

# SmartWallet- Ứng dụng ví điện tử thông minh

Nguyễn Minh Dương -23010441

Lập trình cho thiết bị di động-1-1-25(N04)

Nhóm: 2025\_LTTBDD\_N04\_Nhom\_Duong

Hướng dẫn: ThS. Nguyễn Xuân Quế



# Giới thiệu về SmartWallet

SmartWallet là một ứng dụng ví điện tử hiện đại được phát triển bằng Flutter, cung cấp giải pháp quản lý tài chính cá nhân toàn diện với giao diện thân thiện và các tính năng thông minh, và hỗ trợ tiếng Việt và tiếng Anh

## Nhu cầu tăng cao

Quản lý tài chính cá nhân ngày càng trở nên quan trọng với nhóm tuổi 18-50, khi họ cần một công cụ hiệu quả như SmartWallet để theo dõi và kiểm soát chi tiêu.

## Tại sao chọn ứng dụng

**SmartWallet** được phát triển để đáp ứng **nhu cầu quản lý tài chính cá nhân** ngày càng tăng, giúp người dùng dễ dàng theo dõi chi tiêu và quản lý tài chính.

# Phân tích Yêu cầu Hệ thống

## Yêu cầu chức năng và phi chức năng

### Yêu cầu chức năng

Ứng dụng SmartWallet cần **xác thực người dùng**, cho phép người dùng đăng nhập và đăng ký dễ dàng, đồng thời quản lý ví tiền và theo dõi giao dịch một cách hiệu quả.

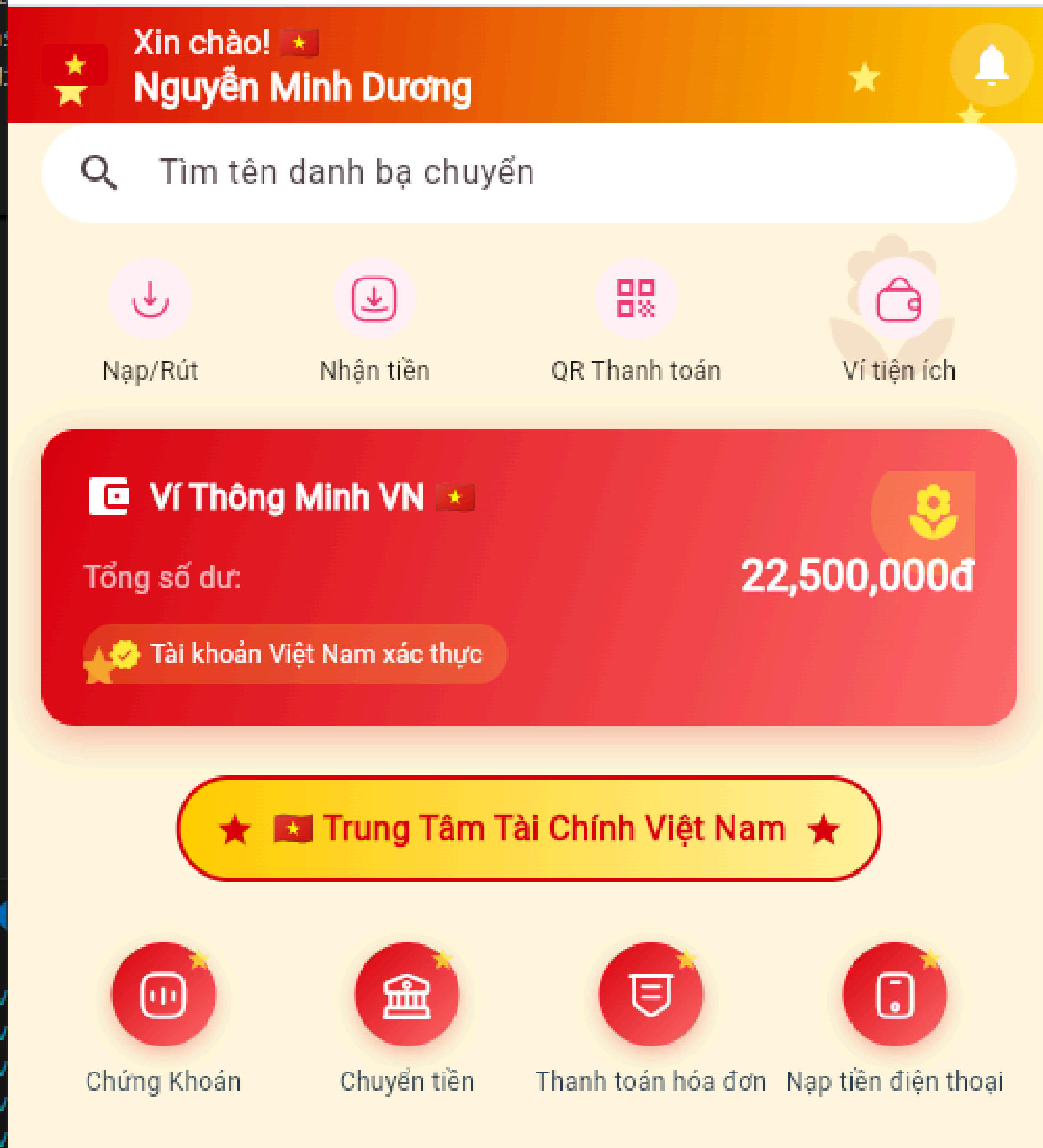
### Yêu cầu phi chức năng

Ứng dụng cần đảm bảo **hiệu năng tối ưu**, với thời gian phản hồi nhanh chóng và tuân thủ các tiêu chuẩn bảo mật cao, đồng thời thân thiện với người dùng trên nhiều thiết bị.

