TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

KHOA CÔNG NGHỆ THÔNG TIN

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_🙞🕮🙜\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Diagram

Description automatically generated with low confidence

BÁO CÁO HỌC PHẦN

PHÁT TRIỂN ỨNG DỤNG TRÊN THIẾT BỊ DI ĐỘNG

XÂY DỰNG ỨNG DỤNG QUẢN LÝ TÀI KHOẢN SINH VIÊN HAUI

TRÊN NỀN TẢNG ANDROID

|  |  |  |
| --- | --- | --- |
| GVHD | : | ThS. Vũ Thị Dương |
| Nhóm | : | 16 |
| Sinh viên thực hiên | : | Bùi Trung Huỳnh Chu Xuân Lộc Ninh Duy Nhật Nguyễn Minh Quang  Nông Minh Trường |
| Lớp: 20242IT6029001 |  | Khóa K16 |

*Hà Nội - Năm 2024*

MỤC LỤC

[**DANH MỤC HÌNH ẢNH** 3](#_Toc169456521)

[**DANH MỤC BẢNG** 4](#_Toc169456522)

[DANH MỤC KÝ HIỆU, VIẾT TẮT 5](#_Toc169456523)

[LỜI CẢM ƠN 6](#_Toc169456524)

[PHẦN 1: MỞ ĐẦU 7](#_Toc169456525)

[1.1 Tổng quan về lập trình trên thiết bị di động 7](#_Toc169456526)

[1.2 Tên đề tài 7](#_Toc169456527)

[1.3 Lý do chọn đề tài 7](#_Toc169456528)

[1.4 Kiến thức, kỹ năng áp dụng 8](#_Toc169456529)

[1.5 Mục tiêu đề tài 8](#_Toc169456530)

[1.6 Tổng quan về báo cáo 8](#_Toc169456531)

[PHẦN 2: KẾT QUẢ NGHIÊN CỨU 9](#_Toc169456532)

[2.1. Tìm hiểu yêu cầu bài toán 9](#_Toc169456533)

[2.2. Xây dựng bài toán 9](#_Toc169456534)

[2.2.1. Phân tích yêu cầu người dùng 9](#_Toc169456535)

[2.2.2. Thiết kế hệ thống 12](#_Toc169456536)

[2.3. Thực hiện bài toán 16](#_Toc169456537)

[2.3.1 Bùi Trung Huỳnh - Đăng nhập, đăng ký, quên mật khẩu, thông tin cá nhân 16](#_Toc169456538)

[2.3.1.1 Đăng nhập 16](#_Toc169456539)

[2.3.1.2 Đăng ký 17](#_Toc169456540)

[2.3.1.3 Quên mật khẩu 19](#_Toc169456541)

[2.3.1.4 Thông tin cá nhân 20](#_Toc169456542)

[2.3.2 Chu Xuân Lộc- Quản lý chi tiêu,Thống kê chi tiêu 22](#_Toc169456543)

[2.3.2.1 Quản lý chi tiêu 22](#_Toc169456544)

[2.3.2.2 Thống kê chi tiêu 25](#_Toc169456545)

[2.3.3 Ninh Duy Nhật – Đổi mật khẩu, Đăng ký môn học 28](#_Toc169456546)

[2.3.3.1 Đổi mật khẩu 28](#_Toc169456547)

[2.3.3.2 Đăng ký môn học 30](#_Toc169456548)

[2.3.4. Nguyễn Minh Quang – Công nợ, nạp tiền công nợ, thanh toán. 31](#_Toc169456549)

[2.3.4.1 Công nợ 31](#_Toc169456550)

[2.3.4.2 Nạp số dư tiền học 33](#_Toc169456551)

[2.3.4.3 Thanh toán 36](#_Toc169456552)

[2.3.5. Nông Minh Trường –Lên kế hoạch, Nạp tiền chi tiêu sinh hoạt. 39](#_Toc169456553)

[2.3.5.1 Màn hình hiển thị kế hoạch 39](#_Toc169456554)

[2.3.5.2 Màn hình thêm kế hoạch 40](#_Toc169456555)

[2.3.5.3 Màn hình xem chi tiết kế hoạch 41](#_Toc169456556)

[2.3.5.4 Màn hình sửa kế hoạch 42](#_Toc169456557)

[2.3.5.5 Nạp tiền chi tiêu sinh hoạt 43](#_Toc169456558)

[2.4 Kiểm thử chương trình 45](#_Toc169456559)

[2.4.1 Kế hoạch kiểm thử 45](#_Toc169456560)

[2.4.2 Kết quả kiểm thử 47](#_Toc169456561)

[**PHẦN 3. KIẾN THỨC LĨNH HỘI VÀ BÀI HỌC KINH NGHIỆM** 50](#_Toc169456562)

[3.1. Nội dung đã thực hiện 50](#_Toc169456563)

[3.2. Hướng phát triển. 51](#_Toc169456564)

[**TÀI LIỆU THAM KHẢO** 52](#_Toc169456565)

# **DANH MỤC HÌNH ẢNH**

[Hình 1. Các Use Case trong ứng dụng 8](#_Toc154365870)

[Hình 2. Các bảng trong cơ sở dữ liệu 9](#_Toc154365871)

[Hình 3. Các màn hình chính(1) 9](#_Toc154365872)

[Hình 4. Các màn hình chính(2) 10](#_Toc154365873)

[Hình 5. Màn hiển thị mục tiêu 15](#_Toc154365874)

[Hình 6. Màn hình thêm mục tiêu mới 22](#_Toc154365875)

[Hình 7. Màn hình chi tiết mục tiêu 28](#_Toc154365876)

[Hình 8. Màn hình sửa mục tiêu 34](#_Toc154365877)

[Hình 9. Màn hình thống kê chi tiêu 39](#_Toc154365878)

[Hình 10. Màn hình liệt kê chi tiêu 43](#_Toc154365879)

[Hình 11. Màn hình thêm chi tiêu 48](#_Toc154365880)

[Hình 12. Màn hình chi tiêu chi tiêu 55](#_Toc154365881)

[Hình 13. Màn hình đăng nhập 62](#_Toc154365882)

[Hình 14. Màn hình đăng ký 68](#_Toc154365883)

[Hình 15. Màn hình các điều khoản ứng dụng 69](#_Toc154365884)

[Hình 16. Màn hình quên mật khẩu 76](#_Toc154365885)

[Hình 17. Màn hình giao diện cài đặt 77](#_Toc154365886)

[Hình 18. Màn hình sửa thông tin, xóa tài khoản người dùng 82](#_Toc154365887)

[Hình 19. Màn hình thay đổi mật khẩu 87](#_Toc154365888)

# **DANH MỤC BẢNG**

[Bảng 1. File XML hiện thị mục tiêu 15](#_Toc154365889)

[Bảng 2. File XML thêm mục tiêu mới 22](#_Toc154365890)

[Bảng 3. File XML màn chi tiết mục tiêu 27](#_Toc154365891)

[Bảng 4. File XML màn sửa mục tiêu 34](#_Toc154365892)

[Bảng 5. File XML màn thống kê 38](#_Toc154365893)

[Bảng 6. File XML màn liệt kê chi tiêu 43](#_Toc154365894)

[Bảng 7. File XML màn thêm chi tiêu 48](#_Toc154365895)

[Bảng 8. File XML màn chi tiết chi tiêu 55](#_Toc154365896)

[Bảng 9. File XML màn đăng nhập 61](#_Toc154365897)

[Bảng 10. File XML màn đăng ký 68](#_Toc154365898)

[Bảng 11. File XML màn quên mật khẩu 75](#_Toc154365899)

[Bảng 12. File XML màn thay đổi thông tin, xóa tài khoản người dùng 82](#_Toc154365900)

[Bảng 13. File XML màn thay đổi mật khẩu 86](#_Toc154365901)

# DANH MỤC KÝ HIỆU, VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Ý nghĩa |
| CSDL | Cơ sở dữ liệu |
| CMS | Content Management System |
| MVC | Model-View-Controller |

# LỜI CẢM ƠN

Với niềm đam mê lập trình cùng với sự dẫn dắt tận tình của cô Vũ Thị Dương cùng các thầy cô trong khoa Công nghệ Thông tin trường Đại học Công nghiệp Hà Nội. Nhóm chúng em Bày tỏ lòng biết ơn với cô thầy bộ môn Phát triển ứng dụng trên thiết bị di động đã nhiệt tình giảng dạy hướng dẫn chúng em kiến thức quý báu. Những kiến thức này không chỉ giúp chúng em trong quá trình thực hiện bài tập lớn mà còn là hành trang tiếp bước cho chúng em trong quá trình học tập và lập nghiệp sau này. Trong quá trình làm bài tập lớn, chúng em đã cố gắng hết sức để hoàn thành nhiệm vụ được giao. Nhóm em hy vọng có thể nhận thêm nhiều lời nhận xét, góp ý từ cô giáo và các bạn!

Nhóm em xin chân thành cảm ơn !

# PHẦN 1: MỞ ĐẦU

## 1.1 Tổng quan về lập trình trên thiết bị di động

Điện thoại thông minh hiện đã là một phần không thể thiếu trong xã hội, việc xây dựng các ứng dụng trên thiết bị di động là một thị trường mở đối với ngành phần mềm. Hiện nay có nhiều phương pháp lập trình như lập trình Native, sử dụng framework đa nền tảng, mới đây là công cụ no code, low code. Nhóm chúng em chọn phương pháp lập trình native với hệ điều hành android để có thể vận dụng và phát triển kiến thức, kỹ năng đã học trong môn phát triển ứng dụng trên thiết bị di động.

