TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**CHUYÊN ĐỀ ASP.NET**

**HỌC KỲ VI, NĂM HỌC 2024-2025**

**ĐỀ TÀI:**

**XÂY DỰNG WEBSITE QUẢN LÝ CHI TIÊU CÁ NHÂN**

*Sinh viên thực hiện:*

Họ tên: ĐẶNG ĐỨC LONG

MSSV: **170124022**

Lớp: **DK24TTG1**

*Giáo viên hướng dẫn:*

Họ tên: ThS. Đoàn Phước Miền

*Vĩnh Long, tháng 10 năm 2025*

**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

Em xin trân trọng gửi lời cảm ơn đến Ban Giám hiệu cùng toàn thể quý thầy cô khoa Công Nghệ Thông Tin, Trường Đại học Trà Vinh đã tận tình truyền đạt những kiến thức và kinh nghiệm quý báu trong suốt quá trình học tập. Đây chính là hành trang quan trọng giúp em có thể vận dụng vào thực tế và hoàn thành đề tài nghiên cứu này.

Đặc biệt, em xin bày tỏ lòng biết ơn sâu sắc đến **ThS. Đoàn Phước Miền**, giảng viên hướng dẫn, người đã luôn tận tâm chỉ dẫn, định hướng và đưa ra nhiều nhận xét, góp ý quý giá trong suốt quá trình thực hiện đề tài **“Xây dựng website quản lý chi tiêu cá nhân”**. Sự hỗ trợ và động viên của thầy là nguồn động lực lớn giúp em vượt qua khó khăn và hoàn thiện đề tài một cách hiệu quả.

Mặc dù đã nỗ lực hoàn thành đề tài trong khả năng của mình, song chắc chắn vẫn còn tồn tại nhiều thiếu sót. Em rất mong nhận được những ý kiến đóng góp từ quý thầy cô để có thể cải thiện và phát triển tốt hơn trong tương lai.

Em xin trân trọng cảm ơn!

*Vĩnh Long, tháng 10 năm 2025*

Đặng Đức Long

MỤC LỤC

[LỜI CẢM ƠN 4](#_Toc210221880)

[MỤC LỤC 5](#_Toc210221881)

[DANH MỤC HÌNH ẢNH 7](#_Toc210221882)

[MỞ ĐẦU 8](#_Toc210221883)

[CHƯƠNG 1 CƠ SỞ LÝ THUYẾT 9](#_Toc210221884)

[1.1. Mô hình MVC 9](#_Toc210221885)

[1.1.1 Lịch sử của MVC 9](#_Toc210221886)

[1.1.2 Khái niệm 9](#_Toc210221887)

[1.1.3 Các thành phần trong MVC 10](#_Toc210221888)

[1.2. Đặc điểm của mô hình MVC 10](#_Toc210221889)

[1.2.1 Ưu điểm mô hình MVC 10](#_Toc210221890)

[1.2.2 Nhược điểm mô hình MVC không hỗ trợ Preview như ASP.NET 11](#_Toc210221891)

[1.3. Nguyên lý hoạt động 12](#_Toc210221892)

[1.4. Công nghệ triển khai ASP.NET Core 12](#_Toc210221893)

[1.4.1 Sự ra đời của ASP.NET Core 12](#_Toc210221894)

[1.4.2 Microsoft .NET Framework với .NET Core và Mono 13](#_Toc210221895)

[CHƯƠNG 2 HIỆN THỰC HÓA NGHIÊN CỨU 15](#_Toc210221896)

[2.1. Mô tả bài toán 15](#_Toc210221897)

[2.2. Mô hình cơ sở dữ liệu 16](#_Toc210221898)

[2.2.1 Bảng Nguoi\_Dung 16](#_Toc210221899)

[2.2.2 Bảng Danh\_Muc 17](#_Toc210221900)

[2.2.3 Bảng Chi\_Tieu 17](#_Toc210221901)

[2.2.4 Bảng Thu\_Nhap 17](#_Toc210221902)

[2.2.5 Bảng Ngan\_Sach 18](#_Toc210221903)

[2.2.6 Bảng Muc\_Tieu 18](#_Toc210221904)

[2.2.7 Bảng Nhac\_Nho 19](#_Toc210221905)

[2.3. Sơ đồ ERD 20](#_Toc210221906)

[2.4. Sơ đồ use case 21](#_Toc210221907)

[2.4.1 Use Case Tổng Quát 21](#_Toc210221908)

[2.4.2 Use case đăng ký 22](#_Toc210221909)

[2.4.3 Use case đăng nhập 23](#_Toc210221910)

[2.4.4 Use case quản lý danh mục 24](#_Toc210221911)

[2.4.5 Use case quản lý chi tiêu 25](#_Toc210221912)

[2.4.6 Use case quản lý thu nhập 26](#_Toc210221913)

[2.4.7 Use case quản lý mục tiêu 27](#_Toc210221914)

[2.4.8 Use case quản lý ngân sách 28](#_Toc210221915)

[2.4.9 Use case quản lý nhắc nhở 29](#_Toc210221916)

[2.4.10 Use case quản lý thống kê 29](#_Toc210221917)

[2.4.11 Use case quản lý tài khoản 30](#_Toc210221918)

[2.5. Sơ đồ tuần tự 31](#_Toc210221919)

[2.5.1 3.5.1. Đăng ký 31](#_Toc210221920)

[2.5.2 3.5.2. Đăng nhập 32](#_Toc210221921)

[2.5.3 3.5.3. quản lý chi tiêu 33](#_Toc210221922)

[2.5.4 3.5.4. quản lý thu nhập 34](#_Toc210221923)

[2.5.5 3.5.5. quản lý danh mục 35](#_Toc210221924)

[2.5.6 3.5.6. quản lý mục tiêu 36](#_Toc210221925)

[2.5.7 3.5.7. quản lý người dùng 37](#_Toc210221926)

[CHƯƠNG 3 KẾT QUẢ NGHIÊN CỨU 38](#_Toc210221927)

[3.1. Giao diện trang chủ 38](#_Toc210221928)

[3.2. Giao diện đăng ký 38](#_Toc210221929)

[3.3. Giao diện đăng nhập 39](#_Toc210221930)

[3.4. Giao diện tổng quan 40](#_Toc210221931)

[3.5. Cập nhật thông tin cá nhân 41](#_Toc210221932)

[3.6. Thêm mục tiêu 42](#_Toc210221933)

[3.7. Giao diện thông báo 43](#_Toc210221934)

[CHƯƠNG 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 44](#_Toc210221935)

[4.1. Kết luận 44](#_Toc210221936)

[4.2. Hướng phát triển 44](#_Toc210221937)

[TÀI LIỆU THAM KHÁO 45](#_Toc210221938)

DANH MỤC HÌNH ẢNH

[Hình 1 : Các thành phần MVC 11](#_Toc210209694)

[Hình 2: Mô hình MVC 13](#_Toc210209695)

[Hình 3: Các phiên bản ASP.NET 15](#_Toc210209696)

[Hình 4 Lược đồ cơ sở dữ liệu 17](#_Toc210209697)

[Hình 5: ER Diagrams 21](#_Toc210209698)

[Hình 7: use case đăng ký 23](#_Toc210209699)

[Hình 8: use case đăng nhập 24](#_Toc210209700)

[Hình 9 Cập nhật thông tin 25](#_Toc210209701)

[Hình 10 use case mua hàng 26](#_Toc210209702)

[Hình 11 use case đánh giá 27](#_Toc210209703)

[Hình 12 Sơ đồ usecase Lịch sử đơn hàng 28](#_Toc210209704)

[Hình 13 Sơ đồ use case quản lý đơn hàng 29](#_Toc210209705)

[Hình 14 Sơ đồ usecase Quản lý đánh giá 30](#_Toc210209706)

[Hình 15 Sơ đồ usecase Quản lý danh mục 30](#_Toc210209707)

[Hình 16 Sơ đồ usecase Quản lý người dùng 31](#_Toc210209708)

[Hình 17 sơ đồ tuần tự đăng ký 32](#_Toc210209709)

[Hình 18 Sơ đồ tuần tự đăng nhập 33](#_Toc210209710)

[Hình 19 sơ đồ tuần tự quản lý chi tiêu 34](#_Toc210209711)

[Hình 20 Sơ đồ tuần tự quản lý thu nhập 35](#_Toc210209712)

[Hình 21 sơ đồ tuần tự quản lý danh mục 36](#_Toc210209713)

[Hình 22 sơ đồ tuần tự quản lý mục tiêu 37](#_Toc210209714)

[Hình 23 sơ đồ tuần tự cập nhật thông tin người dùng 38](#_Toc210209715)

[Hình 24 giao diện trang chủ 39](#_Toc210209716)

[Hình 25 giao diện đăng ký tài khoản 39](#_Toc210209717)

[Hình 26 giao diện đăng nhập 40](#_Toc210209718)

[Hình 27 giao diện tổng quan 41](#_Toc210209719)

[Hình 28 giao diện cập nhật thông tin cá nhân 42](#_Toc210209720)

[Hình 29 giao diện thêm mục tiêu 43](#_Toc210209721)

[Hình 30 giao diện thông báo và nhắc nhở 44](#_Toc210209722)

MỞ ĐẦU

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng các giải pháp số hóa vào quản lý và đời sống cá nhân đã trở thành xu hướng tất yếu. Đặc biệt, trong lĩnh vực quản lý tài chính cá nhân, nhu cầu xây dựng một nền tảng hỗ trợ theo dõi thu nhập, chi tiêu, lập kế hoạch và quản lý ngân sách một cách hiệu quả ngày càng trở nên quan trọng. Tuy nhiên, nhiều người vẫn áp dụng phương pháp ghi chép thủ công, dẫn đến khó khăn trong việc tổng hợp, phân tích và đưa ra quyết định tài chính kịp thời.

Xuất phát từ thực tiễn đó, em đã lựa chọn đề tài **“Xây dựng website quản lý chi tiêu cá nhân”** cho báo cáo môn học *CHUYÊN ĐỀ ASP.NET*. Website được phát triển dựa trên nền tảng **ASP.NET MVC** và sử dụng **SQL Server** để quản lý cơ sở dữ liệu. Báo cáo trình bày chi tiết các bước từ phân tích yêu cầu, thiết kế hệ thống, đến triển khai và thử nghiệm các chức năng chính như quản lý thu nhập, chi tiêu, danh mục, báo cáo thống kê và gợi ý kế hoạch tài chính.

Mặc dù đã nỗ lực hoàn thiện đề tài, nhưng do hạn chế về thời gian và kinh nghiệm, báo cáo khó tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý quý báu từ thầy để cải thiện và hoàn thiện hơn trong tương lai.

Em xin chân thành cảm ơn Thầy **ThS. Đoàn Phước Miền** đã tận tình hướng dẫn và hỗ trợ trong suốt quá trình thực hiện đề tài.

# CƠ SỞ LÝ THUYẾT

## Mô hình MVC

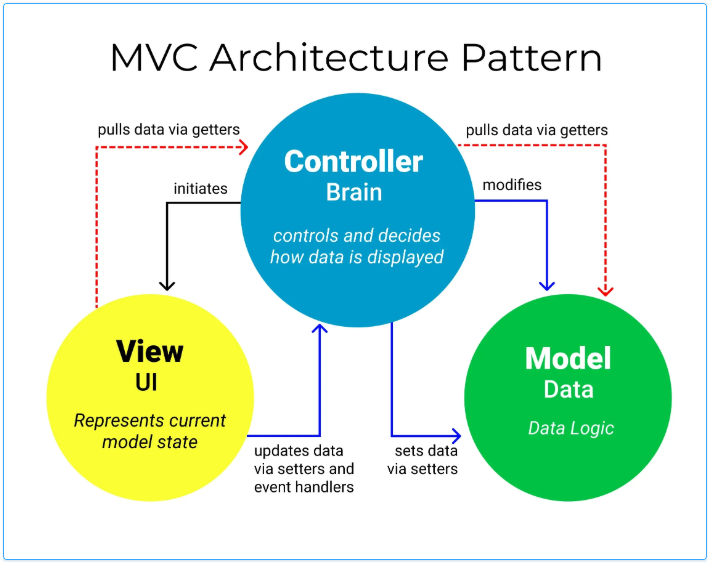
* + 1. Lịch sử của MVC

**MVC** được tiến sĩ Trygve Reenskaug đưa vào [ngôn ngữ lập trình](https://vietnix.vn/ngon-ngu-lap-trinh/) Smalltalk-76 khi ông đến trung tâm Nghiên cứu Xerox Palo Alto (PARC) vào giữa năm 1970. Sau đó, việc triển khai trở nên phổ biến trong các phiên bản khác của Small- Talk. Năm 1988, các bài báo “The Journal of Object Technology” – JOT mang lại bước tranh toàn cảnh về MVC mang liệu sự hiệu quả tốt nhất.

