của JavaScript: JavaScript đóng vai trò quan trọng trong phát triển web hiện đại với nhiều công dụng khác nhau, bao gồm:

Tạo tính năng động cho website: JavaScript giúp cập nhật nội dung trang web theo thời gian thực mà không cần tải lại trang, tạo các hiệu ứng động và cải thiện trải nghiệm người dùng.

Xử lý sự kiện và tương tác người dùng: JavaScript phản hồi nhanh chóng các hành động của người dùng như nhấp chuột, nhập liệu, kéo thả, giúp website trở nên trực quan và thân thiện hơn.

Thay đổi HTML và CSS một cách linh hoạt: Thông qua DOM, JavaScript có thể thay đổi nội dung, thuộc tính HTML và CSS, cho phép ẩn/hiện thành phần, thay đổi màu sắc, kích thước hoặc bố cục trang.

Giao tiếp với máy chủ không đồng bộ: JavaScript hỗ trợ các kỹ thuật như AJAX hoặc Fetch API để gửi và nhận dữ liệu từ máy chủ mà không cần tải lại trang, giúp tăng tốc độ và hiệu quả của ứng dụng web.

Phát triển ứng dụng phía máy chủ: Với Node.js, JavaScript có thể được sử dụng để xây dựng server, API, xử lý dữ liệu và phát triển các ứng dụng web hoàn chỉnh theo mô hình full-stack.

Phát triển ứng dụng đa nền tảng: JavaScript còn được sử dụng để xây dựng ứng dụng di động, ứng dụng desktop và các hệ thống tương tác khác thông qua các framework và thư viện hỗ trợ.

2.4.4. Ưu nhược điểm của JavaScript

JavaScript là một trong những ngôn ngữ lập trình quan trọng trong phát triển web hiện đại, đóng vai trò tạo tính tương tác và xử lý trực tiếp trên trình duyệt. Tuy nhiên, bên cạnh những ưu điểm nổi bật, JavaScript cũng tồn tại một số hạn chế nhất định trong quá trình phát triển và vận hành ứng dụng web.

Ưu điểm của JavaScript:

Giảm tải cho máy chủ: JavaScript được thực thi trực tiếp trên trình duyệt của người dùng, do đó nhiều thao tác xử lý có thể được thực hiện ở phía client mà không

cần gửi yêu cầu liên tục lên server. Điều này giúp giảm tải cho máy chủ và tiết kiệm băng thông hệ thống.

Khả năng tương tác và phản hồi nhanh: JavaScript cho phép xử lý các sự kiện người dùng như nhấp chuột, nhập dữ liệu, cuộn trang một cách tức thời mà không cần tải lại trang, từ đó nâng cao trải nghiệm và tính tương tác của website.

Hỗ trợ đa nền tảng và trình duyệt: JavaScript được hỗ trợ bởi hầu hết các trình duyệt hiện đại trên nhiều hệ điều hành khác nhau, giúp ứng dụng web có khả năng hoạt động ổn định trên nhiều thiết bị như máy tính, máy tính bảng và điện thoại thông minh.

Dễ học và dễ triển khai: JavaScript có cú pháp tương đối đơn giản và có thể được nhúng trực tiếp vào mã HTML, phù hợp với người mới bắt đầu học lập trình web cũng như sinh viên trong quá trình tiếp cận công nghệ web.

Hệ sinh thái phong phú: JavaScript sở hữu cộng đồng phát triển lớn cùng với nhiều thư viện và framework mạnh mẽ như React, Angular, Vue.js, giúp tăng tốc quá trình phát triển ứng dụng và dễ dàng mở rộng chức năng.

Ứng dụng rộng rãi trong nhiều lĩnh vực: Ngoài việc chạy trên trình duyệt, JavaScript còn được sử dụng ở phía máy chủ thông qua Node.js, cho phép xây dựng các ứng dụng web theo mô hình full-stack chỉ với một ngôn ngữ duy nhất.

Nhược điểm của JavaScript:

Mã nguồn dễ bị xem và phân tích: Do JavaScript được tải về và thực thi trên trình duyệt, mã nguồn có thể bị người dùng xem hoặc phân tích, làm tăng nguy cơ lộ logic xử lý nếu không có biện pháp bảo vệ phù hợp.

Hạn chế về bảo mật: Nếu không được lập trình và kiểm soát chặt chẽ, JavaScript có thể trở thành mục tiêu của các lỗ hổng bảo mật như Cross-Site Scripting (XSS), gây ảnh hưởng đến an toàn dữ liệu và hệ thống.

Phụ thuộc vào trình duyệt: Cách các trình duyệt xử lý JavaScript đôi khi không hoàn toàn giống nhau, có thể dẫn đến sự không tương thích và phát sinh lỗi khi ứng dụng chạy trên các môi trường khác nhau.

Hiệu suất phụ thuộc vào thiết bị người dùng: Do JavaScript chạy ở phía client, hiệu suất thực thi phụ thuộc vào cấu hình phần cứng và trình duyệt của người dùng. Trên các thiết bị cấu hình thấp, ứng dụng có thể hoạt động chậm hoặc kém ổn định.

Hạn chế trong xử lý đa luồng: JavaScript chủ yếu hoạt động trên một luồng xử lý chính, gây khó khăn khi xử lý các tác vụ tính toán phức tạp hoặc yêu cầu xử lý song song.

Ảnh hưởng đến tối ưu công cụ tìm kiếm (SEO): Nội dung được tạo động bằng JavaScript đôi khi khó được các công cụ tìm kiếm thu thập và lập chỉ mục đầy đủ, nếu không có giải pháp hỗ trợ phù hợp.

2.5. Ứng dụng mô hình MVC trong hệ thống bán thức ăn nhanh online

2.5.1 Khái quát mô hình MVC

Mô hình MVC là một kiến trúc phần mềm phổ biến, trong đó ứng dụng được chia thành ba thành phần chính:

Model: Chịu trách nhiệm xử lý dữ liệu và các logic nghiệp vụ của hệ thống, đồng thời tương tác trực tiếp với cơ sở dữ liệu.

View: Đảm nhiệm việc hiển thị giao diện và dữ liệu cho người dùng, đồng thời tiếp nhận các thao tác từ phía người dùng.

Controller: Đóng vai trò trung gian, tiếp nhận yêu cầu từ View, xử lý thông qua Model và trả kết quả về View tương ứng.

Việc tách riêng các thành phần này giúp mã nguồn được tổ chức tốt hơn, giảm sự phụ thuộc giữa các phần và thuận tiện cho phát triển theo nhóm.

2.5.2 Áp dụng mô hình MVC trong hệ thống website bán thức ăn nhanh

Model – Xử lý dữ liệu và logic nghiệp vụ:

Model là thành phần chịu trách nhiệm tương tác với cơ sở dữ liệu và xử lý các nghiệp vụ chính của hệ thống. Trong đồ án, Model được thiết kế để quản lý các bảng dữ liệu quan trọng như:

Users (Người dùng/Khách hàng):

- + Đăng ký và đăng nhập tài khoản
- + Quản lý thông tin cá nhân

Products (Sản phẩm/Thức ăn nhanh):

- + Lấy danh sách món ăn
- + Thêm, sửa, xóa sản phẩm

Orders (Đơn hàng):

- + Tạo và lưu thông tin đơn hàng
- + Theo dõi trạng thái đơn hàng

OrderItems (Chi tiết đơn hàng):

- + Liên kết đơn hàng với các sản phẩm
- + Quản lý số lượng và tính tổng tiền

Model sử dụng ngôn ngữ PHP kết hợp với hệ quản trị cơ sở dữ liệu MySQL để thực hiện các thao tác truy vấn như:

- + SELECT: lấy dữ liệu
- + INSERT: thêm dữ liệu
- + UPDATE: cập nhật dữ liệu
- + DELETE: xóa dữ liệu

Các thao tác này được xử lý tập trung trong Model nhằm đảm bảo tính chính xác và nhất quán của dữ liệu trong toàn hệ thống.

View – Giao diện người dùng:

View là phần giao diện mà người dùng trực tiếp tương tác, được xây dựng bằng HTML, CSS và JavaScript. View có nhiệm vụ hiển thị dữ liệu từ Model theo cách trực quan, dễ hiểu và gửi các yêu cầu thao tác của người dùng đến Controller.

Các giao diện chính trong hệ thống bao gồm:

Trang chủ:

- + Hiển thị danh sách món ăn
- + Cho phép chọn món và thêm vào giỏ hàng

Trang giỏ hàng:

- + Hiển thị các sản phẩm đã chọn
- + Cho phép chỉnh sửa số lượng hoặc xóa sản phẩm

Trang đặt hàng:

- + Nhập thông tin giao hàng
- + Xác nhận và gửi đơn hàng

Trang quản trị:

- + Quản lý sản phẩm
- + Quản lý đơn hàng
- + Quản lý tài khoản người dùng

View được thiết kế đảm bảo tính trực quan, dễ sử dụng và tương thích với nhiều thiết bị như máy tính để bàn, máy tính bảng và điện thoại di động.

Controller – Điều phối xử lý và tương tác:

Controller là thành phần trung gian kết nối giữa View và Model. Khi người dùng thực hiện các thao tác trong giao diện như:

- +Thêm sản phẩm vào giỏ hàng
- + Gửi form đăng nhập
- + Hoàn tất đặt món

Controller sẽ tiếp nhận yêu cầu, xử lý logic nghiệp vụ phù hợp bằng cách gọi các phương thức trong Model, sau đó điều hướng và trả kết quả về View tương ứng.

Ví dụ hoạt động của Controller trong hệ thống:

Xem danh sách sản phẩm: Controller gọi Model lấy dữ liệu sản phẩm và truyền dữ liệu sang View để hiển thị.

Đặt hàng: Controller nhận dữ liệu từ form đặt hàng, gọi Model lưu thông tin đơn hàng vào cơ sở dữ liệu và điều hướng sang trang xác nhận đơn hàng.

Nhờ có Controller, phần xử lý nghiệp vụ không bị trộn lẫn với giao diện, giúp mã nguồn dễ đọc, dễ mở rộng và bảo trì.

2.5.3 Lợi ích của việc áp dụng mô hình MVC

Việc áp dụng mô hình MVC trong hệ thống bán thức ăn nhanh online mang lại nhiều lợi ích:

Tách biệt rõ ràng các thành phần: Logic nghiệp vụ, giao diện và điều phối xử lý được phân tách riêng, giúp hệ thống dễ bảo trì và nâng cấp.

Thuận lợi cho phát triển theo nhóm: Nhóm phát triển giao diện tập trung xây dựng View, trong khi nhóm xử lý dữ liệu làm việc với Model và Controller mà không ảnh hưởng lẫn nhau.

Dễ dàng mở rộng hệ thống: Khi cần bổ sung chức năng mới như thanh toán online hoặc quản lý khuyến mãi, hệ thống có thể mở rộng mà không ảnh hưởng đến các chức năng hiện tại.

Tăng khả năng tái sử dụng mã nguồn: Các chức năng xử lý dữ liệu trong Model có thể được sử dụng lại cho nhiều View khác nhau, giúp tiết kiệm thời gian phát triển.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1. Mô tả đề tài

Đề tài “Xây dựng hệ thống bán thức ăn nhanh online” nhằm thiết kế và xây dựng một ứng dụng web thương mại điện tử hỗ trợ việc hiển thị, tìm kiếm, lựa chọn và đặt món ăn nhanh trực tuyến. Hệ thống được phát triển bằng ngôn ngữ lập trình PHP, sử dụng hệ quản trị cơ sở dữ liệu MySQL và triển khai trên môi trường máy chủ XAMPP. Đồng thời, mô hình kiến trúc MVC (Model – View – Controller) được áp dụng nhằm tách biệt rõ ràng giữa logic xử lý, dữ liệu và giao diện người dùng.

Hệ thống hướng đến việc xây dựng một nền tảng đặt món trực tuyến thân thiện, trực quan, giúp người tiêu dùng dễ dàng lựa chọn và đặt món thông qua trình duyệt web. Bên cạnh đó, hệ thống còn cung cấp các công cụ quản lý hiệu quả dành cho chủ cửa hàng nhằm hỗ trợ công tác quản lý hoạt động kinh doanh.

Các chức năng chính của hệ thống bao gồm:

Đăng ký và đăng nhập người dùng: Cho phép người dùng tạo tài khoản mới, đăng nhập và quản lý thông tin cá nhân để sử dụng các chức năng đặt món.

Xem danh sách sản phẩm: Hiển thị thực đơn các món ăn với hình ảnh, thông tin chi tiết và giá cả rõ ràng.

Quản lý giỏ hàng: Cho phép người dùng thêm sản phẩm vào giỏ hàng, chỉnh sửa số lượng hoặc xóa sản phẩm đã chọn.

Đặt món nhanh chóng: Người dùng nhập thông tin giao hàng, xác nhận đơn hàng và gửi yêu cầu đặt món.

Quản trị hệ thống: Người quản lý có thể thêm, sửa, xóa sản phẩm; xem và xử lý đơn hàng; quản lý thông tin người dùng.

Hệ thống được thiết kế để hoạt động ổn định trên các trình duyệt web hiện đại và có thể sử dụng trên nhiều thiết bị khác nhau như máy tính để bàn, điện thoại thông minh và máy tính bảng. Việc xây dựng hệ thống không chỉ giúp người tiêu dùng đặt món một cách thuận tiện mà còn hỗ trợ cửa hàng tối ưu hóa công tác quản lý đơn hàng, theo dõi doanh thu và nâng cao trải nghiệm khách hàng.

Mục tiêu của đề tài là phát triển một hệ thống web đơn giản, dễ sử dụng, phù hợp với quy mô các cửa hàng thức ăn nhanh nhỏ và vừa, đồng thời tạo nền tảng để mở rộng thêm các chức năng nâng cao trong tương lai như thanh toán trực tuyến, đánh giá sản phẩm và tích hợp dịch vụ vận chuyển.

3.2. Đặc tả yêu cầu hệ thống

3.2.1. Yêu cầu chức năng

Yêu cầu chức năng mô tả các chức năng mà hệ thống bán thức ăn nhanh trực tuyến cần thực hiện để đáp ứng nhu cầu của người dùng và quản trị viên. Các chức năng này đảm bảo hệ thống hoạt động đúng mục tiêu đề tài và hỗ trợ đầy đủ các nghiệp vụ cơ bản của một website đặt món ăn online.

Chức năng quản lý người dùng: Hệ thống cho phép người dùng đăng ký tài khoản mới bằng các thông tin cơ bản như họ tên, email, số điện thoại và mật khẩu. Sau khi đăng ký, người dùng có thể đăng nhập và đăng xuất khỏi hệ thống để sử dụng các chức năng đặt món. Hệ thống cũng hỗ trợ chức năng phục hồi mật khẩu trong trường hợp người dùng quên mật khẩu. Ngoài ra, người dùng có thể xem và chỉnh sửa thông tin cá nhân của mình sau khi đăng nhập.

Chức năng quản lý sản phẩm: Hệ thống cung cấp chức năng hiển thị danh sách các món ăn nhanh với đầy đủ thông tin như hình ảnh, tên món, mô tả và giá bán. Người dùng có thể xem chi tiết từng sản phẩm, đồng thời hỗ trợ tìm kiếm hoặc lọc sản phẩm theo tên hoặc danh mục. Đối với quản trị viên, hệ thống cho phép thêm mới sản phẩm, chỉnh sửa thông tin sản phẩm và xóa các sản phẩm không còn kinh doanh.