# Yêu cầu Chức năng

Các yêu cầu chức năng chính của ứng dụng SmartWallet để quản lý tài chính

- Xác thực người dùng qua đăng nhập, đăng ký
- Quản lý ví: tạo, sửa, xóa ví tiền
- Quản lý giao dịch: chuyển tiền và xem lịch sử



# Yêu cầu phi chức năng

## Hiệu năng và Bảo mật

### Hiệu năng ứng dụng

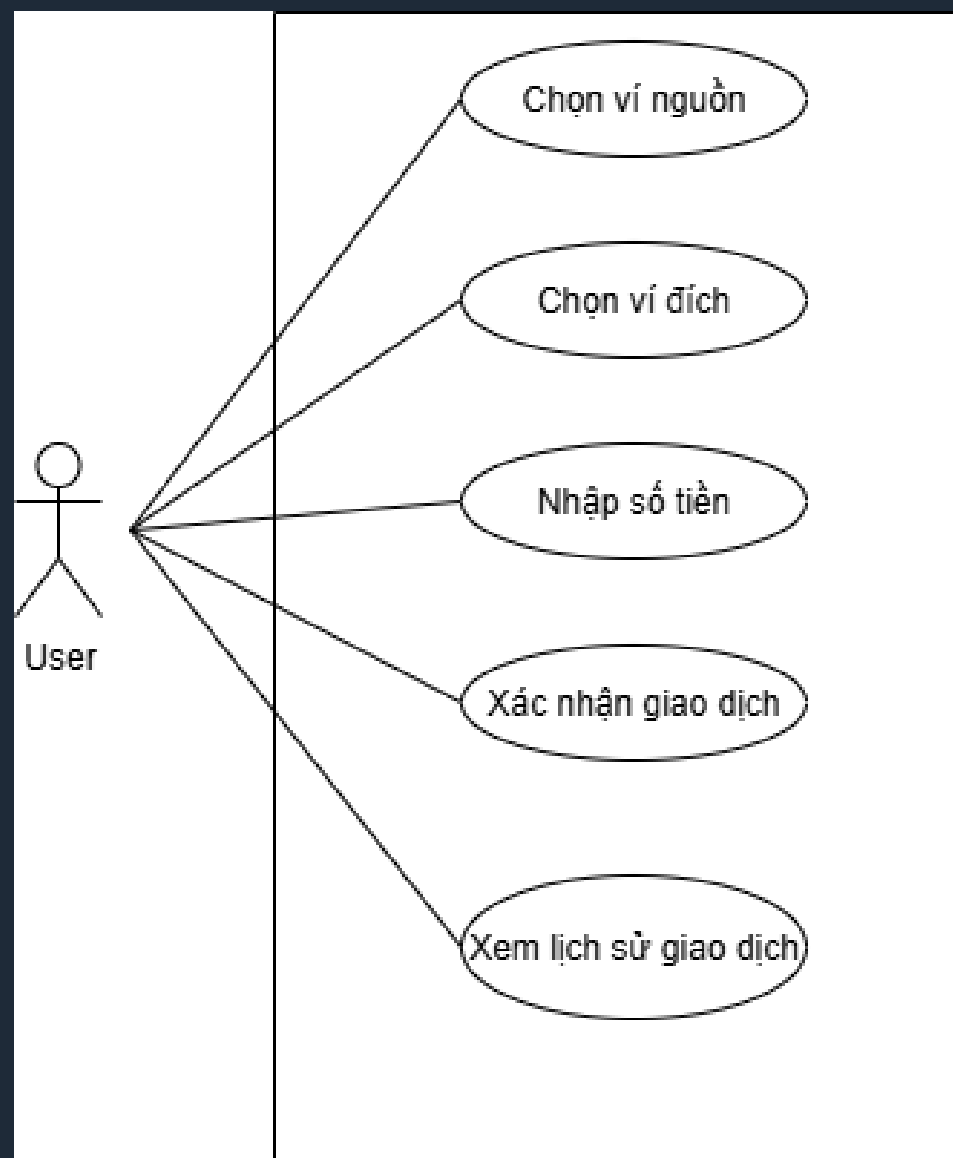
Ứng dụng cần **thời gian phản hồi nhanh** và mượt mà, đảm bảo trải nghiệm người dùng tốt nhất cho quá trình quản lý tài chính cá nhân một cách hiệu quả.

### Bảo mật dữ liệu

Bảo mật là yếu tố quan trọng, với mô phỏng quy trình bảo mật giúp bảo vệ thông tin cá nhân, đảm bảo an toàn cho người dùng khi thực hiện giao dịch.

# Thiết kế Hệ thống

## Mô hình và sơ đồ UML chi tiết



## Tác nhân chính

Hệ thống bao gồm hai tác nhân chính: User và System, mỗi tác nhân có vai trò cụ thể.