* + 1. Khái niệm

**MVC** **(Model-View-Controller**) là một mẫu kiến trúc phần mềm được sử dụng rộng rãi trong việc phát triển ứng dụng để tạo ra các giao diện người dùng trực quan và có khả năng tương tác cao. MVC chia một ứng dụng thành 3 phần chính và mỗi phần có một vai trò riêng biệt:

* Model đại diện cho dữ liệu và quy tắc nghiệp vụ của ứng dụng.
* View chịu trách nhiệm hiển thị dữ liệu cho người dùng một cách trực quan và tương tác.
* Controller đóng vai trò là cầu nối giữa Model và View, xử lý các yêu cầu từ người dùng và cập nhật giao diện tương ứng.
  + 1. Các thành phần trong MVC

****

Hình : Các thành phần MVC

**Model** là lớp đại diện cho dữ liệu của ứng dụng. Nó có thể là một cơ sở dữ liệu, một file cấu hình hoặc một đối tượng phức tạp. Model chịu trách nhiệm lưu trữ, truy xuất và cập nhật dữ liệu.

**View** là lớp giao diện người dùng. Nó hiển thị dữ liệu từ Model cho người dùng và cho phép người dùng tương tác với ứng dụng. View thường được xây dựng bằng các ngôn ngữ template như HTML, JSP hoặc React.

**Controller** là lớp điều khiển luồng của ứng dụng. Nó nhận các yêu cầu từ người dùng, cập nhật Model và chọn View phù hợp để hiển thị.

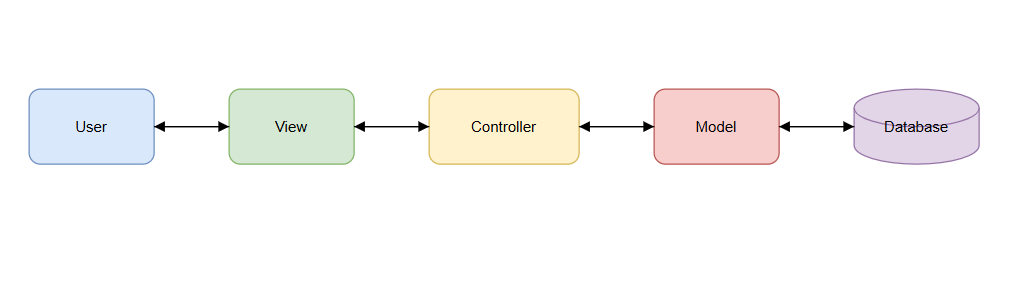
Mối quan hệ giữa các thành phần trong MVC

* **Controller:** Cập nhật Model dựa trên các sự kiện từ View và chọn View phù hợp để hiển thị.
* **Model:** Cung cấp dữ liệu cho View và Controller.
* **View:** Hiển thị dữ liệu từ Model và gửi các sự kiện (event) đến Controller.

## Đặc điểm của mô hình MVC

* + 1. Ưu điểm mô hình MVC
* Đầu tiên, nhắc tới ưu điểm mô hình MVC thì đó là băng thông nhẹ vì không sử dụng viewstate nên khá tiết kiệm băng thông. Việc giảm băng thông giúp website hoạt động ổn định hơn.
* Kiểm tra đơn giản và dễ dàng, kiểm tra lỗi phần mềm trước khi bàn giao lại cho người dùng.
* Một lợi thế chính của MVC là nó tách biệt các phần Model, Controller và View với nhau.
* Sử dụng mô hình MVC chức năng Controller có vai trò quan trọng và tối ưu trên các nền tảng ngôn ngữ khác nhau
* Ta có thể dễ dàng duy trì ứng dụng vì chúng được tách biệt với nhau.
* Có thể chia nhiều developer làm việc cùng một lúc. Công việc của các developer sẽ không ảnh hưởng đến nhau.
* Hỗ trợ [TTD](https://en.wikipedia.org/wiki/Test-driven_development) (test-driven development). Chúng ta có thể tạo một ứng dụng với unit test và viết các won [test case](https://vietnix.vn/test-case-la-gi/).
* Phiên bản mới nhất của MVC hỗ trợ trợ thiết kế responsive website mặc định và các mẫu cho mobile. Chúng ta có thể tạo công cụ View của riêng mình với cú pháp đơn giản hơn nhiều so với công cụ truyền thống.
* Nhược điểm mô hình MVC
* Bên cạnh những ưu điểm MVC mang lại thì nó cũng có một số nhược điểm cần khắc phục.
* MVC đa phần phù hợp với công ty chuyên về website hoặc các dự án lớn thì mô hình này phù hợp hơn so với với các dự án nhỏ, lẻ vì khá là cồng kềnh và mất thời gian.
  + 1. Nhược điểm mô hình MVC không hỗ trợ Preview như ASP.NET
* Không thể Preview các trang như [ASP.NET](https://vietnix.vn/asp-net-la-gi/).
* Khó triển khai.

## Nguyên lý hoạt động



Hình : Mô hình MVC

* Khi một yêu cầu từ **User (Client)** gửi đến **Server**, yêu cầu đó sẽ được **Controller** trong MVC tiếp nhận và phân tích.
* **Controller** xử lý input của User, sau đó giao tiếp với **Model** nếu cần dữ liệu hoặc nghiệp vụ.
* **Model** thực hiện truy xuất/cập nhật **Database**, rồi trả dữ liệu về cho **Controller**.
* **Controller** truyền dữ liệu đó sang **View** để hiển thị cho User.
* **View** chỉ nhận dữ liệu từ Controller và hiển thị, không giao tiếp trực tiếp với Model.
* Tất cả sự tương tác giữa **View** và **Model** đều phải thông qua **Controller**.

**Ví dụ:** User yêu cầu hiển thị thông tin cá nhân.

1. **User** gửi một request (ví dụ /User/Profile) đến **Controller**.
2. **Controller** nhận request, phân tích và gọi đến **Model** để truy xuất dữ liệu người dùng.
3. **Model** lấy dữ liệu từ **Database**, trả về **Controller**.
4. **Controller** gắn dữ liệu vào **ViewModel** rồi chuyển cho **View**.
5. **View** render giao diện và hiển thị thông tin cá nhân cho **User** trên trình duyệt.

## Công nghệ triển khai ASP.NET Core

* + 1. Sự ra đời của ASP.NET Core

ASP.NET Core đã được xuất hiện từ lâu và được biết đến như là một sản phẩm của .Net Framework. ASP.NET được cộng động ủng hộ và phát triển ổn định đến nay. Tuy nhiên đã có sự thay đổi mạnh mẽ khi ASP.NET Core xuất hiện, mang những đặc tính của công nghệ lập trình hiện đại.

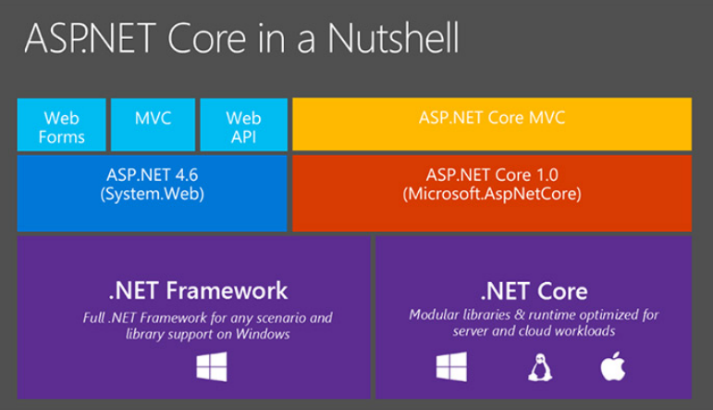
ASP.NET Core là kết quả của sự học hỏi từ các Framework Module hóa khác. Nó chạy trên cơ sở là tập hợp của các gói Module được gọi là Nuget Packages. Nuget Packages cho phép người dùng tối ưu thiết kế ứng dụng chỉ bao gồm những gì cần thiết. Cho nên những ứng dụng được thiết kế trên nền tảng ASP.NET cho ra kết quả có dung lượng nhỏ hơn, bảo mật tốt hơn, hiệu năng hoạt động tốt và ổn định. Quá trình xây dựng cũng không còn quá phức tạp nên vấn đề phát triển và bảo trì sau này cũng dễ hơn nhiều.

Khi làm việc với ASP.NET core người dùng được hỗ trợ những cải tiến đáng kể như:

* Tương thích hệ thống xây dựng Web UI và Web APIs.
* Tích hợp những client –side frameworks hiện đại và những luồng phát triển
* Hệ thống cấu hình dựa trên mô hình đám mây thật sự
* Dependency injection được xây dựng sẵn
* HTTP request được tối ưu nhẹ hơn
* Có thể host trên IIS hoặc self- host trong process của riêng mình
* Được xây dựng trên .NET Core, hỗ trợ thực sự app vesioning
* Chuyển thực thể, thành phần, module như những NuGet packages
* Các công cụ mới để đơn giản hóa quá trình phát triển web hiện đại.
* Xây dựng và chạy đa nền tảng
* Mã nguồn mở, tập trung vào cộng đồng
  + 1. ****Microsoft .NET Framework với .NET Core và Mono****

Bất kỳ ai khi sử dụng ba phiên bản trên kể cả người mới và người sử dụng được thời gian khá dài vẫn có thể nhầm lẫn chúng vì cả ba phiên bản khác nhau mỗi phiên bản đều có Runtime, Libraries và Tooling riêng biệt. Microsoft NET Framework chỉ hoạt động được trên Windows. Mono được phát triển nhằm xây dựng những ứng dụng với giao diện người dùng và được áp dụng với quy mô rộng.

Năm 2013 Microsoft có quyết định đột phá khi quyết định phát triển ASP.NET Core. Nó sử dụng được trong ứng dụng Universal platform và ASP.NET Core. Tùy vào các ứng dụng mà bạn có ý định phát triển để các bạn lựa chọn các phiên bản. Nếu bạn phát triển game dựa trên Unity hay các ứng dụng di động với Xamarin bạn nên chọn sử dụng phiên bản Mono. Còn khi phát triển các Web server bạn có thể sử dụng được cả hai phiên bản .NET Framework và.NET Core .NET Core có thể hoạt động trên đa nền tảng với hiệu năng cao. Bên cạnh đó cũng tồn tại mặt hạn chế là số lượng thư viện hỗ trợ. Microsoft NET Framework có hệ sinh thái và nhiều thư viện hỗ trợ hơn. Điều đáng lưu ý ở đây là không áp dụng Mono cho web server vì Mono không thiết kế để hoạt động với web server nên sẽ gây quá tải cách nhanh chóng.



Hình : Các phiên bản ASP.NET

**ASP.NET Core** là mô hình lập trình mới có thể triển khai trên đám mây và có mục đích sử dụng chạy nhiều nền tảng như: MacOs, Linux và Windows. Nó hỗ trợ được ASP.NET MVC, ASP.NET Core Web API, backend mobile nó còn là một mã nguồn mở.

# HIỆN THỰC HÓA NGHIÊN CỨU

## Mô tả bài toán

Trong quá trình thực hiện đề tài **“Xây dựng website quản lý chi tiêu cá nhân”**, các bước nghiên cứu và triển khai đã được tiến hành như sau:

Khảo sát & phân tích yêu cầu

Xác định nhu cầu thực tế: Người dùng cần một công cụ giúp quản lý thu nhập, chi tiêu, ngân sách và theo dõi mục tiêu tài chính.

Thu thập yêu cầu chức năng: ghi chép thu nhập/chi tiêu, phân loại danh mục, thống kê báo cáo, thiết lập ngân sách, cảnh báo vượt chi tiêu.

Đặc tả yêu cầu hệ thống thông qua **Use Case** và mô tả chức năng.

Thiết kế hệ thống

**Mô hình ca sử dụng (Use Case Diagram):** Biểu diễn các chức năng chính của hệ thống và mối quan hệ với người dùng.

**Biểu đồ lớp (Class Diagram):** Xác định các thực thể chính như User, Category, Income, Expense, Budget, Goal, cùng các thuộc tính và quan hệ giữa chúng.

**Biểu đồ hoạt động (Activity Diagram):** Mô tả luồng công việc của các chức năng tiêu biểu (ví dụ: ghi chép chi tiêu, tạo ngân sách).

**Thiết kế cơ sở dữ liệu:** Dùng mô hình **ERD (Entity Relationship Diagram)** để biểu diễn các bảng như NguoiDung, DanhMuc, ThuNhap, ChiTieu, NganSach, MucTieu, và mối quan hệ.

Cài đặt và hiện thực hóa chương trình

Website được phát triển bằng **ASP.NET MVC**.

Cơ sở dữ liệu được quản lý bởi **SQL Server**.

Các chức năng chính được hiện thực hóa theo đúng đặc tả yêu cầu: quản lý người dùng, quản lý thu nhập – chi tiêu, thiết lập ngân sách, báo cáo thống kê và nhắc nhở.

Giao diện được thiết kế thân thiện, trực quan, giúp người dùng dễ dàng thao tác và theo dõi tài chính cá nhân.

Thử nghiệm & đánh giá

Thử nghiệm các chức năng chính (ghi chép thu chi, thống kê báo cáo, nhắc nhở).

Kiểm tra tính đúng đắn của dữ liệu lưu trữ trong SQL Server.

Đánh giá mức độ đáp ứng nhu cầu thực tế của người dùng.