Chức năng giỏ hàng: Hệ thống cho phép người dùng lựa chọn sản phẩm và thêm vào giỏ hàng trong quá trình duyệt thực đơn. Giỏ hàng hiển thị danh sách các sản phẩm đã chọn, kèm theo số lượng và tổng tiền tạm tính. Người dùng có thể thay đổi số lượng sản phẩm hoặc xóa sản phẩm khỏi giỏ hàng trước khi tiến hành đặt món.

Chức năng đặt hàng: Người dùng có thể tiến hành đặt hàng bằng cách nhập thông tin giao hàng như địa chỉ nhận hàng, số điện thoại liên hệ và ghi chú nếu cần thiết. Hệ thống tự động tính toán tổng giá trị đơn hàng và hiển thị cho người dùng xác nhận. Sau khi người dùng xác nhận, đơn hàng sẽ được lưu vào hệ thống và thông báo đặt hàng thành công được hiển thị.

Chức năng quản lý đơn hàng: Hệ thống hỗ trợ quản trị viên theo dõi và quản lý các đơn hàng đã được đặt. Admin có thể xem danh sách đơn hàng theo từng trạng thái như đơn hàng mới, đang xử lý hoặc đã hoàn tất. Đồng thời, quản trị viên có thể xem chi tiết từng đơn hàng và cập nhật trạng thái đơn hàng trong quá trình xử lý và giao hàng.

Chức năng thống kê và báo cáo: Hệ thống cung cấp các chức năng thống kê cơ bản nhằm hỗ trợ quản lý hoạt động kinh doanh của cửa hàng. Quản trị viên có thể theo dõi doanh thu theo từng khoảng thời gian như ngày, tháng hoặc năm, xem số lượng đơn hàng theo trạng thái và nắm bắt các sản phẩm bán chạy để phục vụ cho việc phân tích và ra quyết định kinh doanh.

3.2.2. Yêu cầu phi chức năng

Yêu cầu phi chức năng mô tả các tiêu chí về chất lượng nhằm đảm bảo hệ thống bán thức ăn nhanh online hoạt động ổn định, an toàn và mang lại trải nghiệm tốt cho người dùng. Các yêu cầu này không trực tiếp tạo ra chức năng mới nhưng có vai trò quan trọng trong việc nâng cao hiệu quả và độ tin cậy của hệ thống.

Hiệu năng (Performance): Hệ thống cần đảm bảo thời gian phản hồi nhanh khi người dùng truy cập các chức năng chính như xem danh sách sản phẩm, thao tác giỏ hàng và đặt hàng. Thời gian tải trang và xử lý yêu cầu nên được giữ trong khoảng cho phép (thường không quá 3 giây) để tránh gây khó chịu cho người dùng. Ngoài ra, hệ thống phải có khả năng phục vụ đồng thời nhiều người dùng truy cập trong điều kiện sử dụng thông thường mà không xảy ra tình trạng chậm hoặc treo hệ thống.

Tính sẵn sàng (Availability): Hệ thống phải hoạt động ổn định và liên tục trong thời gian cửa hàng kinh doanh. Các chức năng quan trọng như xem sản phẩm, đặt hàng và quản lý đơn hàng cần được đảm bảo truy cập xuyên suốt, hạn chế tối đa thời gian gián đoạn hoặc ngừng hoạt động.

Bảo mật (Security): Hệ thống cần đảm bảo an toàn thông tin cho người dùng, đặc biệt là dữ liệu đăng nhập và mật khẩu phải được mã hóa khi lưu trữ trong cơ sở dữ liệu. Bên cạnh đó, hệ thống phải có các biện pháp phòng chống những lỗ hổng bảo mật phổ biến như SQL Injection, Cross-Site Scripting (XSS) và Cross-Site Request Forgery (CSRF). Quyền truy cập của người dùng thường và quản trị viên phải được phân tách rõ ràng để tránh truy cập trái phép.

Tính dễ sử dụng (Usability): Giao diện hệ thống cần được thiết kế đơn giản, trực quan và dễ thao tác, phù hợp với nhiều đối tượng người dùng. Các chức năng như đăng ký, đăng nhập, quản lý giỏ hàng và đặt hàng phải dễ hiểu và dễ sử dụng. Hệ thống cũng cần hiển thị rõ ràng các thông báo lỗi hoặc thông báo thành công để người dùng nắm bắt được trạng thái thao tác của mình.

Khả năng mở rộng (Scalability): Hệ thống phải được thiết kế theo hướng linh hoạt để dễ dàng mở rộng và bổ sung các chức năng mới trong tương lai như thanh toán trực tuyến, đánh giá sản phẩm hoặc theo dõi đơn hàng theo thời gian thực. Cấu trúc mã nguồn và cơ sở dữ liệu cần cho phép mở rộng mà không ảnh hưởng lớn đến các chức năng đã có.

Tính tương thích (Compatibility): Website cần hoạt động tốt và hiển thị chính xác trên các trình duyệt phổ biến hiện nay như Google Chrome, Mozilla Firefox, Microsoft Edge và Safari. Đồng thời, giao diện phải tương thích với nhiều thiết bị khác nhau như máy tính để bàn, máy tính bảng và điện thoại di động.

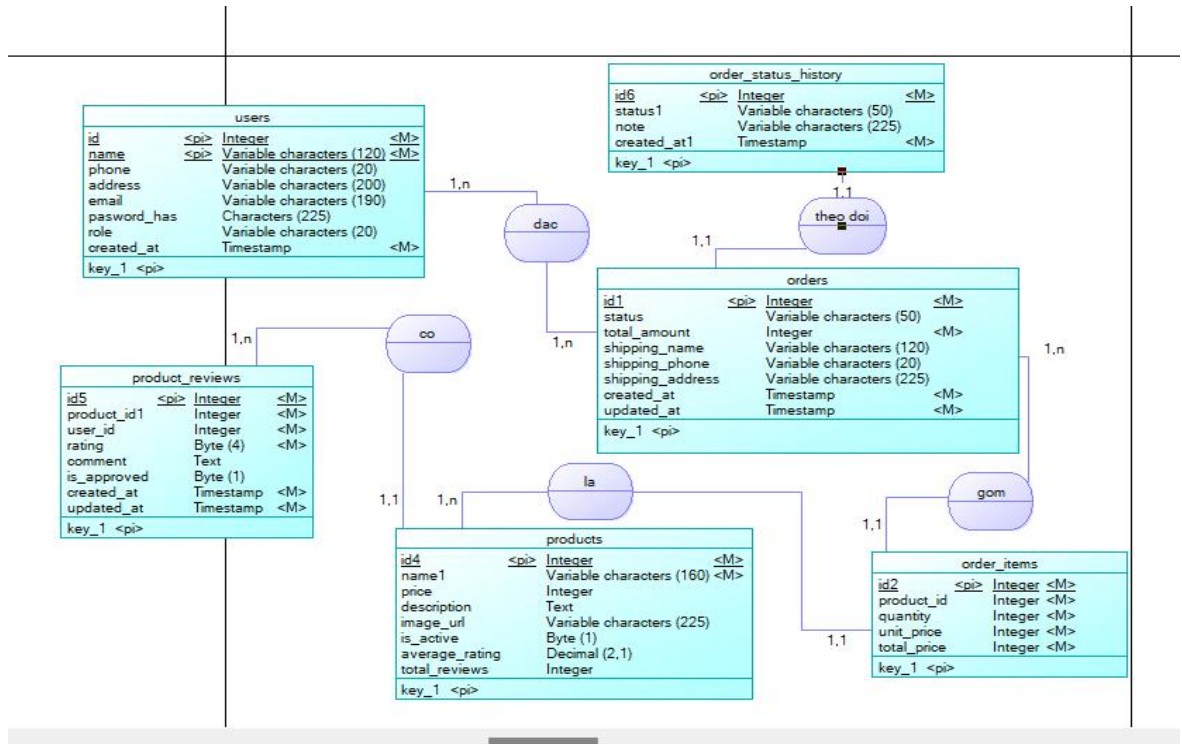
Tính bảo trì (Maintainability): Mã nguồn hệ thống cần được tổ chức rõ ràng, tuân thủ mô hình MVC để thuận tiện cho việc kiểm tra, sửa lỗi và nâng cấp trong tương lai. Cơ sở dữ liệu phải được thiết kế khoa học, có ràng buộc và chú thích đầy đủ nhằm hỗ trợ quá trình bảo trì và mở rộng hệ thống.

Tài liệu và hướng dẫn (Documentation): Hệ thống cần có tài liệu hướng dẫn cài đặt và triển khai cơ bản, giúp người quản trị dễ dàng đưa hệ thống vào sử dụng. Ngoài ra, cần có hướng dẫn sử dụng cho quản trị viên, bao gồm các thao tác quản lý sản phẩm, xử lý đơn hàng và quản lý người dùng.

3.3. Thiết kế dữ liệu

3.3.1. Mô hình thực thể kết hợp (ERD)

Mô hình mô tả các đối tượng và các mối quan hệ giữa các đối tượng.



Hình 3.1.Mô hình thực thể kết hợp

Các thực thể và mối quan hệ:

Sơ đồ cơ sở dữ liệu gồm 6 thực thể chính: users, products, orders, order_items, product_reviews và order_status_history. Trong đó, users lưu thông tin người dùng và quản trị viên; products quản lý danh mục sản phẩm; orders lưu thông tin tổng quát đơn hàng; order_items lưu chi tiết các sản phẩm trong đơn hàng; product_reviews ghi nhận đánh giá của người dùng; và order_status_history theo dõi quá trình thay đổi trạng thái của đơn hàng.

Về mối quan hệ, một người dùng có thể đặt nhiều đơn hàng và viết nhiều đánh giá; mỗi sản phẩm có thể xuất hiện trong nhiều đơn hàng và nhận nhiều phản hồi; mỗi đơn hàng gồm nhiều chi tiết sản phẩm và có nhiều bản ghi lịch sử trạng thái để theo dõi tiến trình xử lý.

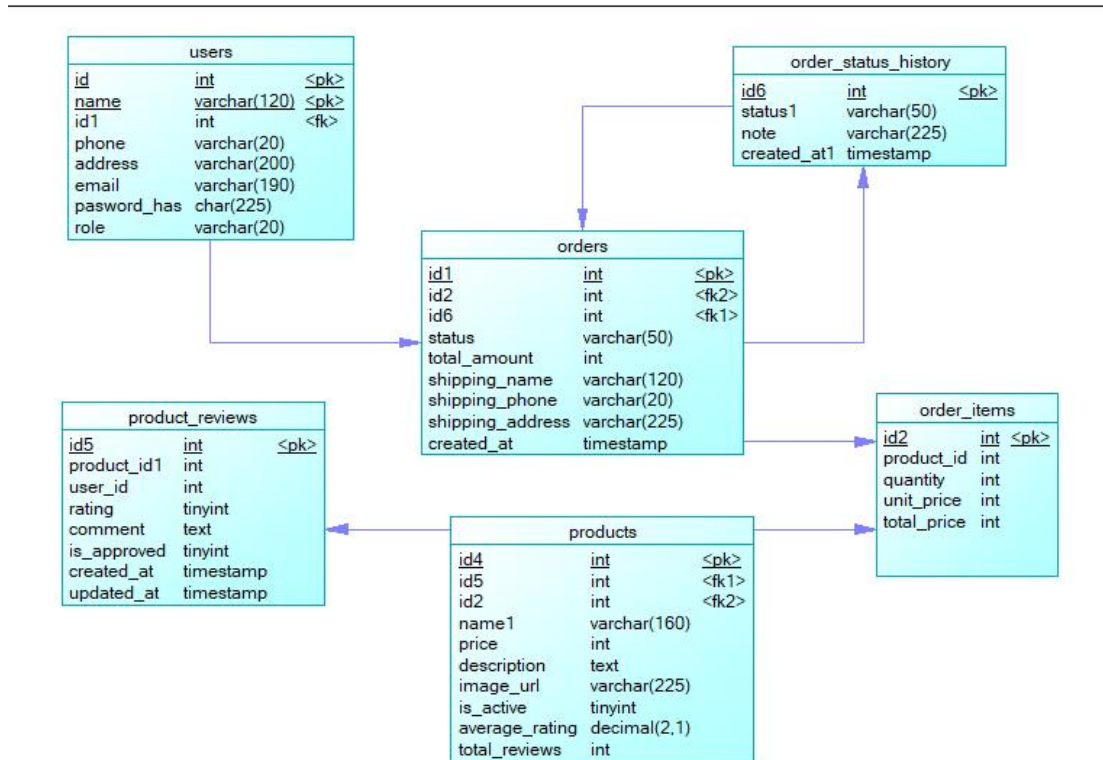
Nhận xét kỹ thuật:

Mỗi thực thể đều được định danh bằng khóa chính, các ràng buộc dữ liệu và kiểu dữ

liệu phù hợp được áp dụng nhằm đảm bảo tính toàn vẹn, ổn định và đáp ứng yêu cầu quản lý của hệ thống bán thức ăn nhanh.

3.3.2. Mô hình vật lý

Mô hình dùng để mô tả cách thức lưu trữ dữ liệu trong cơ sở dữ liệu.



Hình 3.2. Mô hình vật lý

Cấu trúc bảng, ràng buộc và mối quan hệ:

Mô hình cơ sở dữ liệu sử dụng các kiểu dữ liệu phù hợp cho từng loại thông tin, trong đó kiểu int dùng cho khóa định danh và các giá trị số, varchar cho dữ liệu văn bản, timestamp để ghi nhận thời gian tạo hoặc cập nhật, decimal cho các giá trị đánh giá và tinyint cho các trường trạng thái hoặc logic. Mỗi bảng đều được xác định khóa chính để định danh duy nhất bản ghi, đồng thời sử dụng khóa ngoại nhằm đảm bảo tính toàn vẹn dữ liệu giữa các bảng liên quan như người dùng, sản phẩm, đơn hàng, chi tiết đơn hàng, đánh giá và lịch sử trạng thái.

Sơ đồ vật lý thể hiện rõ các luồng dữ liệu chính, trong đó thông tin đơn hàng được liên kết từ người dùng đến đơn hàng và lịch sử trạng thái, còn sản phẩm đóng vai trò trung tâm kết nối với bảng đánh giá và bảng chi tiết đơn hàng. Việc tách riêng thông tin tổng quát đơn hàng và chi tiết sản phẩm giúp mô hình linh hoạt, dễ quản lý và phù hợp cho việc mở rộng hệ thống bán thức ăn nhanh trong thực tế.