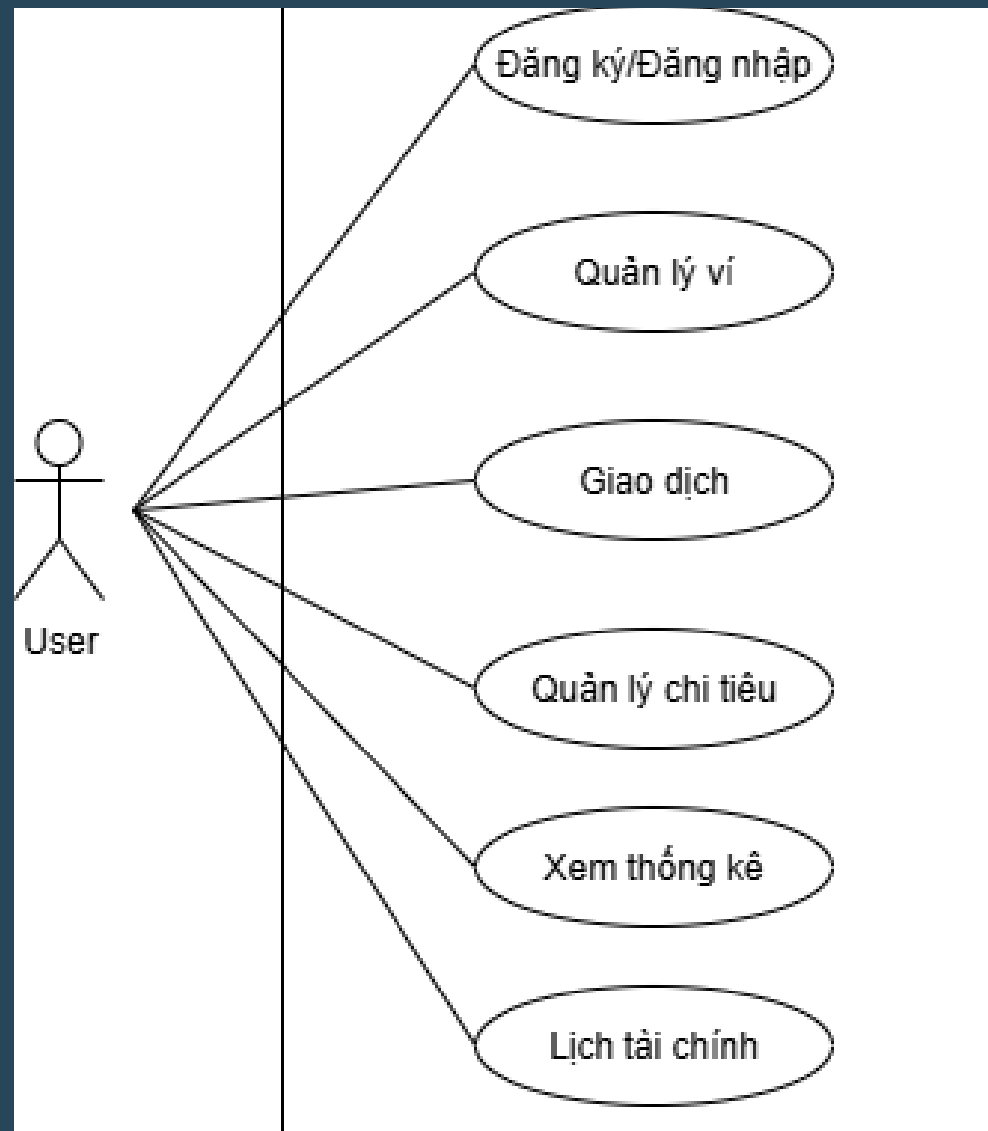
## Các chức năng chính

Sơ đồ Use Case mô tả các chức năng chính như đăng nhập, quản lý ví, và thống kê chi tiêu.

## Các mô hình lớp

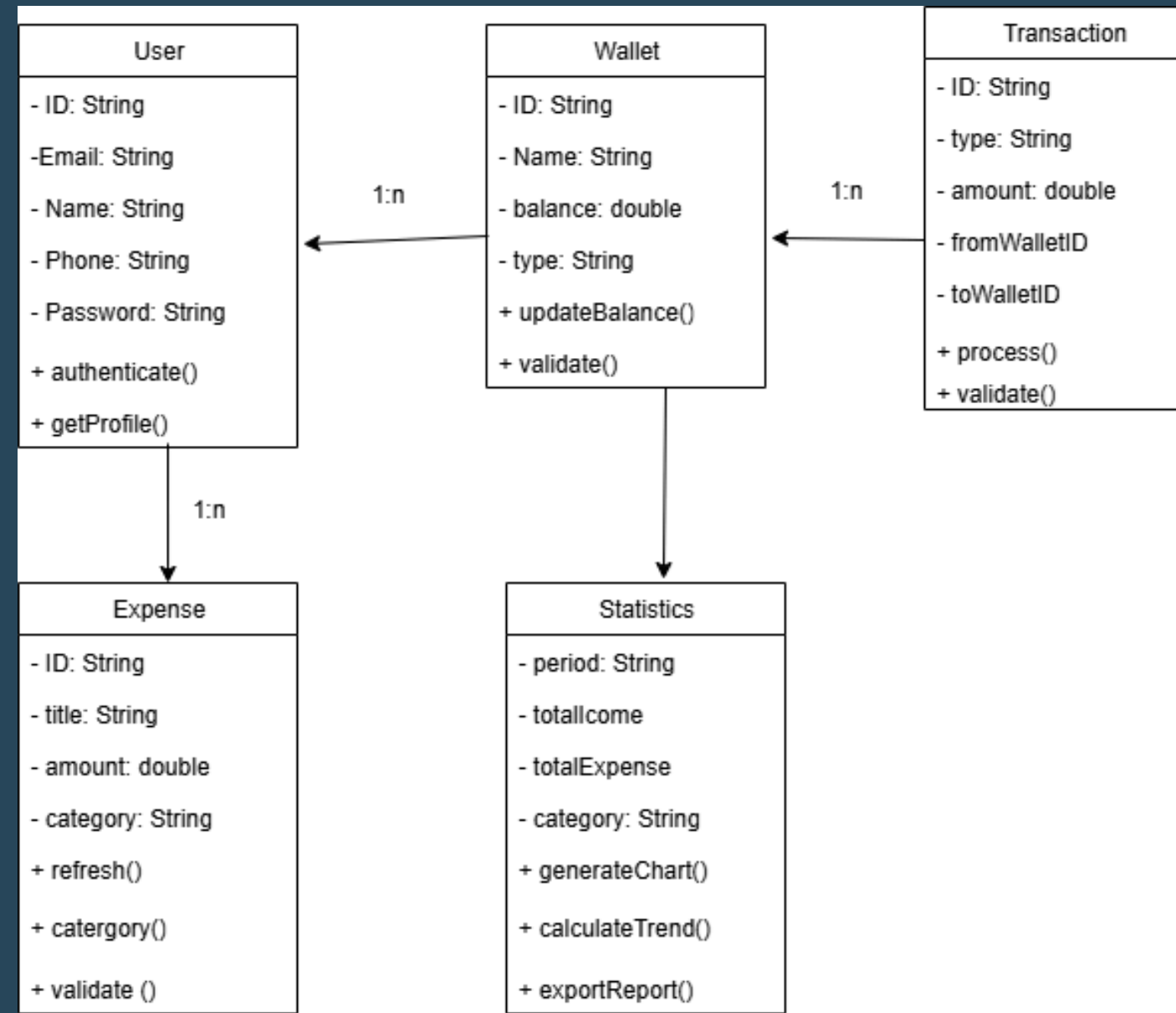
Sơ đồ Lớp trình bày chi tiết các lớp model như User, Wallet, Transaction, và Expense để tổ chức dữ liệu.