## 1.2 Tên đề tài

Xây dựng ứng dụng quản lý tài khoản sinh viên HaUI trên nền tảng Android.

## 1.3 Lý do chọn đề tài

Cuộc sống ngày càng phát triển, nhịp sống của sinh viên khiến ta khó lòng kiểm soát mọi thứ. Vì vậy các công cụ quản lý ra đời giúp sinh viên nói chung và sinh viên HAUI nói riêng dễ dàng kiểm soát chúng hơn. Ý tưởng nhóm chúng em đưa ra là một ứng dụng quản lý tài khoản sinh viên HaUI trong cuộc sống thường nhật. Điện thoại di động đã gắn liền với cuộc sống của chúng ta, việc tích hợp quản lý tài khoản vào ứng dụng điện thoại sẽ mang tới khá nhiều lợi ích và tiện nghi.

Nhận thức được tầm quan trọng của ứng dụng di động hỗ trợ quản lý và cũng như áp dụng kiến thức đã học được của học phần phát triển ứng dụng cho thiết bị di động chúng em xin được áp dụng kiến thức đã học được và tìm hiểu để xây dựng ứng dụng quản lý tài khoản sinh viên HaUI.

## 1.4 Kiến thức, kỹ năng áp dụng

Trong thời gian học tập bộ môn phát triển ứng dụng cho thiết bị di động, chúng em được học về các kiến thức liên quan tới lập trình mobile như lập trình thiết kế activity, các xử lý sự kiện trên màn hình di động, xử lý giao diện người dùng, giao tiếp và lưu trữ giữ liệu trên SQLite, ... Ngoài ra chúng em còn nghiên cứu thêm một số kiến thức khác liên quan đến fragment. Tất cả kiến thức trên được chúng em tiếp thu và vận dụng dụng vào bài tập lớn dưới sự hướng dẫn từ giảng viên bộ môn. Vận dụng các kiến thức liên quan bổ sung đã học trên trường về phân tích đặc tả, lập trình Java, quản lý dự án... Bên cạnh kiến thức chuyên môn chúng em cũng được thực hành kỹ năng khác như kỹ năng giao tiếp, kỹ năng làm việc nhóm, kỹ năng quản lý thời gian. Chúng em đã vận dụng các kỹ năng trên để trong quá trình làm việc hiệu quả hơn.

## 1.5 Mục tiêu đề tài

Mục tiêu đầu ra của chúng em là có thể hiểu được các kiến thức nền tảng về lập trình di động, Áp dụng được các kiến thức, kỹ năng nêu trên để phân tích thiết kế một ứng dụng di động ở mức độ cơ bản. Qua việc hoàn thành bài tập lớn chúng em đã phần nào thực hiện được mục tiêu đã nêu trên.

## 1.6 Tổng quan về báo cáo

Nhóm chúng em tiến hành xây dựng ứng dụng Quản lý tài khoản sinh viên HaUI trên nền tảng Android. Bài báo cáo bao gồm 3 phần:

Phần 1: Mở đầu , tổng quan đề tài

Phần 2: Phần kết quả nghiên cứu

Phần 3: Kiến thức lĩnh hội và bài học kinh nghiệm

# 

# PHẦN 2: KẾT QUẢ NGHIÊN CỨU

## 2.1. Tìm hiểu yêu cầu bài toán

Quản lý tài khoản sinh viên HaUI được thiết kế để giúp sinh viên theo dõi và quản lý chi tiêu cá nhân,đăng kí môn học và thanh toán công nợ một cách hiệu quả. Ý tưởng hình thành xuất phát từ nhu cầu thực tế của việc quản lý tài chính sinh viên trong thời đại số hóa hiện nay.

Ứng dụng này giải quyết các chức năng chính như: theo dõi chi tiêu, lập kế hoạch ngân sách, phân loại chi tiêu, và cung cấp báo cáo chi tiêu,đăng kí môn học và thanh toán công nợ. Tầm quan trọng của các chức năng này nằm ở việc giúp sinh viên kiểm soát tài chính cá nhân và đưa ra quyết định tiêu dùng thông minh.

## 2.2. Xây dựng bài toán

### 2.2.1. Phân tích yêu cầu người dùng

**Yêu cầu chức năng:**

Đăng nhập: Người dùng cần nhập tên đăng nhập và mật khẩu đã đăng ký để truy cập vào ứng dụng. Nếu tên đăng nhập và mật khẩu đúng, người dùng sẽ được chuyển đến màn hình chính của ứng dụng. Nếu không, hệ thống sẽ thông báo lỗi và yêu cầu người dùng nhập lại.

Đăng ký:Người dùng mới cần tạo một tài khoản để sử dụng ứng dụng. Để đăng ký, người dùng cần nhập tên đăng nhập, mật khẩu và xác nhận mật khẩu. Hệ thống sẽ kiểm tra xem tên đăng nhập đã tồn tại hay chưa và xác nhận mật khẩu có khớp với mật khẩu đã nhập hay không. Nếu tất cả thông tin hợp lệ, tài khoản mới sẽ được tạo và người dùng sẽ được chuyển đến màn hình đăng nhập.

Lên kế hoạch :Người dùng có thể thêm, sửa hoặc xóa các mục tiêu trong kế hoạch của mình. Người dùng chọn chức năng tương ứng, nhập hoặc cập nhật thông tin mục tiêu và hệ thống sẽ lưu lại thay đổi.

Nạp tiền chi tiêu sinh hoạt:Người dùng có thể nạp tiền vào tài khoản chi tiêu sinh hoạt của mình. Người dùng chọn chức năng nạp tiền, nhập số tiền cần nạp và hệ thống sẽ cập nhật số dư tài khoản tương ứng.

Thống kê chi tiêu sinh hoạt:Người dùng có thể xem thống kê chi tiêu sinh hoạt theo nhiều tiêu chí như thời gian và loại chi tiêu. Người dùng chọn chức năng thống kê, chọn tiêu chí mong muốn và hệ thống sẽ hiển thị kết quả thống kê.

Quản lý chi tiêu sinh hoạt :Người dùng có thể thêm hoặc sửa các khoản chi tiêu sinh hoạt. Người dùng chọn chức năng thêm hoặc sửa chi tiêu, nhập hoặc cập nhật thông tin chi tiêu và hệ thống sẽ lưu lại thay đổi.

Đổi mật khẩu:Người dùng có thể đổi mật khẩu tài khoản của mình. Người dùng chọn chức năng đổi mật khẩu, nhập mật khẩu hiện tại, mật khẩu mới và xác nhận mật khẩu mới. Hệ thống sẽ kiểm tra tính hợp lệ và cập nhật mật khẩu nếu thông tin hợp lệ.

Đăng ký môn học:Người dùng có thể đăng ký các môn học. Người dùng chọn chức năng đăng ký môn học, chọn các môn học cần đăng ký và hệ thống sẽ lưu lại thông tin đăng ký

Nạp tiền công nợ: Người dùng có thể nạp tiền để thanh toán công nợ. Người dùng chọn chức năng nạp tiền công nợ, nhập số tiền cần nạp và hệ thống sẽ cập nhật số dư tài khoản công nợ.

Thanh toán công nợ:Người dùng có thể thanh toán các khoản công nợ của mình. Người dùng chọn chức năng thanh toán công nợ, chọn khoản công nợ cần thanh toán và hệ thống sẽ xử lý thanh toán và cập nhật trạng thái công nợ.

**Các yêu cầu phi chức năng**:

Ứng dụng cần dễ sử dụng, có khả năng tương thích với nhiều thiết bị và hệ điều hành khác nhau, và đảm bảo an toàn thông tin người dùng.

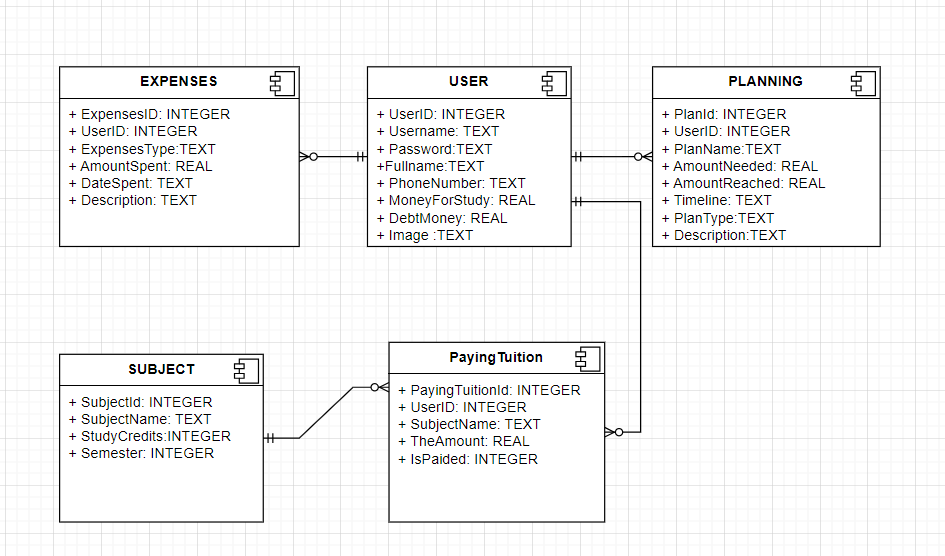
**Biểu đồ usecase**

A diagram of a diagram

Description automatically generated

Hình 1. Các Use Case trong ứng dụng

**Mô tả cơ sở dữ liệu:**



Hình 2. Các bảng trong cơ sở dữ liệu

### 2.2.2. Thiết kế hệ thống

Phần thông tin chung:

Nhóm đã thống nhất về việc thiết kế giao diện với màu sắc nhẹ nhàng với màu xanh dương làm chủ đạo, kích cỡ phù hợp, và bố trí các điều khiển một cách khoa học.

