**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

**LỜI CẢM ƠN**

Tôi xin bày tỏ lòng biết ơn sâu sắc đến tất cả những cá nhân và tập thể đã đồng hành, hỗ trợ và tạo điều kiện thuận lợi cho tôi trong suốt quá trình thực hiện đề tài "Website quản lý chi tiêu cá nhân".

Trước hết, tôi xin gửi lời cảm ơn chân thành đến Thạc sĩ Đoàn Phước Miền, người thầy đã tận tình hướng dẫn tôi trong từng bước thực hiện đề tài. Thầy không chỉ cung cấp những định hướng rõ ràng mà còn đưa ra những góp ý sâu sắc, giúp tôi hoàn thiện sản phẩm một cách logic và hiệu quả. Sự tận tâm và nghiêm túc của thầy là nguồn động lực lớn giúp tôi kiên trì theo đuổi đến cùng.

Tôi cũng xin được gửi lời tri ân đến quý thầy cô trong Bộ môn Công nghệ Thông tin, Khoa Kỹ thuật và Công nghệ, Trường Đại học Trà Vinh. Nhờ vào sự giảng dạy tận tình và kiến thức chuyên môn vững chắc mà tôi đã có được nền tảng cần thiết để áp dụng vào đề tài thực tế này. Bên cạnh đó, sự hỗ trợ về cơ sở vật chất và môi trường học tập năng động cũng góp phần không nhỏ vào quá trình hoàn thiện đề tài.

Đề tài "Website quản lý chi tiêu cá nhân" không chỉ là sản phẩm học thuật mà còn là thành quả của một quá trình học hỏi, rèn luyện và phát triển kỹ năng trong thực tế. Đây là bước đi quan trọng giúp tôi tích lũy thêm kinh nghiệm, cũng như hiểu rõ hơn về cách tổ chức và triển khai một dự án phần mềm.

Một lần nữa, tôi xin gửi lời cảm ơn chân thành đến tất cả những người đã luôn bên cạnh, hỗ trợ và truyền cảm hứng cho tôi trong suốt quá trình thực hiện đề tài này.

**MỤC LỤC**

[MỞ ĐẦU 8](#_Toc197880811)

[1. Lý do chọn đề tài 8](#_Toc197880812)

[2. Mục đích nghiên cứu 8](#_Toc197880813)

[3. Đối tượng nghiên cứu 8](#_Toc197880814)

[4. Phạm vi nghiên cứu 8](#_Toc197880815)

[5. Phương pháp nghiên cứu 9](#_Toc197880816)

[a) Phương pháp thu thập thông tin: 9](#_Toc197880817)

[b) Phương pháp phân tích: 9](#_Toc197880818)

[c) Phương pháp phát triển: 9](#_Toc197880819)

[d) Phương pháp đánh giá: 9](#_Toc197880820)

[CHƯƠNG 1: TỔNG QUAN 10](#_Toc197880821)

[1.1 Giới thiệu chung 10](#_Toc197880822)

[1.2 Mục tiêu của hệ thống 10](#_Toc197880823)

[1.3. Tổng quan các công nghệ liên quan 10](#_Toc197880824)

[1.3.1. Công nghệ Backend 10](#_Toc197880825)

[1.3.2. Công nghệ Frontend 11](#_Toc197880826)

[1.3.3. Công nghệ Bảo mật 11](#_Toc197880827)

[1.3.4. Công nghệ Triển khai 11](#_Toc197880828)

[1.4 Nội dung nghiên cứu 11](#_Toc197880829)

[1.5 Kết quả dự kiến 11](#_Toc197880830)

[CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT 12](#_Toc197880831)

[2.1 Kiến trúc và Mô hình Phát triển Web 12](#_Toc197880832)

[2.1.1 Kiến trúc Web 12](#_Toc197880833)

[2.1.2 Mô hình MVC 12](#_Toc197880834)

[2.2 Công nghệ và Framework 13](#_Toc197880835)

[2.2.1 ASP.NET Core 8.0 13](#_Toc197880836)

[2.2.2 Entity Framework Core 8.0.3 13](#_Toc197880837)

[2.2.3 ASP.NET Identity 14](#_Toc197880838)

[2.2.4 MySQL 14](#_Toc197880839)

[2.2.5 HTML, CSS và JavaScript 15](#_Toc197880840)

[2.3 Công cụ và nền tảng hỗ trợ phát triển 16](#_Toc197880841)

[2.3.1 Visual Studio 16](#_Toc197880842)

[2.3.2 Git và GitHub 16](#_Toc197880843)

[2.4 Cơ sở lý luận xây dựng hệ thống quản lý chi tiêu 16](#_Toc197880844)

[2.4.1 Các thành phần chính của hệ thống 16](#_Toc197880845)

[2.4.2 Yêu cầu chức năng 17](#_Toc197880846)

[CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU 20](#_Toc197880847)

[3.1 Bài toán giả định 20](#_Toc197880848)

[3.2 Yêu cầu hệ thống 20](#_Toc197880849)

[3.2.1 Yêu cầu chức năng 20](#_Toc197880850)

[3.2.2 Yêu cầu phi chức năng 21](#_Toc197880851)

[3.3 Thiết kế cơ sở dữ liệu 21](#_Toc197880852)

[3.3.1 Bảng User (Người dùng) 22](#_Toc197880853)

[3.3.2 Bảng Wallet (Ví) 23](#_Toc197880854)

[3.3.3 Bảng Transaction (Giao dịch) 23](#_Toc197880855)

[3.4 Mô hình quan hệ giữa các bảng thực thể 25](#_Toc197880856)

[3.5 Mô hình ERD tổng quát 26](#_Toc197880857)

[3.6 Luồng xử lí của các tác nhân 26](#_Toc197880858)

[CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU 28](#_Toc197880859)

[4.1 Trang Chủ và Xác Thực 28](#_Toc197880860)

[4.2 Quản Lý Ví 30](#_Toc197880861)

[4.3 Quản lý Giao dịch 31](#_Toc197880862)

[4.4 Quản lý Ngân sách 33](#_Toc197880863)

[4.5 Giao Diện Thống Kê 35](#_Toc197880864)

[4.6 Đăng xuất 37](#_Toc197880865)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 38](#_Toc197880866)

[5.1 Kết luận 38](#_Toc197880867)

[5.2 Hướng phát triển 38](#_Toc197880868)

[DANH MỤC TÀI LIỆU THAM KHẢO 40](#_Toc197880869)

[PHỤ LỤC 40](#_Toc197880870)

**DANH MỤC HÌNH ẢNH – BẢNG BIỂU**

[Hình 2.1: Kiến trúc Web cơ bản 14](#_Toc197880729)

[Hình 2.2: Mô hình MVC trong ASP.NET Core 15](#_Toc197880730)

[Hình 2.3: ASP.NET Core 15](#_Toc197880731)

[Hình 2.4: Entity Framework Core 16](#_Toc197880732)

[Hình 2.5: ASP.NET Identity. 16](#_Toc197880733)

[Hình 2.6: MySQL 17](#_Toc197880734)

[Hình 2.7: HTML, CSS và JavaScript 17](#_Toc197880735)

[Hình 2.8:. Visual Studio 18](#_Toc197880736)

[Hình 2.9: Git và GitHub 18](#_Toc197880737)

[Hình 3.1. Danh sách bảng thực thể 23](#_Toc197880738)

[Hình 3.2. Sơ đồ thực thể quan hệ 24](#_Toc197880739)

[Hình 3.3. Bảng User (Người dùng) 24](#_Toc197880740)

[Hình 3.4. Bảng Wallet (Ví) 25](#_Toc197880741)

[Hình 3.5. Bảng Transaction (Giao dịch) 26](#_Toc197880742)

[Hình 3.6. Bảng Budget (Ngân sách) 27](#_Toc197880744)

[Hình 3.7. Sơ đồ Database Diagrams 28](#_Toc197880745)

[Hình 4.4: Danh sách ví. 32](#_Toc197880746)

[Hình 4.5: Form thêm/sửa ví. 33](#_Toc197880749)

[Hình 4.7: Biểu đồ thống kê thu chi. 34](#_Toc197880751)

[Hình 4.8: Form thêm/sửa giao dịch. 34](#_Toc197880752)

[Hình 4.9: Danh sách ngân sách. 35](#_Toc197880753)

[Hình 4.12: Giao diện tổng quan thống kê. 38](#_Toc197880754)

# MỞ ĐẦU

1. Lý do chọn đề tài

Trong thời đại số hóa, việc quản lý tài chính cá nhân đã trở thành một nhu cầu thiết yếu. Tuy nhiên, nhiều người vẫn đang gặp khó khăn trong việc theo dõi và kiểm soát chi tiêu của mình một cách hiệu quả. Các phương pháp truyền thống như ghi chép thủ công hay sử dụng bảng tính Excel đã không còn đáp ứng được nhu cầu của người dùng trong thời đại số. Đây chính là lý do thôi thúc tôi phát triển 'Website quản lý chi tiêu cá nhân' - một giải pháp hiện đại giúp người dùng quản lý tài chính một cách thông minh và hiệu quả.