# Sơ đồ use-case và lớp của hệ thống



## Use Case

Chức năng cho người dùng



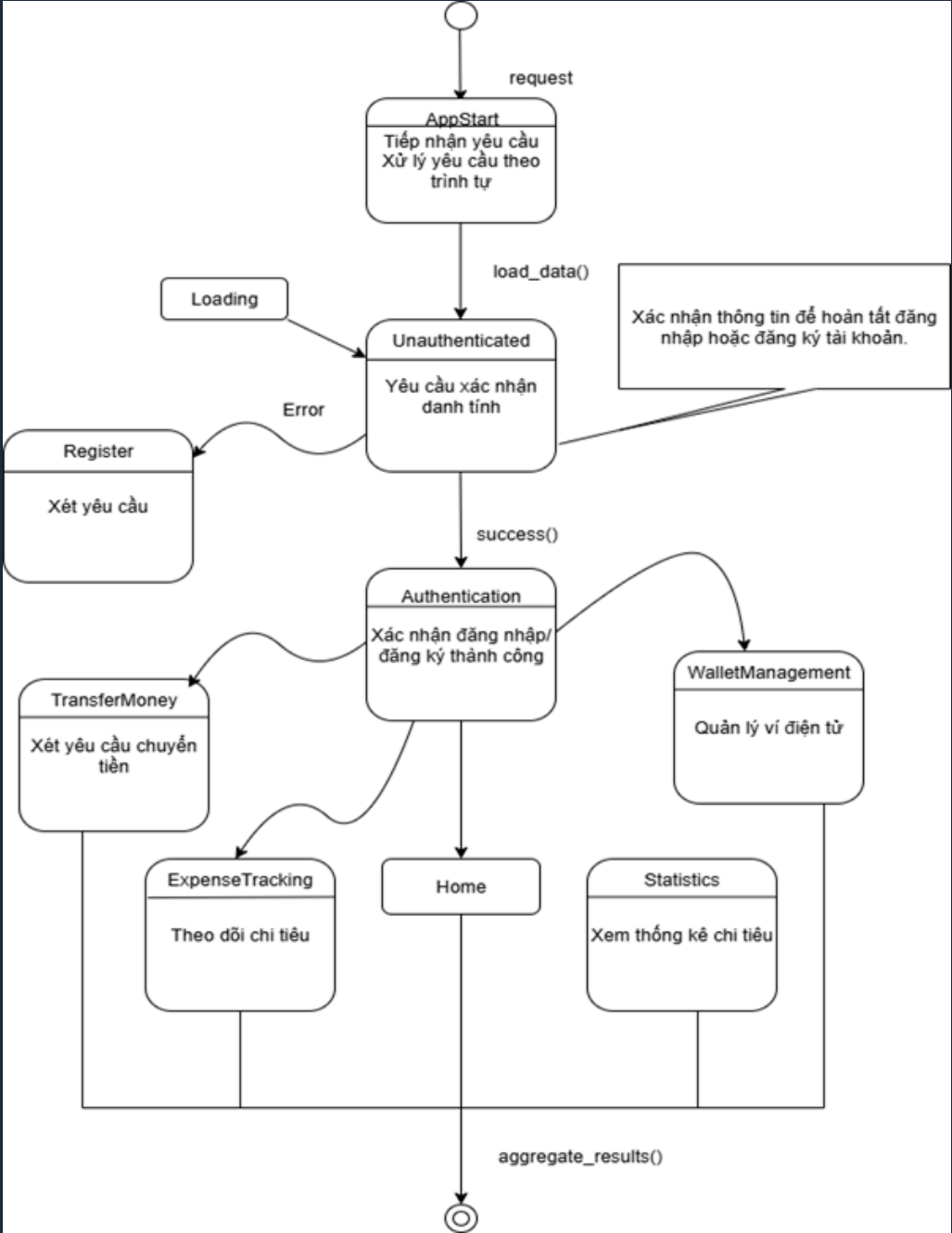
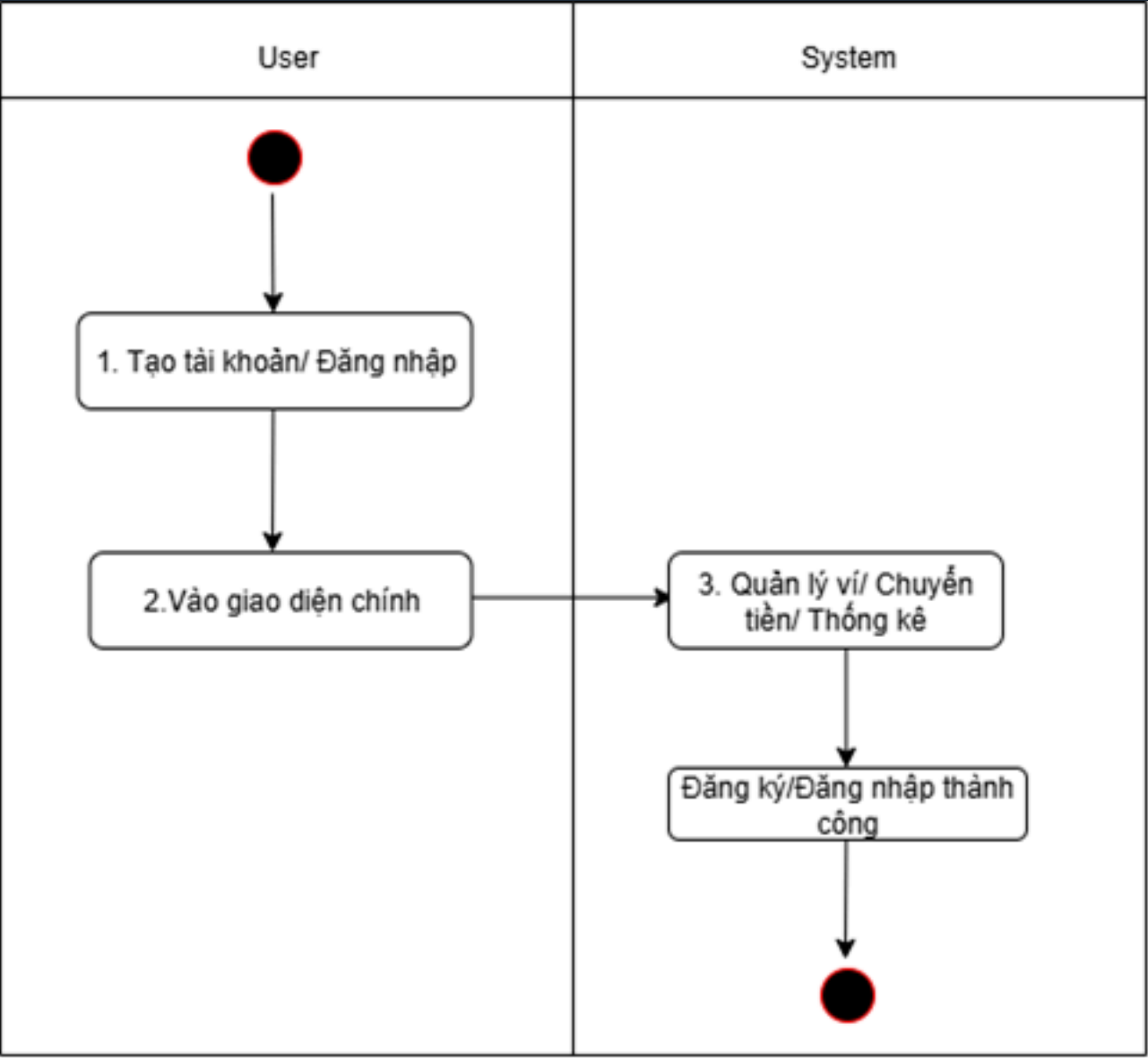
## Sơ đồ lớp