## Mô hình cơ sở dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

Hình Lược đồ cơ sở dữ liệu

* + 1. Bảng Nguoi\_Dung

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| nguoidung\_id | INT | PK, Identity(1,1) | Khóa chính, định danh duy nhất |
| ten\_dang\_nhap | NVARCHAR(50) | NOT NULL, UNIQUE | Tên đăng nhập |
| mat\_khau | NVARCHAR(255) | NOT NULL | Mật khẩu (hash) |
| ho\_ten | NVARCHAR(100) |  | Họ tên người dùng |
| email | NVARCHAR(100) |  | Địa chỉ email |
| so\_dien\_thoai | NVARCHAR(20) |  | Số điện thoại |
| dia\_chi | NVARCHAR(200) |  | Địa chỉ liên hệ |
| anh\_dai\_dien | NVARCHAR(255) |  | Ảnh đại diện |
| lan\_dang\_nhap\_that\_bai | INT | DEFAULT 0 | Số lần đăng nhập sai |
| khoa\_den | DATETIME |  | Thời điểm khóa tài khoản |

* + 1. Bảng Danh\_Muc

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| danhmuc\_id | INT | PK, Identity(1,1) | Khóa chính |
| ten\_danh\_muc | NVARCHAR(100) | NOT NULL | Tên danh mục |
| loai | NVARCHAR(20) | CHECK (ChiTieu / ThuNhap) | Loại danh mục |
| nhom | NVARCHAR(20) |  | Nhóm danh mục (Cố định / Biến đổi) |
| ghi\_chu | NVARCHAR(200) |  | Ghi chú |

* + 1. Bảng Chi\_Tieu

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| chitieu\_id | INT | PK, Identity(1,1) | Khóa chính |
| nguoidung\_id | INT | FK → Nguoi\_Dung | Người chi tiêu |
| danhmuc\_id | INT | FK → Danh\_Muc | Danh mục chi tiêu |
| so\_tien | DECIMAL(18,2) | NOT NULL | Số tiền chi tiêu |
| ngay\_chi | DATE | NOT NULL | Ngày chi tiêu |
| ghi\_chu | NVARCHAR(200) |  | Ghi chú |

* + 1. Bảng Thu\_Nhap

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| thunhap\_id | INT | PK, Identity(1,1) | Khóa chính |
| nguoidung\_id | INT | FK → Nguoi\_Dung | Người nhận thu nhập |
| danhmuc\_id | INT | FK → Danh\_Muc | Danh mục thu nhập |
| so\_tien | DECIMAL(18,2) | NOT NULL | Số tiền |
| ngay\_nhap | DATE | NOT NULL | Ngày nhập thu nhập |
| ghi\_chu | NVARCHAR(200) |  | Ghi chú |
| thang\_thu\_nhap | INT |  | Tháng của khoản thu nhập |
| nam\_thu\_nhap | INT |  | Năm của khoản thu nhập |

* + 1. Bảng Ngan\_Sach

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| ngansach\_id | INT | PK, Identity(1,1) | Khóa chính |
| nguoidung\_id | INT | FK → Nguoi\_Dung | Người dùng đặt ngân sách |
| danhmuc\_id | INT | FK → Danh\_Muc | Danh mục áp dụng ngân sách |
| han\_muc | DECIMAL(18,2) | NOT NULL | Hạn mức chi tiêu |
| thang | INT | NOT NULL | Tháng áp dụng |
| nam | INT | NOT NULL | Năm áp dụng |

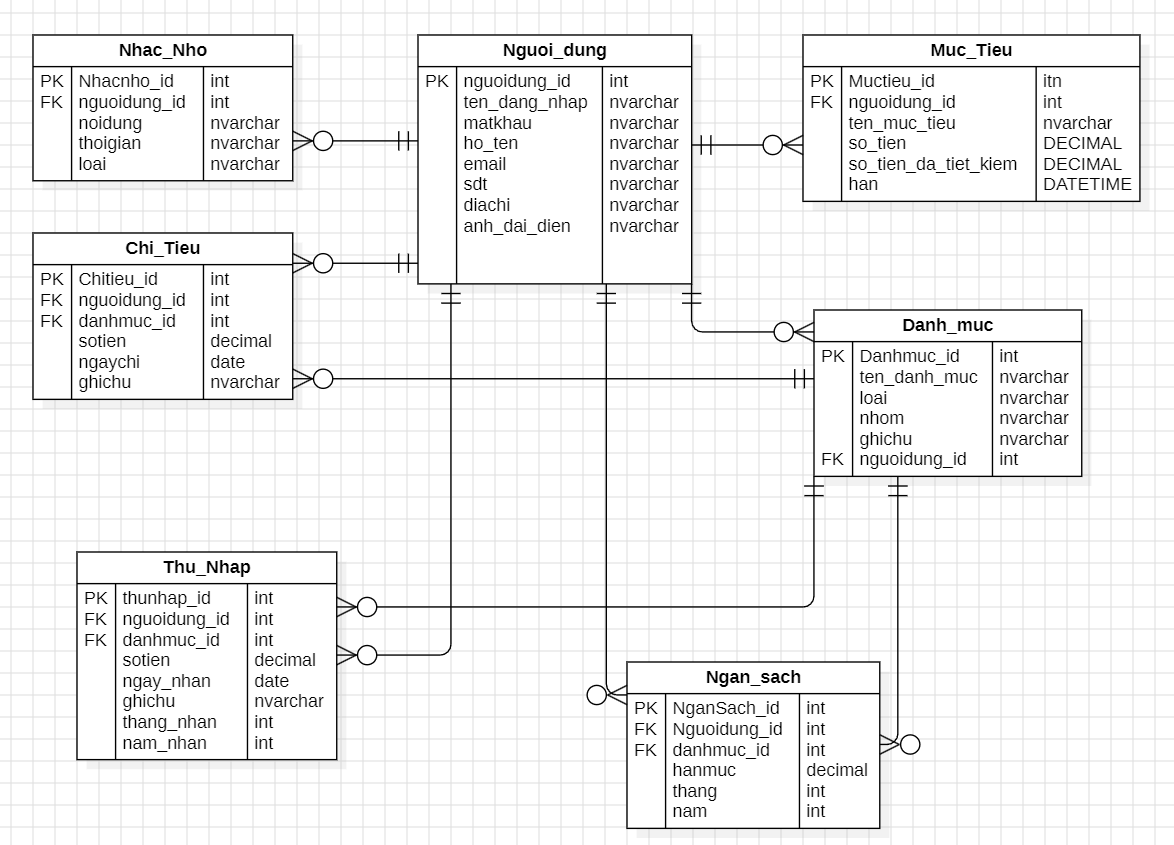
* + 1. Bảng Muc\_Tieu

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| muctieu\_id | INT | PK, Identity(1,1) | Khóa chính |
| nguoidung\_id | INT | FK→ Nguoi\_Dung | Người dùng đặt mục tiêu |
| ten\_muc\_tieu | NVARCHAR(100) | NOT NULL | Tên mục tiêu |
| so\_tien\_can | DECIMAL(18,2) | NOT NULL | Số tiền cần đạt |
| so\_tien\_da\_tiet\_kiem | DECIMAL(18,2) | DEFAULT 0 | Số tiền đã tiết kiệm |
| han | DATETIME |  | Thời hạn đạt mục tiêu |

* + 1. Bảng Nhac\_Nho

|  |  |  |  |
| --- | --- | --- | --- |
| Cột | Kiểu dữ liệu | Ràng buộc | Mô tả |
| nhacnho\_id | INT | PK, Identity(1,1) | Khóa chính |
| nguoidung\_id | INT | FK→ Nguoi\_Dung | Người dùng được nhắc nhở |
| noi\_dung | NVARCHAR(200) |  | Nội dung nhắc nhở |
| thoi\_gian | DATETIME | NOT NULL | Thời gian nhắc nhở |
| loai | NVARCHAR(50) |  | Loại nhắc nhở |

## Sơ đồ ERD



Hình : ER Diagrams

**Các thực thể và mối quan hệ:**

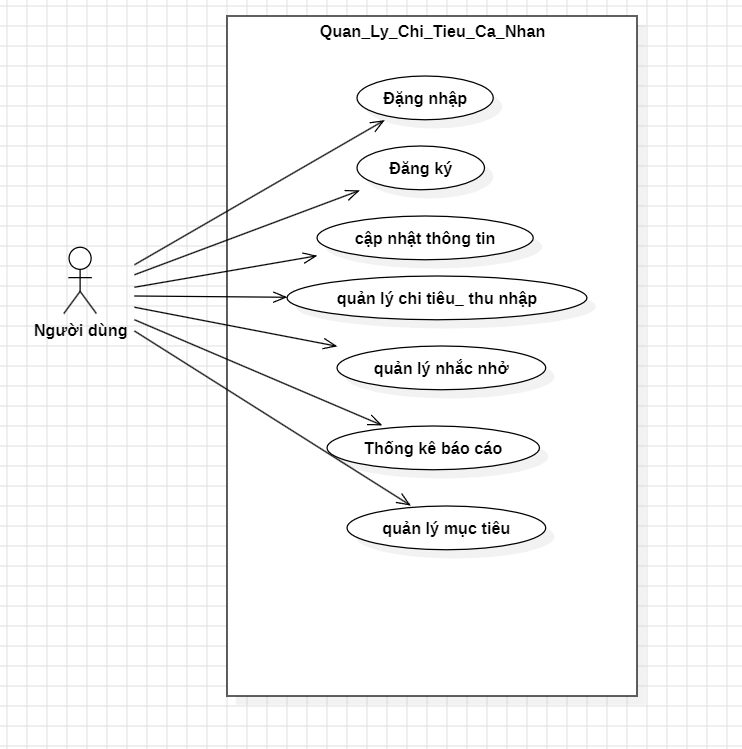
1. **NGUOI\_DUNG** (User) - Thực thể trung tâm lưu trữ thông tin người dùng
2. **DANH\_MUC** (Category) - Phân loại cho thu nhập và chi tiêu (loại: 'ChiTieu' hoặc 'ThuNhap')
3. **CHI\_TIEU** (Expense) - Ghi nhận các khoản chi tiêu
4. **THU\_NHAP** (Income) - Ghi nhận các khoản thu nhập
5. **NGAN\_SACH** (Budget) - Thiết lập hạn mức ngân sách theo tháng/năm
6. **MUC\_TIEU** (Goal) - Đặt mục tiêu tiết kiệm
7. **NHAC\_NHO** (Reminder) - Hệ thống nhắc nhở

**Các mối quan hệ chính:**

* Một người dùng có thể có nhiều danh mục, chi tiêu, thu nhập, ngân sách, mục tiêu và nhắc nhở (quan hệ 1:N)
* Một danh mục có thể được sử dụng cho nhiều chi tiêu, thu nhập và ngân sách (quan hệ 1:N)

