

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Phát triển Ứng dụng trên Thiết bị Di động (CO3043)

Assignment 3

Ứng dụng quản lý chi tiêu Monney

Advisor(s): Hoàng Lê Hải Thanh

Student(s): Trần Quốc Thắng 2110551

Trần Hoàng Đại Sơn 2110509

Lưu Chấn Hưng 2111401

Nguyễn Trần Quốc Uy 1915866

HO CHI MINH CITY, JUNE 2024



Contents

1	Giới thiệu	5
1.1	Vấn đề cần giải quyết	5
1.2	Mục tiêu	5
1.3	Mục đích	5
2	Phân tích thị trường, Phương pháp tiếp cận và Phương pháp luận	6
2.1	Phân tích thị trường	6
2.1.1	Xác định đối tượng người dùng	6
2.1.2	Nghiên cứu thị trường	7
2.2	Phương pháp tiếp cận	8
2.2.1	Tối giản (Minimalism):	8
2.2.2	Tập trung vào người dùng (User-centric):	9
2.3	Phương pháp luận	9
3	Tính năng và chức năng của ứng dụng	10
4	Chi tiết hiện thực ứng dụng	12
4.1	Ngôn ngữ lập trình và Framework	12
4.2	Cơ sở dữ liệu	13
4.3	Framework và thư viện	13
4.4	Môi trường phát triển	13
4.5	Công cụ	14
5	Trải nghiệm người dùng	15
5.1	Đăng nhập / Đăng ký	15
5.2	Xem thống kê, báo cáo	16
5.3	Quản lý giao dịch	17
5.4	Sử dụng Chatbot	18
5.5	Thay đổi gói đăng ký	19
5.6	Gửi phản hồi	20
6	Kiến trúc hệ thống	21
7	Thiết kế Web Service	21
7.1	Kiến trúc	21
7.2	Các thành phần	22



7.3	Cơ sở dữ liệu	22
7.4	Công nghệ sử dụng	22
7.5	Chi tiết về các API	23
8	Triển khai ứng dụng với người dùng cuối	24
9	Khảo sát người dùng	25
9.1	Khảo sát thông tin chung của đối tượng khảo sát	25
9.2	Khảo sát trải nghiệm ứng dụng	26
9.3	Đánh giá chung của người dùng	28
10	Đánh giá mức độ thực hiện đề tài, các vấn đề cần khắc phục	28



1 Giới thiệu

Mooney là một ứng dụng di động được thiết kế để giúp người dùng quản lý tài chính cá nhân một cách hiệu quả thông qua việc theo dõi và quản lý các khoản chi tiêu mỗi ngày.

1.1 Vấn đề cần giải quyết

Trong cuộc sống hiện đại, việc quản lý tài chính cá nhân một cách hiệu quả là một thách thức lớn đối với nhiều người. Việc theo dõi các khoản thu chi hằng ngày, ghi chép các giao dịch và lập kế hoạch chi tiêu cần nhiều thời gian và công sức.

Bên cạnh đó, đối với những người chưa có nhiều kinh nghiệm trong việc lập kế hoạch chi tiêu, việc đưa ra một kế hoạch chi tiêu trong tháng không hề đơn giản. Một số người sẽ gặp khó khăn trong việc duy trì tính kỷ luật và dễ bị rơi vào tình trạng chi tiêu không kiểm soát.

1.2 Mục tiêu

Ứng dụng quản lý thu chi cá nhân của chúng tôi Mooney ra đời nhằm giúp người dùng giải quyết các vấn đề trên bằng cách cung cấp một công cụ hiệu quả và tiện lợi để theo dõi và quản lý tài chính cá nhân. Mục tiêu của ứng dụng là giúp người dùng tiết kiệm thời gian, nâng cao khả năng kiểm soát tài chính và xây dựng thói quen chi tiêu hợp lý.

1.3 Mục đích

Mooney được thiết kế với mục đích giúp đỡ người dùng trong việc quản lý tài chính cá nhân. Bên cạnh những tính năng quen thuộc, Mooney còn có một số tính năng nổi bật như:

- Diễn đàn trao đổi kinh nghiệm: Diễn đàn cho phép người dùng đặt câu hỏi và chia sẻ các phương pháp quản lý thu chi, học hỏi từ cộng đồng và tìm ra những chiến lược tài chính phù hợp nhất cho bản thân.
- Chatbot hỗ trợ người dùng: Người dùng có thể sử dụng Chatbot để giải đáp các câu hỏi liên quan đến quản lý tài chính, cung cấp các mẹo và hướng dẫn chi tiêu, giúp người dùng tối ưu hóa tài chính cá nhân một cách dễ dàng.

- Tự động thêm các giao dịch chỉ bằng chụp các bill thanh toán: Tính năng này giúp tiết kiệm thời gian cho người dùng và thêm các giao dịch vào danh sách chi tiêu một cách chính xác, nhanh chóng.

Với Những tính năng mới mẻ này, nhóm em hi vọng Mooney sẽ trở thành một trợ thủ tiềm năng và đáng tin cậy trong việc quản lý tài chính.

2 Phân tích thị trường, Phương pháp tiếp cận và Phương pháp luận

2.1 Phân tích thị trường

2.1.1 Xác định đối tượng người dùng

1. **Giới trẻ (18-25 tuổi):** Sinh viên, người mới ra trường, bắt đầu tự lập tài chính, chưa có nhiều kinh nghiệm trong việc quản lý chi tiêu.
 - **Nhu cầu:** Theo dõi chi tiêu cơ bản, lập kế hoạch ngân sách đơn giản, tiết kiệm cho mục tiêu ngắn hạn.
 - **Mong muốn:** Giao diện ứng dụng trẻ trung, dễ sử dụng, tích hợp các tính năng nhắc nhở, cảnh báo chi tiêu vượt mức.
2. **Người đi làm (25-40 tuổi):** Đã có thu nhập ổn định, nhu cầu quản lý chi tiêu phức tạp hơn. Có thể đã lập gia đình, có trách nhiệm tài chính với bản thân và gia đình.
 - **Nhu cầu:** Theo dõi thu chi chi tiết, phân loại theo hạng mục, lập kế hoạch ngân sách chi tiết, quản lý nợ, đầu tư.
 - **Mong muốn:** Giao diện trực quan, dễ theo dõi, báo cáo trực quan, đồng bộ dữ liệu trên nhiều thiết bị, kết nối với tài khoản ngân hàng.
3. **Nhóm hộ gia đình:** Cần một công cụ chung để quản lý chi tiêu của cả gia đình.
 - **Nhu cầu:** Phân quyền truy cập cho các thành viên, theo dõi tổng hợp thu chi, lập kế hoạch ngân sách chung, quản lý các khoản vay/cho vay trong gia đình.
 - **Mong muốn:** Dễ sử dụng cho mọi lứa tuổi, đồng bộ dữ liệu theo thời gian thực, đảm bảo tính bảo mật thông tin.

Bằng cách hướng đến đối tượng người dùng đa dạng, ứng dụng quản lý chi tiêu của chúng em mong muốn mang đến giải pháp tài chính cá nhân phù hợp với nhu cầu và đặc điểm của từng nhóm đối tượng.