3.3.3. Mô hình dữ liệu

users (id, name, id1, phone, address, email, password_has, role)

products (id4, id5, id2, name1, price, description, image_url, is_active, average_rating, total_reviews)

orders (id1, id2, id6, status, total_amount, shipping_name, shipping_phone, shipping_address, created_at)

order_items (id2, product_id, quantity, unit_price, total_price)

order_status_history (id6, status1, note, created_at1)

product_reviews (id5, product_id1, user_id, rating, comment, is_approved, created_at, updated_at)

3.3.4. Mô tả các bảng

Bảng 3.1. USER – Người dùng

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã người dùng
2	name	Varchar(120)	NOT NULL	Họ tên người dùng
3	phone	Varchar(20)	NOT NULL	Số điện thoại
4	address	Varchar(200)	NULL	Địa chỉ người dùng
5	email	Varchar(190)	UNIQUE, NOT NULL	Email đăng nhập
6	password_hash	Varchar(225)	NOT NULL	Mật khẩu đã mã hóa
7	role	Varchar(20)	NOT NULL	Vai trò (admin/user)

Bảng 3.2. PRODUCTS – Sản phẩm

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã sản phẩm
2	name	Varchar(160)	NOT NULL	Tên sản phẩm
3	price	Integer	NOT NULL	Giá bán sản phẩm
4	description	Text	NULL	Mô tả chi tiết
5	image_url	Varchar(225)	NULL	Đường dẫn hình ảnh
6	is_active	Byte	NOT NULL	Trạng thái (1: bán, 0: ngưng)

Bảng 3.3.ORDERS – Đơn hàng

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã đơn hàng
2	user_id	Integer	FK, NOT NULL	Người đặt hàng
3	status	Varchar(50)	NOT NULL	Trạng thái đơn hàng
4	total_amount	Integer	NOT NULL	Tổng tiền đơn
5	shipping_name	Varchar(120)	NOT NULL	Tên người nhận
6	shipping_phone	Varchar(20)	NOT NULL	SĐT người nhận

Bảng 3.4.ORDER_ITEMS – Chi tiết đơn hàng

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã chi tiết đơn
2	order_id	Integer	FK, NOT NULL	Mã đơn hàng
3	product_id	Integer	FK, NOT NULL	Mã sản phẩm
4	quantity	Integer	NOT NULL	Số lượng sản phẩm
5	unit_price	Integer	NOT NULL	Giá tại thời điểm mua
6	total_price	Integer	NOT NULL	Thành tiền

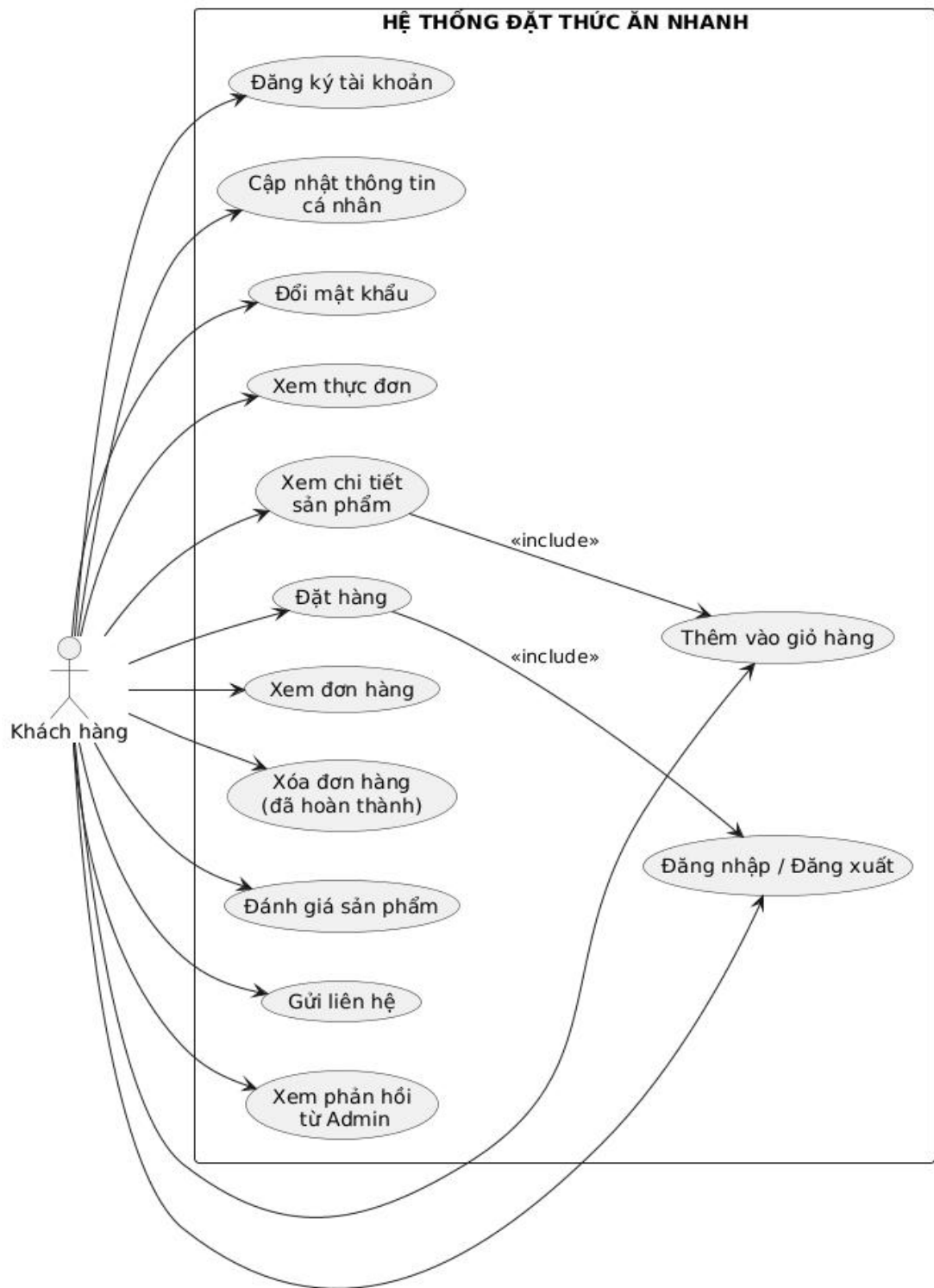
Bảng 3.5.ORDER_STATUS_HISTORY – Lịch sử trạng thái đơn

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã lịch sử
2	order_id	Integer	FK, NOT NULL	Mã đơn hàng
3	status	Varchar(50)	NOT NULL	Trạng thái đơn
4	note	Varchar(225)	NULL	Ghi chú
5	created_at	Timestamp	NOT NULL	Thời gian cập nhật

Bảng 3.6.PRODUCT_REVIEWS – Đánh giá sản phẩm

STT	Thuộc tính	Kiểu dữ liệu	Ràng buộc	Diễn giải
1	id	Integer	PK, NOT NULL	Mã đánh giá
2	product_id	Integer	FK, NOT NULL	Sản phẩm được đánh giá
3	user_id	Integer	FK, NOT NULL	Người đánh giá
4	rating	Byte	NOT NULL	Số sao (1–5)
5	comment	Text	NULL	Nội dung đánh giá
6	is_approved	Byte	DEFAULT 0	Trạng thái duyệt
7	created_at	Timestamp	NOT NULL	Ngày tạo đánh giá
8	updated_at	Timestamp	NULL	Ngày cập nhật

3.3.5. Sơ đồ Use case của User (khách hàng)



Hình 3.3.Sơ đồ Use case của khách hàng (người dùng)

Tác nhân (Actor)

Khách hàng là tác nhân chính của hệ thống. Đây là người trực tiếp tương tác với website để xem thực đơn, lựa chọn món ăn và thực hiện đặt hàng trực tuyến. Ngoài ra, khách hàng còn có thể quản lý thông tin cá nhân, theo dõi đơn hàng và gửi phản hồi đến hệ thống.

Các chức năng (Use Case): Các chức năng của hệ thống được phân thành các nhóm chính như sau:

Quản lý tài khoản:

- Đăng ký tài khoản: Cho phép người dùng mới tạo tài khoản để sử dụng hệ thống.
- Đăng nhập và Đăng xuất: Cho phép người dùng truy cập hoặc thoát khỏi hệ thống; là điều kiện bắt buộc để thực hiện chức năng đặt hàng.
- Cập nhật thông tin cá nhân: Cho phép người dùng chỉnh sửa thông tin cá nhân đã đăng ký.
- Đổi mật khẩu: Cho phép người dùng thay đổi mật khẩu nhằm đảm bảo an toàn thông tin.

Tương tác với thực đơn và sản phẩm:

- Xem thực đơn: Hiển thị danh sách các món ăn trong hệ thống.
- Xem chi tiết sản phẩm: Hiển thị thông tin chi tiết của từng món ăn.
- Đánh giá sản phẩm: Cho phép khách hàng để lại nhận xét và đánh giá món ăn sau khi sử dụng.

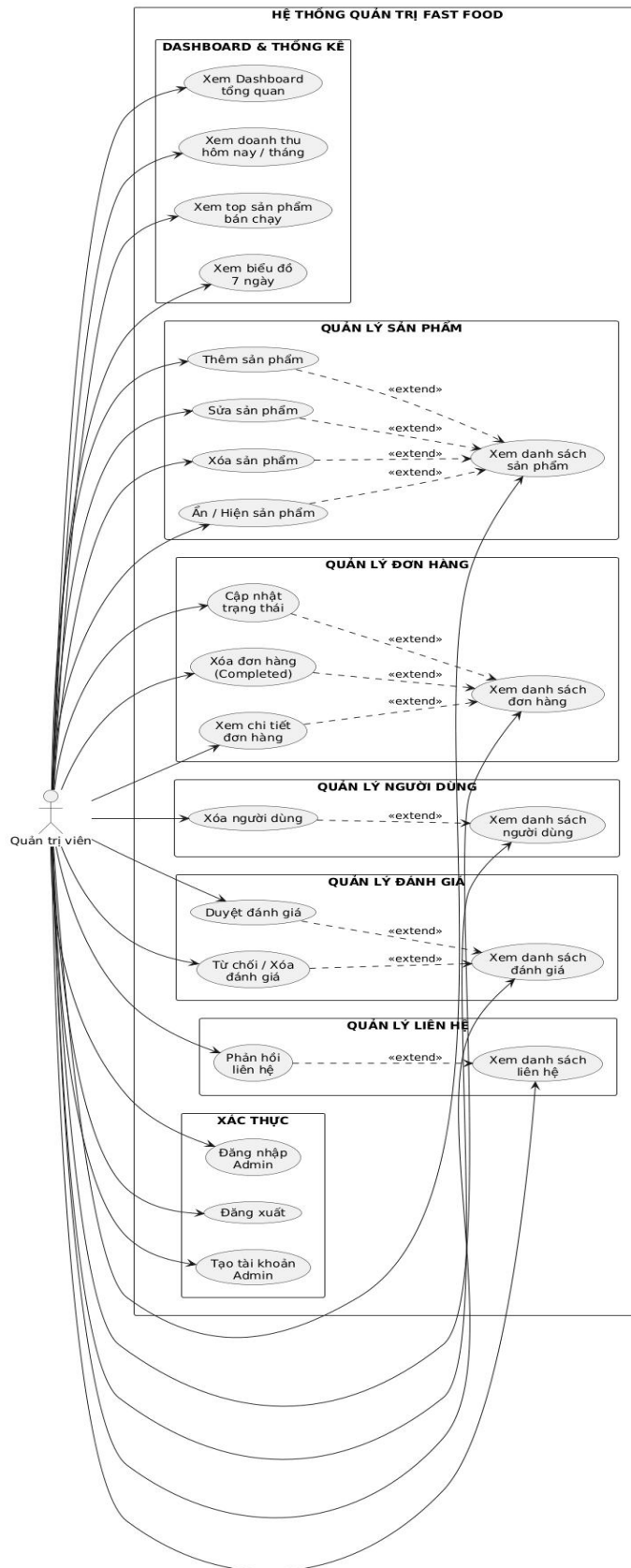
Quy trình đặt hàng:

- Thêm vào giỏ hàng: Lưu trữ các món ăn được chọn trước khi đặt hàng.
- Đặt hàng: Thực hiện quá trình mua hàng dựa trên các sản phẩm trong giỏ hàng.

Quản lý đơn hàng và liên hệ:

- Xem đơn hàng: Theo dõi trạng thái các đơn hàng đã đặt.
- Xóa đơn hàng (đã hoàn thành): Xóa các đơn hàng đã hoàn tất trong lịch sử cá nhân.
- Gửi liên hệ: Gửi yêu cầu hoặc thắc mắc đến hệ thống.
- Xem phản hồi từ Admin: Xem phản hồi của quản trị viên đối với các yêu cầu đã gửi.

3.5.6. Sơ đồ Use case của quản trị (Admin)



Hình 3.4. Sơ đồ Use case của quản trị viên

Tác nhân (Actor)

Quản trị viên (Admin) là người dùng có quyền hạn cao nhất trong hệ thống. Quản trị viên có trách nhiệm quản lý dữ liệu, theo dõi hoạt động kinh doanh, kiểm soát nội dung và đảm bảo hệ thống vận hành ổn định.

Các phân hệ chức năng chính:

Dashboard và thống kê: Phân hệ này hỗ trợ quản trị viên theo dõi nhanh tình hình hoạt động của hệ thống thông qua các thông tin tổng hợp và trực quan. Quản trị viên có thể xem dashboard tổng quan, theo dõi doanh thu theo ngày hoặc theo tháng, xem danh sách các sản phẩm bán chạy và quan sát biểu đồ thống kê trong vòng 7 ngày gần nhất. Các chức năng này giúp quản trị viên nắm bắt hiệu quả kinh doanh và đưa ra quyết định kịp thời.

Quản lý sản phẩm: Phân hệ quản lý sản phẩm cho phép quản trị viên theo dõi và điều chỉnh danh mục món ăn trong hệ thống. Chức năng chính là xem danh sách sản phẩm. Từ danh sách này, quản trị viên có thể thực hiện các chức năng mở rộng như thêm mới sản phẩm, chỉnh sửa thông tin, xóa sản phẩm hoặc thay đổi trạng thái hiển thị của sản phẩm.

Quản lý đơn hàng: Phân hệ này giúp quản trị viên kiểm soát toàn bộ quy trình bán hàng. Quản trị viên có thể xem danh sách các đơn hàng do khách hàng tạo ra. Từ đó, quản trị viên có thể xem chi tiết từng đơn hàng, cập nhật trạng thái đơn hàng (như đang xử lý, đang giao, đã hủy), hoặc xóa các đơn hàng đã hoàn thành.

Quản lý người dùng và đánh giá: Trong phân hệ này, quản trị viên có thể xem danh sách người dùng và thực hiện xóa người dùng khi cần thiết. Ngoài ra, hệ thống còn hỗ trợ quản lý các đánh giá và bình luận của khách hàng. Quản trị viên có thể duyệt, từ chối hoặc xóa các đánh giá không phù hợp nhằm đảm bảo nội dung hiển thị trên hệ thống lành mạnh và chính xác.

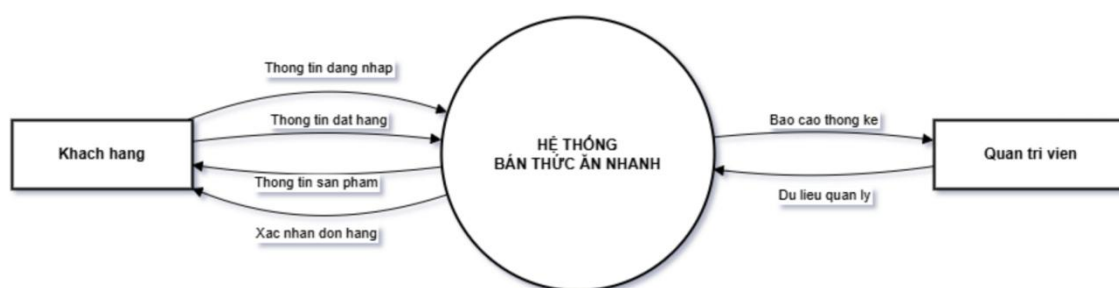
Quản lý liên hệ: Phân hệ quản lý liên hệ cho phép quản trị viên tiếp nhận các yêu cầu, thắc mắc từ khách hàng. Quản trị viên có thể xem danh sách liên hệ và thực hiện phản hồi trực tiếp nhằm hỗ trợ khách hàng kịp thời.