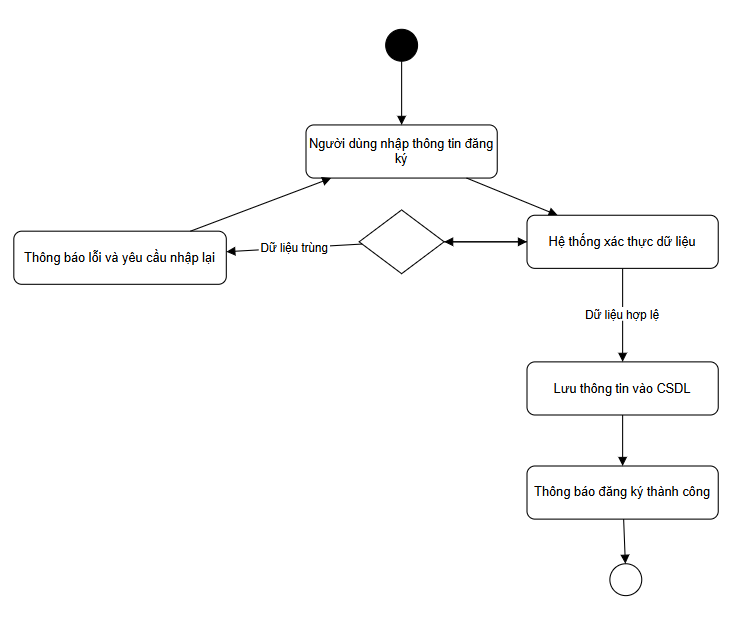
## Sơ đồ use case

* + 1. Use Case Tổng Quát



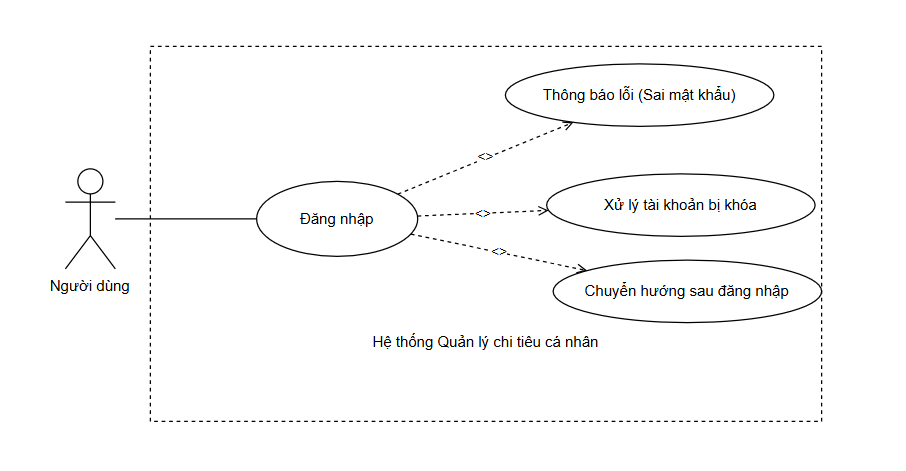
*Hình 6:* *Use tổng quát*

* + 1. Use case đăng ký



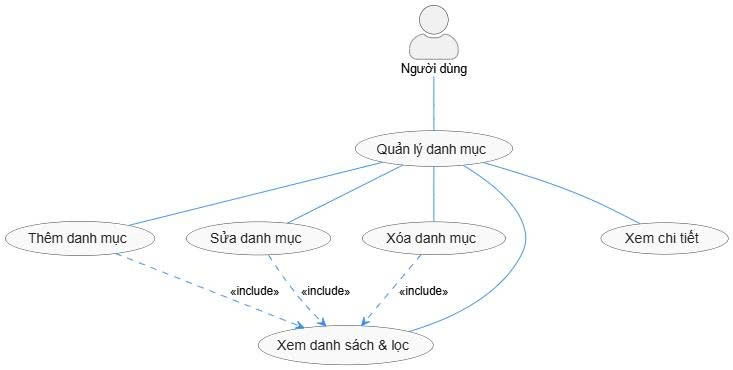
Hình : use case đăng ký

* **Tên Use Case**: Đăng ký tài khoản
* **Tác nhân (Actor)**: Người dùng
* **Mục tiêu**: Cho phép người dùng tạo tài khoản mới để sử dụng hệ thống
* **Mô tả**: Khi người dùng muốn sử dụng các chức năng yêu cầu đăng nhập (như đặt hàng, thêm vào yêu thích,...), họ cần đăng ký tài khoản. Người dùng nhập đầy đủ các thông tin như họ tên, email, tên đăng nhập, mật khẩu,... Hệ thống sẽ kiểm tra thông tin:
* Nếu thông tin **không trùng** với dữ liệu đã có trong CSDL (như email hoặc tên đăng nhập chưa tồn tại), hệ thống sẽ **lưu thông tin mới** và **xác nhận đăng ký thành công**.
* Nếu thông tin **bị trùng** (ví dụ email hoặc tên đăng nhập đã có), hệ thống sẽ **thông báo lỗi và yêu cầu nhập lại**.
  + 1. Use case đăng nhập



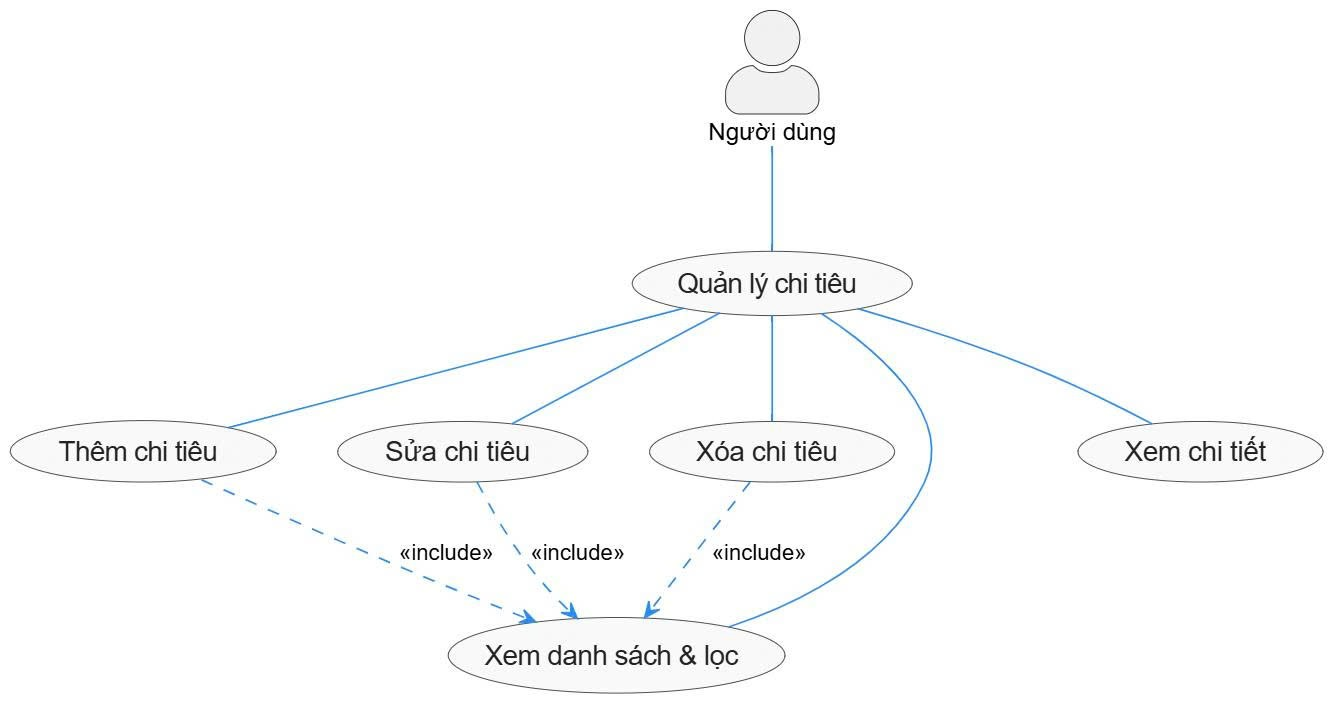
Hình : use case đăng nhập

* **Tên Use Case**: Đăng nhập
* **Tác nhân (Actor)**: Người dùng
* **Mục tiêu**: Cho phép người dùng truy cập vào hệ thống bằng tài khoản đã đăng ký.
* **Mô tả**:Người dùng nhập tên đăng nhập (hoặc email) và mật khẩu để đăng nhập vào hệ thống. Hệ thống kiểm tra thông tin:
  + Nếu đúng: Cho phép truy cập và chuyển đến trang chính.
  + Nếu sai: Hiển thị thông báo lỗi.
    1. Use case quản lý danh mục



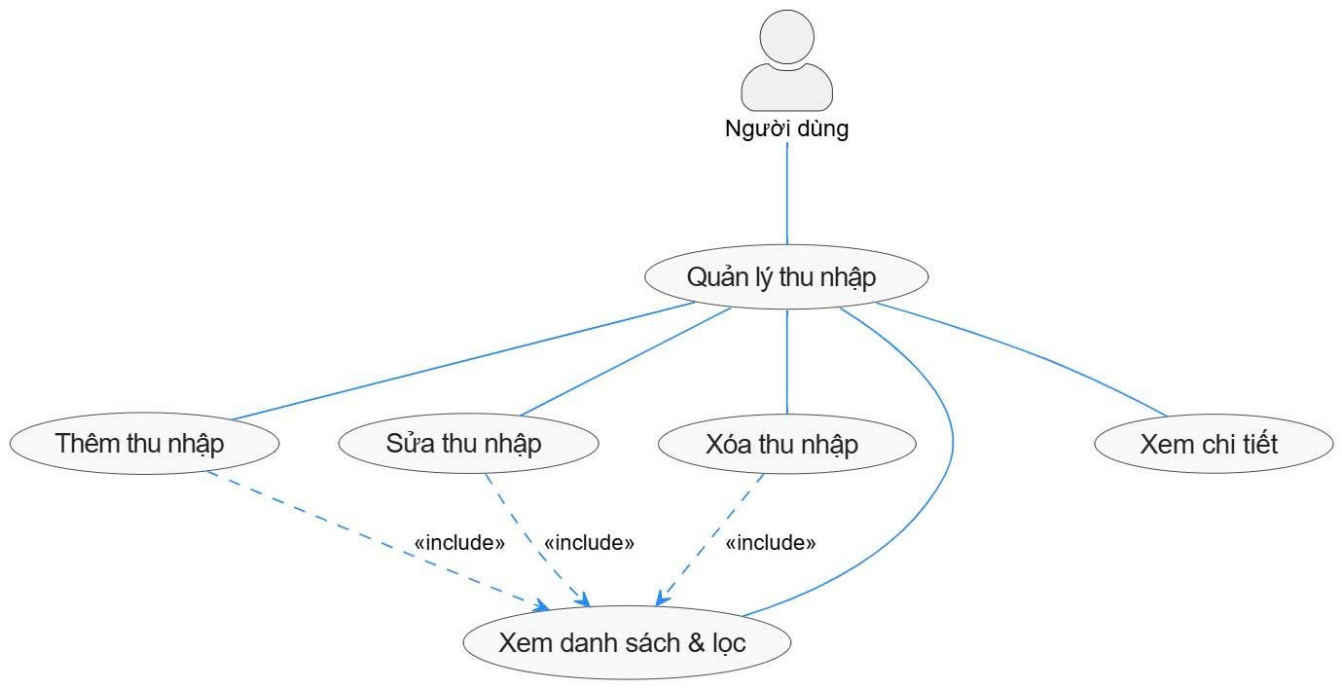
Hình  Cập nhật thông tin

* **Tên Use Case**: Quản lý danh mục
* **Tác nhân (Actor):** Người dùng (có thể là Khách hàng hoặc Quản trị viên tùy hệ thống)
* **Mục tiêu:** Cho phép người dùng quản lý các danh mục trong hệ thống, bao gồm: thêm, sửa, xóa, xem chi tiết, xem danh sách và lọc danh mục.
* **Mô tả:** Sau khi đăng nhập, người dùng có thể truy cập vào chức năng Quản lý danh mục để:
* Thêm danh mục: tạo mới danh mục sản phẩm/dịch vụ.
* Sửa danh mục: chỉnh sửa thông tin của một danh mục có sẵn.
* Xóa danh mục: loại bỏ một danh mục khỏi hệ thống.
* Xem chi tiết: xem đầy đủ thông tin về một danh mục cụ thể.
* Xem danh sách & lọc: hiển thị tất cả danh mục, có thể áp dụng điều kiện lọc để dễ tìm kiếm (ví dụ theo tên, trạng thái).
  + 1. Use case quản lý chi tiêu



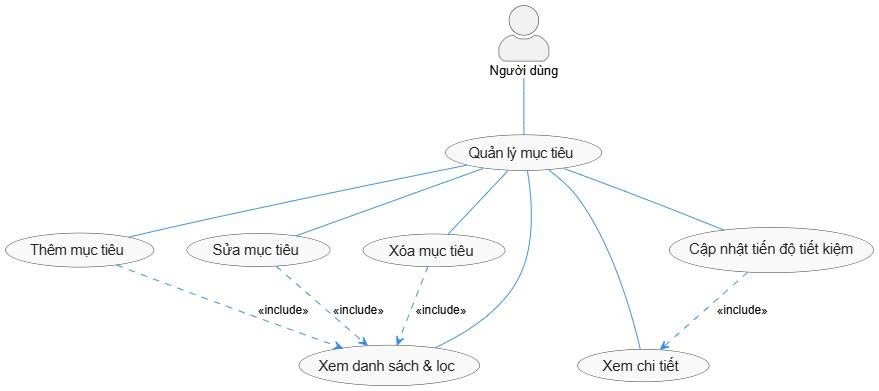
Hình  use case mua hàng

* **Use Case**: Quản lý chi tiêu
* **Tác nhân:** Người dùng
* **Mục tiêu:** Quản lý các khoản chi tiêu cá nhân.
* **Mô tả:** Người dùng có thể:
* Thêm chi tiêu mới.
* Sửa chi tiêu đã ghi.
* Xóa chi tiêu không cần thiết.
* Xem chi tiết từng khoản chi tiêu.
* Xem danh sách chi tiêu và lọc theo điều kiện (thời gian, loại chi tiêu,…).
  + 1. Use case quản lý thu nhập