2.1.2 Nghiên cứu thị trường

Phân tích SWOT của các ứng dụng quản lý chi tiêu phổ biến trên thị trường:

Ứng dụng	Điểm mạnh	Điểm yếu	Cơ hội	Thách thức
Goodbudget	<ul style="list-style-type: none"> - Phương pháp quản lý ngân sách hiệu quả - Hỗ trợ cộng đồng mạnh mẽ - Giao diện thân thiện với người dùng 	<ul style="list-style-type: none"> - Hạn chế tự động hóa - Báo cáo cơ bản - Tích hợp với các tổ chức tài chính 	<ul style="list-style-type: none"> - Tự động hóa, cộng tác và tích hợp - Mở rộng công cụ báo cáo 	<ul style="list-style-type: none"> - Cạnh tranh gay gắt, người dùng ưu tiên giải pháp tự động và tiện lợi
Spendee	<ul style="list-style-type: none"> - Giao diện hấp dẫn - Cung cấp các tính năng sáng tạo 	<ul style="list-style-type: none"> - Thiếu các tính năng mong muốn - Thời gian phản hồi chậm - Phụ thuộc vào Internet - Đồng bộ hóa dữ liệu 	<ul style="list-style-type: none"> - Tự động hóa, cộng tác và tích hợp - Hướng dẫn người dùng 	<ul style="list-style-type: none"> - Cạnh tranh gay gắt, đánh giá không tốt từ người dùng
YouNeedABudget	<ul style="list-style-type: none"> - Tiếp cận mục tiêu, theo dõi thời gian thực, tài nguyên giáo dục - Tích hợp với các tổ chức tài chính 	<ul style="list-style-type: none"> - Mô hình đăng ký, độ phức tạp ban đầu - Tính năng hạn chế cho người dùng quốc tế 	<ul style="list-style-type: none"> - Mở rộng thị trường, tích hợp công nghệ mới nổi, hợp tác với tổ chức tài chính - Đa dạng hóa dịch vụ 	<ul style="list-style-type: none"> - Cạnh tranh gay gắt, đánh giá không tốt từ người dùng
PocketGuard	<ul style="list-style-type: none"> - Tiện lợi, tích hợp nhiều tài khoản, phân loại tự động 	<ul style="list-style-type: none"> - Giới hạn tính năng, yêu cầu kết nối tài khoản 	<ul style="list-style-type: none"> - Mở rộng tính năng, mở rộng thị trường 	<ul style="list-style-type: none"> - Cạnh tranh từ các đối thủ, vấn đề bảo mật và quyền riêng tư

Table 2.1: Phân tích SWOT của các ứng dụng quản lý chi tiêu

Kết luận:

Thị trường ứng dụng quản lý chi tiêu rất cạnh tranh với nhiều ứng dụng mạnh. Để thành công, ứng dụng cần:

- **Cung cấp trải nghiệm người dùng mượt mà:** Giao diện thân thiện, dễ sử dụng, tốc độ xử lý nhanh.
- **Tính năng đa dạng và hữu ích:** Đáp ứng nhu cầu đa dạng của người dùng như theo dõi chi tiêu, lập ngân sách, quản lý nợ, đầu tư,...
- **Tự động hóa:** Tự động đồng bộ dữ liệu từ tài khoản ngân hàng, phân loại giao dịch, gửi thông báo nhắc nhở,...
- **Bảo mật và riêng tư:** Đảm bảo an toàn thông tin tài chính của người dùng.
- **Hỗ trợ người dùng tốt:** Cung cấp tài liệu hướng dẫn chi tiết, giải đáp thắc mắc nhanh chóng.

2.2 Phương pháp tiếp cận

Nhóm chúng em quyết định tiếp cận việc phát triển ứng dụng quản lý chi tiêu dựa trên hai yếu tố chính: Sự tối giản (Minimalism) và Tập trung vào người dùng (User-centric).

2.2.1 Tối giản (Minimalism):

Thay vì cố gắng tích hợp quá nhiều tính năng phức tạp, chúng em ưu tiên tạo ra một ứng dụng **đơn giản, dễ sử dụng** ngay cả với những người dùng ít kinh nghiệm công nghệ.

- Giao diện ứng dụng được thiết kế trực quan, dễ nhìn, loại bỏ các chi tiết thừa.
- Chức năng được tối ưu hóa, tập trung vào các tác vụ quản lý chi tiêu thiết yếu nhất.

Mục tiêu của việc áp dụng phương pháp tối giản là giúp người dùng **dễ dàng làm quen và sử dụng ứng dụng hiệu quả** ngay từ lần đầu tiên.



2.2.2 Tập trung vào người dùng (User-centric):

Trong suốt quá trình phát triển, chúng em luôn đặt **trải nghiệm người dùng** lên hàng đầu.

- Phân tích kỹ lưỡng nhu cầu, thói quen quản lý tài chính của người dùng mục tiêu.
- Ưu tiên phát triển các tính năng thực sự hữu ích, giải quyết các vấn đề cụ thể của người dùng.

Bằng cách tập trung vào người dùng, chúng em mong muốn tạo ra một ứng dụng **thực sự hữu ích và thiết thực**, giúp người dùng dễ dàng kiểm soát chi tiêu, đạt được mục tiêu tài chính của mình.

Tóm lại: Phương pháp tiếp cận của chúng em là kết hợp giữa **tính tối giản và sự tập trung vào người dùng**, nhằm mang đến một giải pháp quản lý chi tiêu **hiệu quả, dễ sử dụng và phù hợp với nhu cầu thực tế** của người dùng.

2.3 Phương pháp luận

Để phát triển ứng dụng quản lý chi tiêu một cách hiệu quả, nhóm chúng em lựa chọn kết hợp hai phương pháp phổ biến trong phát triển phần mềm hiện đại: Agile và Kanban, với sự hỗ trợ đắc lực từ công cụ quản lý dự án Jira.

1. Agile

Chúng em áp dụng mô hình Agile với mục tiêu tạo ra một quy trình phát triển linh hoạt, thích ứng nhanh với các thay đổi và tập trung tối đa vào giá trị mang lại cho người dùng.

- **Sprint:** Dự án được chia thành các chu kỳ phát triển ngắn (sprint) với thời lượng 2 tuần/sprint. Mỗi sprint tập trung vào việc hoàn thành một số tính năng ưu tiên đã được xác định trước.
- **Họp Scrum:** Nhóm thực hiện họp Scrum hàng ngày để cập nhật tiến độ, thảo luận các vấn đề gặp phải và điều chỉnh kế hoạch (nếu cần) cho ngày tiếp theo.
- **Demo & Retrospective:** Sau mỗi sprint, nhóm tổ chức buổi demo sản phẩm với nhau để nhận xét và cải tiến sản phẩm. Đồng thời, nhóm cũng thực hiện buổi họp retrospective để rút kinh nghiệm cho sprint tiếp theo.

2. Kanban

Kanban được sử dụng song song với Agile để trực quan hóa quy trình làm việc, theo dõi tiến độ dự án và quản lý công việc hiệu quả.

- **Kanban board:** Nhóm sử dụng Jira để tạo Kanban board với các cột đại diện cho các trạng thái của công việc (To Do, In Progress, Done).
- **Thẻ Kanban:** Mỗi công việc (task) được thể hiện bằng một thẻ (card) trên bảng Kanban, chứa thông tin chi tiết về task đó (mô tả, người thực hiện, deadline,...).
- **WIP Limit:** Nhóm đặt giới hạn số lượng công việc “đang thực hiện” (Work In Progress limit) cho mỗi cột trên bảng Kanban để tránh lãng phí tài nguyên và tập trung hoàn thành công việc nhanh chóng.

3. Jira

Jira đóng vai trò là công cụ trung tâm hỗ trợ nhóm quản lý dự án theo phương pháp Agile và Kanban.

