ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN

**KHOA KHOA HỌC MÁY TÍNH**



**BÀI TẬP LỚN HỌC PHẦN**

**PHÁT TRIỂN ỨNG DỤNG**

**DI ĐỘNG ĐA NỀN TẢNG**

ĐỀ TÀI:

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ TÀI CHÍNH**

**Sinh viên thực hiện** : Nguyễn Xuân Trường – 21IT247

Nguyễn Quốc An – 21IT325

**Giảng viên hướng dẫn** : TS. Hồ Văn Phi

**Đà Nẵng, tháng 3 năm 2025**

**MỤC LỤC**

Trang

[CHƯƠNG I. LÝ DO CHỌN ĐỀ TÀI 4](#_Toc192965348)

[CHƯƠNG II. THIẾT KẾ GIAO DIỆN 5](#_Toc192965349)

[CHƯƠNG III. THIẾT KẾ CƠ SỞ DỮ LIỆU 18](#_Toc192965350)

[CHƯƠNG IV. CHƯƠNG TRÌNH DEMO 21](#_Toc192965351)

[4.1 Trang đăng nhập 21](#_Toc192965352)

[4.2 Trang đăng ký 22](#_Toc192965353)

[4.3 Trang chính 22](#_Toc192965354)

[4.4 Trang card 23](#_Toc192965355)

[4.5 Trang thống kê 23](#_Toc192965356)

[4.6 Trang cài đặt 24](#_Toc192965357)

[4.7 Trang chuyển tiền 24](#_Toc192965358)

[TÀI LIỆU THAM KHẢO 28](#_Toc192965359)

**DANH MỤC HÌNH ẢNH**

[Hình 4. 2 Cơ sở dữ liệu 18](#_Toc192965169)

[Hình 4. 9 Trang đăng nhập 21](#_Toc192965170)

[Hình 4. 10 Trang đăng ký 22](#_Toc192965171)

[Hình 4. 11 Trang chính 22](#_Toc192965172)

[Hình 4. 12 Trang card 23](#_Toc192965173)

[Hình 4. 13 Trang thống kê 23](#_Toc192965174)

[Hình 4. 14 Trang cài đặt 24](#_Toc192965175)

[Hình 4. 15 Trang chuyển tiền 24](#_Toc192965176)

# CHƯƠNG I. LÝ DO CHỌN ĐỀ TÀI

1. **Nhu cầu thực tế cao**: Trong xã hội hiện đại, việc quản lý tài chính cá nhân và doanh nghiệp là nhu cầu thiết yếu. Nhiều người gặp khó khăn trong việc theo dõi thu chi, lập ngân sách và đạt được mục tiêu tài chính.
2. **Ứng dụng rộng rãi**: Một ứng dụng quản lý tài chính có thể phục vụ nhiều đối tượng khác nhau, từ sinh viên, người đi làm đến doanh nghiệp nhỏ và hộ gia đình.
3. **Xu hướng số hóa tài chính**: Việc chuyển đổi từ quản lý tài chính truyền thống sang các giải pháp số đang phát triển mạnh, đặc biệt sau đại dịch Covid-19.
4. **Cơ hội học tập đa lĩnh vực**: Phát triển ứng dụng này giúp bạn học hỏi về nhiều lĩnh vực khác nhau: lập trình, thiết kế UI/UX, bảo mật dữ liệu, và kiến thức tài chính cơ bản.
5. **Tiềm năng phát triển**: Từ một ứng dụng cơ bản, bạn có thể mở rộng với nhiều tính năng nâng cao như phân tích dữ liệu, dự báo tài chính, kết nối với ngân hàng, hoặc tích hợp trí tuệ nhân tạo để đưa ra gợi ý tài chính.
6. **Giải quyết vấn đề thực tế**: Ứng dụng này giúp người dùng giải quyết các vấn đề như: quản lý chi tiêu quá mức, lập kế hoạch tiết kiệm, theo dõi nợ, và đưa ra quyết định tài chính thông minh hơn.
7. **Có tính bền vững**: Nhu cầu quản lý tài chính luôn tồn tại và ngày càng phức tạp, đảm bảo tính bền vững lâu dài cho dự án của bạn.
8. **Thị trường tiềm năng**: Mặc dù đã có nhiều ứng dụng quản lý tài chính, vẫn luôn có không gian cho các giải pháp mới mang tính cách tân hoặc nhắm vào các phân khúc thị trường cụ thể.
9. **Khả năng thương mại hóa**: Nếu phát triển thành công, ứng dụng có tiềm năng trở thành sản phẩm thương mại thông qua các mô hình như freemium, đăng ký theo gói, hoặc thu phí tính năng cao cấp.