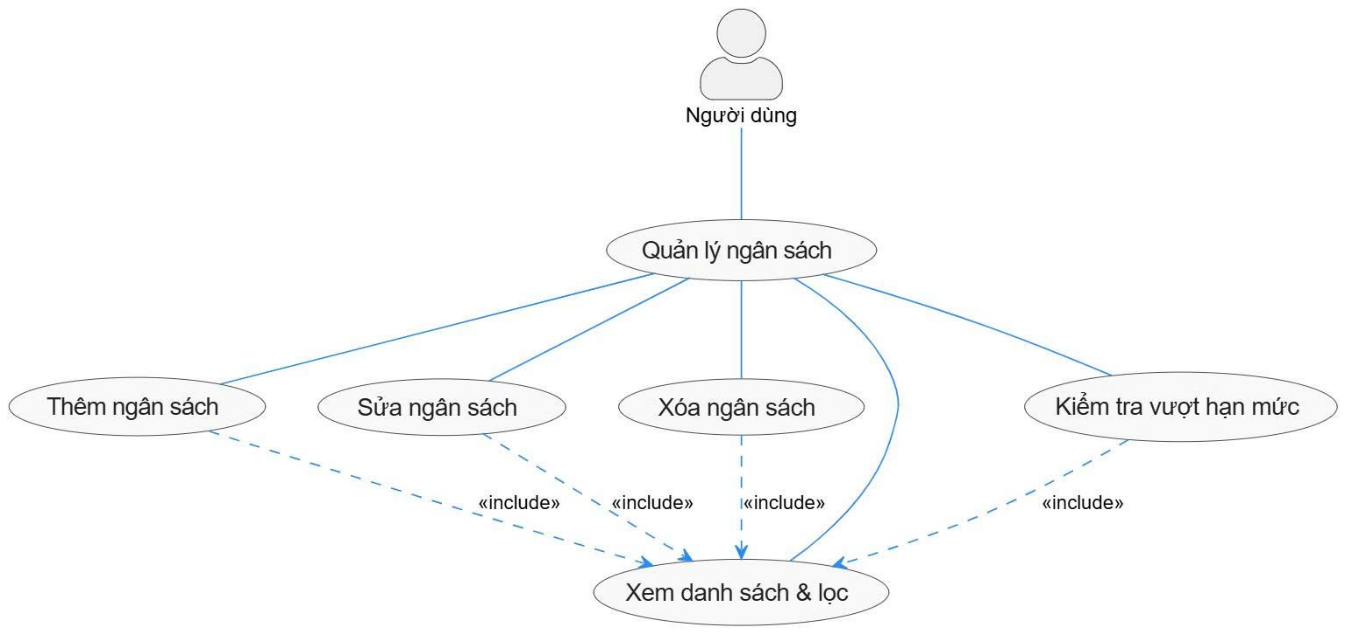
Hình use case đánh giá

* **Tên Use Case**: **Quản lý thu nhập**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Quản lý các khoản thu nhập cá nhân.
* **Mô tả:** Người dùng có thể:
* Thêm thu nhập mới.
* Sửa thông tin thu nhập.
* Xóa thu nhập không cần thiết.
* Xem chi tiết một khoản thu nhập.
* Xem danh sách và lọc thu nhập theo điều kiện (thời gian, loại thu nhập, số tiền,…).
  + 1. Use case quản lý mục tiêu



Hình Sơ đồ usecase Lịch sử đơn hàng

* **Tên Use Case**: **Quản lý mục tiêu**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Hỗ trợ người dùng thiết lập và theo dõi mục tiêu tiết kiệm/chi tiêu.
* **Mô tả:**  
  Người dùng có thể:
* Thêm mục tiêu mới.
* Sửa thông tin mục tiêu.
* Xóa mục tiêu không cần thiết.
* Cập nhật tiến độ tiết kiệm cho mục tiêu.
* Xem chi tiết mục tiêu.
* Xem danh sách mục tiêu và lọc theo điều kiện
  + 1. Use case quản lý ngân sách

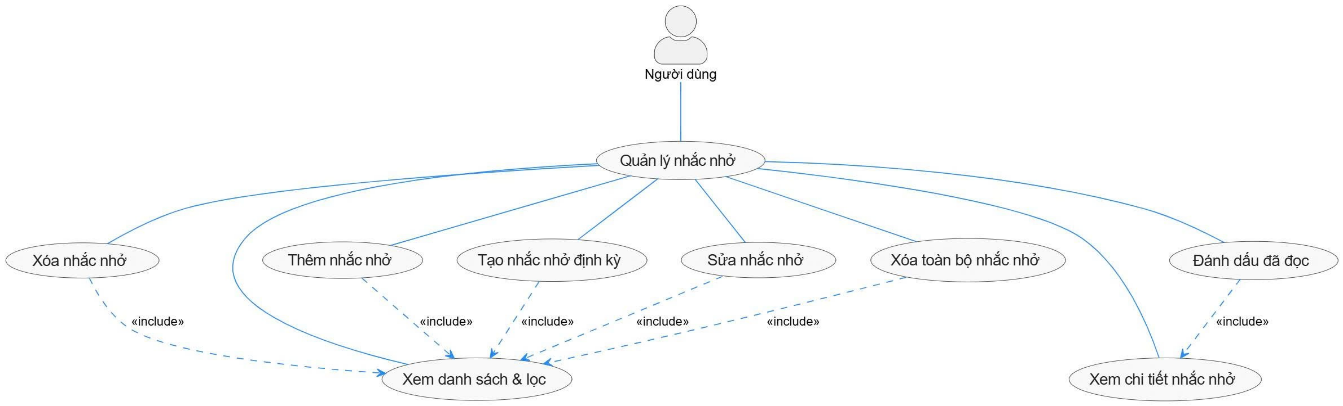


Hình  Sơ đồ use case quản lý đơn hàng

* **Tên Use Case**: **Quản lý ngân sách**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Giúp người dùng thiết lập, theo dõi và kiểm soát ngân sách chi tiêu.
* **Mô tả:**  
  Người dùng có thể:
  + Thêm ngân sách mới.
  + Sửa thông tin ngân sách.
  + Xóa ngân sách không cần thiết.
  + Kiểm tra ngân sách có vượt hạn mức hay không.
  + Xem danh sách ngân sách và lọc theo điều kiện

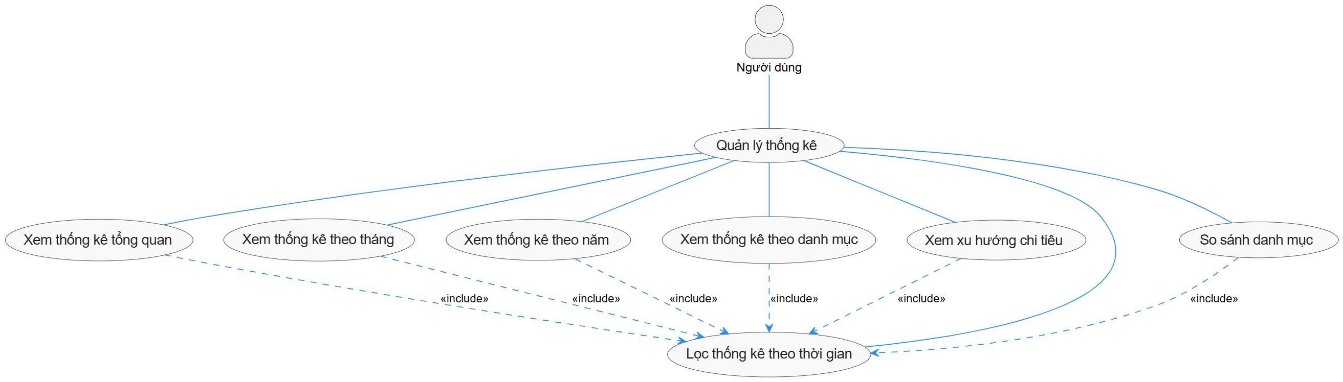
.

* + 1. Use case quản lý nhắc nhở



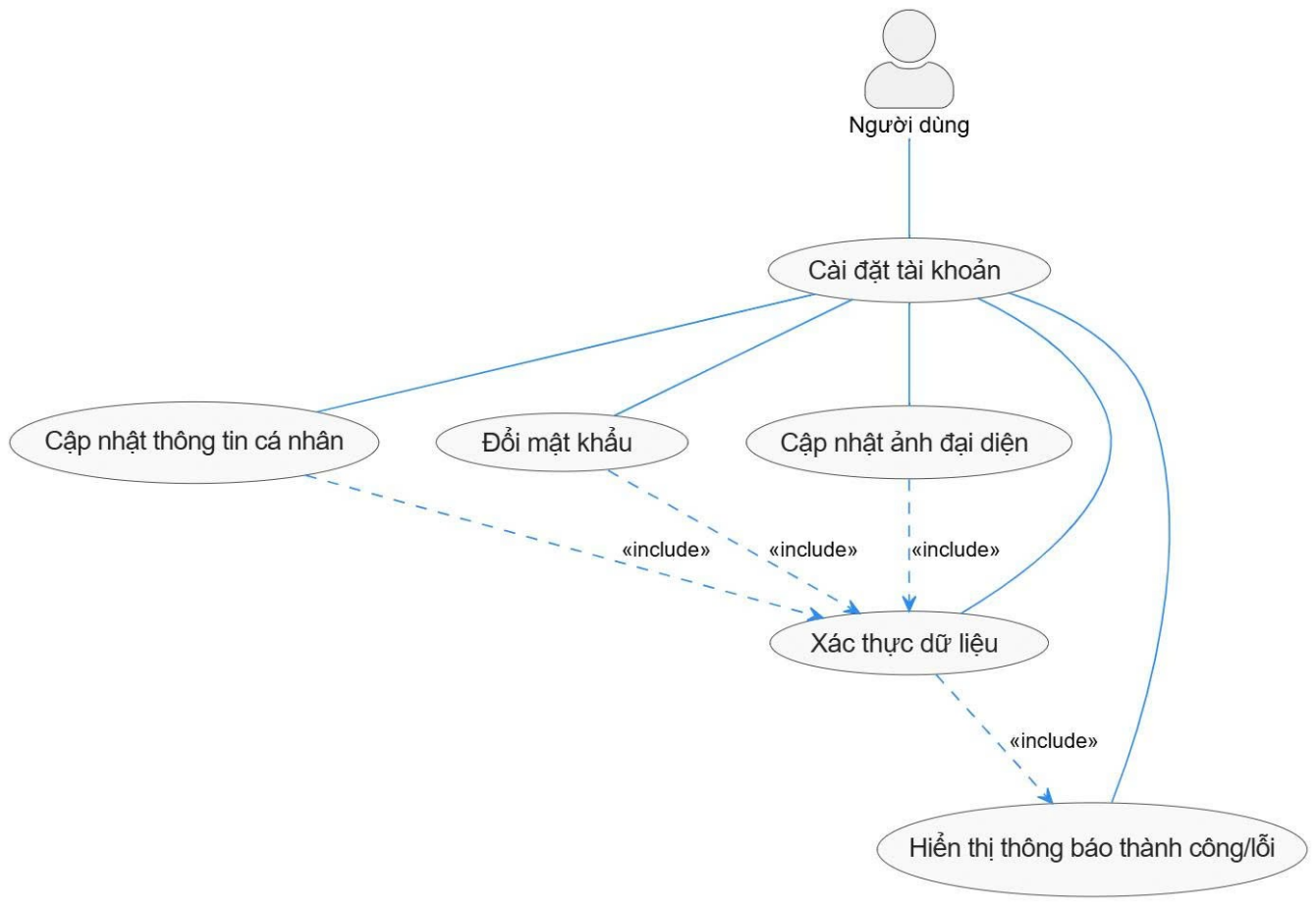
Hình  Sơ đồ usecase Quản lý đánh giá

* **Tên Use Case**: **Quản lý nhắc nhở**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Giúp người dùng tạo và quản lý các nhắc nhở liên quan đến chi tiêu, thu nhập hoặc mục tiêu tài chính.
* **Mô tả:**  
  Người dùng có thể:
* Thêm nhắc nhở mới.
* Tạo nhắc nhở định kỳ.
* Sửa nội dung nhắc nhở.
* Xóa một nhắc nhở hoặc toàn bộ nhắc nhở.
* Đánh dấu nhắc nhở đã đọc.
* Xem chi tiết một nhắc nhở.
  + 1. Use case quản lý thống kê



Hình  Sơ đồ usecase Quản lý danh mục

* **Tên Use Case**: **Quản lý thống kê**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Giúp người dùng theo dõi, phân tích và so sánh dữ liệu thu chi.
* **Mô tả:**  
  Người dùng có thể:
* Xem thống kê tổng quan.
* Xem thống kê theo tháng.
* Xem thống kê theo năm.
* Xem thống kê theo danh mục.
* Xem xu hướng chi tiêu.
* So sánh các danh mục chi tiêu/thu nhập.
  + 1. Use case quản lý tài khoản