- **Quản lý dự án:** Tạo và quản lý dự án, sprint, backlog, issue types.
- **Quản lý công việc:** Tạo và quản lý task, subtask, phân công công việc cho thành viên, theo dõi tiến độ.
- **Báo cáo:** Tạo các báo cáo trực quan về tiến độ dự án, hiệu suất làm việc của nhóm.

Bằng cách kết hợp nhuần nhuyễn Agile, Kanban và Jira, nhóm chúng em đã tạo ra một quy trình phát triển phần mềm linh hoạt, hiệu quả và minh bạch, góp phần quan trọng vào việc hoàn thành ứng dụng quản lý chỉ tiêu đúng tiến độ và đáp ứng tốt các yêu cầu đề ra.

3 Tính năng và chức năng của ứng dụng

Từ những phân tích và nghiên cứu thực tiễn, bằng việc áp dụng các cách tiếp cận đã nêu, nhóm đề xuất các tính năng chính của ứng dụng:

1. Quản lý giao dịch

- **Đối tượng sử dụng chính:** Tính năng chính của ứng dụng, được sử dụng bởi toàn bộ user

- **Vấn đề mà tính năng giải quyết:**

- Giúp người dùng theo dõi và quản lý chi tiêu hàng ngày
- Tạo báo cáo tài chính cá nhân hoặc doanh nghiệp một cách tự động và chính xác.
- Giúp nhận diện xu hướng chi tiêu và tối ưu hóa ngân sách.

- **Mô tả cụ thể:**

Hệ thống giúp người dùng phân loại các giao dịch vào các danh mục như ăn uống, mua sắm, hóa đơn, giải trí... giúp người dùng dễ dàng theo dõi và phân tích chi tiêu. Tạo báo cáo chi tiêu hàng tuần, hàng tháng hoặc hàng năm dưới dạng biểu đồ, giúp người dùng dễ dàng nhận diện các xu hướng chi tiêu của mình. Ngoài ra ứng dụng cũng cung cấp lịch sử giao dịch chi tiết để người dùng có thể kiểm tra lại bất cứ lúc nào.

2. Chatbot hỗ trợ

- **Đối tượng sử dụng chính:** Người dùng cần giải đáp thắc mắc nhanh chóng, không có thời gian hoặc không muốn tương tác trực tiếp với nhân viên hỗ trợ.

- **Vấn đề mà tính năng giải quyết:**

- Cung cấp hỗ trợ khách hàng tự động, nhanh chóng và hiệu quả.
- Giúp người dùng tìm kiếm thông tin và giải đáp các thắc mắc liên quan đến ứng dụng mà không cần chờ đợi lâu.

- **Mô tả cụ thể:**

Sử dụng trí tuệ nhân tạo để hiểu và trả lời các câu hỏi thường gặp về tính năng, cách sử dụng và giải quyết các sự cố phổ biến, hoạt động liên tục, sẵn sàng giải đáp các câu hỏi của người dùng bất kể thời gian nào. Chatbot còn có khả năng nhận diện người dùng và cung cấp các gợi ý hoặc lời khuyên tài chính cá nhân hóa dựa trên lịch sử giao dịch và thông tin tài chính của người dùng.

3. Nhận diện hóa đơn qua hình ảnh

- **Đối tượng sử dụng chính:** Người dùng muốn tiết kiệm thời gian nhập liệu thủ công, bao gồm cá nhân và doanh nghiệp có nhiều hóa đơn cần quản lý.

- **Vấn đề mà tính năng giải quyết:**

- Giảm thiểu thời gian và công sức nhập liệu thủ công

- Tránh sai sót khi nhập dữ liệu từ hóa đơn vào hệ thống

- **Mô tả cụ thể:**

Người dùng chỉ cần chụp ảnh hóa đơn bằng điện thoại thông minh và tải lên ứng dụng. Ứng dụng sau đó sẽ sử dụng công nghệ OCR để tự động nhận diện và trích xuất thông tin từ hóa đơn, bao gồm số tiền, ngày tháng, nhà cung cấp và các mục chi tiêu. Cuối cùng, thông tin từ hóa đơn được tự động ghi nhận vào hệ thống quản lý giao dịch, giúp cập nhật tình hình tài chính của người dùng một cách chính xác và kịp thời

4. Forum chia sẻ kiến thức

- **Đối tượng sử dụng chính:** Người dùng muốn tiết kiệm thời gian nhập liệu thủ công, bao gồm cá nhân và doanh nghiệp có nhiều hóa đơn cần quản lý.
- **Vấn đề mà tính năng giải quyết:**
 - Giảm thiểu thời gian và công sức nhập liệu thủ công
 - Tránh sai sót khi nhập dữ liệu từ hóa đơn vào hệ thống

- **Mô tả cụ thể:**

Người dùng chỉ cần chụp ảnh hóa đơn bằng điện thoại thông minh và tải lên ứng dụng. Ứng dụng sau đó sẽ sử dụng công nghệ OCR để tự động nhận diện và trích xuất thông tin từ hóa đơn, bao gồm số tiền, ngày tháng, nhà cung cấp và các mục chi tiêu. Cuối cùng, thông tin từ hóa đơn được tự động ghi nhận vào hệ thống quản lý giao dịch, giúp cập nhật tình hình tài chính của người dùng một cách chính xác và kịp thời

4 Chi tiết hiện thực ứng dụng

Nhằm xây dựng một ứng dụng quản lý chi tiêu hiệu quả và thân thiện với người dùng, nhóm chúng tôi đã lựa chọn kết hợp các công nghệ tiên tiến sau:

4.1 Ngôn ngữ lập trình và Framework

- **Frontend:** React Native + Expo

Lý do lựa chọn:



- **React Native:** Cho phép xây dựng ứng dụng di động đa nền tảng (iOS và Android) từ một codebase duy nhất, rút ngắn thời gian phát triển và đảm bảo đồng bộ trải nghiệm người dùng trên các nền tảng khác nhau.
- **Expo:** Giúp đơn giản hóa việc khởi tạo dự án, phát triển, build và deploy ứng dụng React Native, đồng thời cung cấp nhiều API và thư viện hữu ích.

- **Backend:** NodeJS + ExpressJS

Lý do lựa chọn:

- **NodeJS:** Môi trường thực thi Javascript phía server, cho phép xây dựng backend nhanh chóng, hiệu năng cao và dễ dàng mở rộng.
- **ExpressJS:** Framework tối giản và phổ biến của NodeJS, giúp xây dựng API và xử lý request từ client một cách dễ dàng và hiệu quả.

4.2 Cơ sở dữ liệu

Firestore Database (Realtime Database)

Lý do lựa chọn: Cơ sở dữ liệu NoSQL, cung cấp khả năng đồng bộ dữ liệu thời gian thực, giúp đơn giản hóa việc phát triển ứng dụng và đồng bộ dữ liệu trên nhiều thiết bị người dùng.

4.3 Framework và thư viện

Axios: Thư viện Javascript phổ biến, được sử dụng để thực hiện các request HTTP đến API từ frontend (React Native).

React Navigation: Giúp xây dựng navigation và routing cho ứng dụng React Native một cách dễ dàng.

Expo SDK: Cung cấp nhiều API built-in cho React Native, giúp đơn giản hóa việc truy cập các tính năng native của thiết bị như Camera, GPS, Notifications,...

4.4 Môi trường phát triển

Hệ điều hành: Android/iOS

Môi trường ảo: Sử dụng Expo CLI, nhóm không cần cài đặt môi trường native mà vẫn có thể phát triển và test ứng dụng React Native trên cả thiết bị thật và simulator/emulator, giúp quá trình phát triển nhanh chóng và thuận tiện hơn.