Xác thực và quản lý tài khoản Admin: Hệ thống cung cấp các chức năng xác thực dành cho quản trị viên bao gồm đăng nhập, đăng xuất và tạo tài khoản quản trị viên mới. Các chức năng này đảm bảo an toàn và phân quyền trong quá trình quản lý hệ thống.

Phân tích mối quan hệ «extend»: Trong sơ đồ Use Case phía quản trị viên, mối quan hệ «extend» (mở rộng) được sử dụng phổ biến. Điều này thể hiện rằng quản trị viên trước tiên sẽ truy cập vào các chức năng chính như xem danh sách sản phẩm hoặc danh sách đơn hàng. Từ các chức năng này, tùy theo nhu cầu cụ thể, quản trị viên mới thực hiện các thao tác chi tiết như thêm, sửa, xóa hoặc cập nhật trạng thái. Cách tổ chức này phù hợp với mô hình quản trị phổ biến của các hệ thống quản lý nội dung và giúp luồng thao tác trở nên rõ ràng, logic.

3.3.6. Sơ đồ DFD

3.3.6.1 Sơ đồ DFD mức 0



Hình 3.5. Sơ đồ DFD mức 0

Mô tả sơ đồ luồng dữ liệu (DFD) mức 0:

Sơ đồ luồng dữ liệu mức 0 mô tả các tiến trình xử lý chính của hệ thống bán thức ăn nhanh, thể hiện cách thức tương tác giữa các thực thể bên ngoài và hệ thống thông qua các luồng dữ liệu cơ bản.

+ Tiến trình 1.0 – Xử lý xác thực:

Tiến trình xử lý xác thực tiếp nhận thông tin đăng nhập từ phía khách hàng. Hệ thống tiến hành kiểm tra và trả về kết quả xác thực cho khách hàng nhằm xác định quyền truy cập vào các chức năng của hệ thống.

+ Tiến trình 2.0 – Quản lý sản phẩm:

Tiến trình quản lý sản phẩm chịu trách nhiệm xử lý các yêu cầu liên quan đến thông tin món ăn.

Đối với khách hàng, hệ thống tiếp nhận yêu cầu xem sản phẩm và phản hồi bằng thông tin sản phẩm bao gồm tên món ăn, giá cả và hình ảnh minh họa.

Đối với quản trị viên, hệ thống tiếp nhận dữ liệu quản lý sản phẩm để cập nhật hoặc điều chỉnh danh mục món ăn trong hệ thống.

+ Tiến trình 3.0 – Xử lý đơn hàng:

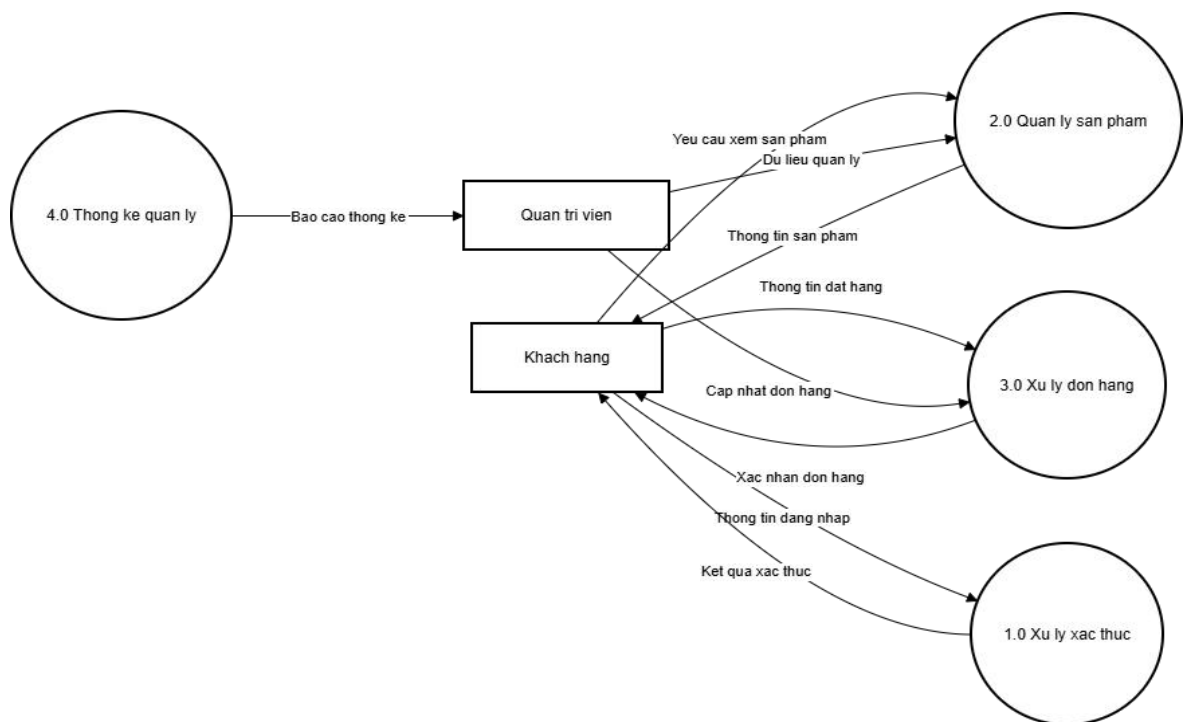
Tiến trình xử lý đơn hàng tiếp nhận thông tin đặt hàng từ khách hàng. Sau khi xử lý, hệ thống gửi xác nhận đơn hàng lại cho khách hàng.

Bên cạnh đó, hệ thống cho phép quản trị viên thực hiện các thao tác cập nhật đơn hàng, chẳng hạn như thay đổi trạng thái đơn hàng sang đang giao hoặc đã hoàn thành.

+ Tiến trình 4.0 – Thống kê và quản lý:

Tiến trình thống kê và quản lý có nhiệm vụ tổng hợp dữ liệu từ toàn bộ hoạt động của hệ thống để tạo ra các báo cáo thống kê cung cấp cho quản trị viên. Các báo cáo này hỗ trợ người quản lý theo dõi doanh thu, xác định các sản phẩm bán chạy và đánh giá hiệu quả kinh doanh của hệ thống.

3.3.6.2. Sơ đồ DFD mức 1



Hình 3.6. Sơ đồ DFD mức 1

Mô tả sơ đồ luồng dữ liệu (DFD) mức 1:

Sơ đồ luồng dữ liệu mức 1 được xây dựng nhằm chi tiết hóa các tiến trình chính của DFD mức 0, giúp làm rõ cách thức xử lý dữ liệu bên trong hệ thống bán thức ăn

nhanh. Sơ đồ thể hiện luồng thông tin giữa các tiến trình con, thực thể bên ngoài và các dữ liệu liên quan.

+ Tiến trình 1.0: Xử lý xác thực

Tiến trình xử lý xác thực tiếp nhận thông tin đăng nhập từ phía khách hàng. Hệ thống tiến hành kiểm tra thông tin xác thực dựa trên dữ liệu người dùng đã lưu trữ và trả về kết quả xác thực cho khách hàng nhằm cho phép hoặc từ chối quyền truy cập vào các chức năng chuyên sâu của hệ thống.

+ Tiến trình 2.0: Quản lý sản phẩm:

Tiến trình quản lý sản phẩm chịu trách nhiệm xử lý các yêu cầu liên quan đến thông tin món ăn trong hệ thống.

Phía khách hàng gửi yêu cầu xem sản phẩm và nhận lại thông tin sản phẩm bao gồm hình ảnh, giá cả và mô tả chi tiết.

Phía quản trị viên cung cấp dữ liệu quản lý sản phẩm để thực hiện các thao tác thêm mới, cập nhật hoặc chỉnh sửa danh mục món ăn.

+ Tiến trình 3.0: Xử lý đơn hàng:

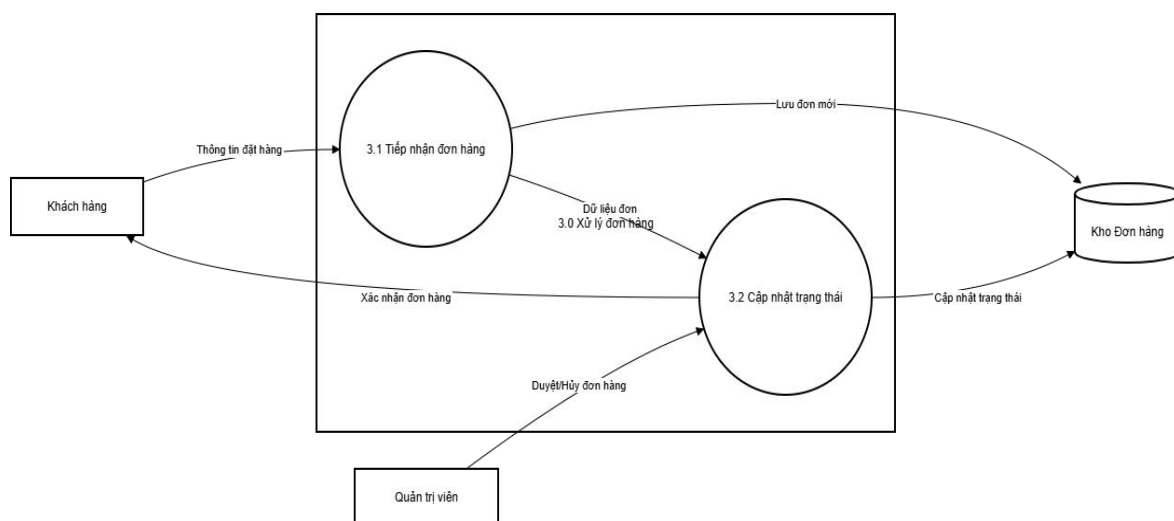
Tiến trình xử lý đơn hàng tiếp nhận thông tin đặt hàng từ khách hàng, bao gồm danh sách món ăn, địa chỉ giao hàng và số điện thoại liên hệ. Sau khi hệ thống xử lý, xác nhận đơn hàng được gửi lại cho khách hàng.

Đồng thời, quản trị viên có thể thực hiện các thao tác cập nhật đơn hàng, bao gồm thay đổi trạng thái đơn hàng như đang chuẩn bị, đang giao hoặc đã hoàn thành, nhằm đảm bảo khách hàng theo dõi được tình trạng đơn hàng.

+ Tiến trình 4.0: Thống kê và quản lý:

Tiến trình thống kê và quản lý có nhiệm vụ tổng hợp dữ liệu từ các tiến trình xử lý trong hệ thống. Kết quả của tiến trình này là các báo cáo thống kê như doanh thu, danh sách sản phẩm bán chạy và biểu đồ tăng trưởng, phục vụ cho công tác theo dõi và ra quyết định của quản trị viên.

Sơ đồ luồng dữ liệu mức 1 giúp mô tả chi tiết cách thức dữ liệu được xử lý và luân chuyển giữa các tiến trình bên trong hệ thống, từ đó làm rõ mối quan hệ giữa khách hàng, quản trị viên và các chức năng của hệ thống.

3.3.6.3. Sơ đồ DFD mức 2**Hình 3.7. Sơ đồ DFD mức 2**

Các thực thể và kho dữ liệu:

Khách hàng: Là thực thể bên ngoài trực tiếp khởi tạo quy trình bằng cách gửi thông tin đặt hàng vào hệ thống.

Quản trị viên: Là thực thể có quyền can thiệp vào quy trình xử lý đơn hàng nhằm kiểm soát và cập nhật trạng thái đơn hàng.

Kho đơn hàng: Là nơi lưu trữ toàn bộ dữ liệu liên quan đến các đơn hàng của hệ thống, bao gồm thông tin đơn hàng và trạng thái xử lý.

Các tiến trình con: Tiến trình 3.0 – Xử lý đơn hàng được phân rã thành các tiến trình con như sau:

+ Tiếp nhận đơn hàng:

Tiến trình này tiếp nhận thông tin đặt hàng từ phía khách hàng. Sau khi kiểm tra dữ liệu hợp lệ, hệ thống thực hiện thao tác lưu đơn hàng mới vào kho đơn hàng để bắt đầu quá trình theo dõi và xử lý.

Dữ liệu sau khi được tiếp nhận sẽ được chuyển tiếp sang tiến trình tiếp theo thông qua luồng dữ liệu phục vụ cho quá trình xử lý đơn hàng.

+ Cập nhật trạng thái đơn hàng:

Tiến trình cập nhật trạng thái đơn hàng thể hiện sự tương tác giữa hệ thống và quản trị viên. Quản trị viên thực hiện các thao tác như duyệt hoặc hủy đơn hàng dựa trên tình trạng thực tế.

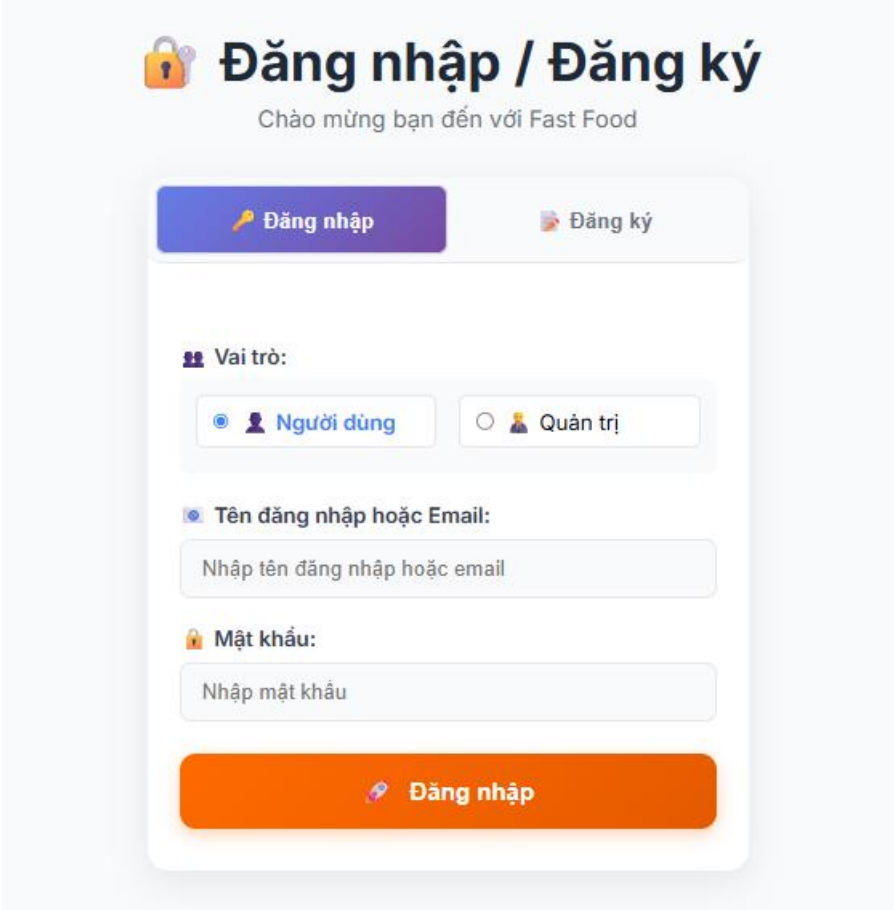
Sau khi có sự thay đổi, hệ thống tiến hành cập nhật trạng thái đơn hàng vào kho đơn hàng để đảm bảo dữ liệu luôn được đồng bộ. Đồng thời, hệ thống gửi xác nhận đơn hàng phản hồi lại cho khách hàng, giúp khách hàng nắm được tình trạng xử lý đơn hàng của mình.