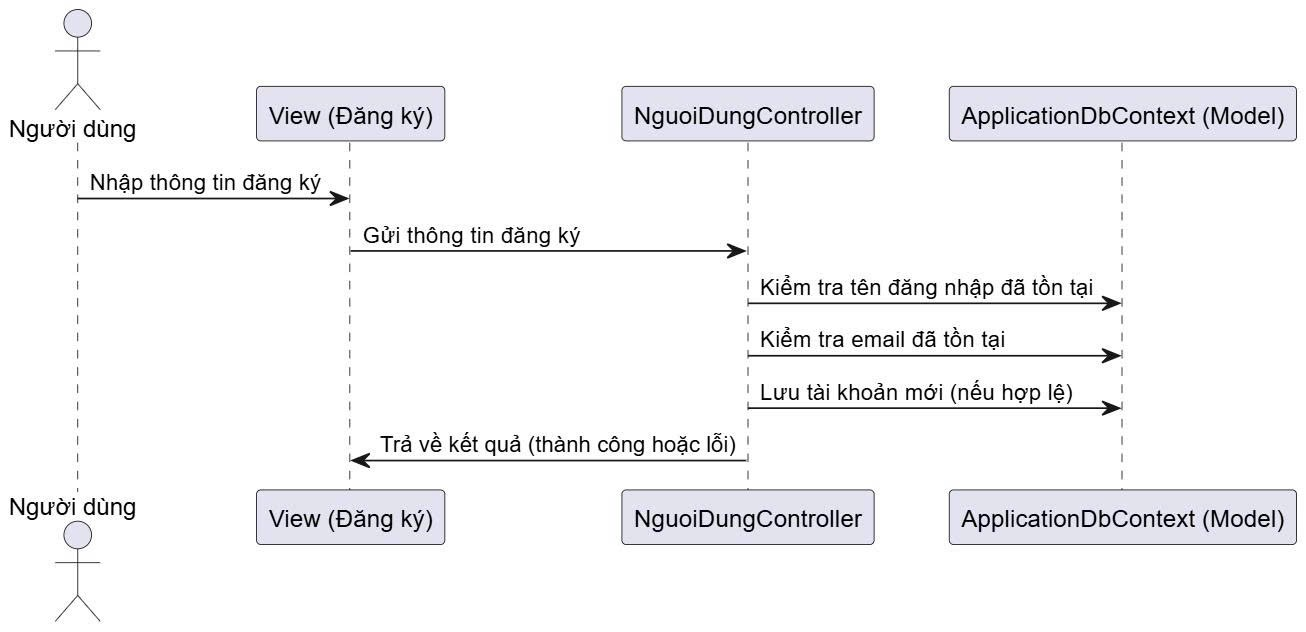


Hình  Sơ đồ usecase Quản lý người dùng

* **Tên Use Case**: **Cài đặt tài khoản**
* **Tác nhân:** Người dùng
* **Mục tiêu:** Cho phép người dùng quản lý và thay đổi thông tin tài khoản cá nhân.
* **Mô tả:**  
  Người dùng có thể:
* Cập nhật thông tin cá nhân (họ tên, email, số điện thoại, địa chỉ,…).
* Đổi mật khẩu đăng nhập.
* Cập nhật ảnh đại diện.
* Thực hiện **xác thực dữ liệu** trước khi lưu thay đổi.
* Nhận **thông báo thành công/thất bại** sau khi thao tác.