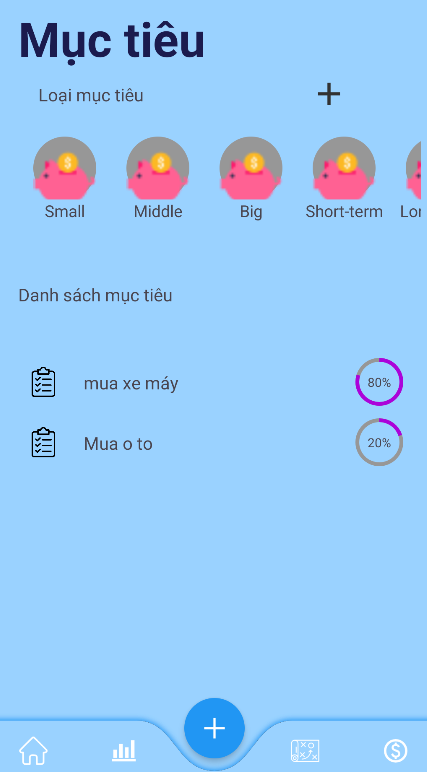
Screens screenshot of a cell phone

Description automatically generated

Hình 3. Các màn hình chính(1)

Giao diện chính của ứng dụng bao gồm một thanh điều hướng ở dưới cùng màn hình, cho phép người dùng dễ dàng chuyển giữa các màn hình chính như “Trang chủ”, “Thống kê”, “Đăng kí”, vv. Mỗi màn hình chính này lại chứa các màn hình con tương ứng với các chức năng cụ thể.

Screens screenshots of a phone

Description automatically generated 

Hình 4. Các màn hình chính(2)

Nhóm cũng đã quy định rõ ràng về việc đặt tên cho các activity, XML layout, menu, values, và cách bố trí các package lưu trữ thông tin của toàn bài và của từng người.

Thiết kế dữ liệu:

Dữ liệu được lưu trữ trong SQLite. Các lớp dữ liệu được ánh xạ thành các bảng trong cơ sở dữ liệu, với mỗi thuộc tính của lớp tương ứng với một trường trong bảng.

Bảng “USER”

Bảng “USER” có các thuộc tính “UserID”, “Username”, “Password”, “PhoneNumber”, “LivingExpenses”, “MoneyForStudying”, “DebtMoney” sẽ được ánh xạ từ các thuộc tính tương ứng trong lớp. Mỗi thuộc tính này sẽ được ánh xạ thành một trường trong bảng, với kiểu dữ liệu tương ứng:

- UserID: INTEGER, PRIMARY KEY, tự động tăng (AUTOINCREMENT).

- Username: TEXT.

- Password: TEXT.

- PhoneNumber: TEXT.

- LivingExpenses: INTEGER.

- MoneyForStudying:REAL.

- DebtMoney: REAL.

-Image:TEXT

Bảng “EXPENSES”

Bảng “EXPENSES” có các thuộc tính “ExpensesID”, “UserID”, “ExpensesType”, “AmountSpent”, “DateSpent”, “Description” sẽ được ánh xạ từ các thuộc tính tương ứng trong lớp. Mỗi thuộc tính này sẽ được ánh xạ thành một trường trong bảng, với kiểu dữ liệu tương ứng:

- ExpensesID: INTEGER, PRIMARY KEY, tự động tăng (AUTOINCREMENT).

- UserID: INTEGER, FOREIGN KEY tham chiếu đến bảng USER.

- ExpensesType: TEXT.

- AmountSpent: REAL.

- DateSpent: TEXT.

- Description: TEXT.

Bảng “PLANNING”

Bảng “PLANNING” có các thuộc tính “PlanId”, “UserID”, “PlanName”, “AmountNeeded”, “AmountReached”, “Timeline”, “PlanType” sẽ được ánh xạ từ các thuộc tính tương ứng trong lớp. Mỗi thuộc tính này sẽ được ánh xạ thành một trường trong bảng, với kiểu dữ liệu tương ứng:

- PlanId: INTEGER, PRIMARY KEY, tự động tăng (AUTOINCREMENT).

- UserID: INTEGER, FOREIGN KEY tham chiếu đến bảng USER.

- PlanName: TEXT.

- AmountNeeded: REAL.

- AmountReached: REAL.

- Timeline: TEXT.

- PlanType: TEXT.

Bảng “SUBJECT”

Bảng “SUBJECT” có các thuộc tính “SubjectId”, “SubjectName”, “StudyCredits”, “Semester” sẽ được ánh xạ từ các thuộc tính tương ứng trong lớp. Mỗi thuộc tính này sẽ được ánh xạ thành một trường trong bảng, với kiểu dữ liệu tương ứng:

- SubjectId: INTEGER, PRIMARY KEY, tự động tăng (AUTOINCREMENT).

- SubjectName: TEXT.

- StudyCredits: INTEGER.

- Semester: TEXT.

Bảng “PayingTuition”

Bảng “PayingTuition” có các thuộc tính “PayingTuitionId”, “UserID”, “SubjectName”, “TheAmount”, “IsPaided” sẽ được ánh xạ từ các thuộc tính tương ứng trong lớp. Mỗi thuộc tính này sẽ được ánh xạ thành một trường trong bảng, với kiểu dữ liệu tương ứng:

- PayingTuitionId: INTEGER, PRIMARY KEY, tự động tăng (AUTOINCREMENT).

- UserID: INTEGER, FOREIGN KEY tham chiếu đến bảng USER.

- SubjectName: TEXT.

- TheAmount: REAL.

- IsPaided: INTEGER (1: đã thanh toán, 0: chưa thanh toán).

## 2.3. Thực hiện bài toán

### Bùi Trung Huỳnh - Đăng nhập, đăng ký, quên mật khẩu, thông tin cá nhân

#### 2.3.1.1 Đăng nhập

Mô tả: Đăng nhập tài khoản có các trường nhập thông tin bao gồm:tên đăng nhập, mật khẩu, textview “Bạn quên mật khẩu?” để chuyển màn hình đặt lại mật khẩu khi người dùng quên, textview “Đăng ký” để chuyển sang màn hình đăng ký; một nút “Đăng nhập” để đăng nhập vào tài khoản.

Nhiệm vụ: Người dùng nhập tên đăng nhập, mật khẩu và kích nút “Đăng nhập” hệ thống sẽ kiểm tra tài khoản tồn tại không, nếu tồn tại sẽ đăng nhập vào tài khoản đó, ngược lại thông báo tên đăng nhập hoặc mật khẩu không đúng.

Giao diện:

A screenshot of a blue screen

Description automatically generated

Hình 13. Màn hình đăng nhập

- Mã nguồn:

Xử lý đăng nhập

|  |
| --- |
|  |

#### 2.3.1.2 Đăng ký

Mô tả: Đăng ký một tài khoản người dùng có các trường nhập dữ liệu bao gồm: họ và tên, tên đăng nhập, mật khẩu, xác nhận lại mật khẩu; một nút “Đăng ký” đăng ký tài khoản mới và một nút “Back” để quay trở về màn hình đăng nhập, textview “Đăng nhập” để chuyển sang màn hình đăng nhập;.

Nhiệm vụ: Người dùng nhập đủ các trường thông tin và kích vào nút “Đăng ký” chương trình sẽ kiểm tra xem tên đăng nhập người dùng nhập tồn tại không? Nếu tồn tại sẽ thông báo tài khoản tồn tại để nhập tên đăng nhập khác. Để quay lại màn hình đăng nhập người dùng chọn nút “Back” hoặc textview “Đăng nhập” để chuyển sang màn hình đăng nhập. Các trường nhập thông tin với các yêu cầu đặc biệt: độ dài mật khẩu tối thiểu 6 ký tự và phải chứa ít nhất 1 ký tự viết hoa và có cả chữ và số, mật khẩu và xác nhận mật khẩu phải giống nhau.

Giao diện:

.xml

|  |
| --- |
|  |

Bảng2. File XML màn đăng ký

Giao diện hoàn thành

A screenshot of a phone

Description automatically generated

Hình 14. Màn hình đăng ký

Mã nguồn:

Xử lý đăng ký

|  |
| --- |
|  |

#### 2.3.1.3 Quên mật khẩu

Mô tả: Màn hình quên mật khẩu có các trường nhập dữ liệu như tên đăng nhập, mật khẩu mới, nhập lại mật khẩu mới; nút “xác nhận”để cập nhật mật khẩu mới, một nút “Back” để quay trở về màn hình đăng nhập.

Nhiệm vụ: Người dùng nhập đầy đủ các trường thông tin và kích nút “Xác nhận”. Hệ thống sẽ kiểm tra trường thông tin gồm tên đăng nhập có khớp với tài khoản đã tồn tại trong cơ sở dữ liệu không? Nếu đã khớp trường thông tin trên thì sẽ cho cập nhật mật khẩu mới, ngược lại thông báo thông tin không chính xác. Trường nhập mật khẩu mới yêu cầu mật khẩu tối thiểu 6 ký tự, ít nhất một ký tự viết hoa và phải có cả chữ cái lẫn chữ số. Trường mật khẩu mới và xác nhận mật khẩu mới phải giống nhau.