# CHƯƠNG II. THIẾT KẾ GIAO DIỆN

|  |  |
| --- | --- |
| Screen | Screen design |
| **Sign In** |  |
| **Sign Up** |  |
| **Home** |  |
| **Card** |  |
| **Statics** |  |
| **Statics** |  |
| **Setting** |  |
| **My profile** |  |
| **Change password** |  |
| **sent** |  |
| **Loan** |  |
| **Deposit savings** |  |

# CHƯƠNG III. THIẾT KẾ CƠ SỞ DỮ LIỆU

A computer screen shot of a computer code

AI-generated content may be incorrect.

Hình 3. 1 Cơ sở dữ liệu

1. Bảng accounts (Tài khoản người dùng)

Chứa thông tin của người dùng trong hệ thống.

Cấu trúc cột:

* id serial PRIMARY KEY: ID tự động tăng, dùng làm khóa chính.
* username TEXT NOT NULL UNIQUE: Tên đăng nhập, phải là duy nhất.
* passwd TEXT NOT NULL: Mật khẩu của người dùng (nên được mã hóa).
* phone TEXT: Số điện thoại của người dùng.
* address TEXT: Địa chỉ của người dùng.
* time\_created TIMESTAMP DEFAULT NOW(): Thời điểm tài khoản được tạo.
* time\_updated TIMESTAMP DEFAULT NOW(): Thời điểm tài khoản được cập nhật lần cuối.
* creator INT DEFAULT 0: ID của người tạo tài khoản.
* updater INT: ID của người cập nhật tài khoản.

2. Bảng cards (Thông tin thẻ)

Lưu trữ thông tin về thẻ thanh toán liên kết với tài khoản người dùng.

Cấu trúc cột:

* id serial PRIMARY KEY: ID duy nhất của thẻ.
* id\_account INT NOT NULL: ID của tài khoản liên kết với thẻ (khóa ngoại từ accounts).
* card\_number TEXT NOT NULL UNIQUE: Số thẻ duy nhất.
* private\_key TEXT NOT NULL: Khóa riêng tư của thẻ (nên được mã hóa).
* pin TEXT: Mã PIN của thẻ.
* cvv TEXT NOT NULL: Mã CVV (bảo mật) của thẻ.
* total\_amount NUMERIC(15, 2) DEFAULT 0.00: Tổng số tiền hiện có trong thẻ.
* time\_created TIMESTAMP DEFAULT NOW(): Thời điểm thẻ được tạo.
* time\_updated TIMESTAMP DEFAULT NOW(): Thời điểm thẻ được cập nhật lần cuối.
* expiration\_date TIMESTAMP DEFAULT (CURRENT\_TIMESTAMP + INTERVAL '5 years'): Ngày hết hạn của thẻ (5 năm kể từ thời điểm tạo).
* Ràng buộc:
  + CONSTRAINT fk\_account FOREIGN KEY (id\_account) REFERENCES accounts (id) ON DELETE CASCADE: Khi tài khoản bị xóa, tất cả các thẻ liên kết cũng bị xóa.

3. Bảng savings (Tiết kiệm)

Lưu trữ thông tin về các khoản tiết kiệm của người dùng.

Cấu trúc cột:

* id serial PRIMARY KEY: ID duy nhất cho mỗi khoản tiết kiệm.
* id\_account INT NOT NULL: ID của tài khoản sở hữu khoản tiết kiệm.
* goal\_name TEXT NOT NULL: Tên mục tiêu tiết kiệm.
* goal\_amount NUMERIC(15, 2) NOT NULL: Số tiền mục tiêu cần tiết kiệm.
* current\_amount NUMERIC(15, 2) DEFAULT 0.00: Số tiền hiện có trong khoản tiết kiệm.
* deadline TIMESTAMP: Thời hạn cần đạt mục tiêu tiết kiệm.
* Ràng buộc:
  + CONSTRAINT fk\_account FOREIGN KEY (id\_account) REFERENCES accounts (id) ON DELETE CASCADE: Khi tài khoản bị xóa, các khoản tiết kiệm cũng bị xóa.

4. Bảng notifications (Thông báo)

Chứa các thông báo gửi đến người dùng.

Cấu trúc cột:

* id serial PRIMARY KEY: ID duy nhất của thông báo.
* id\_account INT NOT NULL: ID của tài khoản nhận thông báo.
* messages TEXT NOT NULL: Nội dung thông báo.
* sent\_at TIMESTAMP DEFAULT NOW(): Thời điểm gửi thông báo.
* Ràng buộc:
  + CONSTRAINT fk\_account FOREIGN KEY (id\_account) REFERENCES accounts (id) ON DELETE CASCADE: Khi tài khoản bị xóa, thông báo liên quan cũng bị xóa.