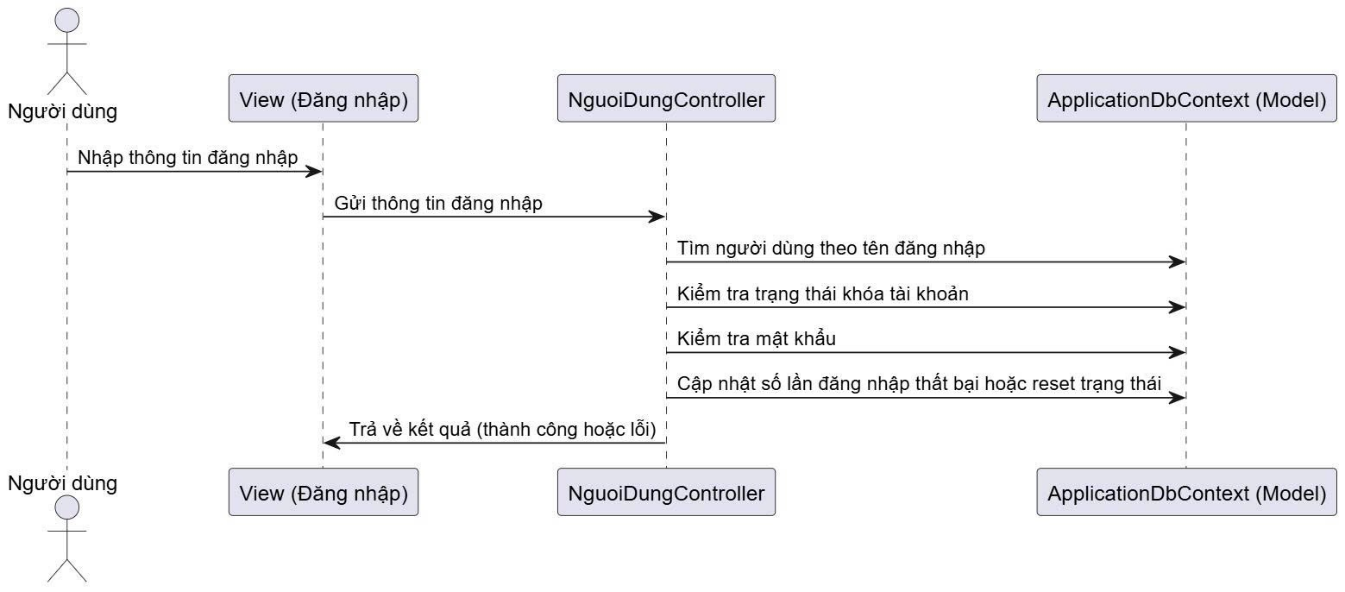
## Sơ đồ tuần tự

* + 1. Đăng ký



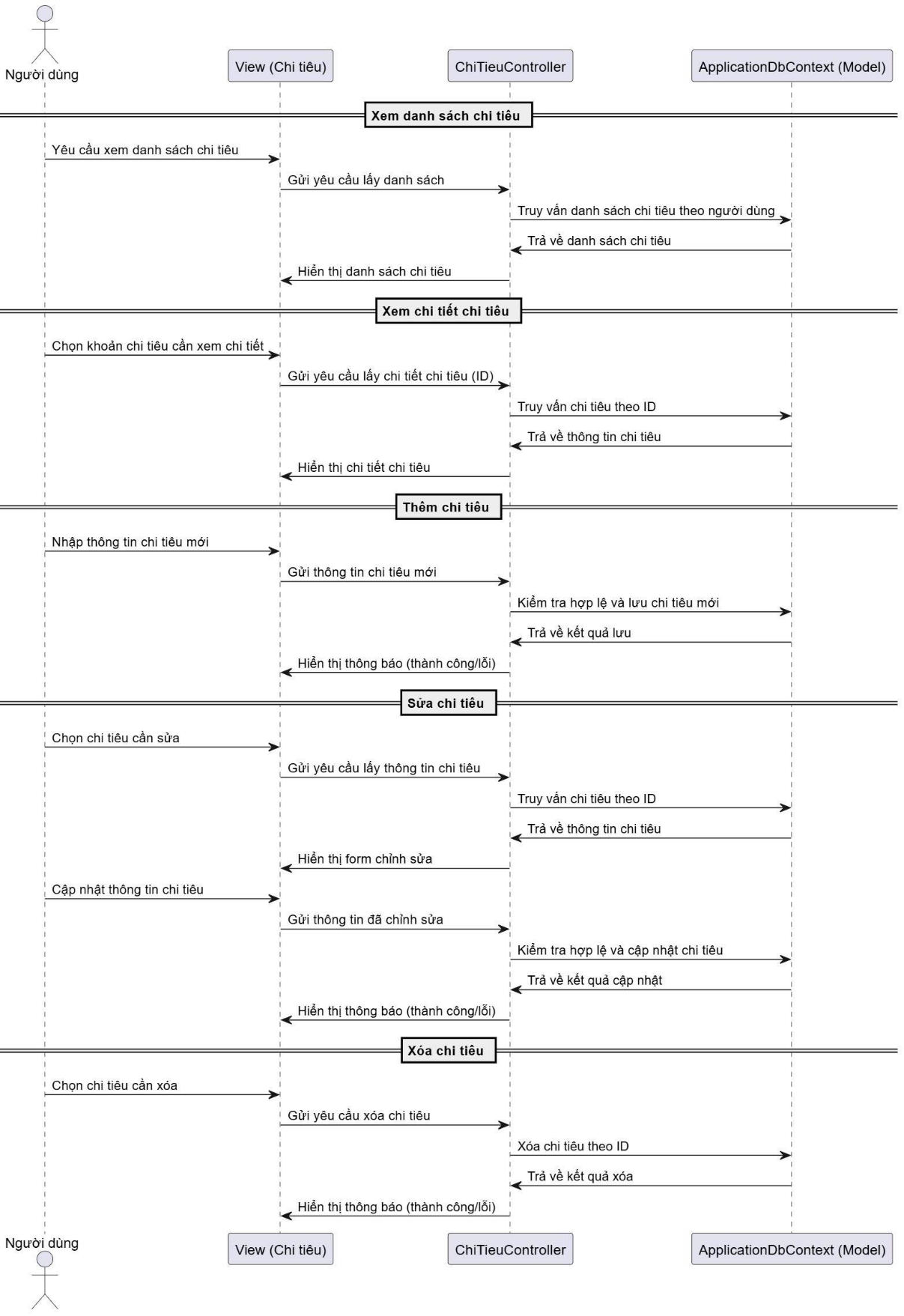
Hình  sơ đồ tuần tự đăng ký

* + 1. Đăng nhập



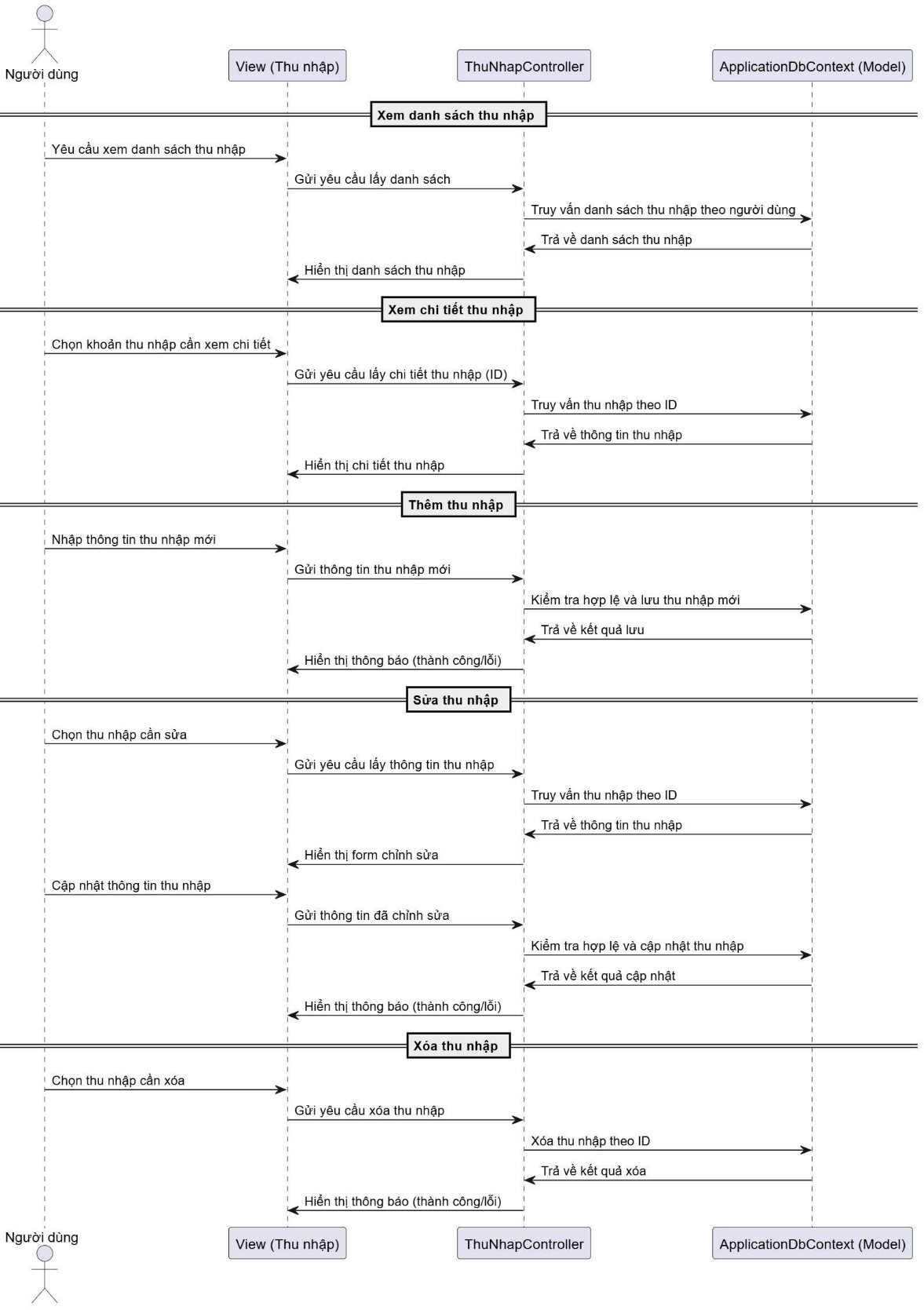
Hình Sơ đồ tuần tự đăng nhập

* + 1. quản lý chi tiêu



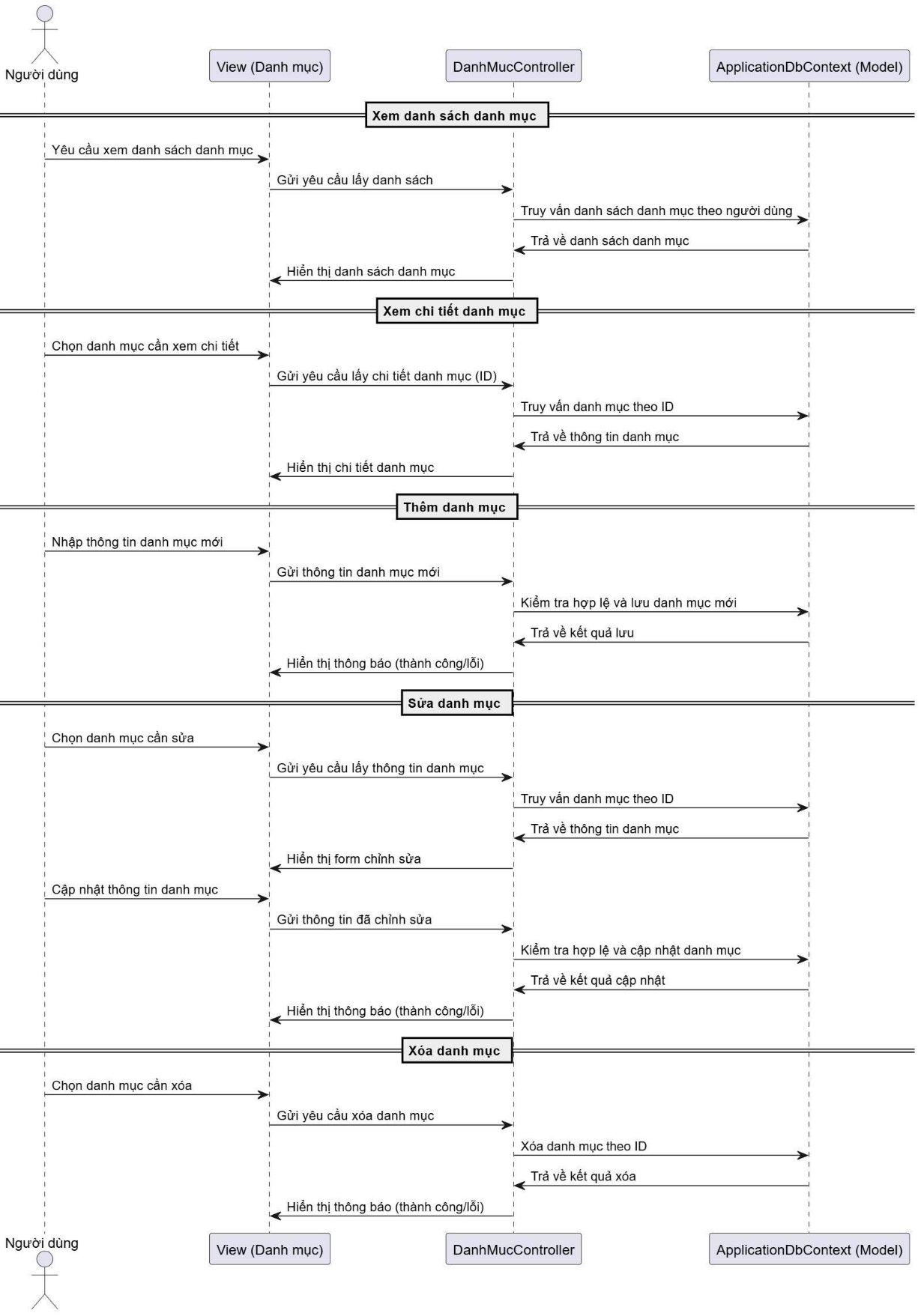
Hình  sơ đồ tuần tự quản lý chi tiêu

* + 1. quản lý thu nhập



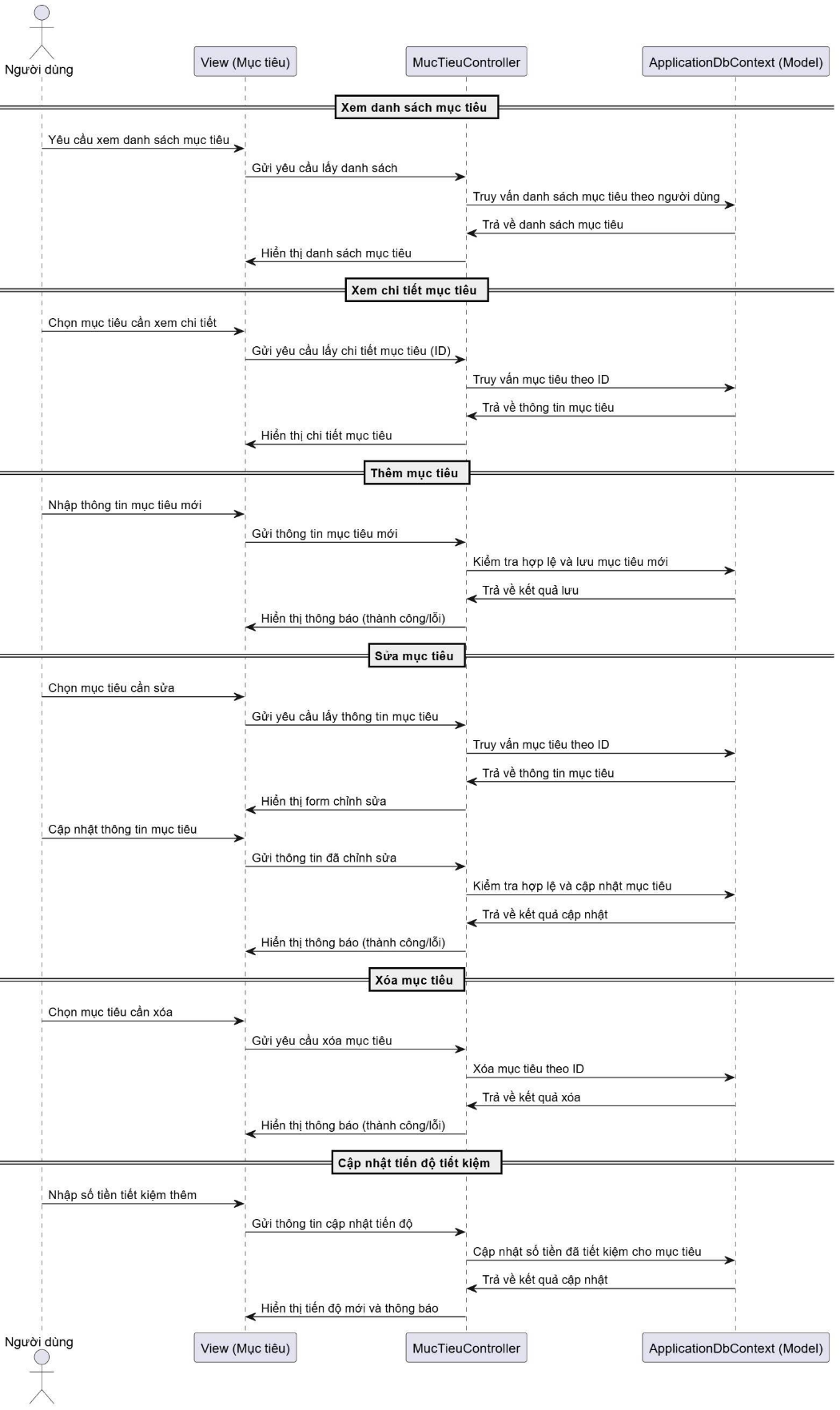
Hình Sơ đồ tuần tự quản lý thu nhập

* + 1. quản lý danh mục



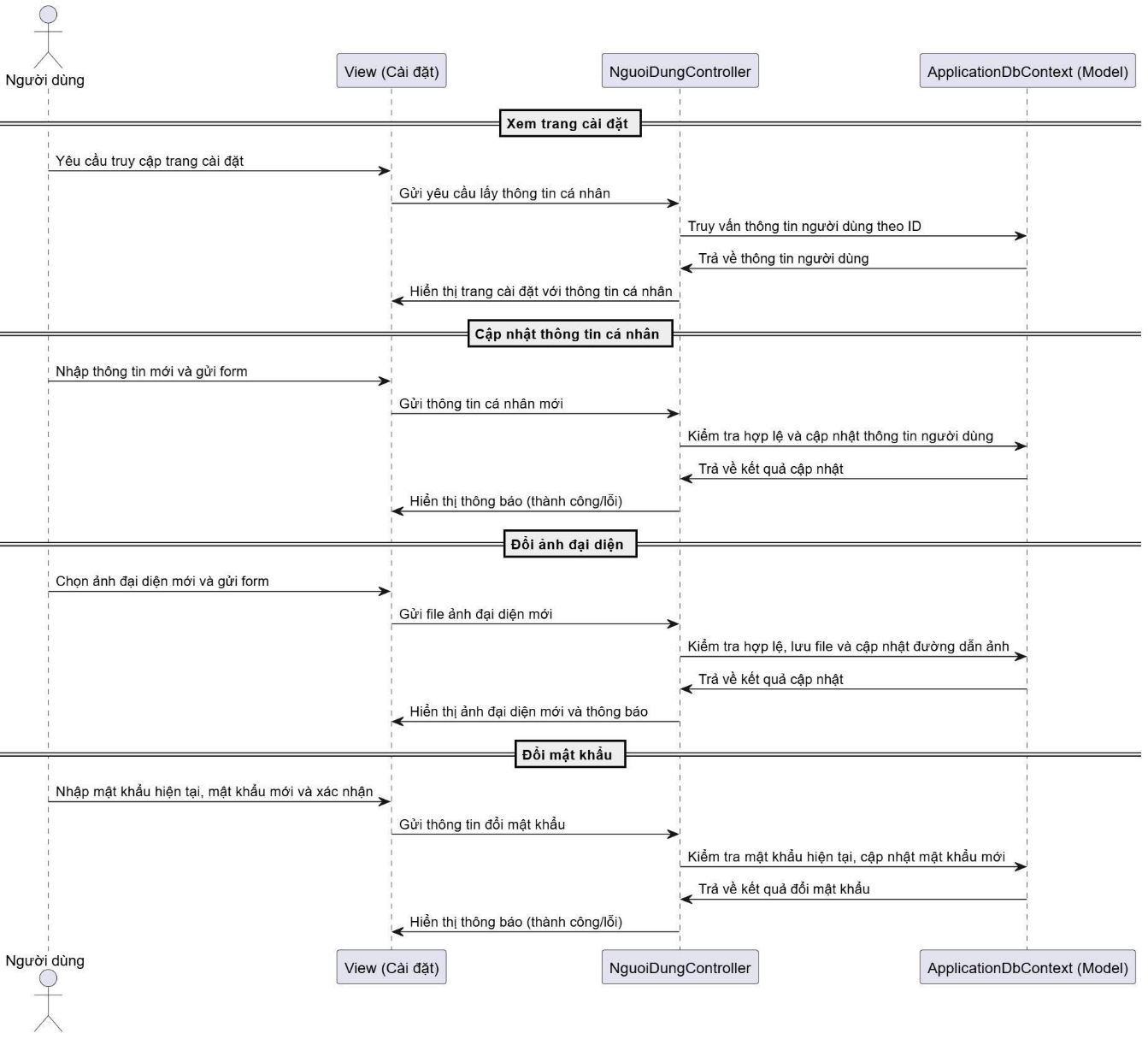
Hình sơ đồ tuần tự quản lý danh mục

* + 1. quản lý mục tiêu



Hình sơ đồ tuần tự quản lý mục tiêu

* + 1. quản lý người dùng



Hình sơ đồ tuần tự cập nhật thông tin người dùng

# KẾT QUẢ NGHIÊN CỨU

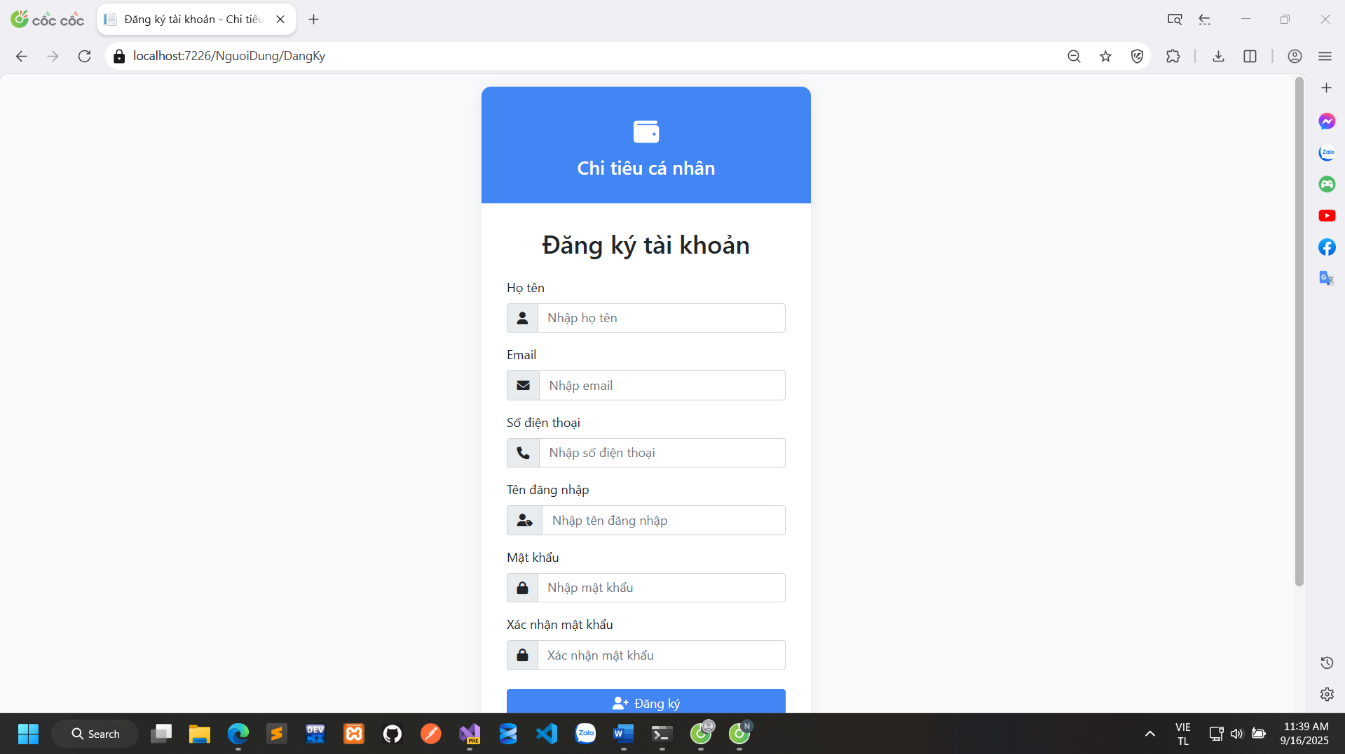
## Giao diện trang chủ

A screenshot of a website

AI-generated content may be incorrect.