Cấu trúc thành phần ứng dụng



# Sơ đồ hoạt động và trạng thái

Hình ảnh mô tả quy trình xử lý đăng nhập, tạo ví và ghi nhận giao dịch trong ứng dụng SmartWallet, giúp người dùng hiểu rõ hơn về cách thức hoạt động và tương tác với hệ thống.





# Công nghệ và Công cụ

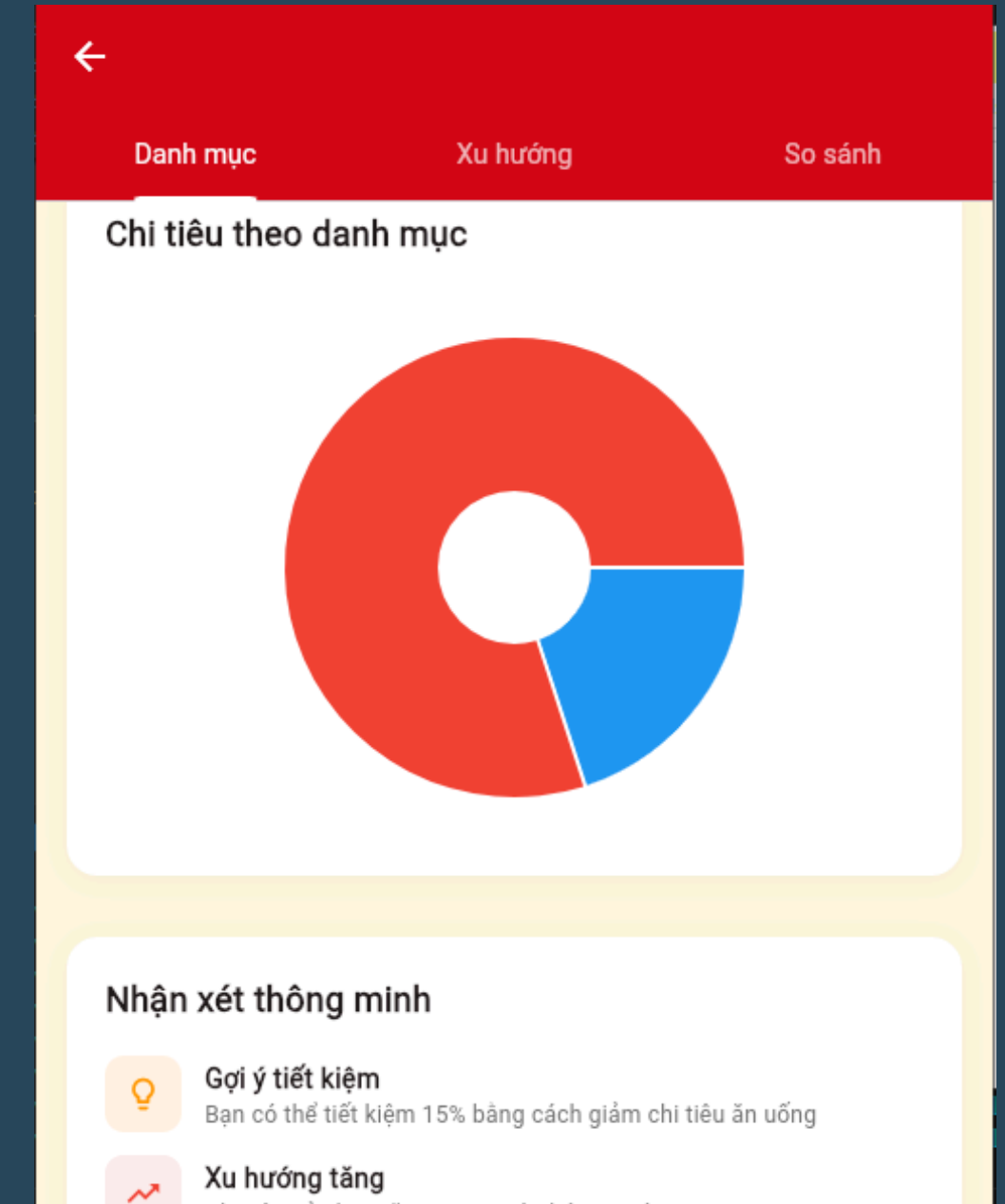
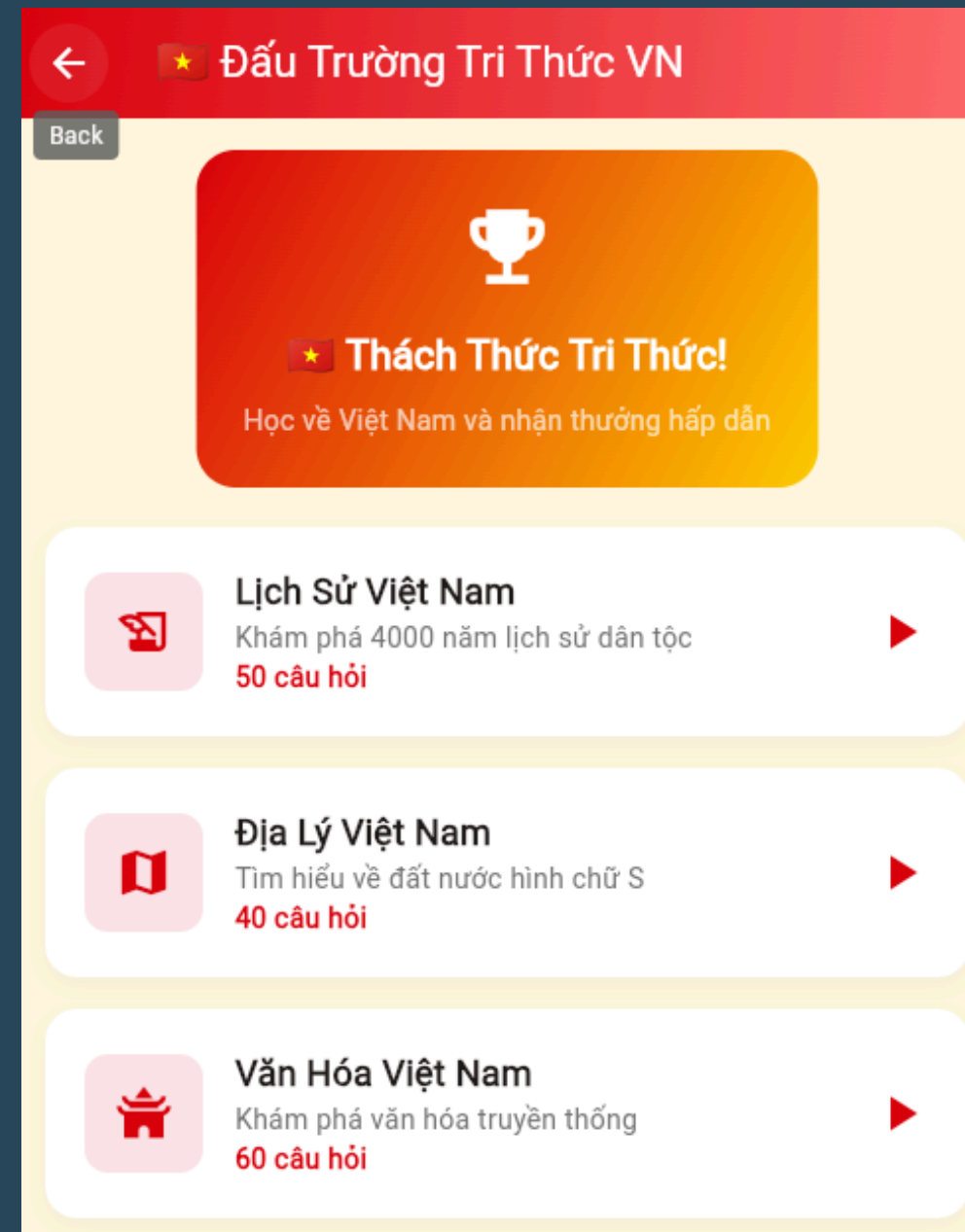
Các nền tảng và thư viện