4.5 Công cụ

IDE: Visual Studio Code, mã nguồn mở, nhẹ nhàng, nhiều extensions hỗ trợ lập trình đa ngôn ngữ.

Quản lý phiên bản: Git (sử dụng GitHub) - công cụ quản lý phiên bản phổ biến, GitHub giúp lưu trữ code và cộng tác nhóm hiệu quả.

Quản lý dự án và task: Jira - theo dõi tiến độ dự án, phân công công việc, quản lý lỗi.

Thiết kế giao diện: Figma - công cụ thiết kế giao diện phổ biến, dễ sử dụng, hỗ trợ làm việc cộng tác.

Ghi chép và chia sẻ tài liệu: Confluence - Nền tảng tạo ghi chú, tài liệu trực tuyến, hỗ trợ làm việc nhóm hiệu quả.

5 Trải nghiệm người dùng

5.1 Đăng nhập / Đăng ký

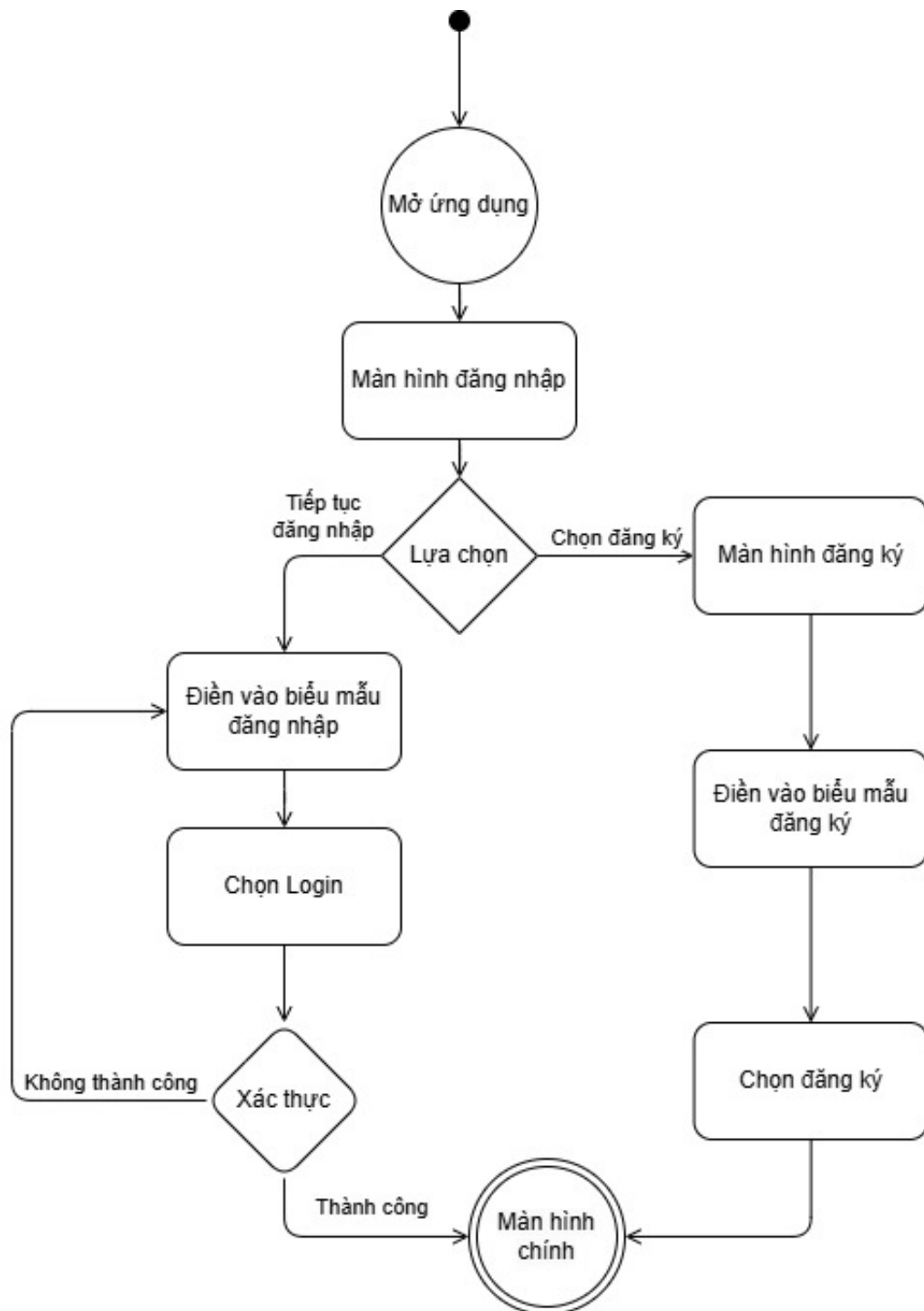


Figure 5.1: User flow Đăng nhập / Đăng ký

5.2 Xem thống kê, báo cáo

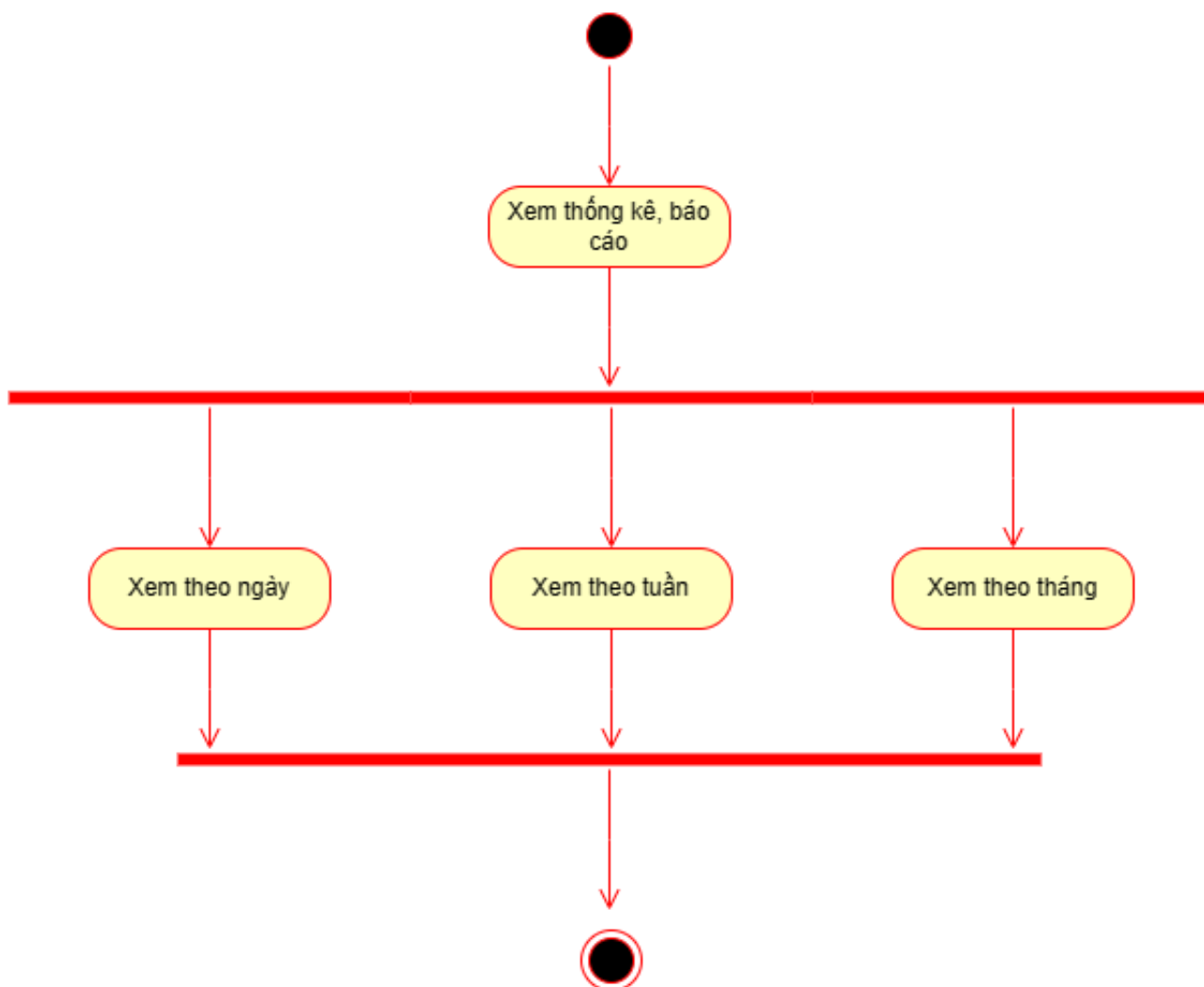


Figure 5.2: User flow Xem thống kê, báo cáo

5.3 Quản lý giao dịch

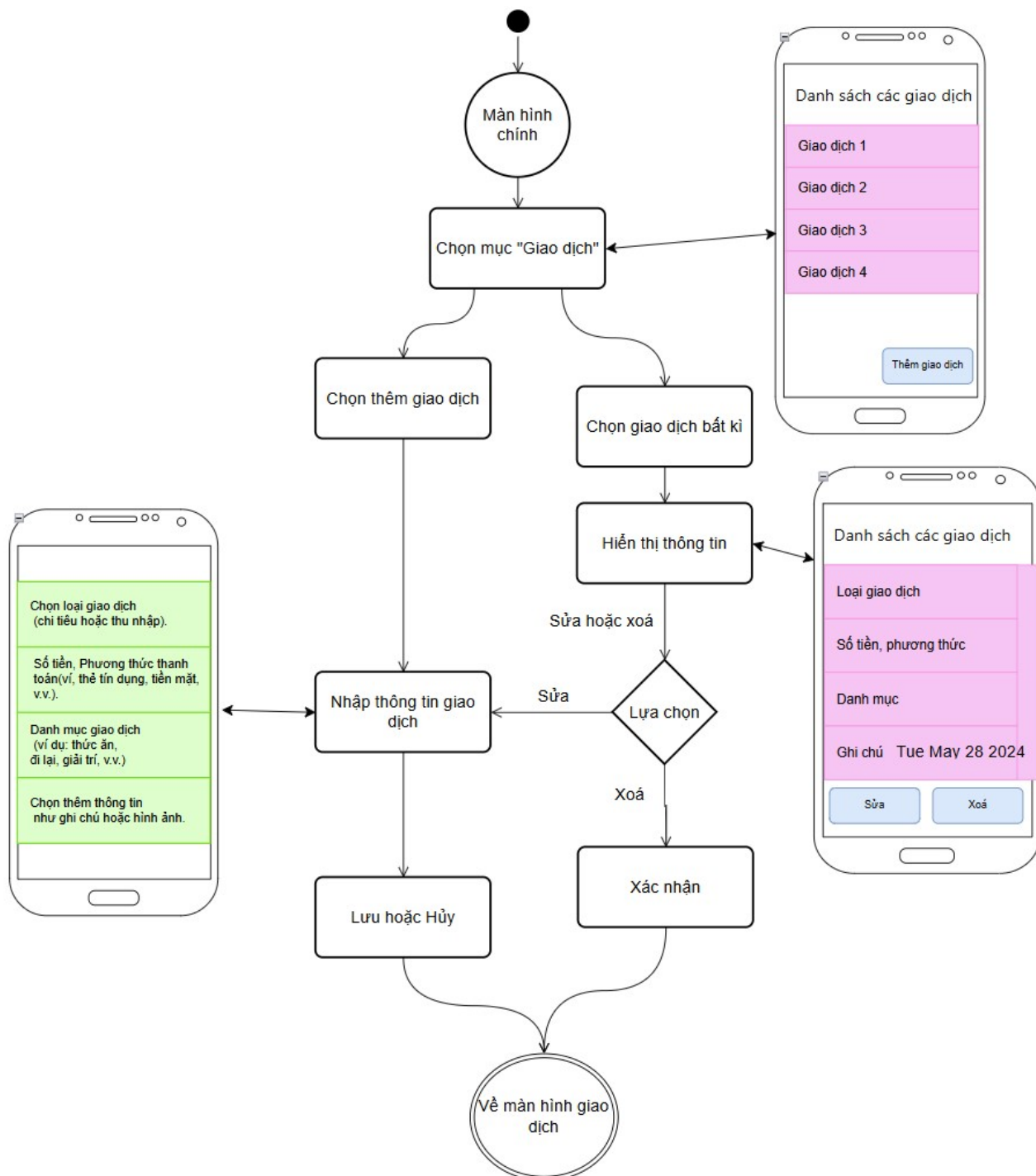


Figure 5.3: User flow Quản lý giao dịch

5.4 Sử dụng Chatbot

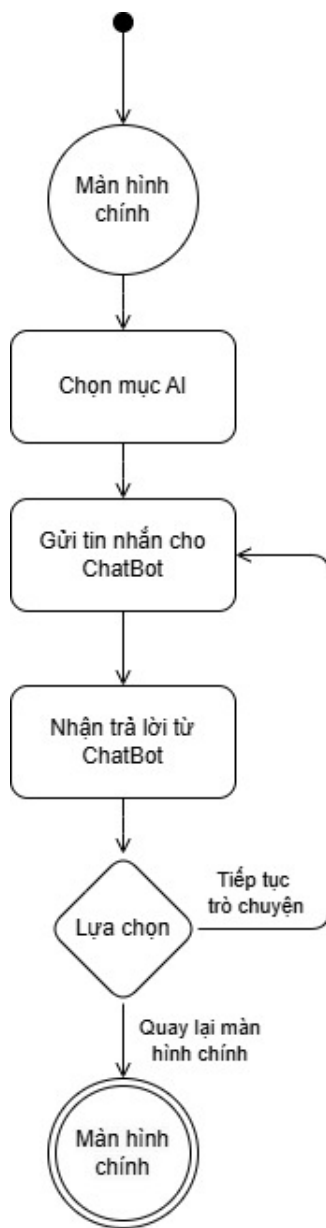


Figure 5.4: User flow Sử dụng Chatbot

5.5 Thay đổi gói đăng ký



Figure 5.5: User flow Thay đổi gói đăng ký

5.6 Gửi phản hồi



Figure 5.6: User flow Gửi phản hồi

6 Kiến trúc hệ thống

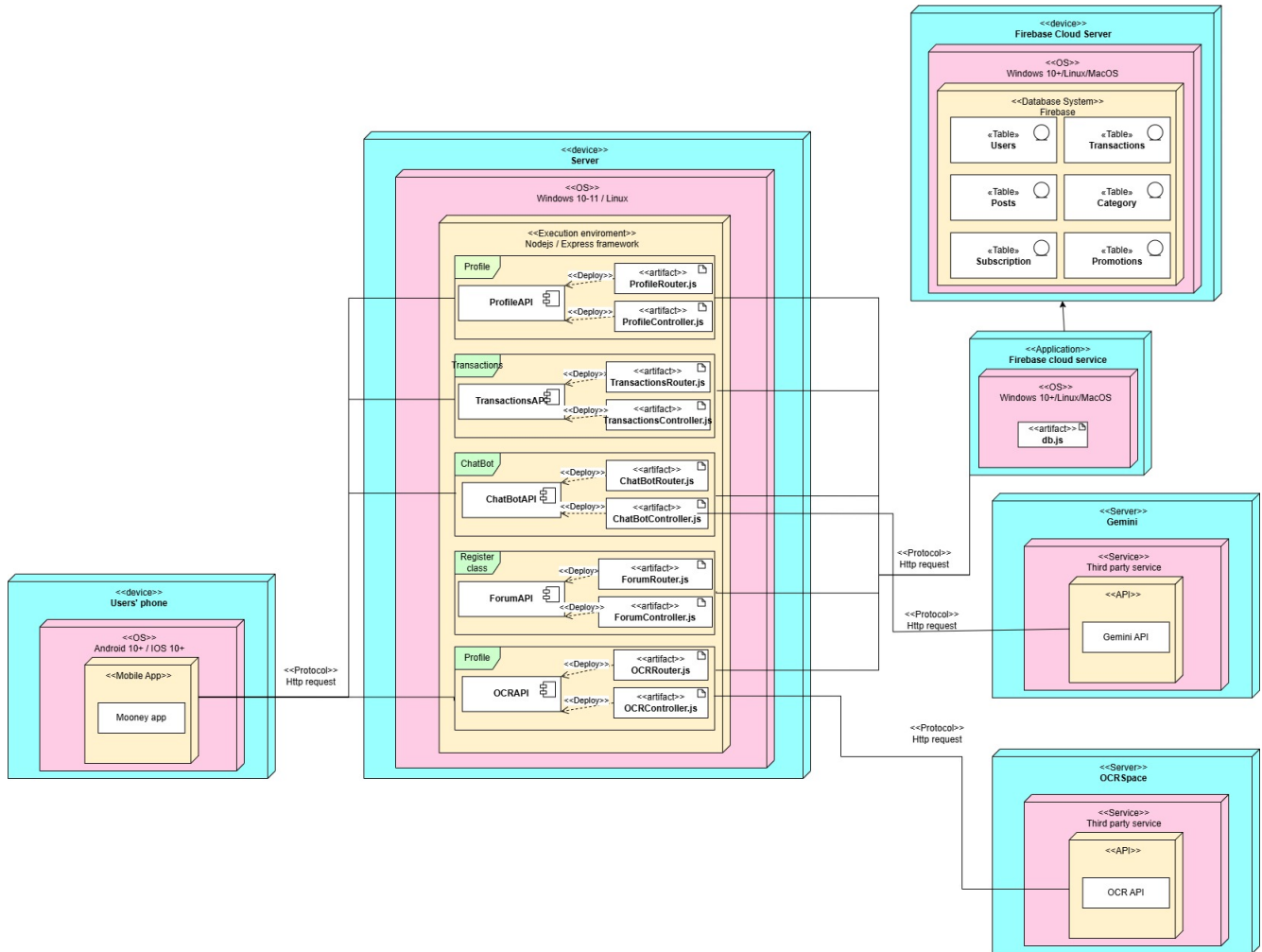


Figure 6.1: Sơ đồ kiến trúc hệ thống

7 Thiết kế Web Service

7.1 Kiến trúc

- **Mô hình client-server:** Ứng dụng sử dụng kiến trúc client-server, trong đó client sẽ gửi yêu cầu đến server và server sẽ xử lý yêu cầu và trả về kết quả cho client.