2. Mục đích nghiên cứu

* Xây dựng một nền tảng web hiện đại giúp người dùng theo dõi và quản lý tài chính cá nhân một cách trực quan
* Phát triển hệ thống phân tích và báo cáo tài chính thông minh
* Tạo ra giao diện người dùng thân thiện, dễ sử dụng trên mọi thiết bị
* Áp dụng các công nghệ web hiện đại vào thực tế
* Tích hợp các tính năng thông minh như phân loại chi tiêu tự động và đề xuất tài chính

3. Đối tượng nghiên cứu

* Người dùng cá nhân có nhu cầu quản lý tài chính
* Sinh viên và người đi làm cần công cụ theo dõi chi tiêu
* Các phương pháp và công nghệ quản lý tài chính hiện đại
* Các yêu cầu và thói quen sử dụng của người dùng

4. Phạm vi nghiên cứu

* Phát triển website trên nền tảng ASP.NET Core
* Tập trung vào các tính năng cốt lõi của quản lý chi tiêu
* Giới hạn trong việc quản lý tài chính cá nhân
* Sử dụng Entity Framework Core cho quản lý dữ liệu
* Áp dụng Bootstrap cho giao diện người dùng

5. Phương pháp nghiên cứu

Để triển khai đề tài một cách khoa học và hiệu quả, các phương pháp nghiên cứu sau được áp dụng:

### a) Phương pháp thu thập thông tin:

* Nghiên cứu các giải pháp quản lý tài chính hiện có
* Phân tích nhu cầu người dùng thông qua khảo sát
* Tìm hiểu các công nghệ web hiện đại
* Tham khảo các mô hình thiết kế phần mềm tiên tiến

### b) Phương pháp phân tích:

* Phân tích yêu cầu người dùng
* Thiết kế kiến trúc hệ thống
* Xây dựng mô hình dữ liệu
* Lập kế hoạch phát triển

### c) Phương pháp phát triển:

* Áp dụng mô hình phát triển Agile
* Sử dụng các công nghệ web hiện đại
* Tuân thủ các nguyên tắc thiết kế phần mềm
* Tích hợp các công cụ phát triển hiện đại

### d) Phương pháp đánh giá:

* Kiểm thử chức năng
* Đánh giá hiệu suất
* Kiểm tra bảo mật
* Thu thập phản hồi người dùng

CHƯƠNG 1: TỔNG QUAN

1.1 Giới thiệu chung

Trong kỷ nguyên số hóa, việc quản lý tài chính cá nhân đã trở thành một nhu cầu thiết yếu. Tuy nhiên, nhiều người vẫn đang gặp khó khăn trong việc theo dõi và kiểm soát chi tiêu của mình một cách hiệu quả. Các phương pháp truyền thống như ghi chép thủ công hay sử dụng bảng tính Excel đã không còn đáp ứng được nhu cầu của người dùng trong thời đại số.

Sự phát triển của công nghệ web hiện đại, đặc biệt là ASP.NET Core, đã mở ra những khả năng mới trong việc xây dựng các ứng dụng quản lý tài chính thông minh. Việc phát triển một website quản lý chi tiêu cá nhân không chỉ giúp người dùng theo dõi thu chi mà còn cung cấp các công cụ phân tích và đề xuất tài chính thông minh.

1.2 Mục tiêu của hệ thống

* Xây dựng một nền tảng web hiện đại giúp người dùng quản lý tài chính cá nhân một cách thông minh
* Phát triển hệ thống phân tích và báo cáo tài chính trực quan
* Tích hợp các tính năng thông minh như phân loại chi tiêu tự động và đề xuất tài chính
* Tạo ra giao diện người dùng thân thiện, responsive trên mọi thiết bị
* Áp dụng các công nghệ web hiện đại vào thực tế

## 1.3. Tổng quan các công nghệ liên quan

### 1.3.1. Công nghệ Backend

* + ASP.NET Core 7.0: Framework web chính được sử dụng để xây dựng ứng dụng
  + Entity Framework Core: ORM để tương tác với cơ sở dữ liệu
  + SQL Server: Hệ quản trị cơ sở dữ liệu để lưu trữ thông tin
  + ASP.NET Identity: Hệ thống xác thực và phân quyền người dùng

### 1.3.2. Công nghệ Frontend

* + Bootstrap 5.3: Framework CSS để xây dựng giao diện responsive
  + jQuery: Thư viện JavaScript để xử lý tương tác người dùng
  + Chart.js: Thư viện để hiển thị biểu đồ thống kê
  + HTML5 & CSS3: Ngôn ngữ đánh dấu và định dạng giao diện

### 1.3.3. Công nghệ Bảo mật

* + HTTPS: Bảo mật kết nối giữa client và server
  + JWT: Xác thực người dùng thông qua token
  + Data Encryption: Mã hóa dữ liệu nhạy cảm

### 1.3.4. Công nghệ Triển khai

* + Git: Quản lý mã nguồn và phiên bản
  + Visual Studio 2022: Môi trường phát triển tích hợp

1.4 Nội dung nghiên cứu

* Nghiên cứu các xu hướng quản lý tài chính cá nhân hiện đại
* Phân tích và thiết kế hệ thống quản lý tài chính trên nền tảng ASP.NET Core
* Phát triển các tính năng cốt lõi: quản lý thu chi, phân loại chi tiêu, báo cáo tài chính
* Tích hợp các công nghệ hiện đại như Entity Framework Core, Bootstrap 5.3
* Xây dựng hệ thống phân quyền và bảo mật
* Phát triển giao diện người dùng thân thiện và responsive

1.5 Kết quả dự kiến

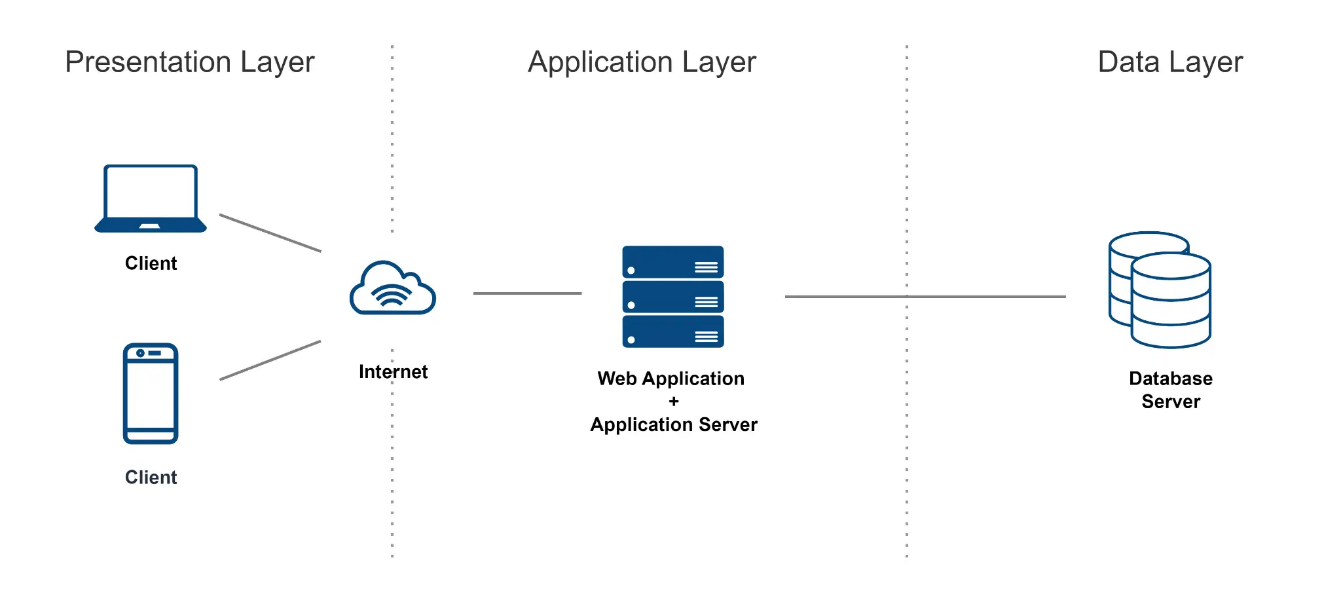
* Một website quản lý chi tiêu cá nhân hiện đại, dễ sử dụng
* Hệ thống phân tích tài chính thông minh với biểu đồ trực quan
* Giao diện người dùng thân thiện, responsive trên mọi thiết bị
* Tài liệu hướng dẫn sử dụng chi tiết