Hình giao diện trang chủ

## Giao diện đăng ký

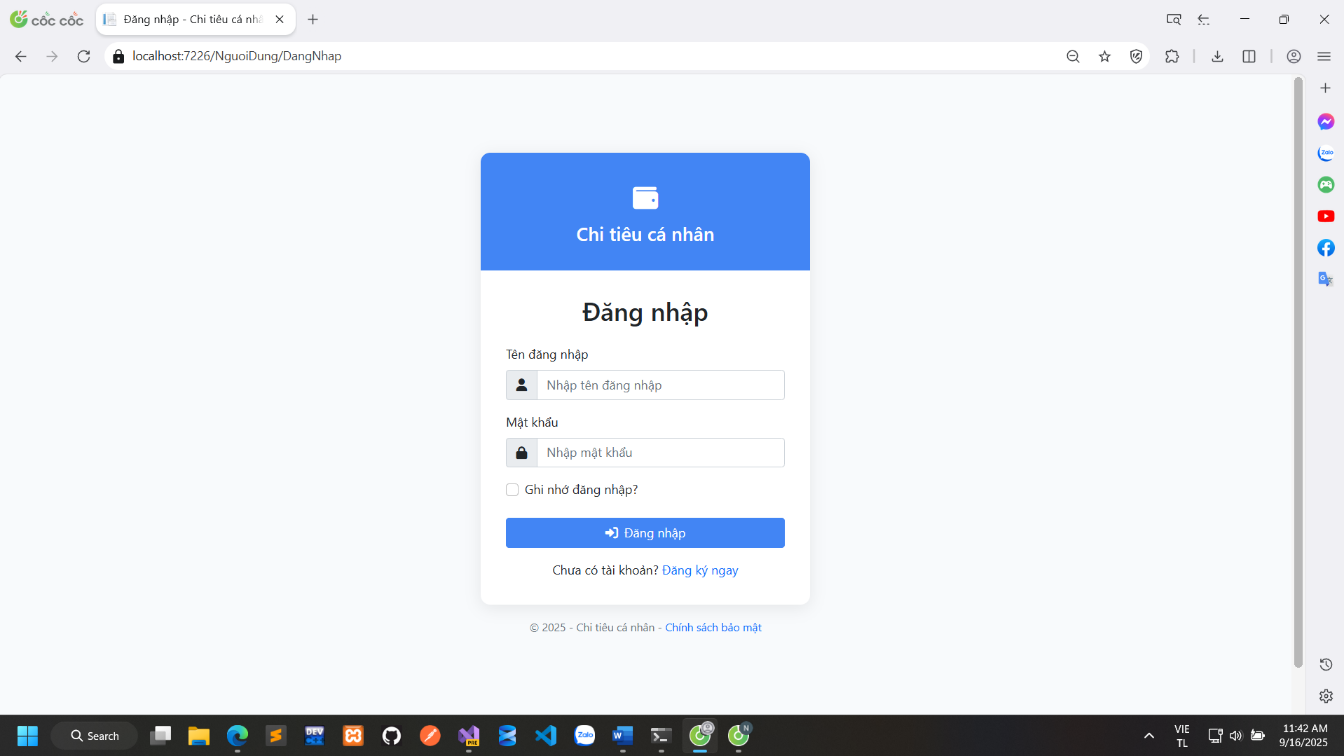


Hình giao diện đăng ký tài khoản

**Chức năng : Đăng ký tài khoản**

* **Mô tả**: Website cung cấp giao diện để người dùng nhập thông tin cá nhân như họ tên, email, số điện thoại, tên đăng nhập và mật khẩu.
* **Kết quả**: Hệ thống sẽ kiểm tra dữ liệu hợp lệ, lưu thông tin vào cơ sở dữ liệu và tạo tài khoản mới cho người dùng. Sau khi đăng ký thành công, người dùng có thể đăng nhập để sử dụng đầy đủ các tính năng của website.

## Giao diện đăng nhập



Hình giao diện đăng nhập

**Chức năng : Đăng nhập**

* **Mô tả:** Website cung cấp giao diện để người dùng nhập thông tin đăng nhập gồm tên đăng nhập và mật khẩu. Ngoài ra, hệ thống hỗ trợ tùy chọn “Ghi nhớ đăng nhập” giúp người dùng không cần nhập lại thông tin ở các lần sau.
* **Kết quả:** Hệ thống sẽ kiểm tra thông tin đăng nhập với dữ liệu trong cơ sở dữ liệu. Nếu hợp lệ, người dùng sẽ được truy cập vào hệ thống và sử dụng các tính năng quản lý chi tiêu cá nhân. Nếu không hợp lệ, hệ thống sẽ thông báo lỗi và yêu cầu nhập lại.

## Giao diện tổng quan

A screenshot of a computer

AI-generated content may be incorrect.

Hình giao diện tổng quan

**Chức năng: Tổng quan tài chính**

* **Mô tả:** Website cung cấp giao diện bảng điều khiển giúp người dùng theo dõi tình hình tài chính cá nhân. Giao diện hiển thị các thông tin tổng hợp như: tổng thu nhập tháng, tổng chi tiêu tháng, số dư khả dụng. Ngoài ra, hệ thống còn trực quan hóa dữ liệu qua biểu đồ tròn (chi tiêu theo danh mục) và biểu đồ cột (xu hướng thu chi theo thời gian). Người dùng có thể dễ dàng thêm nhanh “Chi tiêu” hoặc “Thu nhập” từ màn hình này.
* **Kết quả:** Hệ thống hỗ trợ người dùng quản lý và đánh giá tình hình tài chính một cách trực quan và chi tiết. Người dùng nắm bắt nhanh được số dư, các khoản chi tiêu, thu nhập và xu hướng tài chính để đưa ra quyết định chi tiêu hợp lý.

## Cập nhật thông tin cá nhân

A screenshot of a computer

AI-generated content may be incorrect.

Hình giao diện cập nhật thông tin cá nhân

**Chức năng: Cài đặt tài khoản**

* **Mô tả:** Website cung cấp giao diện để người dùng quản lý và chỉnh sửa thông tin cá nhân. Người dùng có thể:
  + Cập nhật ảnh đại diện bằng cách tải lên tệp hình ảnh (định dạng JPG, PNG, GIF, dung lượng tối đa 5MB).
  + Thay đổi thông tin cơ bản như họ tên, email, số điện thoại và địa chỉ.
  + Truy cập tab “Đổi mật khẩu” để thay đổi mật khẩu đăng nhập.
* **Kết quả:** Hệ thống sẽ kiểm tra dữ liệu nhập vào, nếu hợp lệ sẽ cập nhật thông tin trong cơ sở dữ liệu. Người dùng có thể quản lý thông tin cá nhân của mình một cách thuận tiện và an toàn, đảm bảo tính chính xác và bảo mật khi sử dụng hệ thống.

## Thêm mục tiêu

A screenshot of a computer

AI-generated content may be incorrect.

Hình giao diện thêm mục tiêu

**Chức năng: Tạo mục tiêu tiết kiệm**

* **Mô tả:** Website cung cấp giao diện cho phép người dùng thiết lập các mục tiêu tài chính cá nhân. Người dùng có thể nhập tên mục tiêu (ví dụ: mua xe máy, du lịch), số tiền cần đạt, số tiền đã tiết kiệm và thời hạn hoàn thành mục tiêu. Ngoài ra, hệ thống hiển thị các mẹo thiết lập mục tiêu hiệu quả giúp người dùng dễ dàng quản lý và lập kế hoạch tài chính.
* **Kết quả:** Sau khi nhập thông tin và nhấn “Tạo mục tiêu”, hệ thống sẽ lưu dữ liệu vào cơ sở dữ liệu và thêm mục tiêu mới cho người dùng. Người dùng có thể theo dõi tiến độ tiết kiệm, so sánh với số tiền cần thiết và điều chỉnh mục tiêu khi cần thiết để đạt được kế hoạch đề ra.

## Giao diện thông báo

A screenshot of a computer

AI-generated content may be incorrect.

Hình giao diện thông báo và nhắc nhở

**Chức năng: Nhắc nhở & Cảnh báo**

* **Mô tả:** Website cung cấp giao diện quản lý các nhắc nhở liên quan đến tài chính cá nhân. Người dùng có thể:
  + Tạo nhắc nhở mới hoặc nhắc nhở định kỳ.
  + Theo dõi số lượng nhắc nhở sắp tới, nhắc nhở quá hạn và tổng số nhắc nhở.
  + Quản lý danh sách nhắc nhở với các thông tin chi tiết như: nội dung, thời gian, loại nhắc nhở, trạng thái.
  + Thực hiện các thao tác như xem chi tiết, chỉnh sửa, đánh dấu hoàn thành hoặc xóa nhắc nhở.
* **Kết quả:** Hệ thống giúp người dùng không bỏ lỡ các sự kiện hoặc kế hoạch quan trọng liên quan đến tài chính. Các nhắc nhở hiển thị rõ ràng, trực quan và có cảnh báo khi đến hạn, hỗ trợ người dùng chủ động hơn trong việc quản lý chi tiêu và mục tiêu.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Đề tài đã xây dựng được hệ thống quản lý chi tiêu cá nhân với các chức năng chính như: đăng ký, đăng nhập, quản lý thu nhập – chi tiêu, thiết lập mục tiêu tiết kiệm, nhắc nhở tài chính và báo cáo tổng quan. Hệ thống hỗ trợ người dùng theo dõi tình hình tài chính một cách trực quan, giúp nâng cao khả năng quản lý và lập kế hoạch chi tiêu.

Kết quả đạt được:

* Xây dựng giao diện thân thiện, dễ sử dụng.
* Tích hợp các chức năng quản lý tài chính cá nhân cơ bản.
* Cung cấp báo cáo trực quan bằng biểu đồ và bảng thống kê.
* Đảm bảo lưu trữ, cập nhật và xử lý dữ liệu người dùng một cách an toàn.

## Hướng phát triển

Trong tương lai, hệ thống có thể được mở rộng và nâng cấp theo các hướng sau:

* Phát triển ứng dụng di động để tăng tính tiện lợi cho người dùng.
* Tích hợp trí tuệ nhân tạo (AI) để phân tích, dự đoán xu hướng chi tiêu và đưa ra gợi ý tối ưu.
* Hỗ trợ liên kết với tài khoản ngân hàng để tự động cập nhật thu chi.
* Bổ sung tính năng chia sẻ và so sánh chi tiêu trong nhóm hoặc gia đình.
* Nâng cao khả năng bảo mật và xác thực nhiều lớp nhằm đảm bảo an toàn cho dữ liệu người dùng.

# TÀI LIỆU THAM KHÁO

**Sách**

|  |  |
| --- | --- |
| [1] | J. Chadwick, T. Snyder and H. Panda, Programming ASP.NET MVC 4: Developing Real-World Web Applications with ASP.NET MVC, Sebastopol, CA: O'Reilly Media, Inc., 2012. |
| [2] | S. Walther, ASP.NET MVC Framework Unleashed, Indianapolis: Sams Publishing, 2009.  **Websites** |
| [3] | Udemy, "Mastering ASP.NET MVC 5," 8 2020. [Online]. Available: https://www.udemy.com/course/masteringaspnetmvc5/. [Accessed 17 6 2025]. |
| [4] | Kteam, "Lập trình website với ASP.NET MVC cơ bản," HowKteam.vn, 1 6 2015. [Online]. Available: https://howkteam.vn/course/lap-trinh-website-voi-aspnet-mvc-co-ban-50. [Accessed 17 6 2025]. |