- **RESTful API:** Server cung cấp RESTful API để client có thể tương tác với ứng dụng. Mỗi API sẽ có một endpoint cụ thể và sử dụng các phương thức HTTP (GET, POST, PUT, DELETE) để thực hiện các chức năng khác nhau.

7.2 Các thành phần

- **Controllers:** Nhận request từ client, xử lý logic nghiệp vụ và gọi đến các service tương ứng.
 - Ví dụ: `UserController` xử lý các request liên quan đến người dùng như tạo tài khoản mới (`createUser`), lấy thông tin người dùng (`getUserById`), cập nhật thông tin người dùng (`updateUserbyId`)...
- **Services:** Chứa logic nghiệp vụ của ứng dụng, tương tác với model để truy xuất và xử lý dữ liệu.
 - Ví dụ: `UserService` cung cấp các phương thức để tạo người dùng mới (`createNewUser`), lấy thông tin người dùng (`getUser`), cập nhật thông tin người dùng (`updateUser`)...
- **Models:** Định nghĩa cấu trúc dữ liệu và cung cấp các phương thức để tương tác với cơ sở dữ liệu.
 - Ví dụ: `UserModel` định nghĩa cấu trúc dữ liệu cho người dùng bao gồm các trường `email`, `name`, `uid`... và cung cấp các phương thức để thêm, lấy, cập nhật và xóa người dùng trong cơ sở dữ liệu.
- **Routes:** Định nghĩa các endpoint cho API và ánh xạ chúng với các controller tương ứng.
 - Ví dụ: `/user/create_user` được ánh xạ đến phương thức `createUser` trong `UserController`.

7.3 Cơ sở dữ liệu

- Sử dụng Firebase Firestore làm cơ sở dữ liệu.