5. Bảng categories (Danh mục giao dịch)

Lưu trữ danh mục giao dịch (ví dụ: ăn uống, mua sắm, tiết kiệm, v.v.).

Cấu trúc cột:

* id serial PRIMARY KEY: ID duy nhất của danh mục.
* type\_category INT NOT NULL CHECK (type\_category IN (0, 1)): Loại danh mục (0: thu nhập, 1: chi tiêu).
* name\_category TEXT NOT NULL: Tên danh mục.
* icon TEXT: Biểu tượng của danh mục.
* note TEXT: Ghi chú cho danh mục.
* time\_created TIMESTAMP DEFAULT NOW(): Thời điểm danh mục được tạo.
* time\_updated TIMESTAMP DEFAULT NOW(): Thời điểm danh mục được cập nhật.

6. Bảng transactions (Giao dịch)

Lưu trữ các giao dịch tài chính của người dùng.

Cấu trúc cột:

* id serial PRIMARY KEY: ID duy nhất của giao dịch.
* type\_transaction INT NOT NULL CHECK (type\_transaction IN (0,1)): Loại giao dịch (0: chi tiêu, 1: thu nhập).
* transaction\_hash VARCHAR(256) NOT NULL UNIQUE: Mã giao dịch duy nhất (có thể dùng blockchain hash).
* account\_receiver INT NOT NULL: ID tài khoản nhận tiền.
* name\_receiver VARCHAR(255) NOT NULL: Tên người nhận.
* sender\_id INT NOT NULL: ID tài khoản gửi tiền.
* sender\_name VARCHAR(255) NOT NULL: Tên người gửi.
* amount NUMERIC(10, 2) NOT NULL CHECK (amount >= 0): Số tiền giao dịch (phải >= 0).
* messages TEXT: Ghi chú hoặc mô tả giao dịch.
* timestamps TIMESTAMPTZ DEFAULT NOW(): Thời điểm thực hiện giao dịch.
* category\_id INT: ID danh mục giao dịch (liên kết với bảng categories).
* card\_id INT NOT NULL: ID của thẻ thực hiện giao dịch.
* Ràng buộc:
  + CONSTRAINT fk\_category FOREIGN KEY (category\_id) REFERENCES categories (id): Liên kết với danh mục giao dịch.
  + CONSTRAINT fk\_card FOREIGN KEY (card\_id) REFERENCES cards (id): Liên kết với thẻ thực hiện giao dịch.

# CHƯƠNG IV. CHƯƠNG TRÌNH DEMO

### 4.1 Trang đăng nhập

A screenshot of a login screen

AI-generated content may be incorrect.

Hình 4. 1 Trang đăng nhập

Người dùng nhập số điện thoại và mật khảu để đăng nhập, nếu chưa có tài khoản thì ấn vào phần “Sign Up” để tạo tài khoản mới.

### 4.2 Trang đăng ký

A screenshot of a phone number and address

AI-generated content may be incorrect.

Hình 4. 2 Trang đăng ký

Người dùng nhập các thông tin như tên đầy đủ, số điện thoại, địa chỉ và mật khẩu và ấn “Sign Up” để tạo tài khoản, sau khi đăng ký tài khoản thành công, người dùng được chuyển về trang “Sign In”.

### 4.3 Trang chính

A screenshot of a phone

AI-generated content may be incorrect.

Hình 4. 3 Trang chính

Sau khi đăng nhập thành công, người dùng được đưa đến trang chính, nơi hiển thị các thông tin cơ bản như: số dư, số tài khoản, mã cvv, thời gian hết hạn của thẻ và các giao dịch đã thực hiện.

### 4.4 Trang card

A screenshot of a mobile phone

AI-generated content may be incorrect.

Hình 4. 4 Trang card

Tại trang card, người dùng xem các thông tin của thẻ và có thêm tính năng giới hạn số tiền chi tiêu.

### 4.5 Trang thống kê

A screenshot of a phone

AI-generated content may be incorrect.

Hình 4. 5 Trang thống kê

Người dùng sẽ xem được thống kê giao dịch theo danh mục có sẵn được đồ thị hóa, giúp kiểm soát dễ hơn chi tiêu.

### 4.6 Trang cài đặt

A screenshot of a phone

AI-generated content may be incorrect.

Hình 4. 6 Trang cài đặt

Người dùng có thể xem thông tin cá nhân, liên hệ với doanh nghiệp, thay đổi mật khẩu tài khoản, xem điều khoản riêng tư.

### 4.7 Trang chuyển tiền

A screenshot of a phone

AI-generated content may be incorrect.