## Flutter và Dart

Flutter 3.x cùng với ngôn ngữ Dart cho phép phát triển ứng dụng di động **một cách nhanh chóng** và hiệu quả.

## Quản lý trạng thái

Sử dụng thư viện Provider giúp quản lý trạng thái **mượt mà và hiệu quả**, tạo trải nghiệm người dùng tốt hơn.



# Cấu trúc và Nguyên tắc UI/UX

## Cấu trúc thư mục

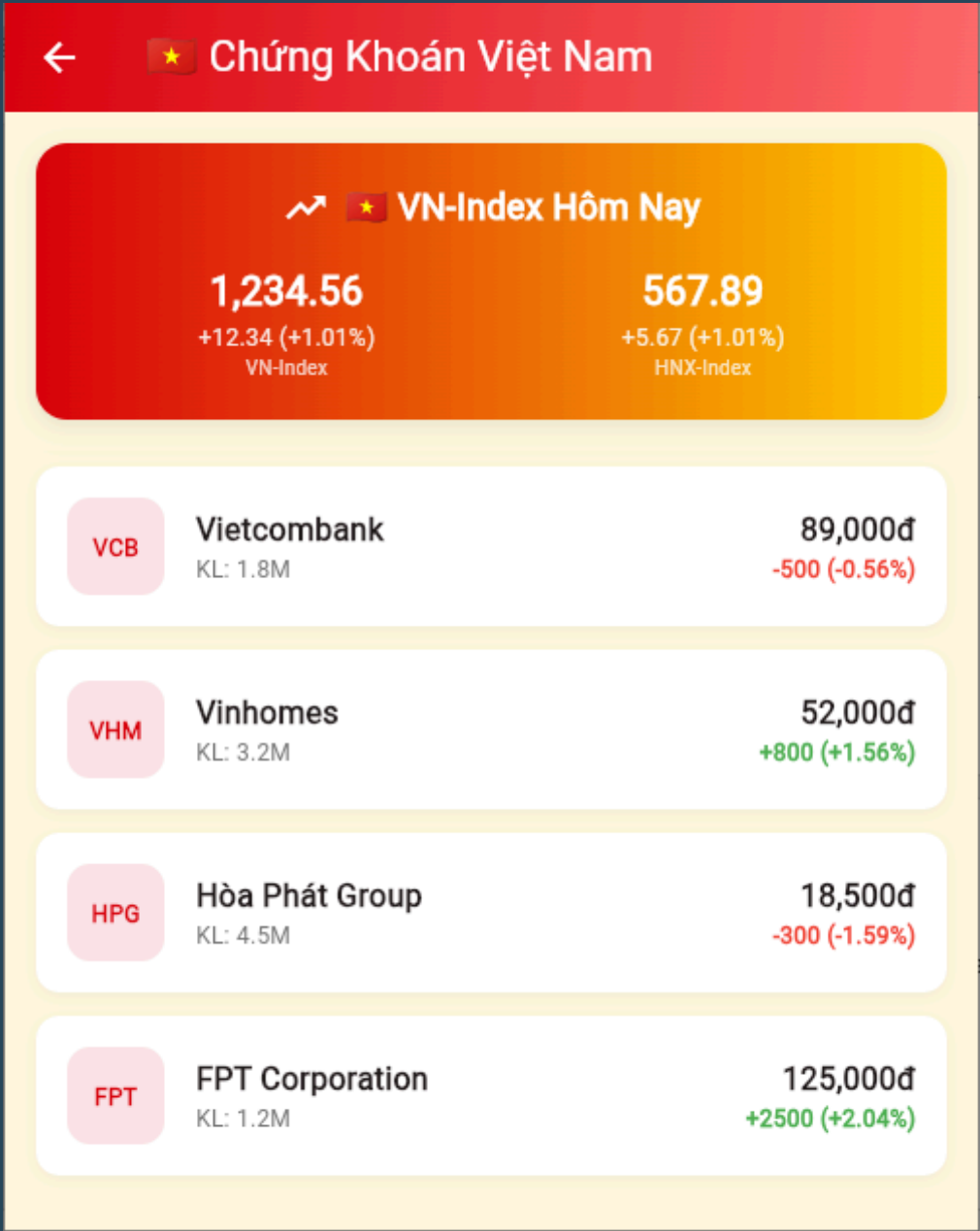
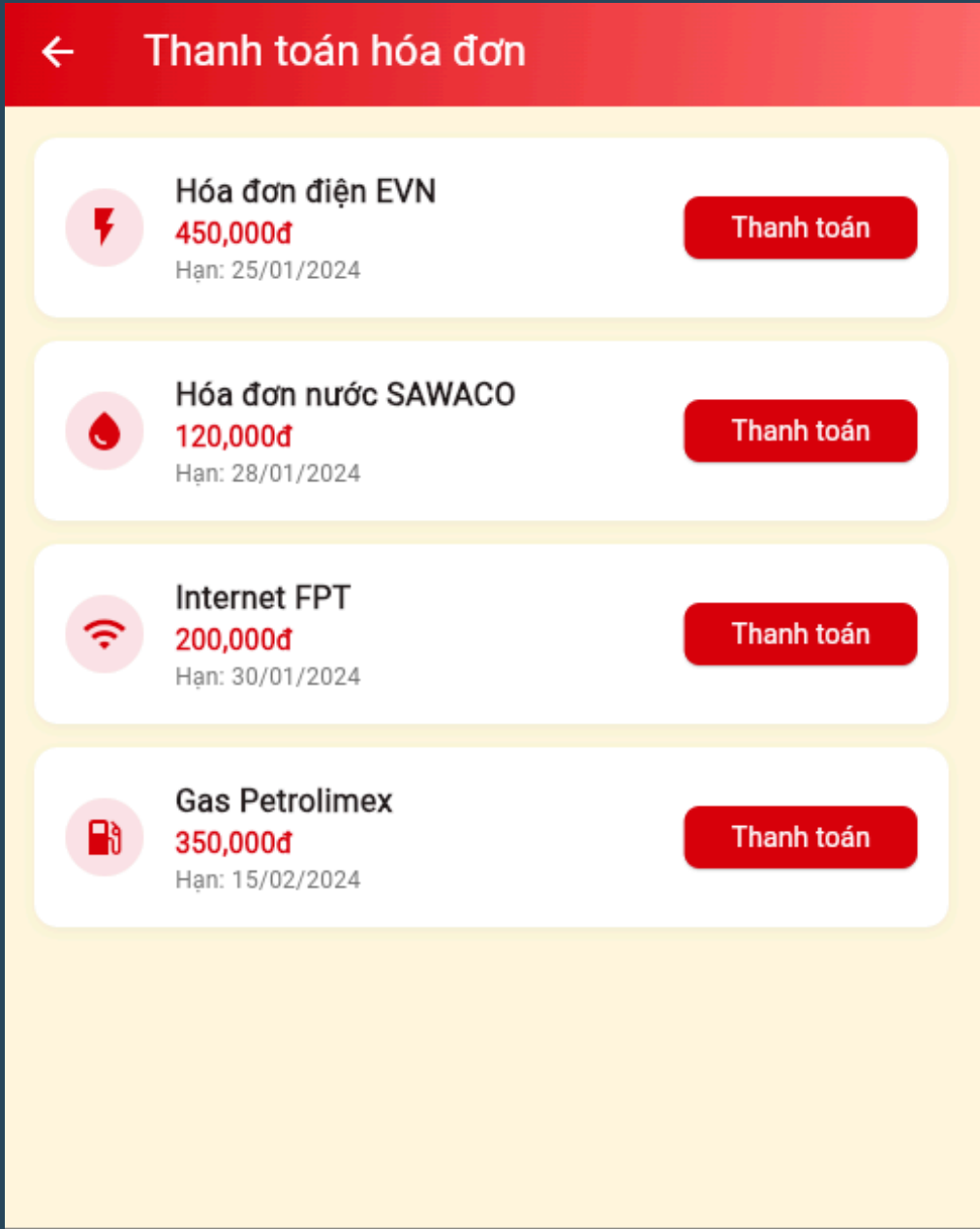
Dựa trên Feature-based kết hợp Clean Architecture.