7.4 Công nghệ sử dụng

- **Node.js:** Nền tảng để xây dựng server.
- **Express.js:** Framework để xây dựng web application và API.
- **Firebase Firestore:** Cơ sở dữ liệu NoSQL.



7.5 Chi tiết về các API

Dịch vụ	API	Chức năng
User Service	/user	
	POST /create_user	Tạo người dùng mới
	GET /get_user/:uid	Lấy thông tin người dùng theo <i>uid</i>
	PUT /update_user/:uid	Cập nhật thông tin người dùng theo <i>uid</i>
Promotion Service	/promotions	
	POST /	Tạo chương trình khuyến mãi mới
	GET /:id	Lấy thông tin chương trình khuyến mãi theo <i>id</i>
	GET /	Lấy danh sách tất cả chương trình khuyến mãi
Subscription Service	/subscription	
	GET /:id	Lấy thông tin gói đăng ký theo <i>id</i>
	GET /	Lấy danh sách tất cả gói đăng ký
	GET /ref/:id	Lấy <i>reference</i> của gói đăng ký theo <i>id</i>
Feedback Service	/feedback	
	POST /	Tạo phản hồi mới
Transaction Service	/transactions	
	POST /create_transaction	Tạo giao dịch mới
	GET /get_transactions/:uid	Lấy danh sách giao dịch theo <i>uid</i> của người dùng
Post Service	/post	
	POST /create_post	Tạo bài đăng mới
	GET /get_posts	Lấy danh sách tất cả bài đăng

Table 7.1: Chi tiết các API của Web Service



8 Triển khai ứng dụng với người dùng cuối

Phát hành ứng dụng miễn phí trên App Store (iOS) và CH Play (Android). Áp dụng mô hình kinh doanh freemium: Cung cấp miễn phí các tính năng cơ bản. Mở khóa các tính năng nâng cao bằng cách đăng ký gói premium.