Giao diện:

.xml:

|  |
| --- |
|  |

Bảng 11. File XML màn quên mật khẩu

Giao diện hoàn thành:

A screenshot of a blue screen

Description automatically generated

Hình 16. Màn hình quên mật khẩu

Mã nguồn:

|  |
| --- |
|  |

#### 2.3.1.4 Thông tin cá nhân

Mô tả: Là một Màn hình gồm: thông tin cơ bản người dùng ( Họ và tên, ảnh đại diện, số dư hiện tại), các item menu chức năng(Thay đổi ảnh đại diện, thay đổi mật khẩu, chức năng lưu mật khẩu,chức năng đăng nhập bằng vân tay, chức năng nhạp tiền sinh hoạt ,đăng xuất).

Nhiệm vụ: Người dùng kích vào ảnh đại diện trong màn hình home.

Giao diện thực tế:

A screenshot of a cell phone

Description automatically generated

Hình 17. Màn hình giao diện cài đặt

Giao diện:

.xml

|  |
| --- |
|  |

Mã nguồn:

|  |
| --- |
|  |

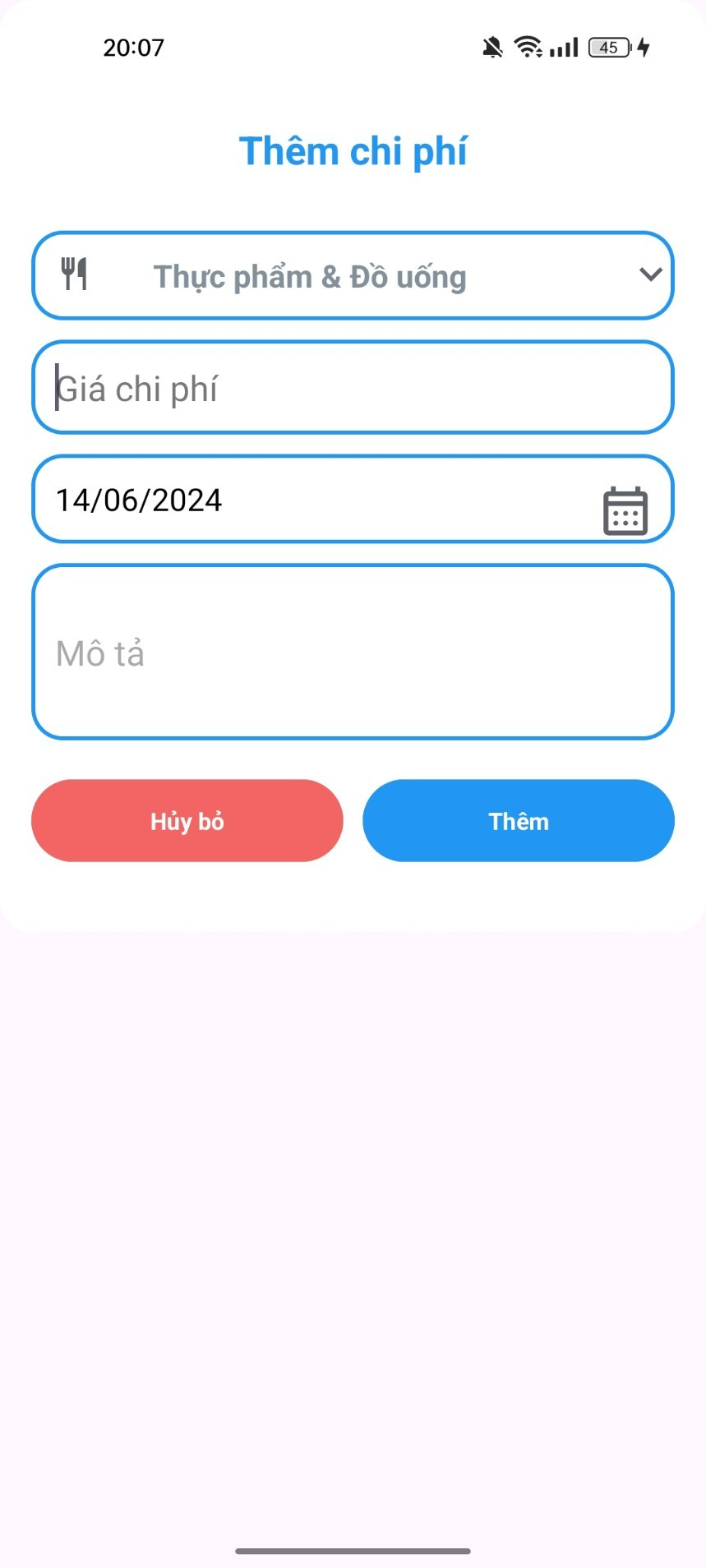
### Chu Xuân Lộc- Quản lý chi tiêu,Thống kê chi tiêu

#### 2.3.2.1 Quản lý chi tiêu

Mô tả: Màn chi tiết chi tiêu có các sẽ hiện thông tin chi tiêu như số tiền, ngày, loại, mô tả, cho phép người dùng sửa trực tiếp các trường, một nút Thêm để lưu những thay đổi chi tiêu và một nút Hủy bỏ để quay lại màn liệt kê chi tiêu,.

Nhiệm vụ: người dùng sửa các thông tin của chi tiêu sau đó bấm nút Lưu, chương trình sẽ kiểm tra dữ liệu có hợp lệ hay không, nếu có chi tiêu sẽ được sửa,nếu không sẽ hiện lên thông báo, người dùng sẽ trở về màn liệt kê chi tiêu khi bấm nút back hoặc sửa thành công.

Giao diện hoàn thành



Hình 12. Màn hình chi tiêu chi tiêu

Mã nguồn:

Xử lý sự kiện chọn ngày bằng DatePickerDialog:

|  |
| --- |
| date\_picker.setOnClickListener(new View.OnClickListener() { @Override public void onClick(View v) {  Calendar calendar = Calendar.getInstance();  int ngay = calendar.get(Calendar.DATE);  int thang = calendar.get(Calendar.MONTH);  int nam = calendar.get(Calendar.YEAR);  DatePickerDialog dialog = new DatePickerDialog(add\_chi\_phi.this, android.R.style.Theme\_Holo\_Light\_Dialog, new DatePickerDialog.OnDateSetListener() { @Override public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {  try {  Date date = new Date(year - 1900, month, dayOfMonth);  DateFormat df = new SimpleDateFormat("dd/MM/yyyy");  String dateString = df.format(date);  thoi\_gian\_mua.setText(dateString);  } catch (Exception e) {  Toast.makeText(add\_chi\_phi.this, e.toString(), Toast.LENGTH\_SHORT).show();  }  }  }, nam, thang, ngay);  dialog.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));  dialog.setTitle("Chọn thời gian");  dialog.show();  }  }); |

Xử lý sự kiện khi nút thêm chi phí được nhấn:

|  |
| --- |
| nhanThemChiPhi.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (xulyDauVao()) {  finish();  }  }  }); |

Phương thức xử lý dữ liệu đầu vào:

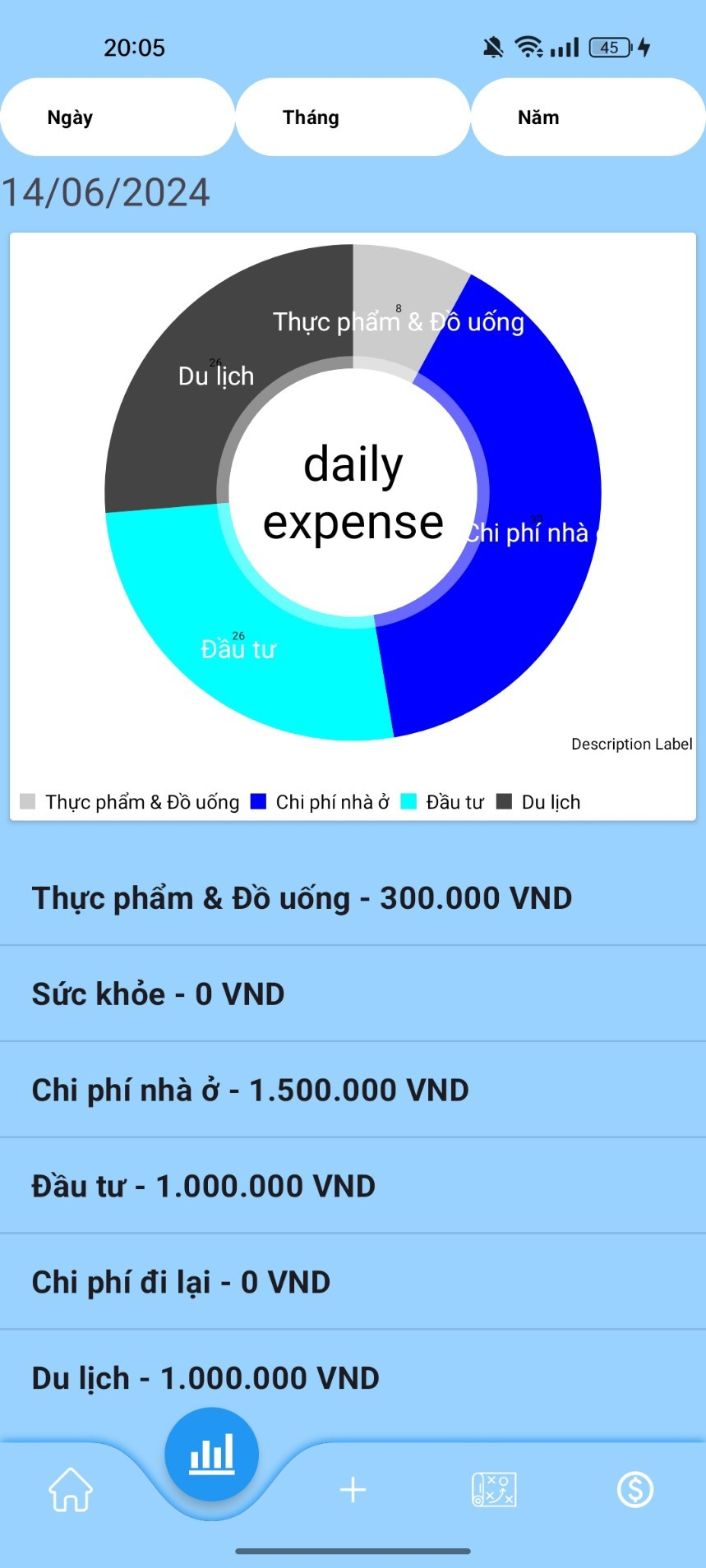
|  |
| --- |
| private boolean xulyDauVao() {  if (mo\_ta\_chi\_phi.getText().toString().isEmpty() || gia\_chi\_phi.getText().toString().isEmpty()) {  Toast.makeText(this, "Phải điền đầy đủ thông tin", Toast.LENGTH\_SHORT).show();  return false;  } else {  double gia = Double.parseDouble(gia\_chi\_phi.getText().toString());  if (gia < 1000) {  Toast.makeText(this, "Chi phí phải lớn hơn 1000", Toast.LENGTH\_SHORT).show();  return false;  } else {  UserObject userObject = new UserObject();  userObject = getDataUserName();  String ngaymua = thoi\_gian\_mua.getText().toString();  databaseHelper = new DatabaseHelper(add\_chi\_phi.this);  double sodumoi = userObject.getLivingExpenses() - gia;  if (sodumoi >= 0) {  databaseHelper.update\_LivingExpenses(userObject.getUserID(), sodumoi);  databaseHelper.insertUser\_KhoanChi(userObject.getUserID(), loaichi, gia, ngaymua, mo\_ta\_chi\_phi.getText().toString());  return true;  } else {  Toast.makeText(this, "Số dư không đủ", Toast.LENGTH\_SHORT).show();  return false;  }  }  }  } |