## Kiến trúc 3 lớp

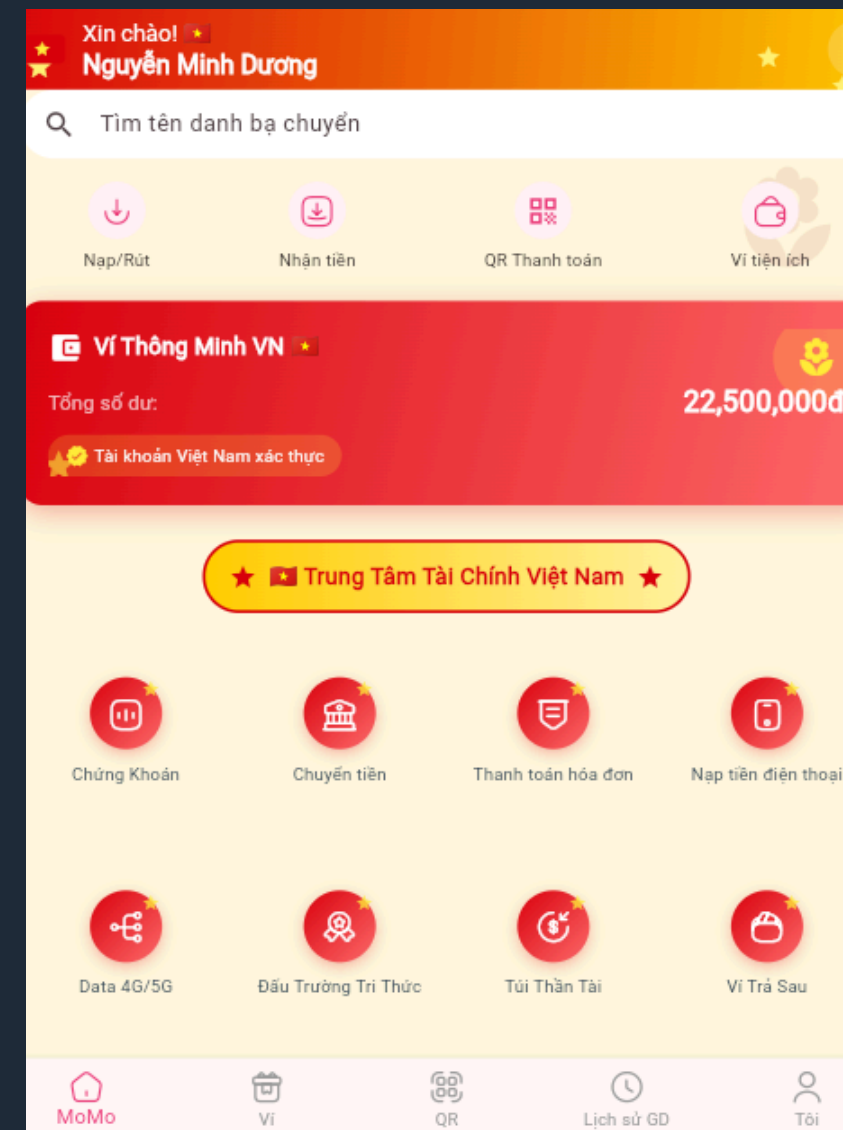
Gồm Presentation Layer, Business Logic Layer và Data Layer.

## Nguyên tắc thiết kế

Tuân thủ Material Design 3, đơn giản và nhất quán.

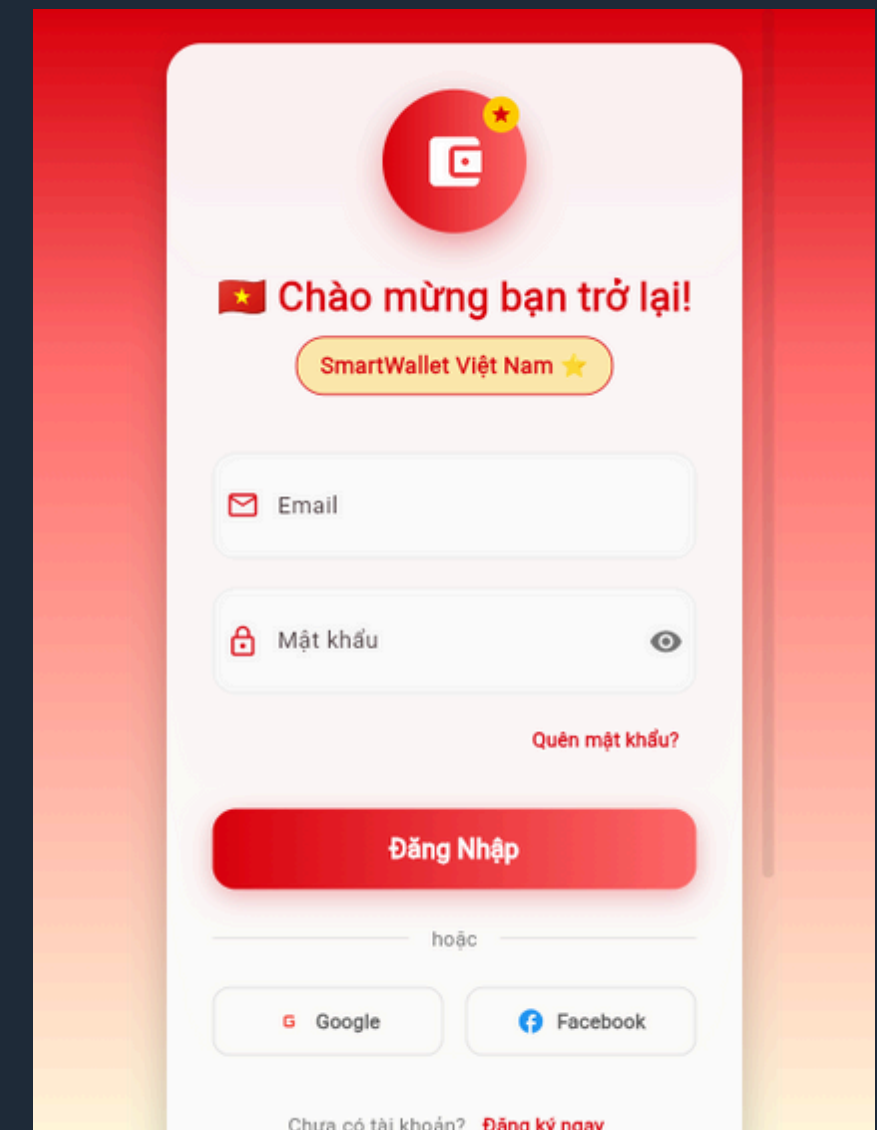


# Giao diện ứng dụng SmartWallet



## Màn hình chính

Quản lý chi tiêu và giao dịch cá nhân



## Màn hình đăng nhập

Đơn giản và dễ dàng truy cập ứng dụng

# Kiểm thử và Đánh giá

## Kết quả dự án

### Chiến lược kiểm thử

Chiến lược kiểm thử áp dụng Kim tự tháp Kiểm thử, với Unit Tests chiếm 70%, đảm bảo độ tin cậy của tính năng chính và giảm thiểu lỗi.

### Kết quả hoàn thành

Dự án đạt 100% tính năng cốt lõi, 95% giao diện và demo tốt bằng mock data, thể hiện sự hiệu quả trong thiết kế UI và kiến trúc ứng dụng.

# Kết luận

Tóm tắt dự án và đề xuất hướng phát triển tương lai

- Hiện thực hóa đầy đủ các chức năng cốt lõi gồm:
- Quản lý nhiều loại ví.
- Ghi nhận thu chi theo danh mục.
- Chuyển tiền nội bộ giữa các ví.
- Xem lịch sử giao dịch qua lịch tương tác.
- Phân tích dữ liệu bằng biểu đồ trực quan.