9 Khảo sát người dùng

9.1 Khảo sát thông tin chung của đối tượng khảo sát

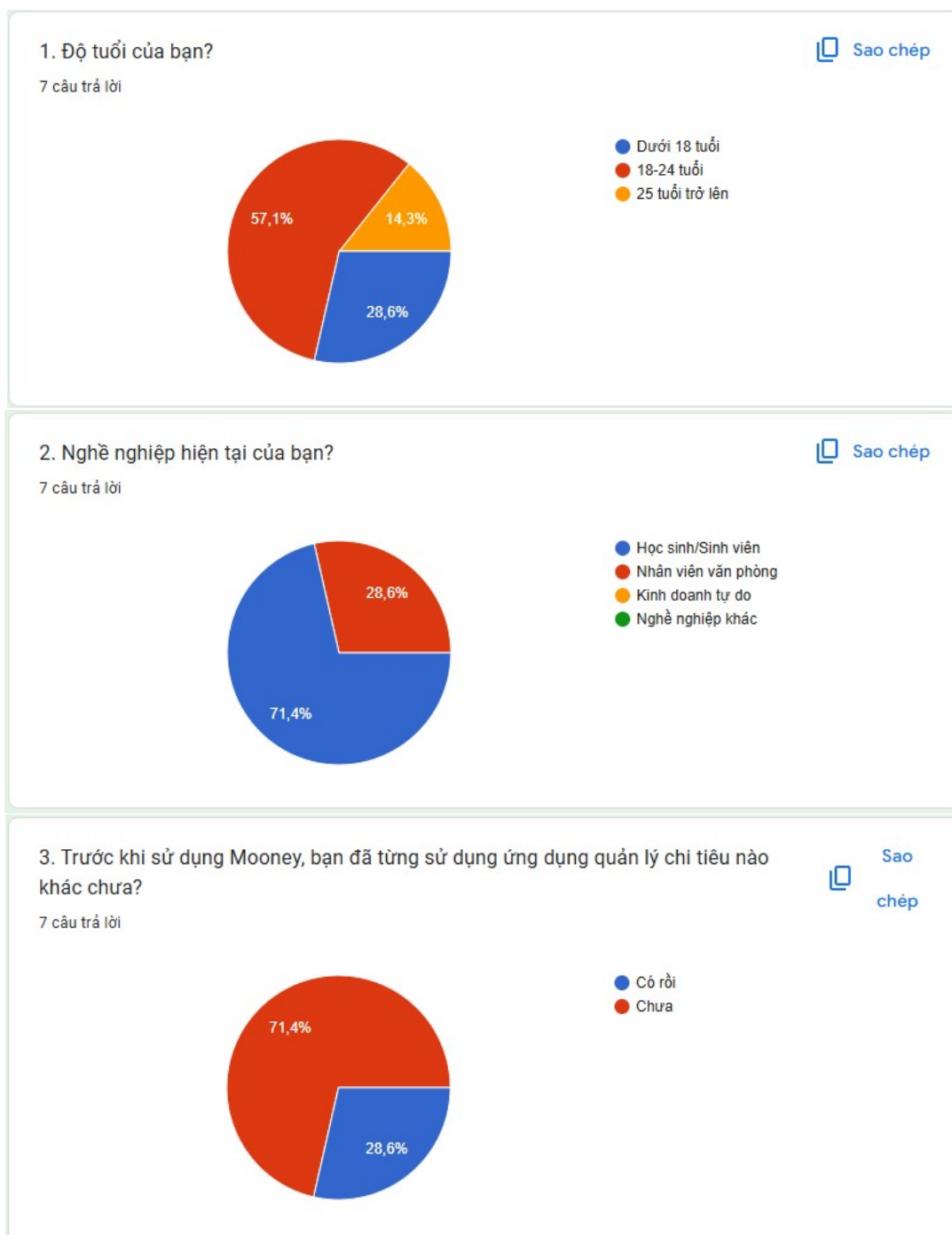


Figure 9.1: Thông tin chung

9.2 Khảo sát trải nghiệm ứng dụng



Figure 9.2: Khảo sát trải nghiệm ứng dụng

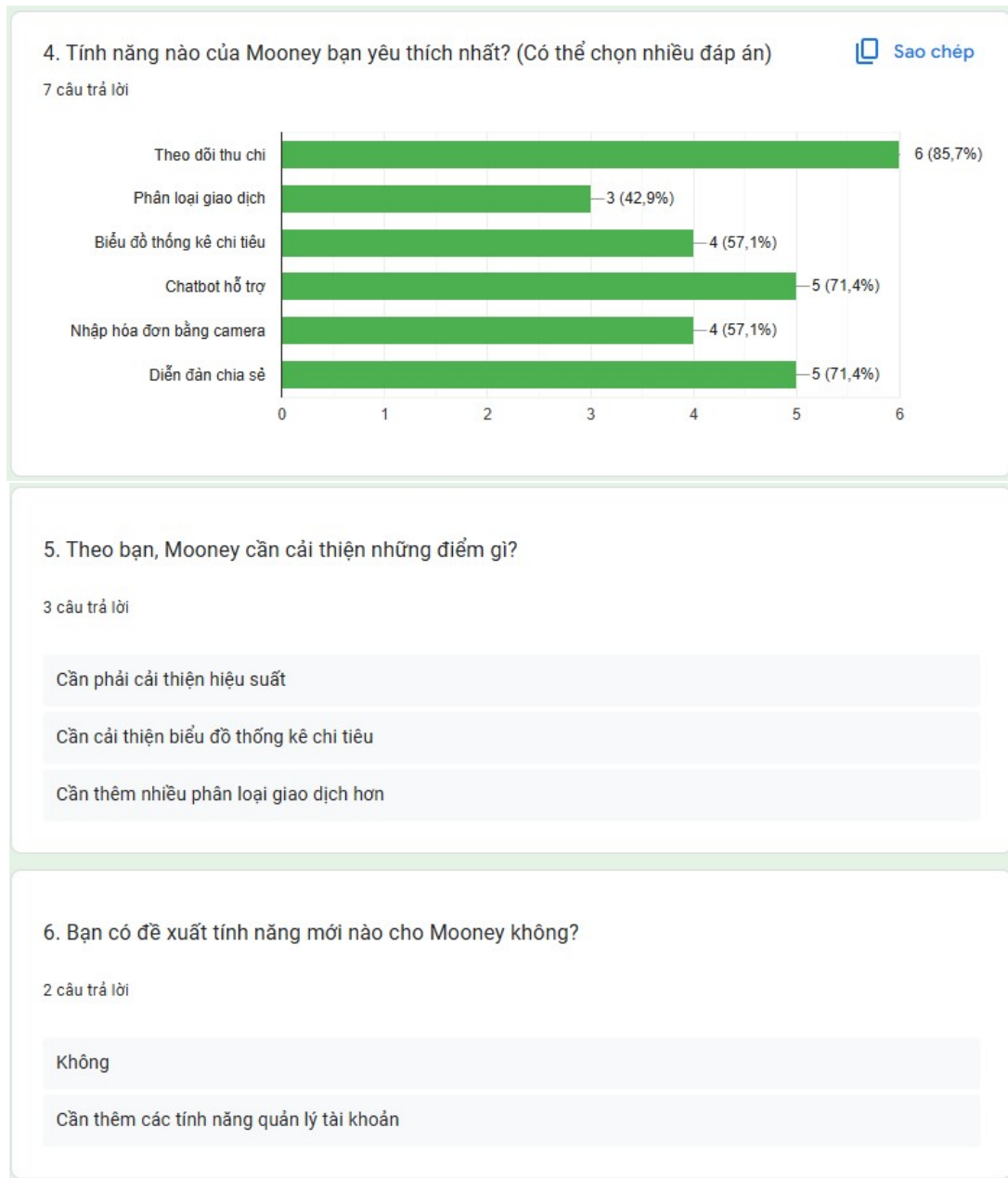


Figure 9.3: Khảo sát trải nghiệm ứng dụng

9.3 Đánh giá chung của người dùng

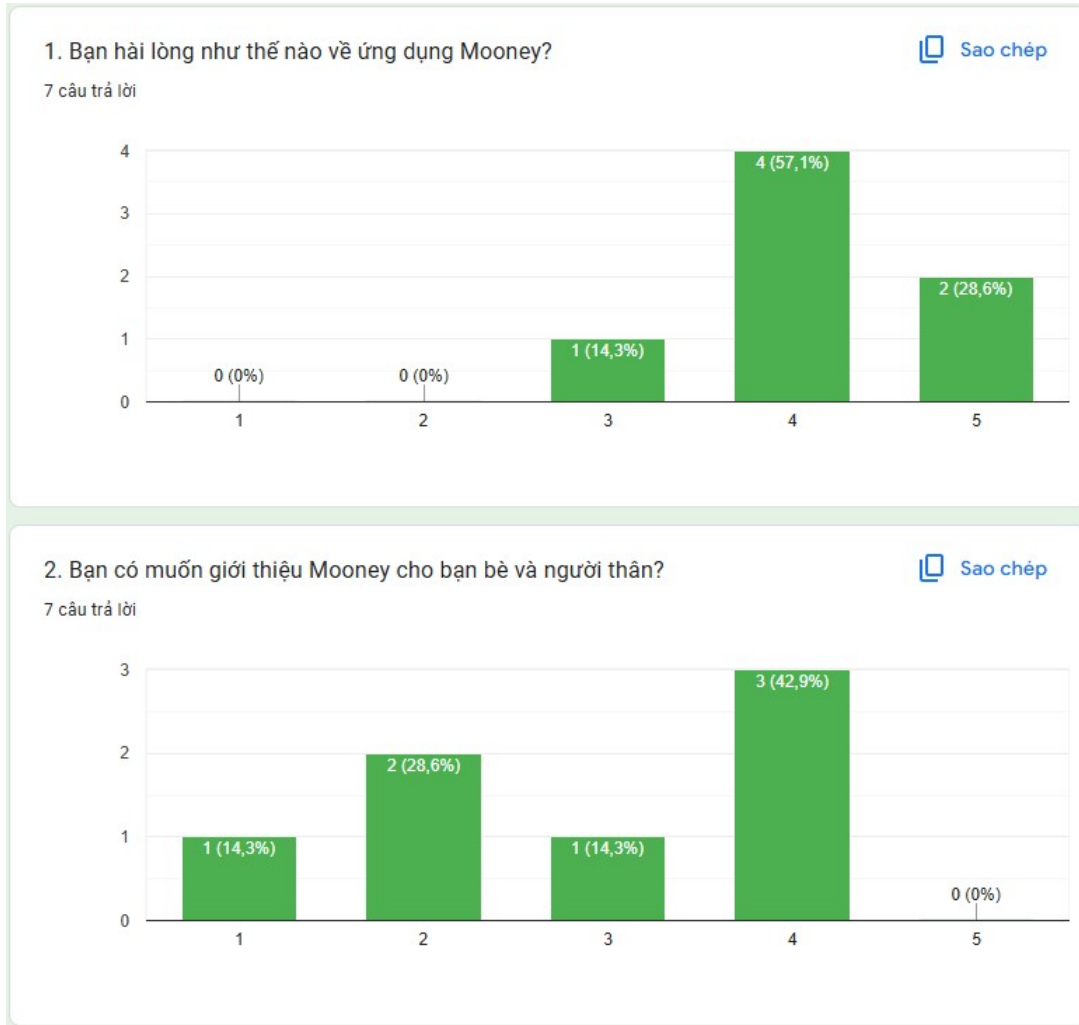


Figure 9.4: Khảo sát đánh giá chung của người dùng

10 Đánh giá mức độ thực hiện đề tài, các vấn đề cần khắc phục

- Đề tài "Ứng dụng quản lý chỉ tiêu Mooney" đã hoàn thành các mục tiêu đề ra ban đầu, bao gồm:
 - Xây dựng thành công ứng dụng di động đa nền tảng (iOS và Android) hỗ trợ người dùng quản lý chỉ tiêu cá nhân một cách hiệu quả.
 - Ứng dụng cung cấp các tính năng theo dõi thu chi, phân loại giao dịch, tạo

báo cáo trực quan, tích hợp chatbot hỗ trợ và diễn đàn chia sẻ kiến thức.

- Giao diện ứng dụng được thiết kế theo hướng tối giản, thân thiện và dễ sử dụng, phù hợp với đa dạng đối tượng người dùng.
- Kết quả khảo sát người dùng cho thấy ứng dụng được đánh giá cao về tính hữu ích, giao diện thân thiện và có tiềm năng phát triển.

• **Tuy nhiên, bên cạnh những kết quả đạt được, đề tài vẫn còn tồn tại một số hạn chế cần được khắc phục và cải thiện trong tương lai:**

- **Hiệu suất ứng dụng:** Ứng dụng cần được tối ưu hơn nữa để cải thiện tốc độ tải và xử lý dữ liệu, đặc biệt là khi người dùng có lượng giao dịch lớn.
- **Ví người dùng:** Cần cho phép một người dùng có thể tạo và quản lý nhiều ví (wallet) khác nhau.