#### 2.3.2.2 Thống kê chi tiêu

Mô tả: Mã xml sẽ gồm 3 linearLayout tương ứng với 3 lựa chọn xem thống kê là là “theo ngày”, “theo tháng”,”theo năm”. Cùng lớp 3 nút bấm tương ứng, 1 màn hình sẽ được hiển thị và 2 màn hình còn lại sẽ được ẩn đi theo tùy chọn. Mỗi Linear sẽ gồm 1 TextView dùng để hiển thị khoảng thời gian mà người dùng đã chọn để xem thống kê, 1 biểu đồ tròn và 1 ListView dùng để hiển thị danh sách.

Nhiệm vụ: Hiển thị các thống kê chi tiêu của người dùng lên màn hình thông qua biểu đồ và danh sách các hạng mục. Các thống kê sẽ được hiện thị theo thời gian tương ứng với các nút bấm ở phía trên màn hình.

Giao diện hoàn thành:



Hình 9. Màn hình thống kê chi tiêu

Mã nguồn:

**Phương thức lấy dữ liệu chi tiêu hàng ngày và cập nhật biểu đồ:**

|  |
| --- |
| private void dailydata1(String ngayhientai){  databaseHelper= new DatabaseHelper(getActivity());   displayDate.setText(ngayhientai);  PieChart dailyPieChart;  ListView dailyList;   UserObject userObject= new UserObject();  userObject=getDataUserName();  List<ExpensesObject> list=databaseHelper.getExpensesObjectOfDate(userObject.getUserID(),ngayhientai);  Collator collator = Collator.*getInstance*(Locale.*FRENCH*);  collator.setStrength(Collator.*PRIMARY*);  float n1=0,n2=0,n3=0,n4=0,n5=0,n6=0,n7=0;  for(int i=0;i<list.size();i++){  if(collator.compare(list.get(i).getExpensesType(),"Thực phẩm & Đồ uống")==0){  n1+= (float) list.get(i).getAmountSpent();  } else if (collator.compare(list.get(i).getExpensesType(),"Sức khỏe")==0) {  n2+= (float) list.get(i).getAmountSpent();  } else if (collator.compare(list.get(i).getExpensesType(),"Chi phí nhà ở")==0) {  n3+= (float) list.get(i).getAmountSpent();  }else if (collator.compare(list.get(i).getExpensesType(),"Đầu tư")==0) {  n4+= (float) list.get(i).getAmountSpent();  }else if (collator.compare(list.get(i).getExpensesType(),"Chi phí đi lại")==0) {  n5+= (float) list.get(i).getAmountSpent();  }else if (collator.compare(list.get(i).getExpensesType(),"Du lịch")==0) {  n6+= (float) list.get(i).getAmountSpent();  }else if (collator.compare(list.get(i).getExpensesType(),"Chi phí khác")==0) {  Log.*e*("check dữ liệu chi phí khác",n7+"");  n7+= (float) list.get(i).getAmountSpent();  }   }   ArrayList<PieEntry> dataVals = new ArrayList<>();  dataVals.add(new PieEntry(n1,"Thực phẩm & Đồ uống"));  dataVals.add(new PieEntry(n2,"Sức khỏe"));  dataVals.add(new PieEntry(n3,"Chi phí nhà ở"));  dataVals.add(new PieEntry(n4,"Đầu tư"));  dataVals.add(new PieEntry(n5,"Chi phí đi lại"));  dataVals.add(new PieEntry(n6,"Du lịch"));  dataVals.add(new PieEntry(n7,"Chi phí khác"));  for(int i=0;i<dataVals.size();i++){  if(dataVals.get(i).getValue()==0){  dataVals.remove(i);  }  }    //Nhập dữ liệu cho biểu đồ thống kê ngày //  dailyPieChart = binding.dailyPieChart;  PieDataSet piedataset = new PieDataSet(dataVals,"");  piedataset.setColors(colorClassArray);  PieData pieData = new PieData(piedataset);  dailyPieChart.setData(pieData);  dailyPieChart.setCenterTextSize(25);  dailyPieChart.setUsePercentValues(true);  dailyPieChart.setCenterText("daily expense");  dailyPieChart.invalidate();   NumberFormat numberFormat = NumberFormat.*getInstance*();  ArrayList<String> list1 = new ArrayList<>();  list1.add("Thực phẩm & Đồ uống - "+numberFormat.format(n1) +" VND");  list1.add("Sức khỏe - "+numberFormat.format(n2)+" VND");  list1.add("Chi phí nhà ở - "+numberFormat.format(n3)+" VND");  list1.add("Đầu tư - "+numberFormat.format(n4)+" VND");  list1.add("Chi phí đi lại - "+numberFormat.format(n5)+" VND");  list1.add("Du lịch - "+numberFormat.format(n6)+" VND");  list1.add("Chi phí khác - "+numberFormat.format(n7)+" VND");   ArrayList<String> List = new ArrayList<>();  List.addAll(list1);   ArrayAdapter<String> adapter1 = new ArrayAdapter<String>(getActivity(), android.R.layout.*simple\_list\_item\_1*, List);  dailyList = binding.listDailyExpense;  dailyList.setAdapter(adapter1);  adapter1.notifyDataSetChanged(); } |

### 2.3.3 Ninh Duy Nhật – Đổi mật khẩu, Đăng ký môn học

#### 2.3.3.1 Đổi mật khẩu

Mô tả: Màn hình Đổi mật khẩu có các trường nhập dữ liệu như Mật khẩu hiện tại, Mật khẩu mới, Xác nhận mật khẩu; nút Đổi mật khẩu để thay đổi mật khẩu mới, một nút Hủy bỏ để quay trở về màn hình đăng nhập.

Nhiệm vụ: Người dùng nhập đầy đủ các trường thông tin và kích nút “Đổi mật khẩu”. Hệ thống sẽ kiểm tra trường Mật khẩu hiện tại có khớp với mật khẩu của tài khoản trong cơ sở dữ liệu không? Nếu đã khớp thì sẽ cho cập nhật mật khẩu mới, ngược lại thông báo thông tin không chính xác. Trường nhập mật khẩu mới yêu cầu mật khẩu tối thiểu 6 ký tự, một ký tự đầu tiên viết hoa, mật khẩu có cả chữ lẫn số, trường Mật khẩu mới và Xác nhận mật khẩu phải giống nhau, trường Mật khẩu mới và trường Mật khẩu hiện tại không được trùng nhau, nếu nhập sai yêu cầu thì sẽ hiện thông báo nhập lại để thỏa mãn yêu cầu tương ứng với lỗi sai.

Giao diện :

A screenshot of a chat

Description automatically generated

Mã nguồn:

Java:

#### 2.3.3.2 Đăng ký môn học

Mô tả: Màn hình Đăng ký môn học cho phép người dùng Đăng ký các môn học mới, người dùng chọn học kỳ và chọn các môn cần đăng ký, nút Đăng ký để đăng ký môn học mới.

Nhiệm vụ: người dùng chọn học kỳ và chọn các môn cần đăng ký và kích nút Đăng ký. Hệ thống sẽ thông báo thành công và thêm thông tin của môn đăng ký và trong phần thanh toán của người dùng. Nếu số tiền đăng ký tín vượt qua 15.000.000 thì sẽ không được đăng ký nữa, nếu môn học đã được đăng ký trong khoảng thời gian hiện tại thì sẽ không được đăng ký nữa.