Hình 4. 7 Trang chuyển tiền

Người dùng nhập các thông liên quan như: địa chỉ thẻ người nhận, hệ thống tự động lấy tên của người nhận, nội dung chuyển khoản, danh mục, số tiền cần chuyện, sau đó hệ thống kiểm tra số dư và thực hiện giao dịch.

**A screenshot of a cell phone

AI-generated content may be incorrect.**

Hình 4. 8 Saving Accounts

A screenshot of a mobile banking account

AI-generated content may be incorrect.

Hình 4. 9 Deposit savings

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

1. **Những mặt chưa làm được**

Mặc dù dự án đã đạt được nhiều kết quả quan trọng, vẫn còn một số hạn chế cần được cải thiện:

Hiệu suất và tối ưu hóa

Ứng dụng chưa được tối ưu hóa hoàn toàn cho các thiết bị có cấu hình thấp.

Thời gian phản hồi của một số truy vấn blockchain vẫn còn chậm.

Tích hợp Blockchain

Chưa triển khai đầy đủ các chức năng nâng cao như Smart Contract phức tạp.

Chưa thử nghiệm hệ thống với một lượng lớn giao dịch thực tế.

Bảo mật dữ liệu

Chưa có cơ chế mã hóa mạnh mẽ cho dữ liệu lưu trữ trong PostgreSQL.

Cần cải thiện khả năng chống tấn công DDoS và các lỗ hổng bảo mật khác.

Trải nghiệm người dùng (UI/UX)

Giao diện người dùng chưa được tối ưu hoàn toàn, cần cải thiện về mặt thiết kế và tính trực quan.

Một số chức năng chưa thân thiện với người dùng và cần điều chỉnh.

1. **Những mặt đã làm được**

Dự án đã đạt được nhiều kết quả quan trọng, bao gồm:

Xây dựng ứng dụng tài chính cá nhân với Flutter

Hoàn thành giao diện cơ bản với Flutter.

Đã triển khai một số chức năng như ghi chép thu/chi, phân tích tài chính.

Tích hợp Blockchain vào hệ thống quản lý tài chính

Hoàn thành Smart Contract cơ bản trên nền tảng Ethereum sử dụng Hardhat.

Lưu trữ các giao dịch tài chính trên Blockchain để đảm bảo tính minh bạch.

Phát triển cơ sở dữ liệu với PostgreSQL

Xây dựng hệ thống lưu trữ dữ liệu tài chính.

Kết nối cơ sở dữ liệu với ứng dụng để lưu và truy xuất thông tin hiệu quả.

Xây dựng nền tảng bảo mật cơ bản

Triển khai xác thực người dùng.

Đảm bảo dữ liệu không thể bị thay đổi một cách trái phép nhờ cơ chế Blockchain.

1. **Hướng phát triển**

Trong tương lai, có thể mở rộng và nâng cấp dự án theo các hướng sau:

Tối ưu hiệu suất và mở rộng quy mô

Cải thiện tốc độ xử lý giao dịch bằng cách tối ưu hóa thuật toán và sử dụng các giải pháp Blockchain Layer 2.

Hỗ trợ nhiều loại tiền tệ và tích hợp API ngân hàng để tự động cập nhật giao dịch.

Nâng cao bảo mật

Ứng dụng mã hóa dữ liệu cấp độ trường (Field-level Encryption) trong PostgreSQL.

Nâng cấp cơ chế xác thực bằng sinh trắc học hoặc xác thực hai yếu tố (2FA).

Phát triển tính năng AI hỗ trợ tài chính

Tích hợp Machine Learning để phân tích mô hình chi tiêu và đề xuất tiết kiệm hiệu quả.

Phát triển chatbot hỗ trợ tư vấn tài chính tự động.

Hỗ trợ đa nền tảng

Mở rộng ứng dụng lên nền tảng web và desktop để tiếp cận nhiều người dùng hơn.

Hỗ trợ ứng dụng trên iOS với hiệu suất tối ưu.

Để hoàn thành đề tài này, một lần nữa chúng em xin chân thành cảm ơn thầy giáo TS. Hồ Văn Phi đã quan tâm, giúp đỡ chúng em trong suốt thời gian qua để chúng em hoàn thành được đồ án môn học. Chúng em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

[1] *Flutter documentation*. (n.d.). <https://docs.flutter.dev/>

[2] *Getting started with Hardhat | Ethereum development environment for professionals by Nomic Foundation*. (n.d.). https://hardhat.org/hardhat-runner/docs/getting-started#overview

[3] *PostgreSQL: documentation*. (n.d.). The PostgreSQL Global Development Group. https://www.postgresql.org/docs/

[4] *Solidity — Solidity 0.8.29 documentation*. (n.d.). https://docs.soliditylang.org/en/latest/