Sơ đồ luồng dữ liệu mức 2 giúp làm rõ chi tiết cách thức tiến trình xử lý đơn hàng được thực hiện bên trong hệ thống, đồng thời thể hiện rõ vai trò của khách hàng, quản trị viên và kho dữ liệu trong toàn bộ quy trình.

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

4.1 Giao diện của người dùng

4.1.1. Giao diện đăng nhập

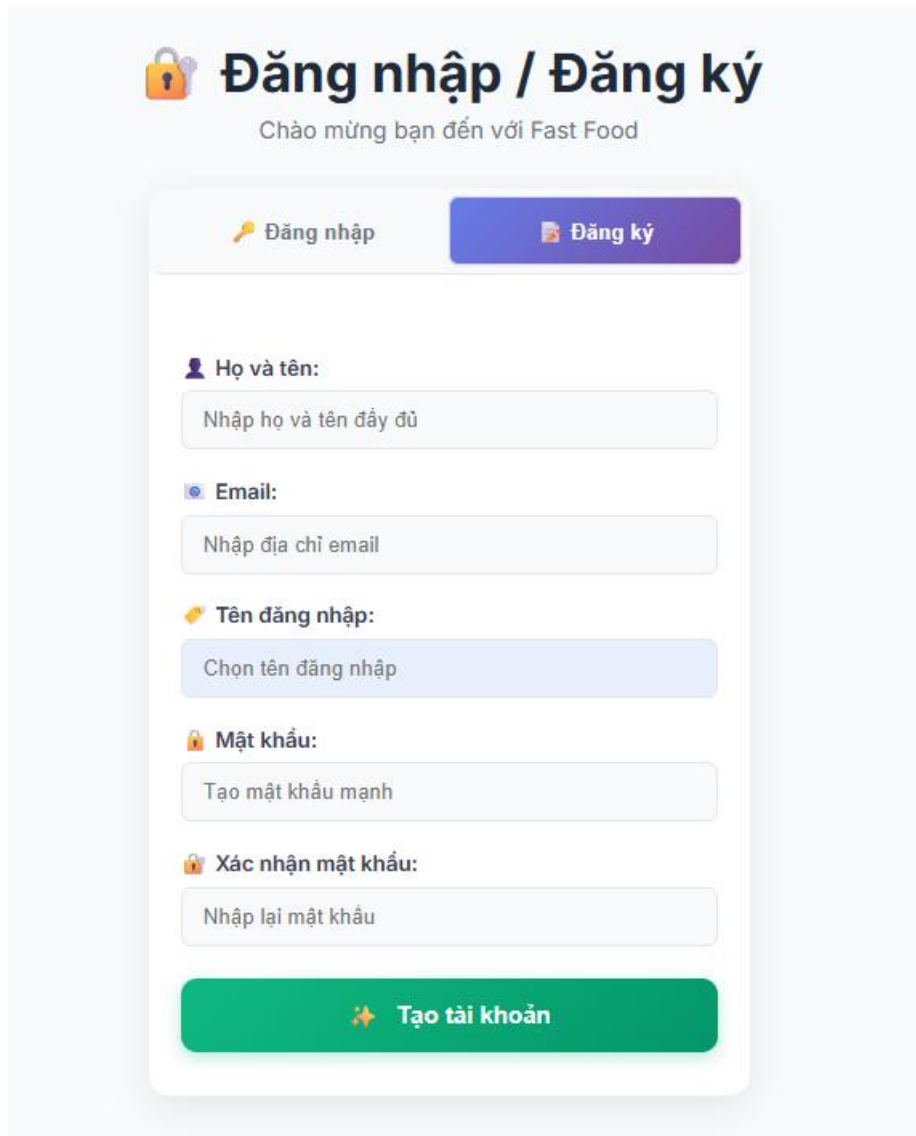


The screenshot shows a web interface for a fast-food system. At the top, there's a title "Đăng nhập / Đăng ký" (Login / Register) with a subtitle "Chào mừng bạn đến với Fast Food" (Welcome to Fast Food). Below this, there are two tabs: "Đăng nhập" (Login) and "Đăng ký" (Register). The "Đăng nhập" tab is active. Under the tabs, there's a section for "Vai trò:" (Role) with two radio buttons: "Người dùng" (User) and "Quản trị" (Admin). The "Người dùng" option is selected. Below the role selection, there's a label "Tên đăng nhập hoặc Email:" (Username or Email) and a text input field with the placeholder "Nhập tên đăng nhập hoặc email". Below that, there's a label "Mật khẩu:" (Password) and a text input field with the placeholder "Nhập mật khẩu". At the bottom, there's a large orange button labeled "Đăng nhập" (Login).

Hình 4.1. Giao diện đăng nhập

Giao diện đăng nhập là chức năng xác thực, cho phép người dùng truy cập vào hệ thống website bán thức ăn nhanh. Người dùng chọn vai trò *Người dùng*, nhập tên đăng nhập hoặc email và mật khẩu, sau đó nhấn nút Đăng nhập. Hệ thống kiểm tra thông tin và chuyển người dùng vào giao diện chính nếu xác thực thành công.

4.1.2. Giao diện đăng ký

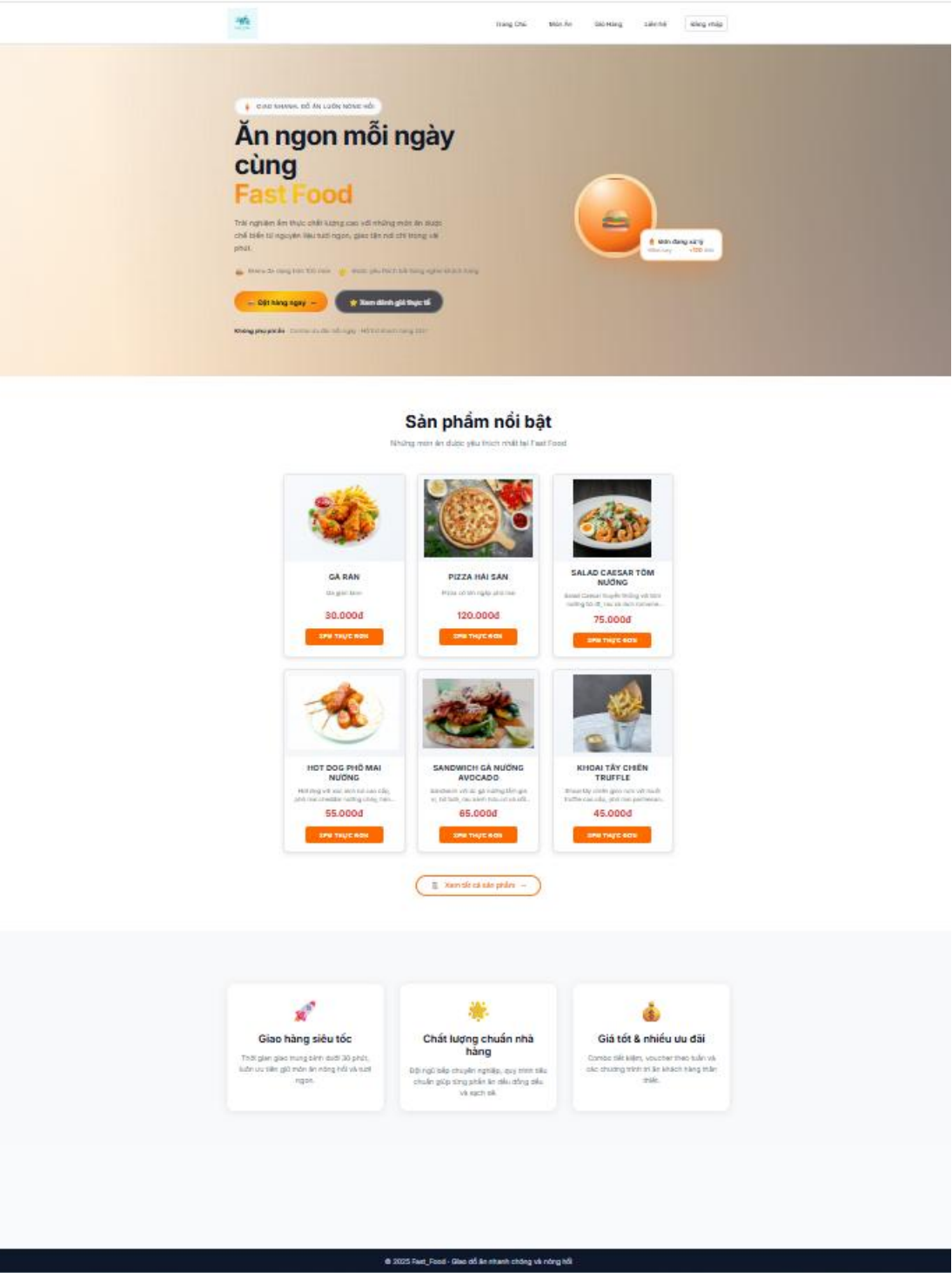


The screenshot displays a registration form titled "Đăng nhập / Đăng ký" (Login / Register) with the subtitle "Chào mừng bạn đến với Fast Food" (Welcome to Fast Food). The form is divided into two main sections: "Đăng nhập" (Login) and "Đăng ký" (Register). The "Đăng ký" section is highlighted with a purple background. The registration fields include: "Họ và tên:" (Full name) with a placeholder "Nhập họ và tên đầy đủ"; "Email:" with a placeholder "Nhập địa chỉ email"; "Tên đăng nhập:" (Username) with a placeholder "Chọn tên đăng nhập"; "Mật khẩu:" (Password) with a placeholder "Tạo mật khẩu mạnh"; and "Xác nhận mật khẩu:" (Confirm password) with a placeholder "Nhập lại mật khẩu". A green button labeled "Tạo tài khoản" (Create account) is located at the bottom of the registration section.

Hình 4.2. Giao diện đăng kí

Giao diện đăng ký tài khoản cho phép người dùng mới tạo tài khoản để sử dụng hệ thống Fast Food. Người dùng nhập các thông tin cơ bản gồm họ và tên, email, tên đăng nhập và mật khẩu (kèm xác nhận mật khẩu), sau đó nhấn nút Tạo tài khoản. Hệ thống kiểm tra tính hợp lệ của dữ liệu, lưu thông tin vào cơ sở dữ liệu và cho phép người dùng đăng nhập vào hệ thống với vai trò Người dùng.

4.1.3. Giao diện Trang chủ

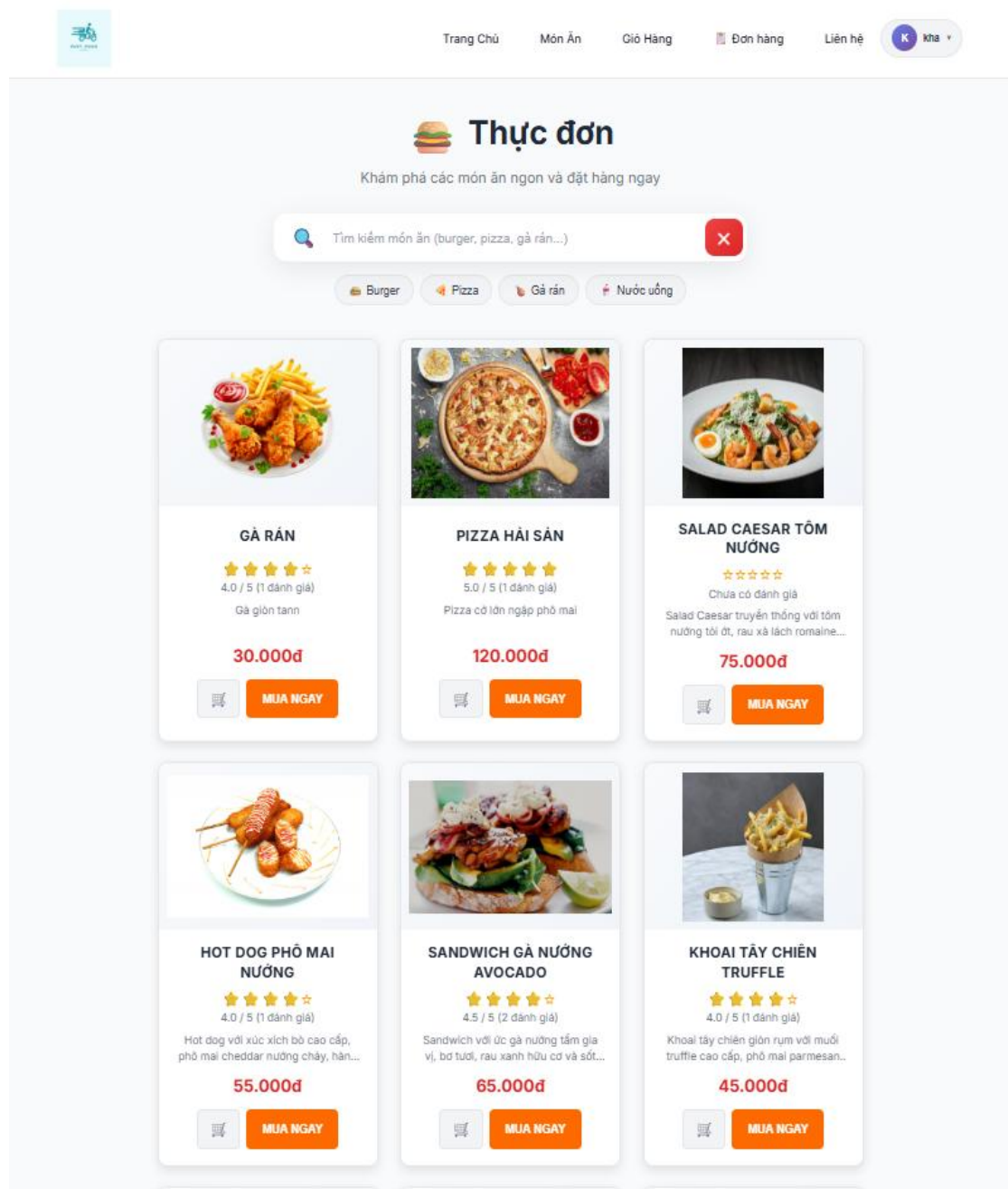


Hình 4.3. Giao diện trang chủ

Giao diện trang chủ là nơi cung cấp cái nhìn tổng quan về hệ thống Fast Food và các sản phẩm nổi bật. Trang chủ hiển thị banner giới thiệu, thanh điều hướng và danh sách sản phẩm tiêu biểu kèm hình ảnh, tên món và giá bán. Người dùng có thể dễ dàng truy cập các chức năng như xem thực đơn, giỏ hàng, liên hệ và đăng nhập/đăng

ký. Từ trang chủ, khách hàng có thể lựa chọn sản phẩm và tiếp tục thực hiện quy trình đặt hàng.