Giao diện:

Thực tế:

A screenshot of a cell phone

Description automatically generated

Mã nguồn:

Java:

### 2.3.4. Nguyễn Minh Quang – Công nợ, nạp tiền công nợ, thanh toán.

#### 2.3.4.1 Công nợ

Mô tả: là màn hình gồm: số dư tiền học, số tiền công nợ, 2 nút chức năng( Nạp tiền, Thanh toán) và lịch sử thanh toán.

Nhiệm vụ: Người dùng kích vào biểu tượng công nợ trên màn hình.

Giao diện hoàn thành:

A screenshot of a phone

Description automatically generated

Hình 13. Màn hình công nợ

- Mã nguồn:

Lấy thông tin người dùng; Xử lý OnClick của nút Thanh toán và Nạp tiền

|  |
| --- |
| databaseHelper = new DatabaseHelper(getContext());   SharedPreferences sharedPreferences = getActivity().getSharedPreferences("LUU\_TRANG\_THAI\_NGUOI\_DUNG",*MODE\_PRIVATE*);  String user= sharedPreferences.getString("UserName","");   int userId = databaseHelper.getUserByUsername(user).getUserID();  loadUserData(userId);  loadTransactionHistory(userId);  btnThanhToan.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  ThanhToan\_Fragment chiTietFragment = new ThanhToan\_Fragment();  FragmentTransaction transaction = getActivity().getSupportFragmentManager().beginTransaction();  transaction.replace(R.id.*fragment\_container*, chiTietFragment);   transaction.addToBackStack(null);  transaction.commit();  }  });   btnNaptien.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  Recharge\_Fragment rechargeFragment = new Recharge\_Fragment();  FragmentTransaction transaction = getActivity().getSupportFragmentManager().beginTransaction();  transaction.replace(R.id.*fragment\_container*, rechargeFragment);   transaction.addToBackStack(null);  transaction.commit();  }  });   return view;  } |

Phương thức lấy số dư tiền học và tiền công nợ

|  |
| --- |
| private void loadUserData(int userId) {  UserObject user = databaseHelper.getUserById(userId);  if (user != null) {  soDuTextView.setText(String.*format*("%,.0f VND", user.getMoneyForStudying()));  soTienConNoTextView.setText(String.*format*("%,.0f VND", user.getDebtMoney()));  }  } |

Phương thức tải lại lịch sử thanh toán

|  |
| --- |
| private void loadTransactionHistory(int userId) {  transactionList.clear();  transactionList.addAll(databaseHelper.getPaidTuitionList(userId));  transactionAdapter.notifyDataSetChanged();  } |

#### 2.3.4.2 Nạp số dư tiền học

Mô tả: màn hình bao gồm 1 textview hiển thị tiền công nợ, 1 edittext nhập số tiền, 1 spinner chọn nạp tiền trực tiếp hoặc nạp tiền bằng ngân hàng, ngoài ra nếu chọn nạp bằng ngân hàng sẽ có 3 edittext để nhập thông tin khác. Cuối cùng là 2 nút nạp tiền và quay lại.

Nhiệm vụ: Ngươi dùng cần nhập đúng định dạng số tiền. Nếu chọn nạp tiền theo ngân hàng sẽ có 3 edittext hiện ra và tất cả các edittext đều phải nhập mới có thể ấn nạp tiền. Nếu không nhập đủ sẽ thông báo chưa nhập đủ thông tin. Ấn nạp tiền sẽ thông báo nạp tiền thành công. Nếu ấn quay lại sẽ quay lại màn hình công nợ.

Giao diện hoàn thành

A screenshot of a phone

Description automatically generated A screenshot of a phone

Description automatically generated

Hình 14. Màn hình Nạp số dư tiền học

Mã nguồn:

Xử lý khi chọn phương thức thanh toán

|  |
| --- |
| phuongThucThanhToanSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {  @Override  public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {  String selectedMethod = parent.getItemAtPosition(position).toString();  if (selectedMethod.equals("Nạp tiền bằng ngân hàng")) {  bankTransferLayout.setVisibility(View.*VISIBLE*);  } else {  bankTransferLayout.setVisibility(View.*GONE*);  }  }   @Override  public void onNothingSelected(AdapterView<?> parent) {  bankTransferLayout.setVisibility(View.*GONE*);  } }); |

Xử lý OnClick nút Nạp

|  |
| --- |
| buttonNap.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  try {  double tienNap = Double.*parseDouble*(editText.getText().toString());  if (tienNap <= 0) {  showDialog("Số tiền nhập không hợp lệ. Vui lòng nhập số tiền lớn hơn 0");  editText.setText("");  } else {  String selectedMethod = phuongThucThanhToanSpinner.getSelectedItem().toString();  if (selectedMethod.equals("Nạp tiền bằng ngân hàng")) {  String soTaiKhoan = soTaiKhoanEditText.getText().toString().trim();  String tenNganHang = tenNganHangEditText.getText().toString().trim();  String chiNhanhNganHang = chiNhanhNganHangEditText.getText().toString().trim();  if (soTaiKhoan.isEmpty() || tenNganHang.isEmpty() || chiNhanhNganHang.isEmpty()) {  showDialog("Vui lòng nhập đầy đủ thông tin ngân hàng");  return;  }  }   userObject.setMoneyForStudying(tienNap + userObject.getMoneyForStudying());  databaseHelper.updateUser(userObject);  editText.setText("");  ArrayAdapter<CharSequence> adapter = ArrayAdapter.*createFromResource*(requireContext(),  R.array.*payment\_methods*, android.R.layout.*simple\_spinner\_item*);  adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*);  phuongThucThanhToanSpinner.setAdapter(adapter);   showDialog("Nạp tiền thành công");  }  } catch (NumberFormatException e) {  showDialog("Vui lòng nhập số tiền hợp lệ");  editText.setText("");  }  }  }); |

#### 2.3.4.3 Thanh toán

Mô tả: Màn hình quên mật khẩu có các trường nhập dữ liệu như firstname, lastname, email, newpassword, confirm password; nút update để cập nhật mật khẩu mới, một nút cancel để quay trở về màn hình đăng nhập.

Nhiệm vụ: Người dùng nhập đầy đủ các trường thông tin và kích nút “update”. Hệ thống sẽ kiểm tra các trường thông tin gồm firstname, lastname, email có khớp với tài khoản đã tồn tại trong cơ sở dữ liệu không? Nếu đã khớp 3 trường thông tin thì sẽ cho cập nhật mật khẩu mới, ngược lại thông báo thông tin không chính xác. Trường nhập mật khẩu mới yêu cầu mật khẩu tối thiểu 5 ký tự, ít nhất một ký tự viết hoa. Trường new password và confirm password phải giống nhau.

Giao diện hoàn thành

A screenshot of a cell phone

Description automatically generated

Hình 14. Màn hình Nạp số dư tiền học

Mã nguồn:

Lấy thông tin người dùng và xử lý OnClick

|  |
| --- |
| SharedPreferences sharedPreferences = getActivity().getSharedPreferences("LUU\_TRANG\_THAI\_NGUOI\_DUNG", getContext().*MODE\_PRIVATE*);  String userName = sharedPreferences.getString("UserName", "");  currentUser = databaseHelper.getUserByUsername(userName);  recyclerView.setLayoutManager(new LinearLayoutManager(getContext()));  loadData();   rbChonTatCa.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  if (adapter.getSelectedPositions().size() == payingTuitionList.size()) {  adapter.getSelectedPositions().clear();  rbChonTatCa.setChecked(false);  } else {  for (int i = 0; i < payingTuitionList.size(); i++) {  adapter.getSelectedPositions().add(i);  }  }  adapter.notifyDataSetChanged();  }  });   btnThanhToanFinal.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  paySelectedTuition();  }  });   btnQuayLai.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  FragmentManager fragmentManager = getFragmentManager();  fragmentManager.popBackStack();  }  });   return view;  } |

Phương thức tải lại data

|  |
| --- |
| private void loadData() {  payingTuitionList = databaseHelper.getUnpaidTuitionList(currentUser.getUserID());  adapter = new PayingTuitionAdapter(getContext(), payingTuitionList);  recyclerView.setAdapter(adapter); } |

Phương thức xử lý thanh toán và hiển thị thông báo

|  |
| --- |
| private void paySelectedTuition() {  Set<Integer> selectedPositions = adapter.getSelectedPositions();  double totalPayment = 0;  if (selectedPositions.isEmpty()) {  showDialog("Vui lòng chọn ít nhất một môn học để thanh toán");  return;  }   for (int position : selectedPositions) {  PayingTuitionObject tuition = payingTuitionList.get(position);  totalPayment += tuition.getAmount();  }  if (currentUser.getMoneyForStudying() < totalPayment) {  showDialog("Số dư không đủ để thanh toán");  return;  }   for (int position : selectedPositions) {  PayingTuitionObject tuition = payingTuitionList.get(position);  tuition.setPaided(true);  databaseHelper.updatePayingTuition(tuition);  }   double newDebt = currentUser.getDebtMoney() - totalPayment;  currentUser.setMoneyForStudying(currentUser.getMoneyForStudying() - totalPayment);  currentUser.setDebtMoney(newDebt);  databaseHelper.updateUser(currentUser);  showDialog("Thanh toán thành công!");  rbChonTatCa.setChecked(false);  loadData();  }  private void showDialog(String message) {  AlertDialog.Builder builder = new AlertDialog.Builder(getContext());  builder.setMessage(message)  .setPositiveButton("OK", null)  .create()  .show();  } |

### 2.3.5. Nông Minh Trường –Lên kế hoạch, Nạp tiền chi tiêu sinh hoạt.

#### 2.3.5.1 Màn hình hiển thị kế hoạch

Mô tả: Màn hình gồm Nút “+” giúp người dùng chuyển qua giao diện tạo một kế hoạch mới. Màn hình gồm 2 recyclerView. RecyclerView thứ nhất hiển thị các loại kế hoạch theo kỳ hạn : khi nhấn vào item ở đây sẽ phân loại danh sách ở dưới theo loại đã chọn. RecyclerView thứ 2 hiển thị danh sách các kế hoạch đã tạo: khi nhấn vào item ở đây sẽ đi tới giao diện chi tiết kế hoạch.