- **Biểu đồ thống kê:** Các biểu đồ trực quan hóa dữ liệu cần được thiết kế đa dạng và trực quan hơn, giúp người dùng dễ dàng nắm bắt thông tin chi tiêu.
- **Phân loại giao dịch:** Mở rộng thêm các danh mục chi tiêu phổ biến và cho phép người dùng tùy chỉnh danh mục theo nhu cầu cá nhân.
- **Tính năng quản lý tài khoản:** Bổ sung các tính năng cho phép người dùng thay đổi thông tin cá nhân, mật khẩu và nâng cao bảo mật tài khoản.

• **Trong tương lai, nhóm phát triển sẽ tập trung vào việc:**

- Nâng cấp hiệu suất ứng dụng, tối ưu code và xử lý lỗi hiệu quả hơn.
- Phát triển tính năng quản lý nhiều ví cho phép người dùng tạo mới, đặt tên, tùy chỉnh, và chuyển đổi giữa các ví một cách dễ dàng.
- Cải thiện giao diện người dùng, bổ sung thêm các biểu đồ thống kê trực quan và dễ hiểu.
- Mở rộng tính năng quản lý tài khoản, phân loại giao dịch.

Với những nỗ lực cải tiến và phát triển không ngừng, nhóm phát triển tin tưởng rằng ứng dụng Mooney sẽ ngày càng hoàn thiện và trở thành một công cụ hữu ích, đồng hành cùng người dùng trong việc quản lý tài chính cá nhân hiệu quả.



Các liên kết

- Mã nguồn:
 - Frontend: <https://github.com/Hungww/Expense-Tracker.git>
 - Backend: https://github.com/Hungww/ExpenseTracker_server.git
- Behance: <https://www.behance.net/gallery/195433669/Mooney>
- Figma: <https://www.figma.com/file/FrvX7oBF4rvjakOKKNemHi/BTL-MOBILE?type=design&node-id=0%3A1&mode=design&t=JuLOdom2qJwz76j0-1>
- Survey link: <https://forms.gle/Y4BFTQo2j6GXuDsd9>
- Apk link: https://drive.google.com/drive/folders/1f4ZAPyXLB3ZIDzln_CLPL67pSexYvaEW?usp=drive_link