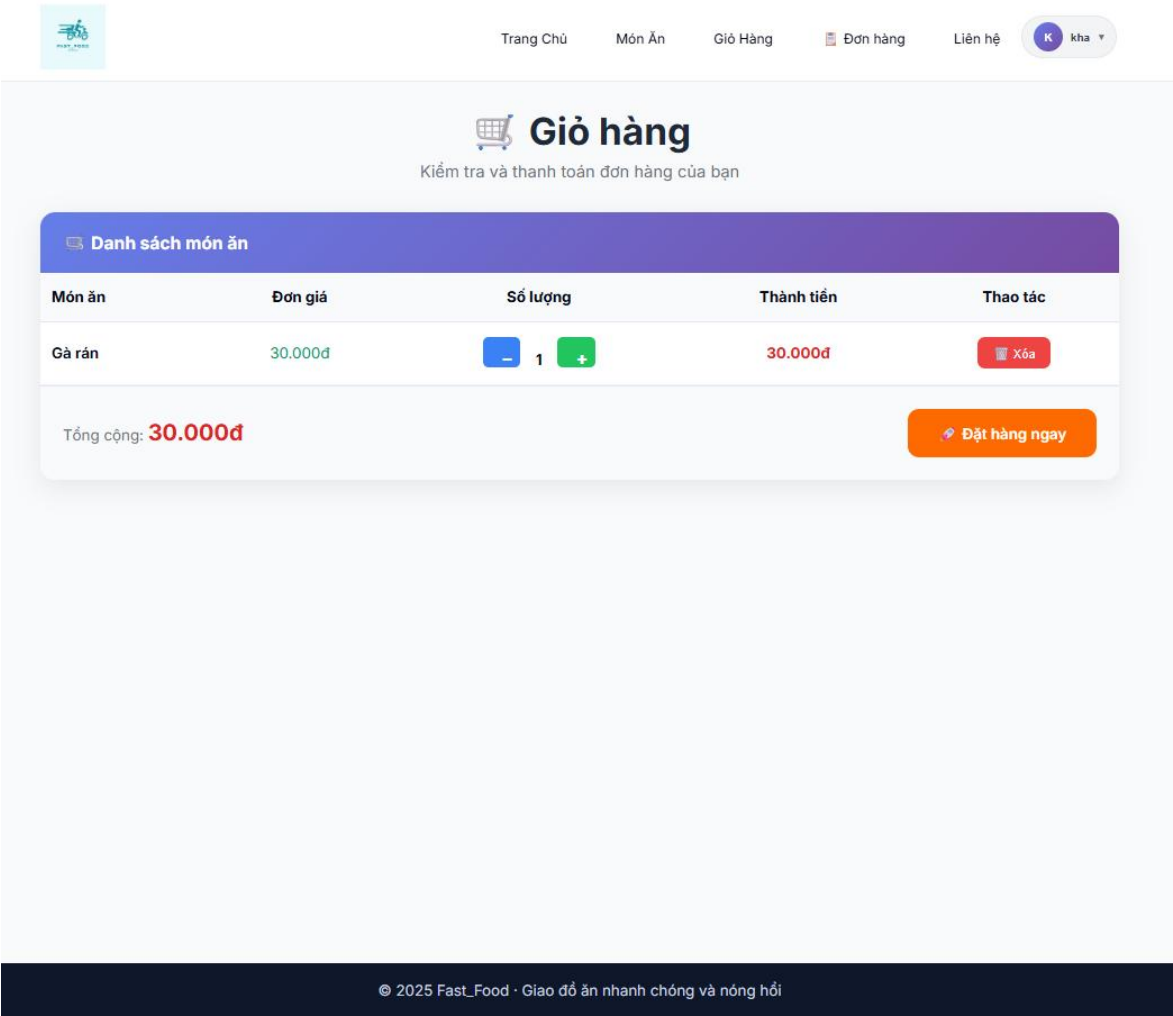
4.1.4. Giao diện Món ăn



Hình 4.4. Giao diện món ăn

Giao diện thực đơn là nơi hiển thị danh sách các sản phẩm hiện có, hỗ trợ người dùng tìm kiếm và lựa chọn món ăn để thực hiện đặt hàng. Giao diện được thiết kế trực quan, giúp người dùng dễ dàng tiếp cận thông tin sản phẩm.

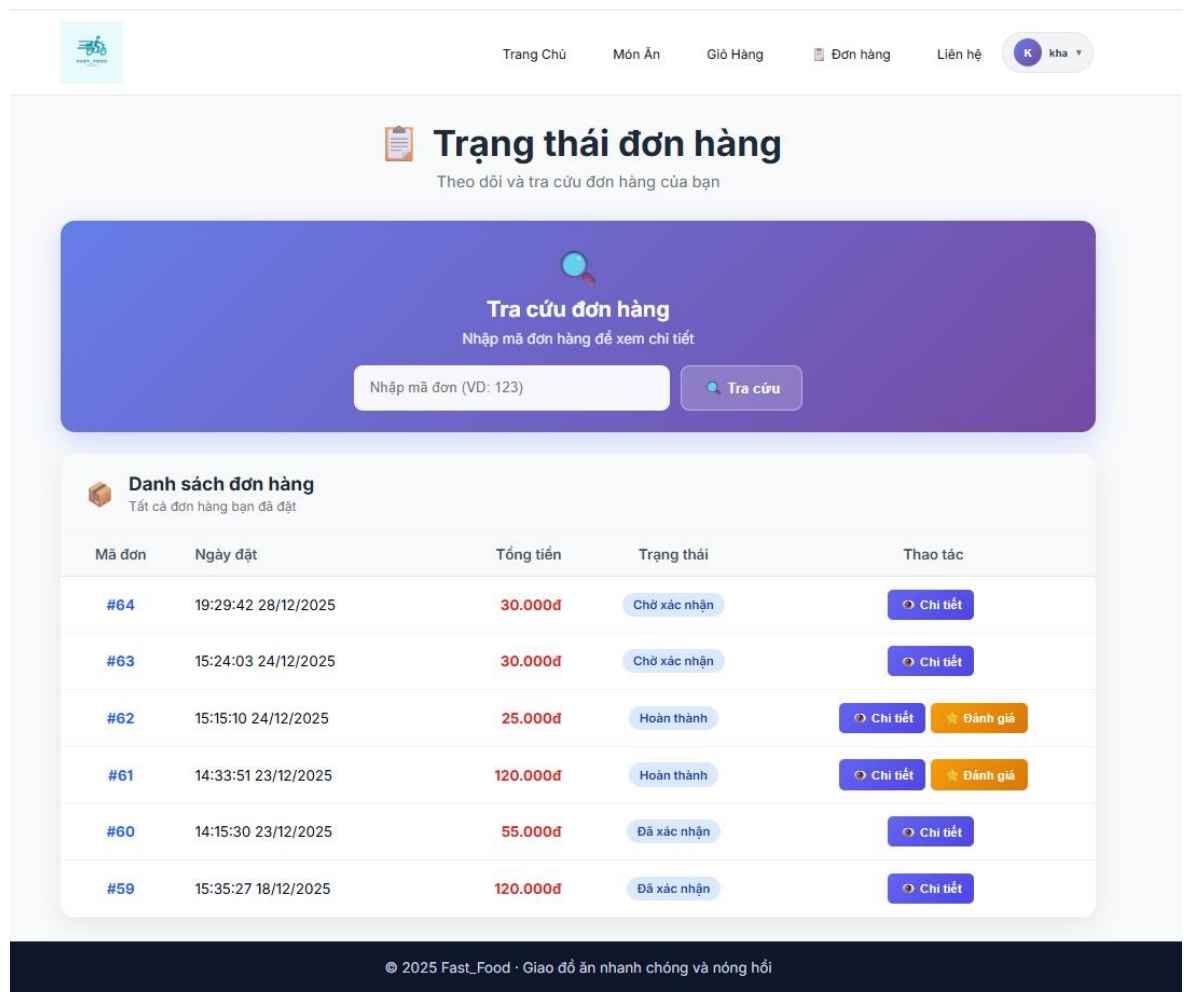
4.1.5. Giao diện Giỏ hàng



Hình 4.5. Giao diện giỏ hàng

Giao diện giỏ hàng cho phép người dùng xem lại các sản phẩm đã chọn trước khi tiến hành thanh toán, đồng thời hỗ trợ điều chỉnh số lượng, xóa sản phẩm và theo dõi tổng giá trị đơn hàng. Sau khi kiểm tra thông tin, người dùng có thể thực hiện đặt hàng để chuyển sang bước xác nhận và hoàn tất giao dịch.

4.1.6. Giao diện Đơn hàng



Hình 4.6. Giao diện đơn hàng

Giao diện trạng thái đơn hàng cho phép người dùng theo dõi tiến độ xử lý các đơn hàng đã đặt thông qua danh sách lịch sử giao dịch và công cụ tra cứu theo mã đơn. Thông tin hiển thị bao gồm thời gian đặt hàng, tổng tiền và trạng thái hiện tại của đơn, giúp khách hàng chủ động nắm bắt quá trình xử lý. Ngoài ra, hệ thống hỗ trợ xem chi tiết và đánh giá đơn hàng sau khi hoàn thành, góp phần nâng cao tính minh bạch và trải nghiệm người dùng.

4.1.7. Giao diện Liên hệ

Liên hệ với chúng tôi

Nếu bạn có thắc mắc về đơn hàng, góp ý về món ăn hoặc cần hỗ trợ, hãy gửi thông tin cho admin qua form dưới đây.

Họ và tên *

Email *

Số điện thoại

Nội dung *

Gửi liên hệ

Thông tin liên hệ khác:
Email: admin@fastfood.local

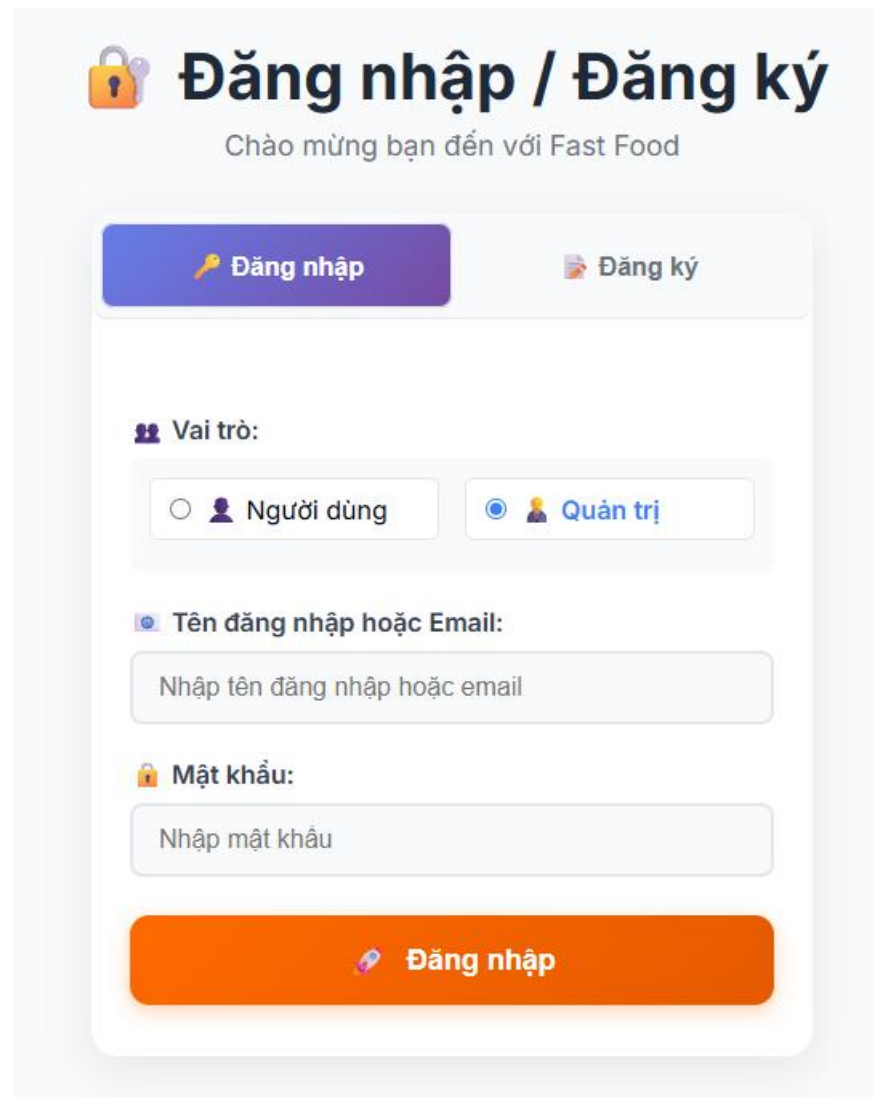
© 2025 Fast_Food · Giao đồ ăn nhanh chóng và nóng hổi

Hình 4.7. Giao diện liên hệ

Giao diện liên hệ cho phép người dùng gửi thắc mắc, góp ý hoặc yêu cầu hỗ trợ trực tiếp đến quản trị viên thông qua biểu mẫu thu thập thông tin cơ bản như họ tên, email, số điện thoại và nội dung liên hệ. Sau khi gửi, dữ liệu được chuyển đến hệ thống quản lý để xử lý và phản hồi kịp thời. Ngoài ra, hệ thống cung cấp địa chỉ email hỗ trợ trực tiếp nhằm tăng tính linh hoạt trong giao tiếp, góp phần nâng cao chất lượng dịch vụ và sự hài lòng của khách hàng.