Nhiệm vụ: Màn hiển thị danh sách mục kế hoạch có nhiệm vụ hiển thị danh sách kế hoạch, có gọi đến màn hình thêm kế hoạch thông qua ImageView “+”.

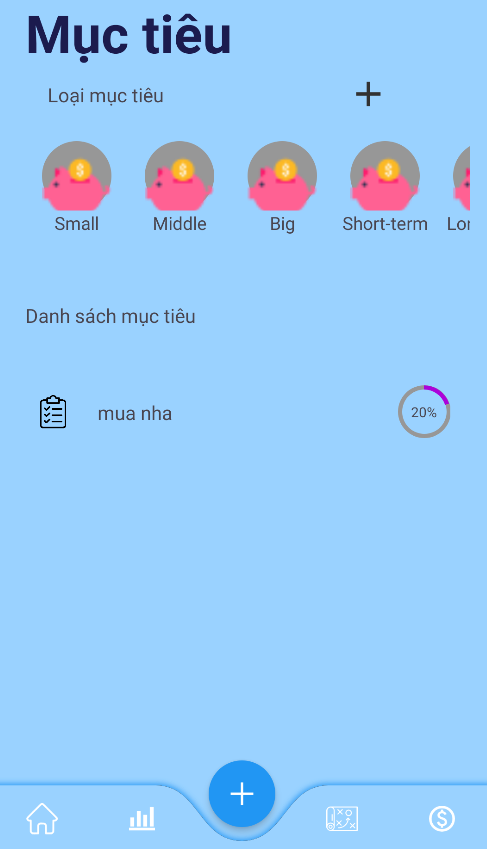
Giao diện:

.xml

|  |
| --- |
|  |

Bảng 11. File XML màn hình lên kế hoạch

Giao diện hoàn thành:



Hình 15. Màn hình lên kế hoạch

- Mã nguồn:

|  |
| --- |
|  |

#### 2.3.5.2 Màn hình thêm kế hoạch

Mô tả: Bao gồm các trường nhập dữ liệu cho mục tiêu mới và 2 nút button xử lý sự kiện.

Nhiệm vụ: Màn thêm kế hoạch có nhiệm vụ thêm kế hoạch mới, có gọi đến màn hiển thị danh sách kế hoạch sau khi thêm kế hoạch hoàn tất.

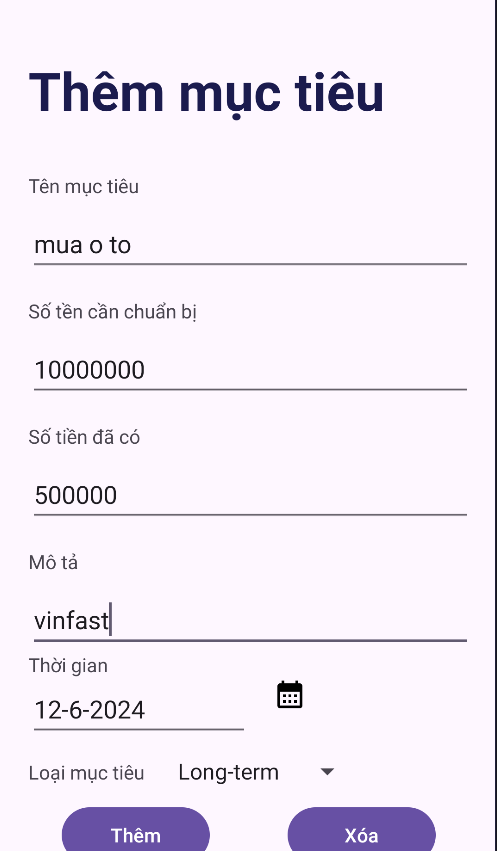
Giao diện:

.xml

|  |
| --- |
|  |

Bảng 12. File XML màn hình thêm kế hoạch

Giao diện hoàn thành:



Hình 16. Màn hình thêm kế hoạch

- Mã nguồn:

|  |
| --- |
|  |

#### 2.3.5.3 Màn hình xem chi tiết kế hoạch

Mô tả: Hiển thị các thông tin của kế hoạch và 2 nút button xử lý sự kiện.

Nhiệm vụ: Màn chi tiết kế hoạch hiển thị thông tin chi tiết của kế hoạch, có chức năng xóa kế hoạch đang hiển thị. có gọi đến màn danh sách kế hoạch sau khi xóa kế hoạch và màn chỉnh sửa sau khi nhấn vào nút “Cập nhật”.

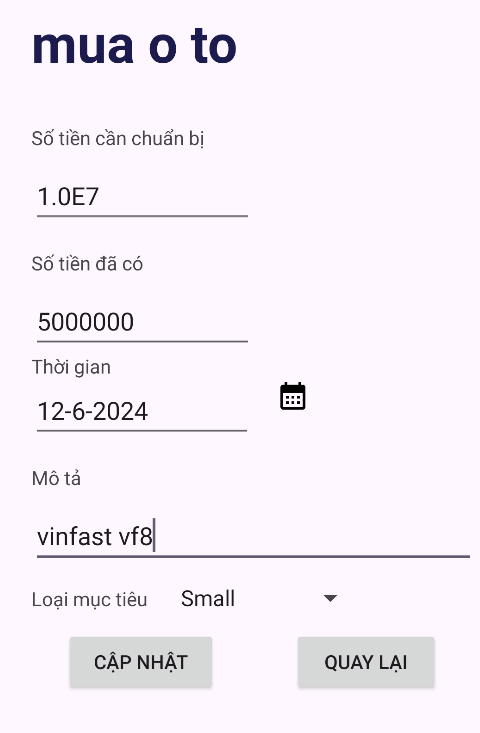
Giao diện:

.xml

|  |
| --- |
|  |

Bảng 13. File XML màn hình xem chi tiết kế hoạch

Giao diện hoàn thành:



Hình 17. Màn hình xem chi tiết kế hoạch

- Mã nguồn:

|  |
| --- |
|  |

#### 2.3.5.4 Màn hình sửa kế hoạch

Mô tả: Hiển thị nội dung cần sửa và 2 button xử lý sự kiện.

Nhiệm vụ: Màn hình Chỉnh sửa kế hoạch hiển thị các trường sửa thông tin, có liên kết với màn danh sách kế hoạch thông qua nút “Quay lại”.

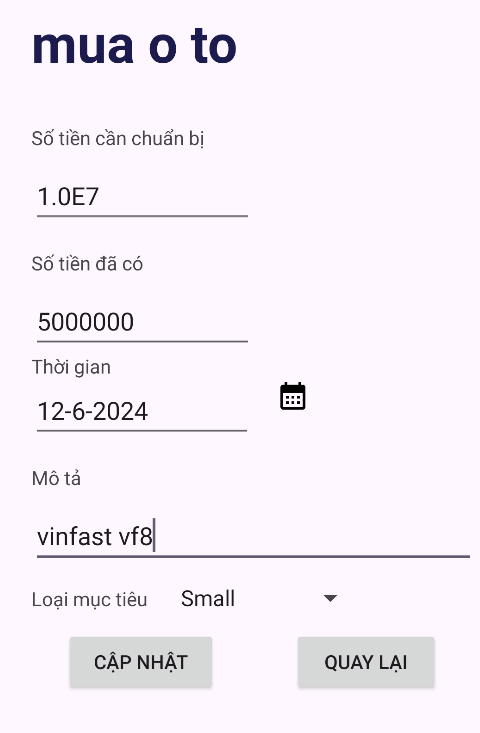
Giao diện:

.xml

|  |
| --- |
|  |

Bảng 14. File XML màn hình sửa kế hoạch

Giao diện hoàn thành:



Hình 18. Màn hình sửa kế hoạch

- Mã nguồn:

|  |
| --- |
|  |

#### 2.3.5.5 Nạp tiền chi tiêu sinh hoạt

Mô tả: Màn hình bao gồm 1 edittext hiển thị tiền nạp và 2 button xử lý sự kiện.

Nhiệm vụ: Người dùng cần nhập đúng định dạng số tiền, ấn nạp tiền sẽ thông báo nạp tiền thành công. Nếu ấn hủy bỏ sẽ quay lại màn hình thông tin cá nhân.