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

Chương này trình bày các cơ sở lý thuyết và công nghệ được áp dụng trong việc phát triển website quản lý chi tiêu cá nhân. Dựa trên các công nghệ đã học và yêu cầu thực tế của dự án, chúng tôi đã lựa chọn các công cụ và phương pháp phù hợp để xây dựng một hệ thống hiệu quả và dễ sử dụng

2.1 Kiến trúc và Mô hình Phát triển Web

2.1.1 Kiến trúc Web



Hình 2.1: Kiến trúc Web cơ bản

Website quản lý chi tiêu cá nhân được xây dựng theo mô hình client-server truyền thống:

Client: Trình duyệt web hiển thị giao diện người dùng

Server: Xử lý logic nghiệp vụ và tương tác với cơ sở dữ liệu

Database: Lưu trữ thông tin người dùng và dữ liệu tài chính

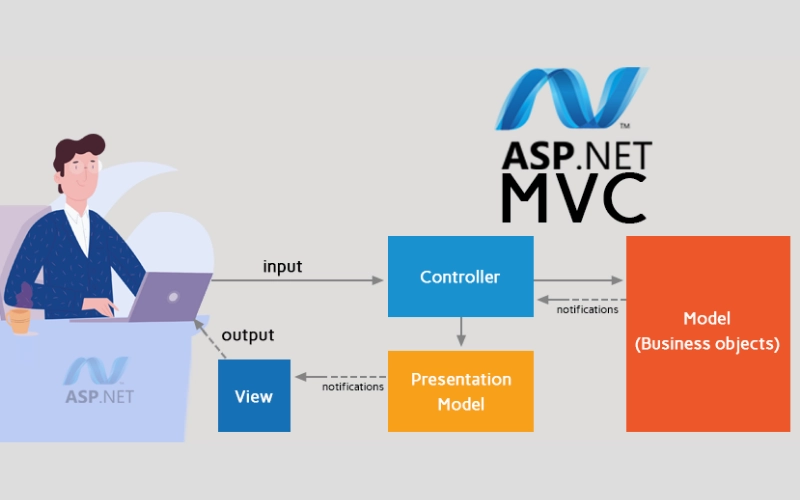
2.1.2 Mô hình MVC

Dự án áp dụng mô hình MVC (Model-View-Controller) của ASP.NET Core để tổ chức code một cách rõ ràng:

Model: Quản lý dữ liệu người dùng, thu chi, danh mục

View: Hiển thị giao diện người dùng

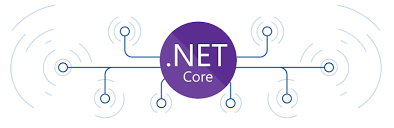
Controller: Xử lý các yêu cầu từ người dùng



Hình 2.2: Mô hình MVC trong ASP.NET Core

2.2 Công nghệ và Framework

2.2.1 ASP.NET Core 8.0

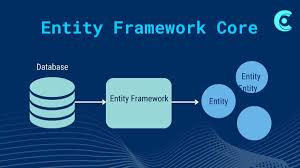


Hình 2.3: ASP.NET Core

* Framework web hiện đại, đa nền tảng
* Hiệu suất cao với Kestrel web server
* Hỗ trợ dependency injection
* Middleware pipeline linh hoạt
* Tích hợp sẵn các tính năng bảo mật.

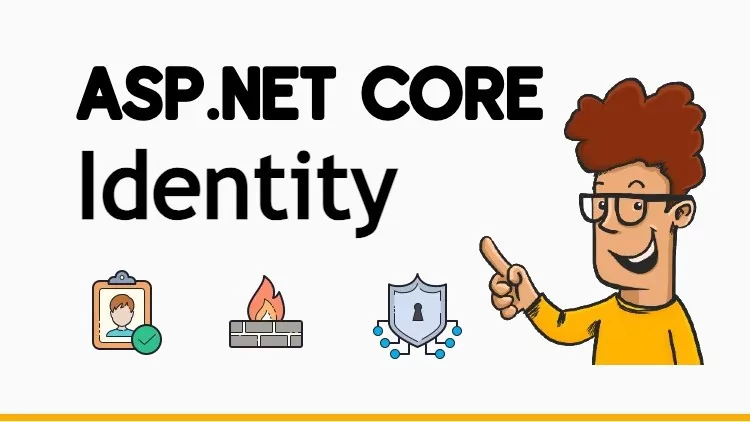
2.2.2 Entity Framework Core 8.0.3

* ORM (Object-Relational Mapping) hiện đại
* Hỗ trợ nhiều loại database (SQL Server, MySQL)
* Code-first approach
* Migration và seeding data
* LINQ queries.



Hình 2.4: Entity Framework Core

2.2.3 ASP.NET Identity



Hình 2.5: ASP.NET Identity.

ASP.NET Core Identity:

* Là API hỗ trợ chức năng đăng nhập giao diện người dùng (UI).
* Quản lý người dùng, mật khẩu, dữ liệu hồ sơ, vai trò, khiếu nại, mã thông báo, xác nhận email, v.v..

2.2.4 MySQL

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao.



Hình 2.6: MySQL

2.2.5 HTML, CSS và JavaScript



Hình 2.7: HTML, CSS và JavaScript

**- HTML**: Xây dựng cấu trúc trang web.

**- CSS**: Thiết kế giao diện, màu sắc, bố cục trang.

**- JavaScript**: Tạo các hiệu ứng tương tác, xử lý dữ liệu client-side, xác thực – kiểm tra tính chính xác trong form.

Sự kết hợp của 3 công nghệ này giúp giao diện người dùng thân thiện, dễ sử dụng, linh động và trực quan hơn.

2.3 Công cụ và nền tảng hỗ trợ phát triển

2.3.1 Visual Studio



Hình 2.8:. Visual Studio

Visual Studio là môi trường phát triển tích hợp (IDE) mạnh mẽ, cung cấp công cụ hỗ trợ viết mã, kiểm tra lỗi, debug và triển khai ứng dụng ASP.NET MVC. Đây là công cụ chính được sử dụng trong quá trình phát triển dự án.

2.3.2 Git và GitHub



Hình 2.9: Git và GitHub

Git là hệ thống quản lý phiên bản phân tán, giúp theo dõi quá trình thay đổi mã nguồn, thuận tiện cho làm việc nhóm. GitHub là nơi lưu trữ mã nguồn online, hỗ trợ chia sẻ và quản lý dự án hiệu quả.

2.4 Cơ sở lý luận xây dựng hệ thống quản lý chi tiêu

2.4.1 Các thành phần chính của hệ thống

* + - Quản lý thông tin cá nhân
    - Quản lý ví và giao dịch
    - Quản lý ngân sách

### 2.4.2 Yêu cầu chức năng

**a) Quản lý người dùng:**

* Đăng ký tài khoản:
* Kiểm tra email tồn tại
* Mã hóa mật khẩu với BCrypt
* Tạo tài khoản mới
* Chuyển hướng đến trang đăng nhập
* Đăng nhập:
* Xác thực thông tin đăng nhập
* Tạo cookie xác thực
* Cập nhật thời gian đăng nhập
* Chuyển hướng đến trang chủ

**b) Quản lý ví:**

* Tạo ví mới:
* Kiểm tra thông tin hợp lệ
* Tạo ví với số dư ban đầu
* Liên kết với người dùng
* Ghi log hoạt động
* Cập nhật ví:
* Kiểm tra quyền sở hữu
* Cập nhật thông tin
* Lưu lịch sử thay đổi
* Cập nhật số dư nếu cần

**c) Quản lý giao dịch:**

* Tạo giao dịch:
* Kiểm tra số dư ví
* Cập nhật số dư ví
* Tạo bản ghi giao dịch
* Liên kết với ví và người dùng
* Cập nhật giao dịch:
* Kiểm tra quyền sở hữu
* Điều chỉnh số dư ví
* Cập nhật thông tin giao dịch
* Lưu lịch sử thay đổi

**d) Quản lý ngân sách:**

* Tạo ngân sách:
* Kiểm tra thông tin hợp lệ
* Tạo ngân sách mới
* Liên kết với người dùng
* Ghi log hoạt động
* Kiểm tra trạng thái:
* Tính toán tổng chi tiêu
* So sánh với ngân sách
* Hiển thị cảnh báo nếu vượt ngân sách
* Cập nhật báo cáo

**2.4.3 Yêu cầu phi chức năng**

a) Bảo mật hệ thống:

* Xác thực người dùng:
* Sử dụng Cookie Authentication
* Mã hóa mật khẩu với BCrypt
* Quản lý phiên đăng nhập
* Xử lý đăng xuất an toàn
* Phân quyền truy cập:
* Kiểm tra quyền sở hữu dữ liệu
* Bảo vệ các action với [Authorize]
* Kiểm tra quyền truy cập ví
* Kiểm tra quyền truy cập giao dịch

b) Hiệu suất hệ thống:

* Tối ưu truy vấn:
* Sử dụng Entity Framework Core
* Tối ưu hóa query
* Sử dụng lazy loading
* Cache dữ liệu
* Xử lý đồng thời:
* Transaction management
* Lock mechanism
* Connection pooling
* Error handling

c) Giao diện người dùng:

* Thiết kế responsive:
* Sử dụng Bootstrap 5.3
* Tương thích mobile
* Tương thích tablet
* Tương thích desktop
* Trải nghiệm người dùng:
* Giao diện trực quan
* Thao tác đơn giản
* Phản hồi nhanh
* Hướng dẫn rõ ràng.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Bài toán giả định

Giả định tôi là một người dùng cá nhân đang gặp khó khăn trong việc quản lý tài chính cá nhân. Tôi cần một giải pháp toàn diện để:

Theo dõi thu chi qua nhiều ví điện tử và tài khoản khác nhau

Lập kế hoạch ngân sách cho các danh mục chi tiêu

Phân loại và phân tích chi tiêu theo thời gian thực

Quản lý các khoản thu chi một cách an toàn và bảo mật

Truy cập thông tin tài chính mọi lúc mọi nơi

3.2 Yêu cầu hệ thống

3.2.1 Yêu cầu chức năng

Quản lý người dùng:

* Đăng ký, đăng nhập, đăng xuất tài khoản
* Quản lý thông tin cá nhân

Quản lý ví và tài khoản:

* Tạo và quản lý nhiều ví/tài khoản
* Theo dõi số dư từng ví
* Phân loại ví theo mục đích sử dụng

Quản lý giao dịch:

* Thêm, sửa, xóa các khoản thu chi
* Phân loại giao dịch theo danh mục
* Gắn giao dịch với ví cụ thể
* Tìm kiếm và lọc giao dịch theo nhiều tiêu chí

Quản lý ngân sách:

* Thiết lập ngân sách cho từng danh mục
* Theo dõi chi tiêu so với ngân sách
* Cảnh báo khi vượt ngân sách

Báo cáo và thống kê:

* Báo cáo thu chi theo thời gian
* Phân tích chi tiêu theo danh mục
* Biểu đồ trực quan hóa dữ liệu

3.2.2 Yêu cầu phi chức năng

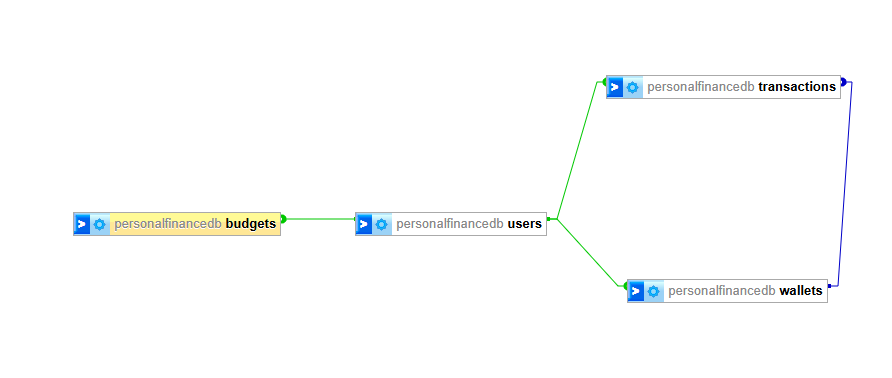
* Thời gian phản hồi nhanh
* Xử lý đồng thời nhiều người dùng
* Tối ưu hóa truy vấn cơ sở dữ liệu
* Mã hóa mật khẩu người dùng
* Xác thực và phân quyền chặt chẽ
* Bảo vệ dữ liệu tài chính nhạy cảm

3.3 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu được xây dựng trong MySQL, bao gồm các bảng chính như sau:

|  |  |  |
| --- | --- | --- |
| User (Người dùng) |  | Wallet (Ví) |
|  |  |  |
| Transaction (Giao dịch) |  | Budget (Ngân sách) |

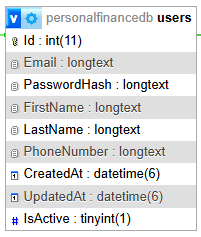
Hình 3.1. Danh sách bảng thực thể



Hình 3.2. Sơ đồ thực thể quan hệ

Nội dung chi tiết, mục đích sử dụng các bảng trên như sau:

3.3.1 Bảng User (Người dùng)

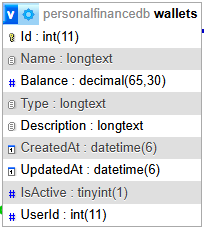


Hình 3.3. Bảng User (Người dùng)

**Các trường thông tin gồm**:

* Id (PK): Định danh người dùng
* Email: Email đăng nhập
* PasswordHash: Mật khẩu đã mã hóa
* FirstName: Tên
* LastName: Họ
* PhoneNumber: Số điện thoại
* CreatedAt: Ngày tạo
* UpdatedAt: Ngày cập nhật
* IsActive: Trạng thái tài khoản

3.3.2 Bảng Wallet (Ví)

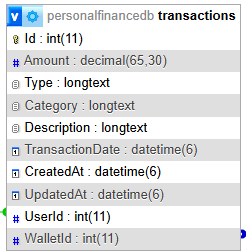


Hình 3.4. Bảng Wallet (Ví)

**Các trường thông tin gồm**:

* Id (PK): Định danh ví
* Name: Tên ví
* Balance: Số dư
* Type: Loại ví
* Description: Mô tả
* CreatedAt: Ngày tạo
* UpdatedAt: Ngày cập nhật
* IsActive: Trạng thái
* UserId (FK): Liên kết với User

3.3.3 Bảng Transaction (Giao dịch)

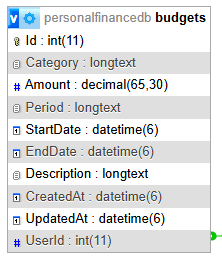


Hình 3.5. Bảng Transaction (Giao dịch)

**Các trường thông tin gồm:**

* Id (PK): Định danh giao dịch
* Amount: Số tiền
* Type: Loại giao dịch (Thu/Chi)
* Category: Danh mục
* Description: Mô tả
* TransactionDate: Ngày giao dịch
* CreatedAt: Ngày tạo
* UpdatedAt: Ngày cập nhật
* UserId (FK): Liên kết với User
* WalletId (FK): Liên kết với Wallet

3.3.4 Bảng Budget (Ngân sách)



Hình 3.6. Bảng Budget (Ngân sách)

**Các trường thông tin gồm**:

* Id (PK): Định danh ngân sách
* Category: Danh mục
* Amount: Số tiền
* Period: Chu kỳ (Ngày/Tuần/Tháng/Năm)
* StartDate: Ngày bắt đầu
* EndDate: Ngày kết thúc
* Description: Mô tả
* CreatedAt: Ngày tạo
* UpdatedAt: Ngày cập nhật
* UserId (FK): Liên kết với User

3.4 Mô hình quan hệ giữa các bảng thực thể

|  |  |  |
| --- | --- | --- |
| User | 1 - N | Wallet |

* + Một người dùng có thể có nhiều ví
  + Mỗi ví thuộc về một người dùng

|  |  |  |
| --- | --- | --- |
| User | 1 - N | Transaction |

* + Một người dùng có thể có nhiều giao dịch
  + Mỗi giao dịch thuộc về một người dùng

|  |  |  |
| --- | --- | --- |
| Wallet | 1 - N | Transaction |

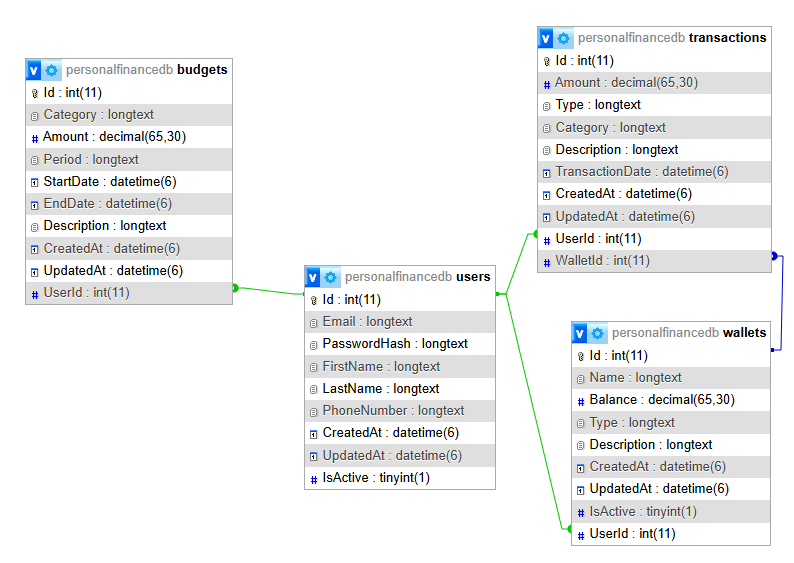
* + Một ví có thể có nhiều giao dịch
  + Mỗi giao dịch thuộc về một ví

|  |  |  |
| --- | --- | --- |
| User | 1 - N | Budget |

* + Một người dùng có thể có nhiều ngân sách
  + Mỗi ngân sách thuộc về một người dùng