4.2. Giao diện của quản trị viên

4.2.1. Giao diện đăng nhập



Đăng nhập / Đăng ký
Chào mừng bạn đến với Fast Food

Đăng nhập **Đăng ký**

Vai trò:

☐ Người dùng ☒ Quản trị

Tên đăng nhập hoặc Email:
Nhập tên đăng nhập hoặc email

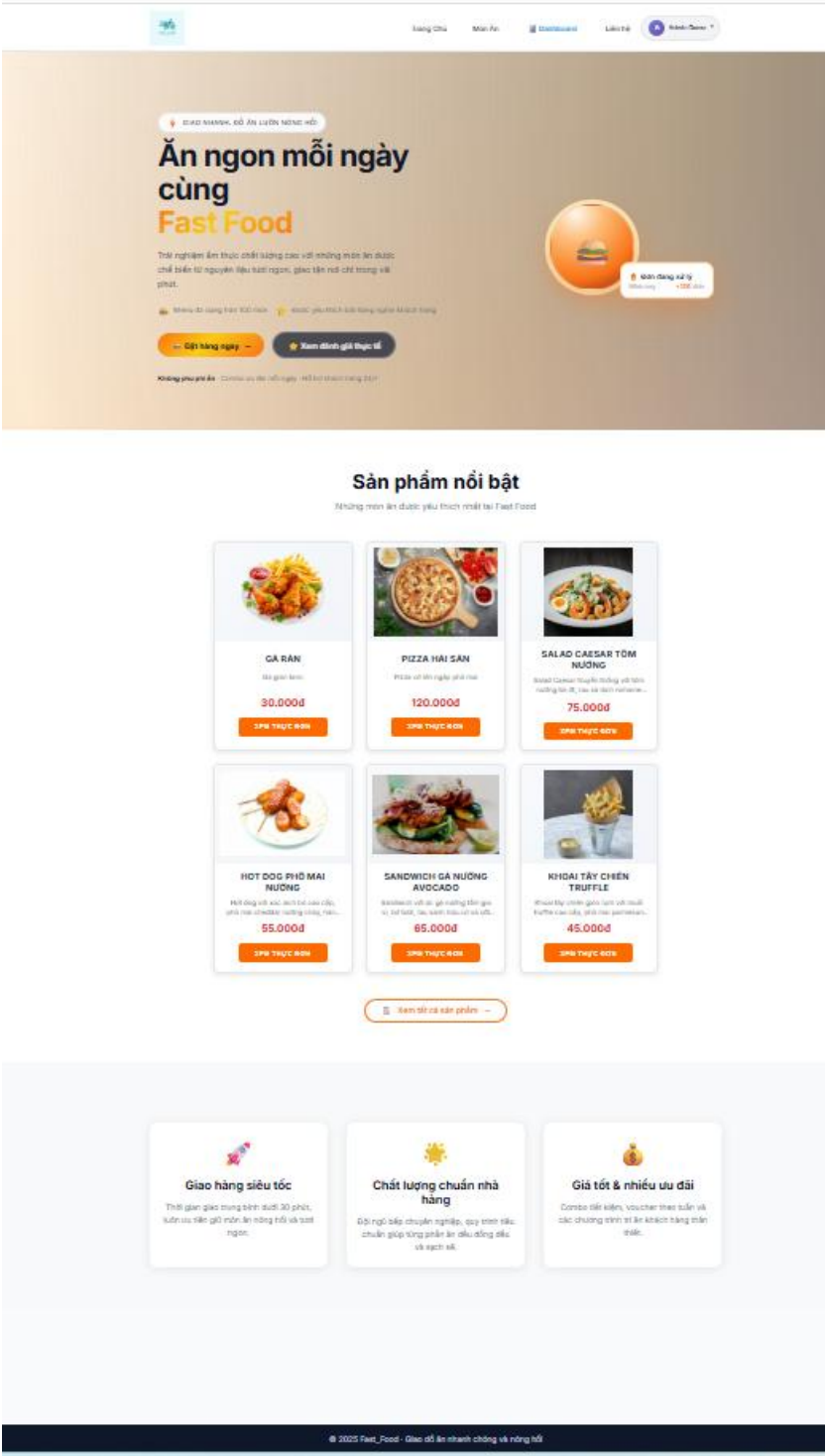
Mật khẩu:
Nhập mật khẩu

Đăng nhập

Hình 4.8. Giao diện đăng nhập

Giao diện đăng nhập quản trị cho phép người dùng có quyền điều hành truy cập vào hệ thống với vai trò Quản trị viên. Tại đây, người dùng lựa chọn vai trò “Quản trị”, nhập tên đăng nhập hoặc email và mật khẩu để thực hiện xác thực. Sau khi thông tin hợp lệ, hệ thống sẽ điều hướng trực tiếp trang chủ và có thể mở trang Dashboard quản trị. Cơ chế đăng nhập theo vai trò này giúp đảm bảo tính bảo mật, kiểm soát quyền truy cập và an toàn dữ liệu cho toàn bộ hệ thống Fast Food.

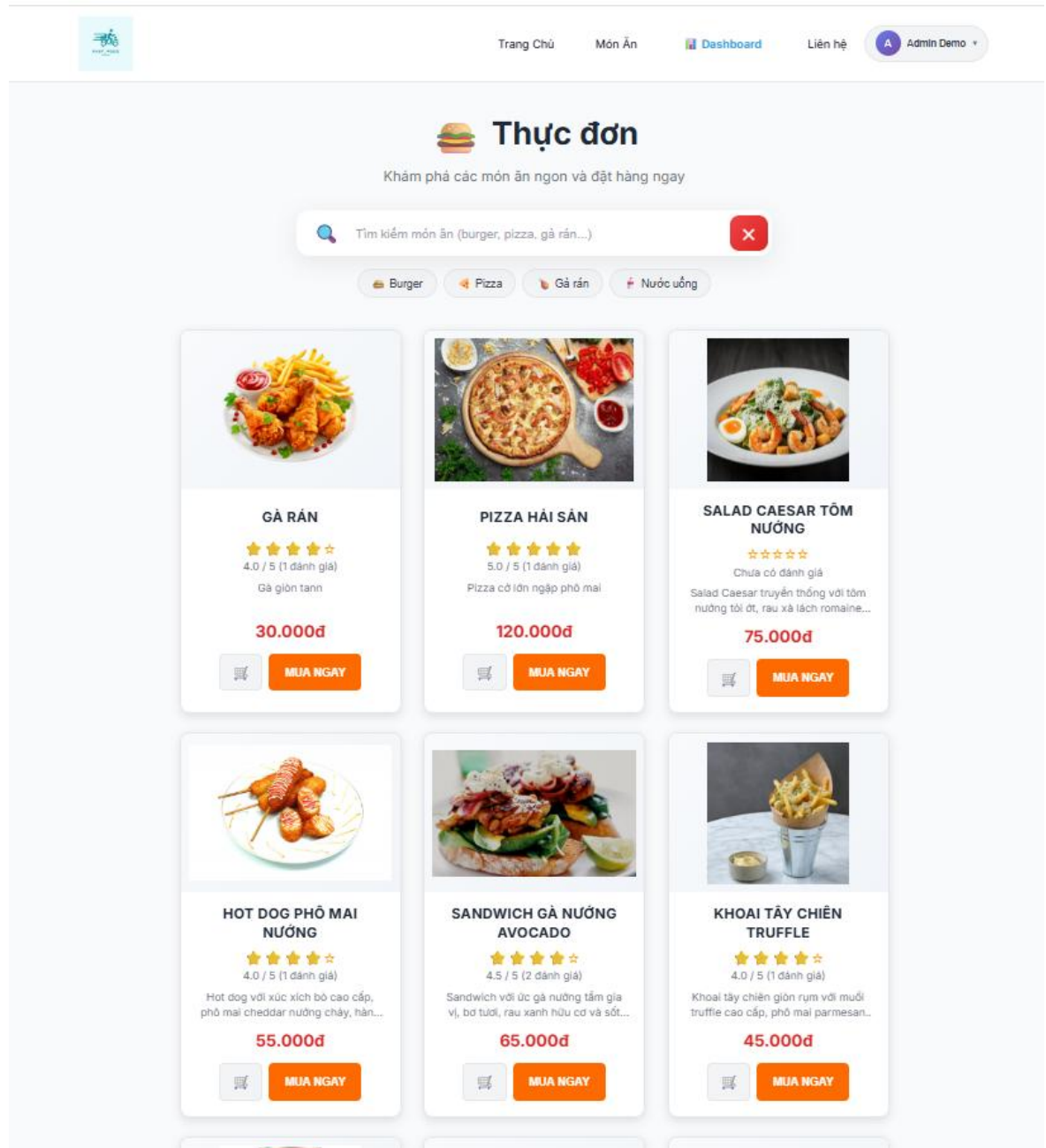
4.2.2. Giao diện Trang chủ



Hình 4.9. Giao diện trang chủ

Ở trang món ăn này cũng giống của người dùng admin có trang chủ giống nhau ,nhưng có thêm trang Dashboard .

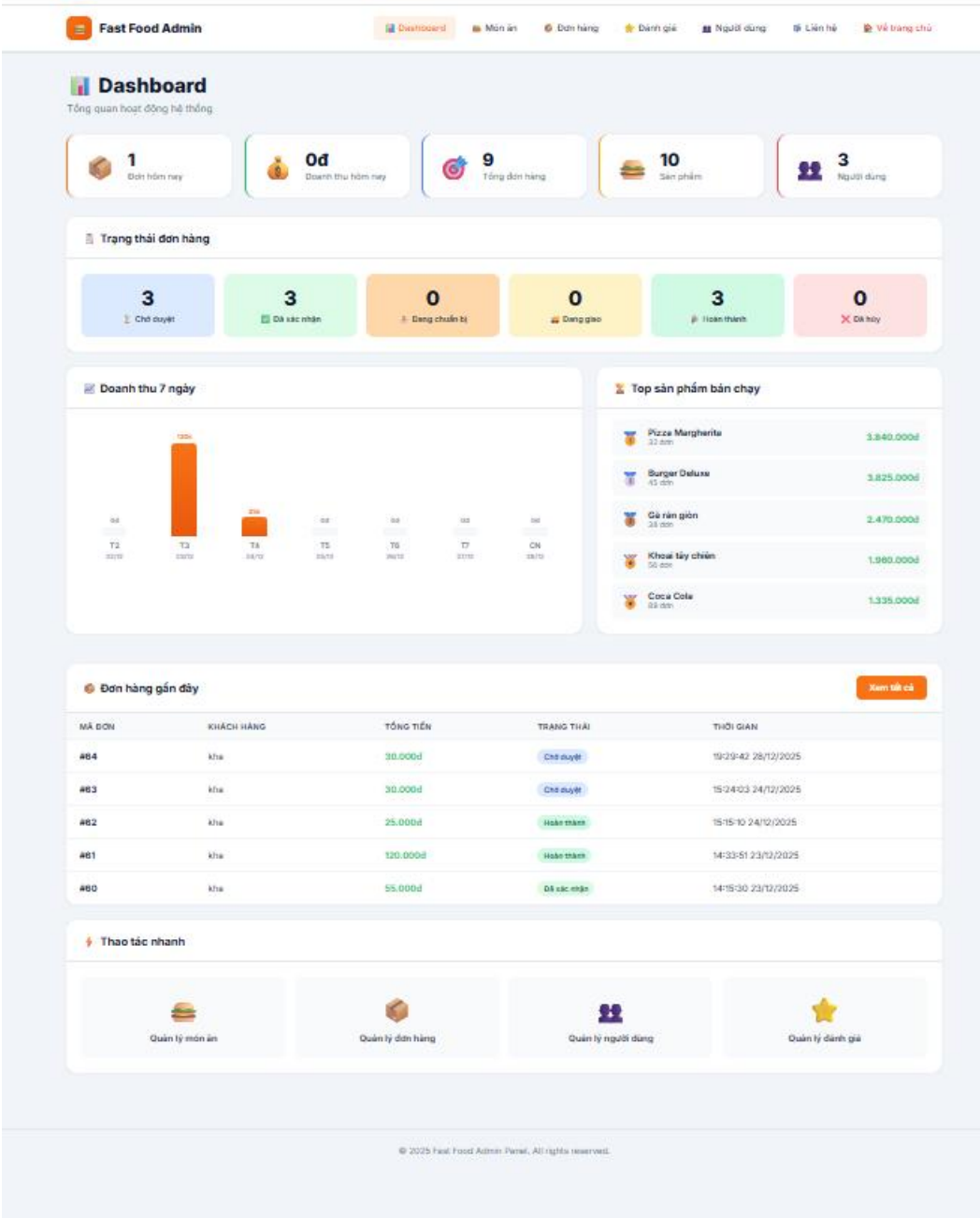
4.2.3. Giao diện Món ăn



Hình 4.10. Giao diện món ăn

Ở trang món ăn này cũng giống của người dùng, admin có thể xem được những đánh giá của sản phẩm và món ăn nhưng không thêm vào giỏ hàng được. Việc này để thuận lợi cho admin khi thêm sản phẩm và admin theo dõi sản phẩm có bị lỗi gì không.

4.2.4. Giao diện Dashboard



Hình 4.11. Giao diện Dashboard

Giao diện Dashboard là trung tâm quản lý của hệ thống, cung cấp cái nhìn tổng quan về tình hình hoạt động kinh doanh Fast Food thông qua các chỉ số và biểu đồ thống kê trực quan. Tại đây, quản trị viên có thể theo dõi nhanh số lượng đơn hàng, doanh thu, trạng thái xử lý đơn, danh sách sản phẩm bán chạy và các giao dịch gần nhất. Dữ liệu được cập nhật theo thời gian thực, hỗ trợ việc giám sát hiệu suất vận hành và đưa ra quyết định quản lý kịp thời. Bố cục dạng thẻ giúp thông tin được trình bày khoa học, dễ quan sát và thuận tiện cho thao tác quản trị.

4.2.5.1 Giao diện thêm Món ăn

Fast Food Admin

Dashboard

Món ăn

Đơn hàng

Đánh giá

Người dùng

Liên hệ

Về trang chủ

Quản lý món ăn

Thêm, sửa và quản lý menu món ăn trong hệ thống

Thêm món ăn mới

Tên món

Nhập tên món ăn

Giá (đ)

Nhập giá

Mô tả món ăn

Nhập mô tả chi tiết về món ăn...

Ảnh món ăn

Chọn tệp

Không có tệp nào được chọn

Thêm món

Danh sách món ăn

10 món

ID	TÊN MÓN	GIÁ	MÔ TẢ	ẢNH	TRẠNG THÁI	HÀNH ĐỘNG
#10	<div>Gà rán</div>	<div>30000</div>	<div>Gà giòn tan</div>	<div></div> <div>Đổi ảnh</div>	<div>Đang bán</div>	<div>Lưu</div> <div>Xóa</div>
#9	<div>Pizza Hải Sản</div>	<div>120000</div>	<div>Pizza cỡ lớn ngập phô mai</div>	<div></div> <div>Đổi ảnh</div>	<div>Đang bán</div>	<div>Lưu</div> <div>Xóa</div>

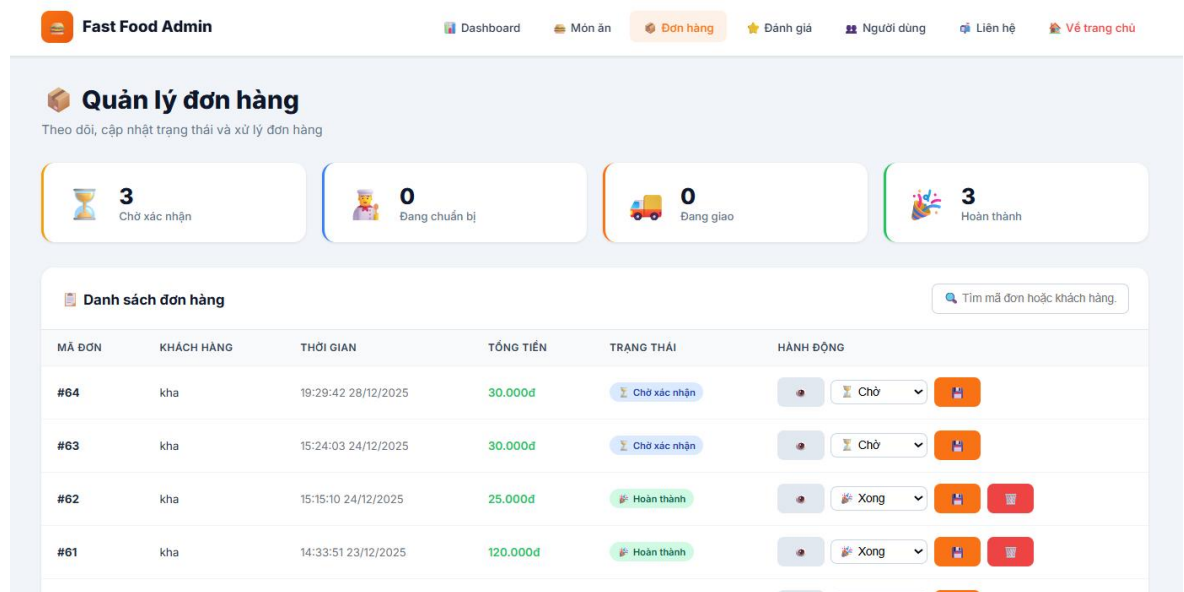
Hình 4.12. Giao diện thêm món ăn

Giao diện quản trị danh mục món ăn cho phép quản trị viên thực hiện đầy đủ các nghiệp vụ quản lý món ăn trong hệ thống. Giao diện được chia thành hai phân vùng chính gồm khu vực thêm món mới với các trường thông tin cơ bản như tên món, giá bán, mô tả chi tiết và hình ảnh minh họa; cùng khu vực danh sách món ăn hiển thị toàn bộ sản phẩm đang kinh doanh dưới dạng bảng. Tại đây, quản trị viên có thể theo dõi thông tin từng món, chỉnh sửa dữ liệu, thay đổi trạng thái kinh doanh hoặc xóa sản phẩm khi cần thiết. Cách tổ chức này giúp việc cập nhật và kiểm soát thực đơn diễn ra nhanh chóng, chính xác và đảm bảo tính nhất quán giữa dữ liệu quản trị và thông tin hiển thị cho người dùng.