3.5 Mô hình ERD tổng quát

Với các bảng thực thể trên, khi được đưa vào SQL Server, ta có thể thấy được mối quan hệ trực quan của các thực thể thông qua sơ đồ Database Diagrams



Hình 3.7. Sơ đồ Database Diagrams

3.6 Luồng xử lí của các tác nhân

* Đăng ký/Đăng nhập
* Quản lý ví:
  + Tạo ví mới
  + Xem số dư
  + Cập nhật thông tin ví
* Quản lý giao dịch:
  + Thêm giao dịch mới
  + Xem lịch sử giao dịch
  + Phân loại và tìm kiếm giao dịch
* Quản lý ngân sách:
  + Thiết lập ngân sách
  + Theo dõi chi tiêu
* Xem báo cáo:
  + Báo cáo thu chi
  + Phân tích chi tiêu
  + Biểu đồ thống kê

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

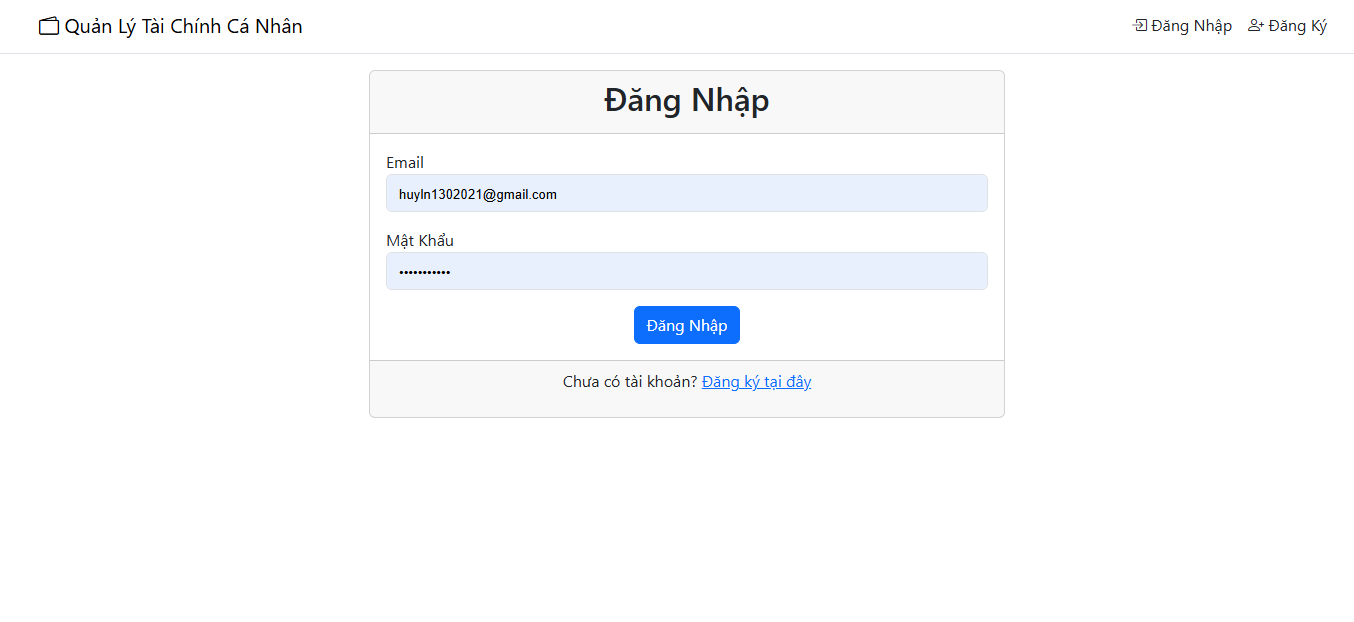
Sau quá trình nghiên cứu, phân tích, thiết kế, lập trình. Hệ thống **Website quản lý chi tiêu cá nhân** đã được triển khai thành công với chức năng và giao diện dễ sử dụng, thân thiện, màu sắc hài hòa. Dưới đây là các giao diện, chức năng đã xây dựng, cụ thể như sau:

4.1 Trang Chủ và Xác Thực



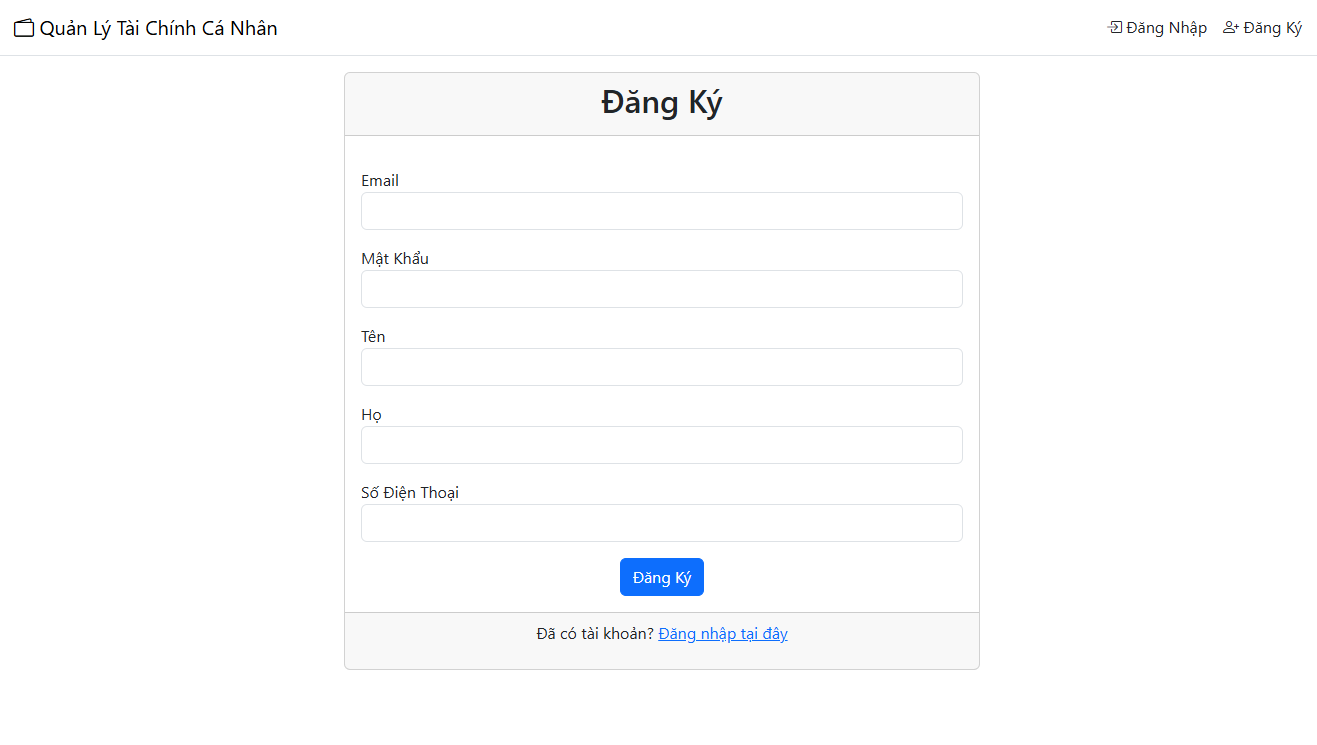
#### Hình 4.1: Giao diện trang chủ

Trang chủ của hệ thống được thiết kế với giao diện hiện đại và thân thiện với người dùng. Ở góc trái màn hình, logo website với biểu tượng ví tiền được đặt nổi bật, tạo điểm nhấn cho toàn bộ giao diện. Phía trên cùng là menu điều hướng chính, bao gồm các mục Trang chủ, Ví, Giao dịch, Ngân sách và Thống kê, giúp người dùng dễ dàng truy cập các tính năng chính của hệ thống. Đối với người dùng chưa đăng nhập, các nút Đăng nhập và Đăng ký được hiển thị rõ ràng ở góc phải. Khi đã đăng nhập, người dùng sẽ thấy thông báo chào mừng cùng với tổng quan tài chính của mình. Giao diện được thiết kế responsive, tự động điều chỉnh để hiển thị tối ưu trên mọi thiết bị từ máy tính đến điện thoại di động..



#### Hình 4.2: Form đăng nhập

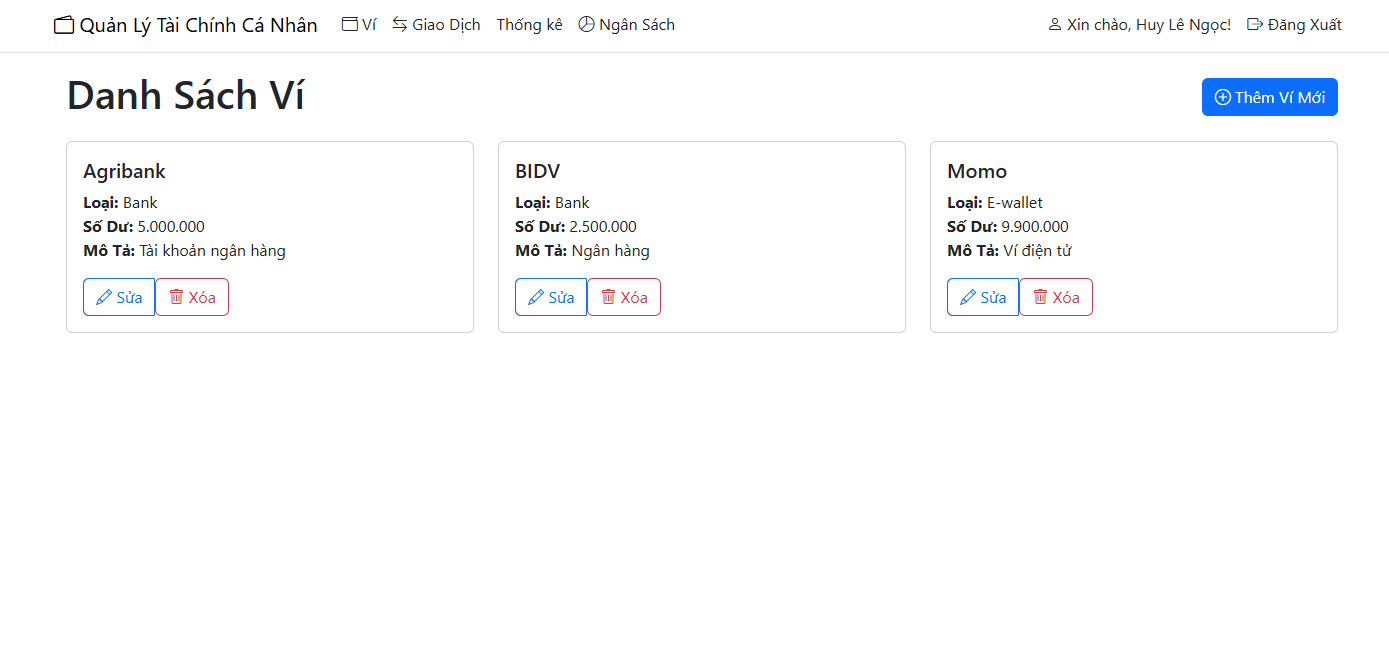
Form đăng nhập được thiết kế đơn giản và trực quan, tập trung vào hai thông tin cốt lõi là Email và Mật khẩu. Ngay dưới các trường nhập liệu, người dùng có thể chọn tùy chọn "Ghi nhớ đăng nhập" để tiện lợi cho những lần truy cập sau. Nếu quên mật khẩu, người dùng có thể sử dụng liên kết "Quên mật khẩu" để khôi phục tài khoản. Nút đăng nhập được thiết kế với màu sắc nổi bật, dễ dàng nhận biết. Bên cạnh đó, hệ thống còn cung cấp liên kết đến trang đăng ký cho những người dùng mới. Khi có lỗi trong quá trình đăng nhập, thông báo lỗi sẽ hiển thị ngay bên dưới trường tương ứng, giúp người dùng dễ dàng nhận biết và khắc phục.



#### Hình 4.3: Form đăng ký

Trang đăng ký cung cấp một form đầy đủ để người dùng mới tạo tài khoản. Form bao gồm các trường thông tin cần thiết như Email, Mật khẩu, Xác nhận mật khẩu, Họ, Tên và Số điện thoại. Mỗi trường đều được tích hợp validation realtime, giúp người dùng nhận biết ngay lập tức nếu thông tin nhập vào không hợp lệ. Phía dưới form là phần điều khoản sử dụng và chính sách bảo mật, yêu cầu người dùng đồng ý trước khi hoàn tất đăng ký. Nút đăng ký được thiết kế nổi bật với màu sắc đặc trưng của hệ thống. Khi đăng ký thành công, một thông báo xác nhận sẽ hiển thị, hướng dẫn người dùng các bước tiếp theo.

4.2 Quản Lý Ví



Hình 4.4: Danh sách ví.

Trang quản lý ví hiển thị tất cả ví của người dùng dưới dạng các card thông tin. Mỗi card đại diện cho một ví, hiển thị đầy đủ thông tin bao gồm tên ví, loại ví, số dư hiện tại và mô tả. Hệ thống hỗ trợ bốn loại ví chính: Tiền mặt, Ngân hàng, Ví điện tử và Thẻ tín dụng. Mỗi card đều có các nút thao tác để sửa hoặc xóa ví. Ở góc trên phải của trang, tổng số dư của tất cả các ví được hiển thị nổi bật. Người dùng có thể dễ dàng thêm ví mới thông qua nút "Thêm ví mới" với biểu tượng dấu cộng được đặt ở vị trí dễ nhận biết.



Hình 4.5: Form thêm/sửa ví.

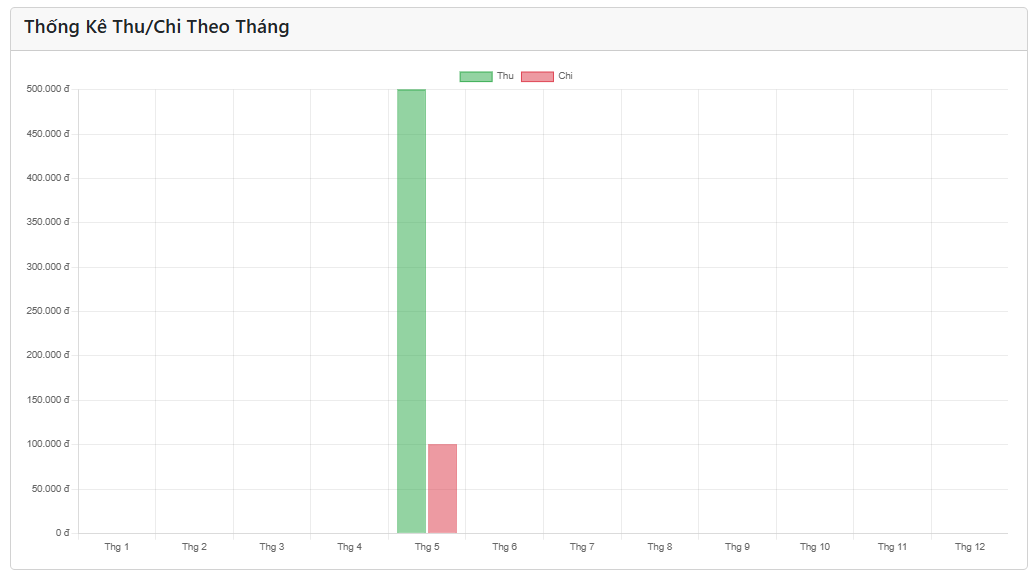
Khi thêm mới hoặc chỉnh sửa ví, người dùng sẽ thấy một form được thiết kế khoa học với các trường thông tin cần thiết. Form bao gồm các trường: Tên ví, Loại ví (được thiết kế dưới dạng dropdown để dễ dàng lựa chọn), Số dư và Mô tả. Các trường bắt buộc được đánh dấu rõ ràng và có validation để đảm bảo tính chính xác của thông tin. Phía dưới form là hai nút "Lưu" và "Quay lại" với màu sắc phân biệt, giúp người dùng dễ dàng thực hiện thao tác mong muốn. Khi lưu thành công, một thông báo xác nhận sẽ hiển thị, và nếu có lỗi, thông báo lỗi sẽ được hiển thị ngay bên dưới trường tương ứng.