Đào Công Duy

63

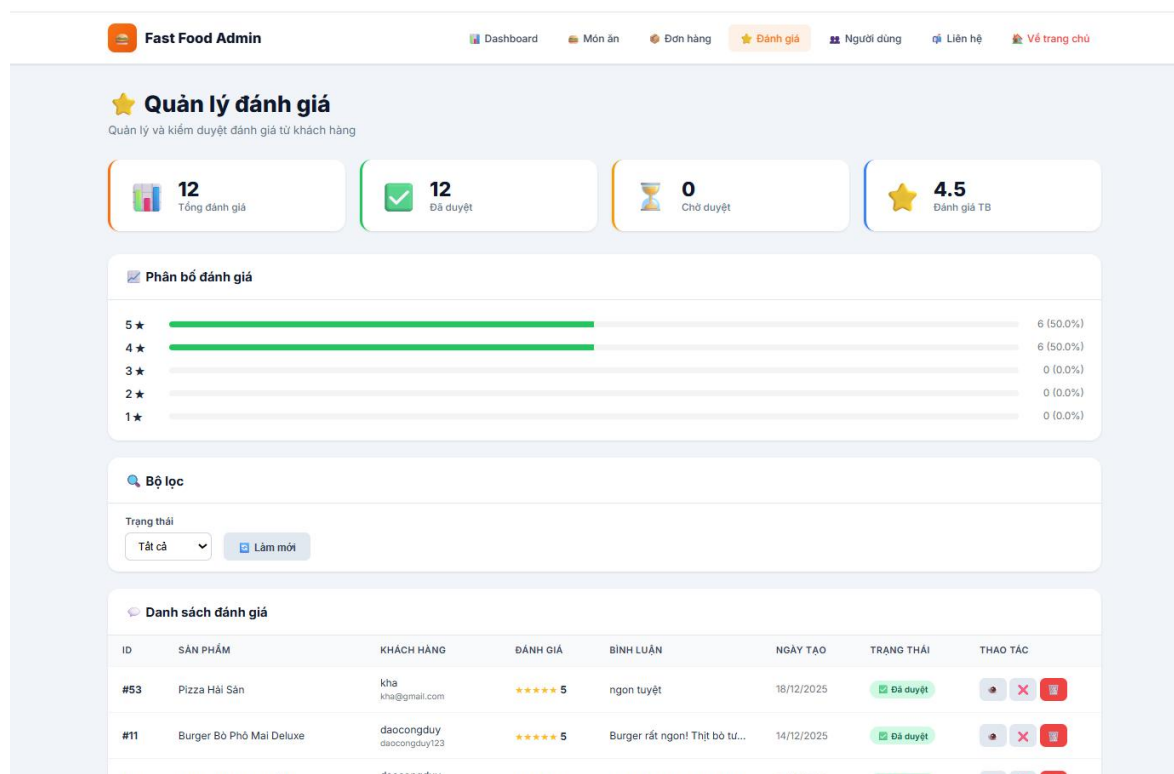
4.2.5.2. Giao diện đơn hàng



Hình 4.13. Giao diện đơn hàng

Giao diện quản lý đơn hàng cho phép quản trị viên theo dõi và điều phối toàn bộ quy trình xử lý đơn hàng trên hệ thống thông qua các chỉ số trạng thái tức thời và bảng dữ liệu chi tiết. Hệ thống hiển thị số lượng đơn hàng theo từng giai đoạn như chờ xác nhận, đang chuẩn bị, đang giao và đã hoàn thành, giúp quản trị viên nắm bắt nhanh tình hình vận hành. Danh sách đơn hàng được trình bày dưới dạng bảng với các thông tin chính gồm mã đơn, tên khách hàng, thời gian giao dịch, tổng tiền và trạng thái hiện tại. Ngoài ra, giao diện hỗ trợ tìm kiếm đơn hàng theo mã hoặc tên khách hàng, cho phép cập nhật trạng thái xử lý, xem chi tiết hoặc xóa các đơn hàng phát sinh lỗi. Việc tập trung các chức năng quản lý tại một giao diện duy nhất giúp nâng cao hiệu quả điều hành và đảm bảo tính chính xác trong quá trình xử lý đơn hàng và thống kê doanh thu.

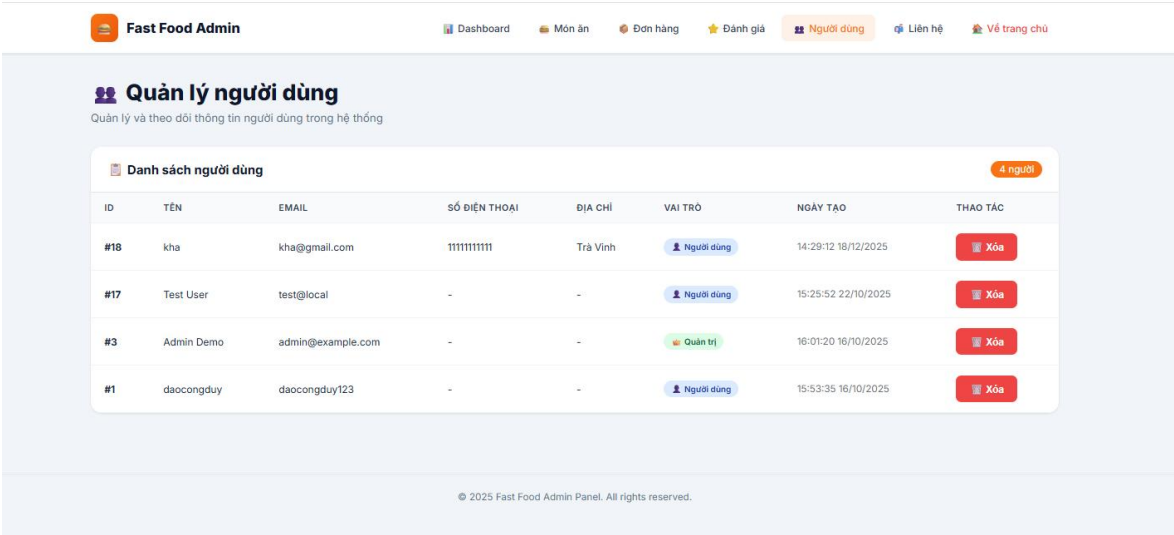
4.2.5.3. Giao diện đánh giá



Hình 4.14. Giao diện đánh giá

Giao diện thống kê và kiểm duyệt phản hồi cho phép quản trị viên theo dõi tổng quan mức độ hài lòng của khách hàng thông qua các chỉ số như tổng số đánh giá, số phản hồi đã duyệt, chờ duyệt và điểm đánh giá trung bình của hệ thống. Dữ liệu được trực quan hóa bằng biểu đồ phân bố xếp hạng từ 1 đến 5 sao, hỗ trợ việc phân tích xu hướng phản hồi. Danh sách đánh giá được hiển thị dưới dạng bảng chi tiết theo sản phẩm, khách hàng, số sao, nội dung bình luận, thời gian và trạng thái. Ngoài ra, giao diện tích hợp các chức năng tìm kiếm, lọc trạng thái và thao tác kiểm duyệt như xem chi tiết, phê duyệt, hủy bỏ hoặc xóa đánh giá, góp phần đảm bảo tính minh bạch và nâng cao chất lượng dịch vụ của hệ thống Fast Food.

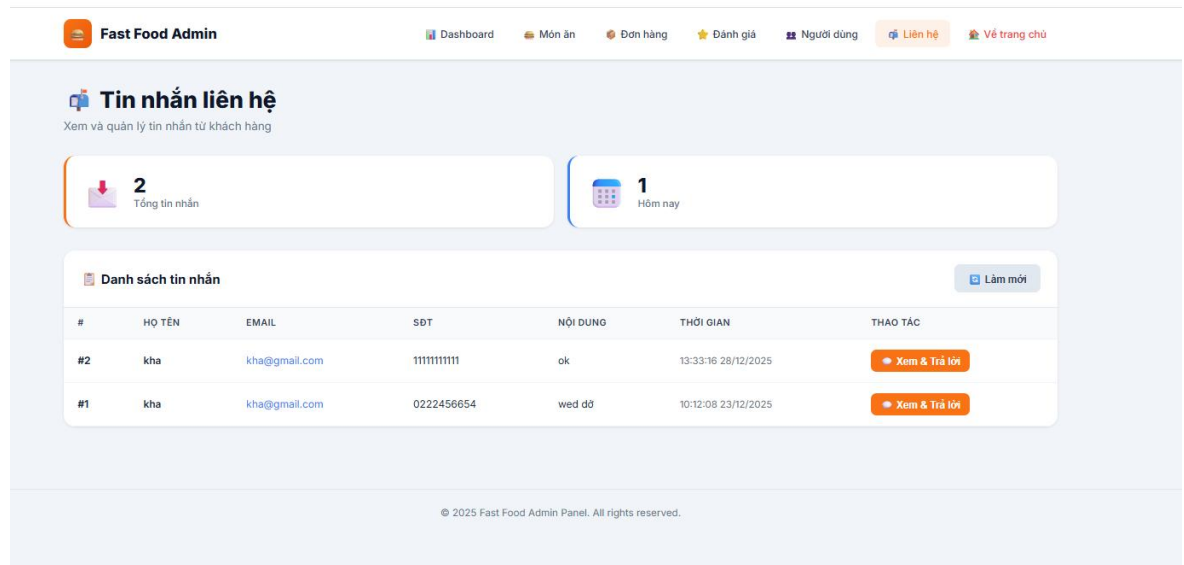
4.2.5.4. Giao diện người dùng



Hình 4.15. Giao diện quản lý khách hàng (người dùng)

Giao diện quản lý người dùng cho phép quản trị viên theo dõi và kiểm soát toàn bộ danh sách tài khoản tham gia hệ thống Fast Food. Thông tin người dùng được hiển thị dưới dạng bảng dữ liệu gồm mã định danh, thông tin cá nhân, vai trò truy cập và thời điểm tạo tài khoản, giúp quản trị viên dễ dàng nắm bắt và phân loại đối tượng sử dụng hệ thống. Ngoài ra, giao diện cung cấp thống kê tổng số tài khoản hiện có và cho phép thực hiện thao tác xóa các tài khoản không hợp lệ hoặc vi phạm quy định. Việc quản lý tập trung phân hệ người dùng góp phần nâng cao tính bảo mật, đảm bảo phân quyền rõ ràng và hỗ trợ hiệu quả công tác vận hành hệ thống.

4.2.5.5. Giao diện liên hệ



Hình 4.16. Giao diện liên hệ

Giao diện quản lý tin nhắn liên hệ cho phép quản trị viên tiếp nhận, theo dõi và phản hồi các yêu cầu hỗ trợ hoặc góp ý từ phía người dùng trong quá trình sử dụng hệ thống Fast Food. Giao diện cung cấp các chỉ số tổng quan như tổng số tin nhắn và số tin nhắn phát sinh trong ngày, giúp quản trị viên chủ động phân bổ và ưu tiên xử lý công việc. Danh sách tin nhắn được hiển thị dưới dạng bảng với các thông tin gồm họ tên, email, số điện thoại, nội dung và thời gian gửi. Thông qua chức năng xem và phản hồi, quản trị viên có thể nhanh chóng tiếp cận chi tiết yêu cầu và thực hiện hỗ trợ kịp thời, góp phần nâng cao chất lượng dịch vụ và mức độ hài lòng của khách hàng.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1.Kết luận

Trong quá trình thực hiện đồ án, hệ thống website bán thức ăn nhanh đã được tiến hành phân tích yêu cầu, thiết kế và xây dựng dựa trên mô hình kinh doanh trực tuyến thực tế. Hệ thống được phát triển nhằm đáp ứng các nhu cầu cơ bản của người dùng và nhà quản lý, bao gồm các chức năng quản lý sản phẩm, đặt hàng trực tuyến, theo dõi trạng thái đơn hàng, quản lý người dùng, quản lý đánh giá và tiếp nhận liên hệ hỗ trợ.

Bên cạnh các chức năng nghiệp vụ chính, giao diện người dùng và giao diện quản trị được thiết kế theo hướng trực quan, dễ sử dụng, giúp nâng cao trải nghiệm cho khách hàng cũng như hỗ trợ quản trị viên trong công tác điều hành, thống kê và kiểm soát dữ liệu hiệu quả. Thông qua quá trình triển khai và hoàn thiện, hệ thống cho thấy tính ổn định, khả thi và phù hợp với yêu cầu của một website bán thức ăn nhanh trong thực tế, góp phần nâng cao hiệu quả quản lý, tối ưu quy trình vận hành và cải thiện chất lượng dịch vụ cho người sử dụng.

5.2.Hướng phát triển

Trong thời gian tới, hệ thống website bán thức ăn nhanh có thể tiếp tục được mở rộng và hoàn thiện nhằm đáp ứng tốt hơn nhu cầu thực tế của người dùng và doanh nghiệp. Trước hết, hệ thống có thể tích hợp thêm các phương thức thanh toán trực tuyến phổ biến như ví điện tử, thẻ ngân hàng hoặc QR Code để tăng tính tiện lợi trong quá trình đặt hàng.

Bên cạnh đó, việc phát triển các chức năng nâng cao như quản lý khuyến mãi, mã giảm giá, thông báo trạng thái đơn hàng theo thời gian thực và lịch sử mua hàng chi tiết sẽ giúp nâng cao trải nghiệm người dùng. Hệ thống cũng có thể được mở rộng sang nền tảng thiết bị di động thông qua ứng dụng Mobile App nhằm tăng khả năng tiếp cận khách hàng.

Ngoài ra, việc áp dụng các công nghệ phân tích dữ liệu và trí tuệ nhân tạo để gợi ý món ăn, dự đoán xu hướng tiêu dùng hoặc tối ưu quy trình quản lý đơn hàng sẽ giúp nâng cao hiệu quả vận hành và khả năng cạnh tranh của hệ thống trong môi trường kinh doanh trực tuyến ngày càng phát triển.

DANH MỤC TÀI LIỆU THAM KHẢO