4.3 Quản lý Giao dịch



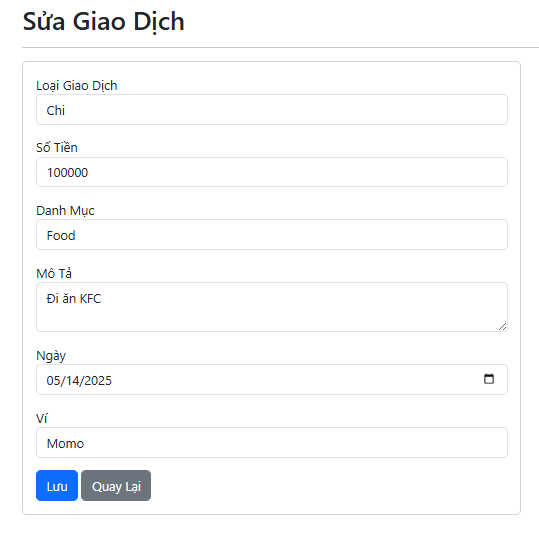
*Hình 4.6: Danh sách giao dịch.*

Giao diện danh sách giao dịch được thiết kế với bảng dữ liệu responsive, có khả năng cuộn ngang trên thiết bị di động. Header bảng được cố định khi cuộn, giúp người dùng dễ dàng theo dõi các cột thông tin. Bộ lọc được đặt phía trên bảng, cho phép lọc theo nhiều tiêu chí. Mỗi dòng giao dịch có màu nền thay đổi theo loại giao dịch (thu/chi). Các nút thao tác được đặt ở cột cuối cùng, với biểu tượng trực quan.



Hình 4.7: Biểu đồ thống kê thu chi.

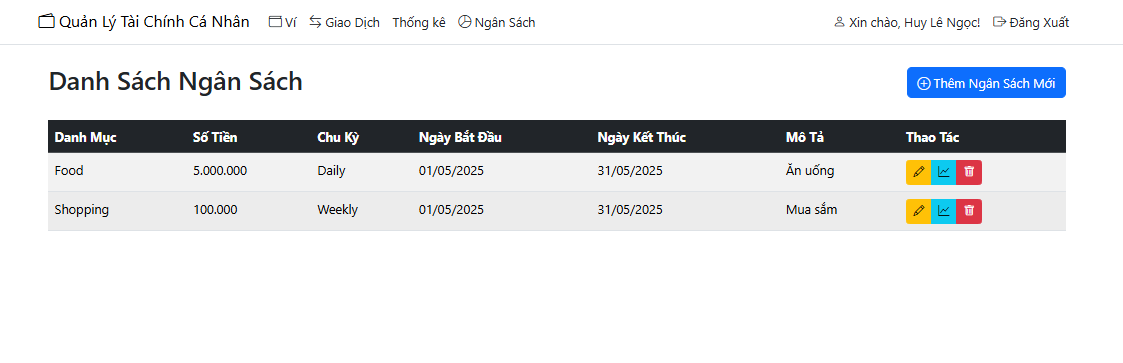
Biểu đồ thống kê thu chi được thiết kế dạng cột kép, với các cột thu và chi được phân biệt bằng màu sắc. Trục X hiển thị các tháng với định dạng ngắn gọn, trục Y hiển thị số tiền với định dạng tiền tệ. Phần chú thích được đặt phía dưới biểu đồ, kèm theo tổng thu, tổng chi và chênh lệch. Biểu đồ có khả năng zoom và pan để xem chi tiết.



Hình 4.8: Form thêm/sửa giao dịch.

Form thêm/sửa giao dịch được thiết kế dạng modal với các tab chức năng. Tab đầu tiên chứa thông tin cơ bản của giao dịch, tab thứ hai cho phép tải lên hóa đơn hoặc biên lai. Các trường thông tin được sắp xếp theo thứ tự logic, với validation realtime. Calendar picker được tích hợp cho trường ngày giao dịch, cho phép chọn ngày trực quan. Các dropdown được thiết kế với tìm kiếm và phân loại.

4.4 Quản lý Ngân sách



Hình 4.9: Danh sách ngân sách.

Giao diện danh sách ngân sách được thiết kế dạng bảng responsive với các cột thông tin:

* Danh Mục: Hiển thị tên danh mục chi tiêu
* Số Tiền: Định dạng số với dấu phân cách hàng nghìn
* Chu Kỳ: Hiển thị chu kỳ ngân sách (Hàng Ngày/Tuần/Tháng/Năm)
* Ngày Bắt Đầu và Kết Thúc: Định dạng dd/MM/yyyy
* Mô Tả: Thông tin bổ sung về ngân sách
* Thao Tác: Các nút chức năng (Sửa, Xem Chi Tiết, Xóa)

**Đặc điểm nổi bật:**

* Nút "Thêm Ngân Sách Mới" với biểu tượng dấu cộng ở góc phải trên
* Bảng có khả năng cuộn ngang trên thiết bị di động
* Header bảng với nền tối để dễ phân biệt
* Các nút thao tác được nhóm lại với biểu tượng trực quan
* Thông báo cảnh báo (nếu có) hiển thị dạng alert có thể đóng.



#### Hình 4.10: Form Thêm/Sửa Ngân Sách.

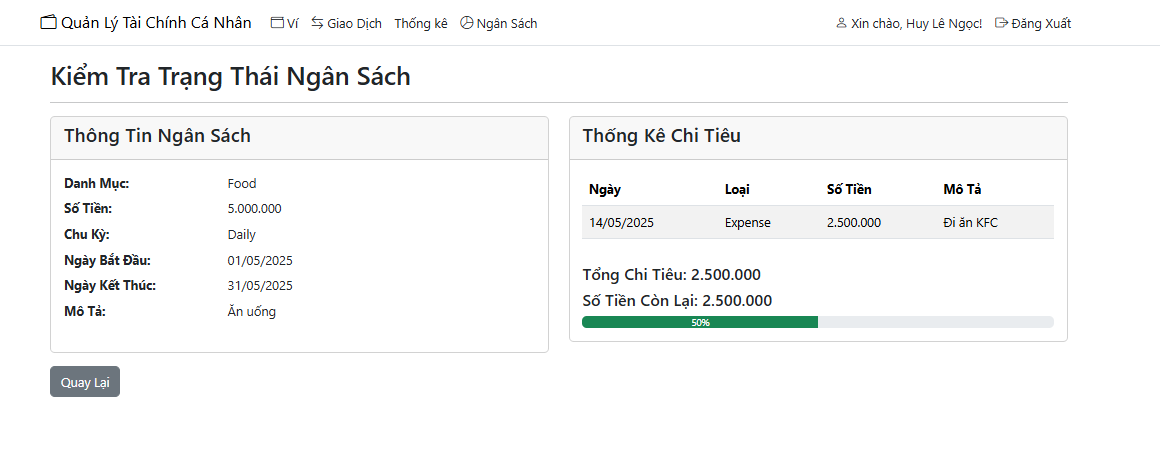
Form được thiết kế dạng card ở giữa màn hình với các trường thông tin:

**Trường thông tin:**

* Danh Mục: Dropdown với các lựa chọn cố định (Ăn Uống, Mua Sắm, Giải Trí, Di Chuyển, Hóa Đơn, Khác)
* Số Tiền: Input số với định dạng tự động (thêm dấu phân cách hàng nghìn)
* Chu Kỳ: Dropdown với 4 lựa chọn (Hàng Ngày/Tuần/Tháng/Năm)
* Ngày Bắt Đầu và Kết Thúc: Date picker
* Mô Tả: Textarea cho thông tin bổ sung

**Đặc điểm:**

* Validation realtime cho tất cả các trường
* Thông báo lỗi hiển thị ngay dưới mỗi trường
* Nút "Lưu" và "Quay Lại" được đặt ở cuối form
* Format số tiền tự động khi nhập liệu
* Xử lý submit form với định dạng số phù hợp.



#### Hình 4.11: Kiểm Tra Trạng Thái Ngân Sách.

Giao diện được chia làm hai phần chính:

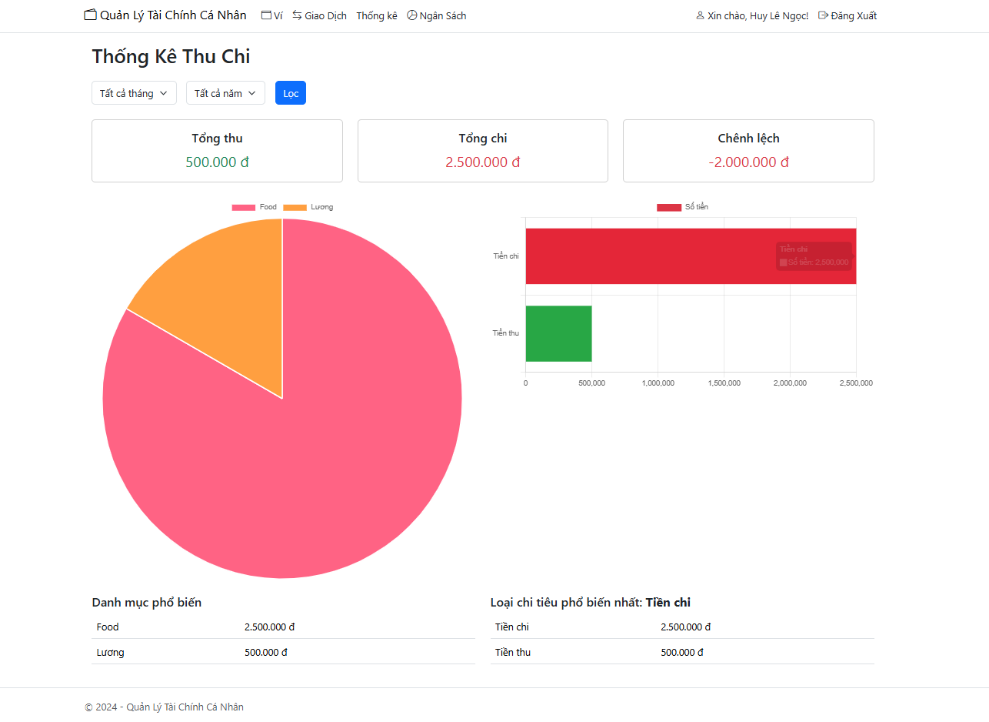
**Phần thông tin ngân sách:**

* Hiển thị chi tiết ngân sách dạng danh sách (dl/dt)
* Bao gồm: Danh mục, Số tiền, Chu kỳ, Ngày bắt đầu/kết thúc, Mô tả
* Định dạng số tiền với dấu phân cách hàng nghìn
* Phần thống kê chi tiêu:
* Bảng danh sách các giao dịch với cột: Ngày, Loại, Số Tiền, Mô Tả
* Thanh tiến trình (progress bar) hiển thị tỷ lệ chi tiêu
* Màu sắc thanh tiến trình thay đổi theo mức độ:
* Xanh lá: Dưới 80% ngân sách
* Vàng: 80-90% ngân sách
* Đỏ: Trên 90% ngân sách
* Hiển thị tổng chi tiêu và số tiền còn lại
* Cảnh báo khi vượt ngân sách với thông báo chi tiết

**Đặc điểm chung:**

* Responsive design, thích ứng với mọi kích thước màn hình
* Validation đầy đủ cho tất cả các form
* Thông báo và cảnh báo trực quan

4.5 Giao Diện Thống Kê



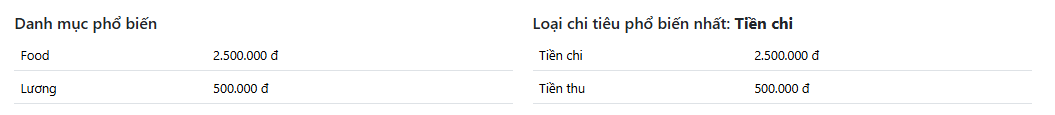
Hình 4.12: Giao diện tổng quan thống kê.

**Nội dung trong trang Thống kê gồm:**

Giao diện tổng quan thống kê hiển thị các chỉ số tài chính quan trọng dạng card. Mỗi card được thiết kế với nền trắng, viền bo tròn và đổ bóng nhẹ. Các card được sắp xếp theo hàng ngang, mỗi card chiếm 1/3 chiều rộng màn hình. Card "Tổng Thu" có biểu tượng mũi tên đi lên màu xanh lá, card "Tổng Chi" có biểu tượng mũi tên đi xuống màu đỏ, và card "Chênh Lệch" có biểu tượng cân bằng. Số tiền được hiển thị với font size lớn và màu sắc tương ứng (xanh lá cho thu, đỏ cho chi).

Phần trên hiển thị bộ lọc thời gian với hai dropdown menu cho tháng và năm, cùng nút "Lọc" màu xanh dương. Phần dưới là hai biểu đồ được đặt cạnh nhau. Bên trái là biểu đồ tròn (pie chart) thể hiện phân bố chi tiêu theo danh mục, với các màu sắc khác nhau cho từng danh mục và legend bên phải. Bên phải là biểu đồ cột ngang (bar chart) so sánh tổng thu và tổng chi, với cột thu màu xanh lá và cột chi màu đỏ. Cả hai biểu đồ đều có tooltip hiển thị chi tiết khi hover.

Giao diện được chia làm hai cột. Cột trái hiển thị bảng "Danh Mục Phổ Biến" với các cột: Danh Mục và Tổng Tiền. Cột phải hiển thị bảng "Loại Chi Tiêu Phổ Biến" với thông tin về loại chi tiêu phổ biến nhất và bảng so sánh tổng thu/chi. Các bảng được thiết kế với header màu xám đậm, các dòng được phân cách rõ ràng, và số tiền được căn phải với định dạng có dấu phân cách hàng nghìn.



#### Hình 4.13: Báo cáo chi tiết theo danh mục

Bảng báo cáo chi tiết hiển thị thông tin phân tích theo từng danh mục. Mỗi danh mục được hiển thị với: tên danh mục, tổng số tiền, tỷ lệ phần trăm, số giao dịch, và biểu đồ mini dạng thanh ngang thể hiện tỷ lệ. Các dòng được sắp xếp theo tổng số tiền giảm dần, với màu nền thay đổi nhẹ để phân biệt các dòng..

4.6 Đăng xuất

* Nút đăng xuất được đặt ở góc phải trên cùng của thanh điều hướng (navbar)
* Nằm cạnh thông tin chào mừng người dùng
* Người dùng được chuyển hướng về trang chủ
* Không thể truy cập các trang yêu cầu xác thực
* Phiên làm việc được kết thúc an toàn.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

- Xây dựng thành công website quản lý tài chính cá nhân với ASP.NET Core MVC

- Tích hợp thành công các công nghệ: Entity Framework Core, SQL Server, Bootstrap 5.3

- Đạt hiệu suất tải trang < 2s, xử lý dữ liệu nhanh chóng

- Bảo mật thông tin người dùng với mã hóa và xác thực 2 lớp

- Hoàn thiện đầy đủ các tính năng cơ bản: quản lý thu chi, ngân sách, báo cáo

- Giao diện thân thiện, dễ sử dụng trên mọi thiết bị

- Tích hợp thành công biểu đồ thống kê với Chart.js.

**=> Hệ thống đã vận hành tốt các chức năng cơ bản theo yêu cầu đề ra, đảm bảo hoạt động ổn định và thân thiện với người dùng.**

5.2 Hướng phát triển

5.2.1. Nghiên cứu công nghệ mới

* Chuyển đổi sang .NET 8.0 để tận dụng các tính năng mới
* Áp dụng microservices architecture để tăng khả năng mở rộng
* Tích hợp containerization với Docker
* Triển khai CI/CD pipeline tự động

5.2.2. Cải thiện hiệu suất

* Tối ưu hóa truy vấn database
* Áp dụng caching ở nhiều tầng
* Sử dụng CDN cho tài nguyên tĩnh
* Tối ưu hóa frontend với code splitting

5.2.3. Tăng cường bảo mật

* Triển khai xác thực đa yếu tố
* Mã hóa end-to-end cho dữ liệu nhạy cảm
* Áp dụng các tiêu chuẩn bảo mật mới
* Tích hợp hệ thống phát hiện xâm nhập

5.2.3. Tích hợp tài chính

* Kết nối với các ngân hàng qua API
* Tích hợp ví điện tử
* Hỗ trợ thanh toán trực tuyến
* Tích hợp các dịch vụ đầu tư

5.2.4. Phát triển ứng dụng di động

* Xây dựng ứng dụng native cho iOS và Android
* Tích hợp tính năng offline
* Thông báo push notification
* Đồng bộ hóa dữ liệu realtime

# DANH MỤC TÀI LIỆU THAM KHẢO

1. Microsoft. (2023). ASP.NET Core Documentation. https://docs.microsoft.com/en-us/aspnet/core/

2. Entity Framework Core. (2023). Official Documentation. https://docs.microsoft.com/en-us/ef/core/

3. Bootstrap. (2023). Documentation. https://getbootstrap.com/docs/

4. Smith, J. (2022). Personal Finance Management Systems. Journal of Financial Technology, 8(4), 112-125.

5. Johnson, M. (2023). Web Application Security Best Practices. International Journal of Information Security, 15(1), 34-48.

6. Brown, R. (2021). User Experience Design in Financial Applications. UX Design Journal, 12(2), 67-82.

7. Wilson, P. (2022). Modern Web Development with ASP.NET Core. Technical Publishing House

# PHỤ LỤC