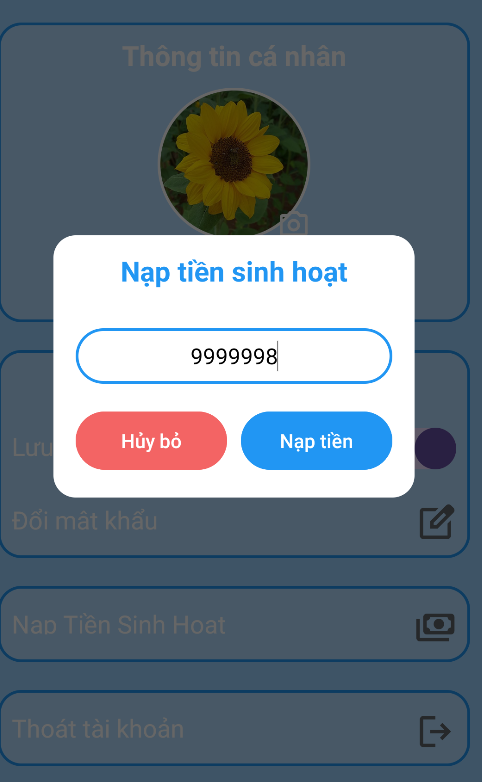
Giao diện:

.xml

|  |
| --- |
|  |

Bảng 15. File XML màn nạp tiền chi tiêu sinh hoạt

Giao diện hoàn thành



Hình 19. Màn hình nạp tiền chi tiêu sinh hoạt

Mã nguồn:

|  |
| --- |
| : 0 VND");  }  }  } |

## 2.4 Kiểm thử chương trình

### 2.4.1 Kế hoạch kiểm thử

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mốc công việc | Sản phẩm | Thời gian | Bắt đầu | Kết thúc |
| Lập kế hoạch kiểm thử | Test plan | 1 ngày | 7/6/2024 | 7/6/2024 |
| Xem lại các tài liệu | Test plan | 1 ngày | 8/6/2024 | 8/6/2024 |
| Thiết kế các testcase | Test case | 1 ngày | 9/6/2024 | 9/6/2024 |
| Viết các testcase | Test case | 2 ngày | 10/6/2024 | 11/6/2024 |
| Xem lại các testcase | Test case | 1 ngày | 12/6/2024 | 12/6/2024 |
| Thực thi các testcase | Test case | 1 ngày | 13/6/2024 | 13/6/2024 |
| Ghi nhận và đánh giá kết quả kiểm thử | Test report | 1 ngày | 14/6/2024 | 14/6/2024 |

|  |  |  |  |
| --- | --- | --- | --- |
| Người kiểm thử | Tên kiểm thử | Thời gian | Công cụ |
| Bùi Trung Huỳnh | Kiểm thử chức năng đăng nhập | 13/6/2024 | Kiểm thử thủ công |
| Bùi Trung Huỳnh | Kiểm thử chức năng đăng ký tài khoản | 13/6/2024 | Kiểm thử thủ công |
| Chu Xuân Lộc | Kiểm thử chức năng nhập chi tiêu | 13/6/2024 | Kiểm thử thủ công |
| Chu Xuân Lộc | Kiểm thử chức năng xem thống kê | 13/6/2024 | Kiểm thử thủ công |
| Nông Minh Trường | Kiểm thử chức năng lập kế hoạch | 13/6/2024 | Kiểm thử thủ công |
| Ninh Duy Nhật | Kiểm thử chức năng đổi mật khẩu | 13/6/2024 | Kiểm thử thủ công |
| NôngMinh Trường | Kiểm thử chức năng nạp tiền sinh hoạt | 13/6/2024 | Kiểm thử thủ công |
| Nguyễn Minh Quang | Kiểm thử chức năng nạp tiền học phí | 13/6/2024 | Kiểm thử thủ công |
| Ninh Duy Nhật | Kiểm thử chức năng đăng kí môn học | 13/6/2024 | Kiểm thử thủ công |
| Nguyễn Minh Quang | Kiểm thử chức năng thanh toán công nợ | 13/6/2024 | Kiểm thử thủ công |

### 2.4.2 Kết quả kiểm thử

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Case | Đầu vào | Đầu ra mong muốn | Kết quả |
| 1 | Kiểm tra chức năng đăng nhập | Nhập tài khoản và mật khẩu hợp lệ | Đăng nhập thành công, chuyển trang với chế độ đã đăng nhập | Đạt |
| Nhập tài khoản và mật khẩu không hợp lệ | Thông báo đăng nhập không thành công | Đạt |
| 2 | Kiểm tra chức năng đăng ký | Nhập thông tin hợp lệ | Thông báo đăng ký thành công | Đạt |
| Nhập thông tin không hợp lệ | Nhập lại | Đạt |
| 3 | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất khỏi tài khoản, chuyển sang trang chủ | Đạt |
| 5 | Kiểm tra chức năng nhập chi tiêu | Nhập vào các hạng mục của khoản chi tiêu | Hiển thị nhập thành công | Đạt |
| 6 | Kiểm tra chức năng xem thống kê | Nhấn chọn mục xem thống kê theo ngày, tuần, tháng, năm | Hiển thị các khoản đã chi tiêu theo danh sách từng hạng mục | Đạt |
| 7 | Kiểm tra chức năng lập kế hoạch | Nhấn chọn lập kế hoạch | Hiển thị lập kế hoạch thành công và hiển thị kế hoạch đã lập lên màn hình | Đạt |
|  |  |  |  |  |
| 8 | Kiểm thử chức năng đổi mật khẩu | Nhấn đổi mật khẩu và nhập mật khẩu mới | Mật khẩu được thay đổi | Đạt |
| 9 | Kiểm thử chức năng nạp tiền sinh hoạt | Nhấn nạp tiền và nhập số tiền | Số tiền được tăng lên theo số tiền nạp vào | Đạt |
| 10 | Kiểm thử chức năng nạp tiền học phí | Nhấn nạp tiền và nhập số tiền | Số tiền được tăng lên theo số tiền nạp vào | Đạt |
| 11 | Kiểm thử chức năng đăng kí môn học | Chọn môn học và nhấn đăng kí | Môn học được chọn vào danh sách đăng kí | Đạt |
| 12 | Kiểm thử chức năng thanh toán công nợ | Ấn các khoản nợ và ấn thanh toán | Các khoản nợ chọn được thanh toán | Đạt |

- Tỉ lệ test case đạt (passed): 96%

- Tỉ lệ test case không đạt (failed): 4%

- Hệ thống chạy ổn định, đúng theo ý định ban đầu => Sản phẩm có thể sử dụng được

# 

# **PHẦN 3. KIẾN THỨC LĨNH HỘI VÀ BÀI HỌC KINH NGHIỆM**

## 3.1. Nội dung đã thực hiện

Tổng kết lại trong bài báo này này chúng em đã thực hiện phát triển một dự án công nghệ thông tin với đề tài “Ứng dụng quản lý tài khoản sinh viên Haui”. Các công nghệ chúng em sử dụng để phát triển dự án đa phần là các công nghệ quen thuộc và dễ dàng phát triển một ứng dụng di động Android như Android Studio, ngôn ngữ lập trình Java, dữ liệu được lưu trữ vào SQLite. Trong quá trình thực hiện, nhóm chúng em đã học hỏi và lĩnh hội được rất nhiều kiến thức và kỹ năng. Chúng em học được quy trình xây dựng một ứng dụng di động, cách thức hoạt động, các luồng logic và cách các thành phần trong một ứng dụng tương tác với nhau. Bên cạnh đó, chúng em học được cách sử dụng hiệu quả hơn Android Studio trong quá trình gặp lỗi, củng cố kiến thức về ngôn ngữ lập trình Java, tương tác với cơ sở dữ liệu SQLite, cũng như nâng cao kỹ năng lập trình của mình. Để hoàn thành được đề tài này, kỹ năng làm việc nhóm là vô cùng quan trọng. Kể từ khi dự án bắt đầu, nhóm chúng em đã tích cực cùng nhau lên kế hoạch, tổ chức họp và phân chia công việc một cách chuyên nghiệp. Chúng em đã sử dụng công cụ Git và GitHub để quản lý, lưu trữ và hạn chế lỗi trong quá trình viết code của cả nhóm. Ngoài những cuộc họp đã được lên kế hoạch, nhóm còn tổ chức những cuộc họp tức thì để giúp giải quyết các thắc mắc, vấn đề của các thành viên để đảm bảo dự án có thể hoạt động một cách trơn tru và đúng tiến độ. Trong quá trình thực, đã phát sinh những xung đột, những vấn đề không mong muốn, cả nhóm cùng nhau ngồi lại phân tích và đưa ra những giải pháp nhằm hướng đến một mục tiêu chung.

Một lần nữa, chúng em xin gửi lời cảm ơn chân thành tới cô Vũ Thị Dương đã hỗ trợ, hướng dẫn chúng em thực hiện đề tài này một cách thuận lợi.

Chúng em xin chân thành cảm ơn!

## 3.2. Hướng phát triển.

Ứng dụng quản lý tài khoản sinh viên là một chủ đề rất thiết thực trong cuộc sống cùng với quy mô ứng dụng trong trường đại học, nhóm chúng em mong muốn Ứng dụng quản lý tài khoản sinh viên Haui sẽ được biết đến nhiều hơn và tạo được nhiều giá trị hơn với mục tiêu là thương mại hóa được trên thị trường. Một số hướng phát triển của ứng dụng:

Nâng cấp ứng dụng mượt mà hơn.

Kiểm tra các trường hợp còn thiếu sót.

Nâng cao chế độ bảo mật cho ứng dụng.

Xây dựng thêm nhiều chức năng để hệ thống đầy đủ và toàn vẹn hơn.

Tìm hiểu xu hướng của người dùng để chỉnh sửa, bổ sung hệ thống phù hợp hơn.

Phát triển thêm để có thể phù hợp với nhiều trường đại học hơn, nhằm mở rộng quy mô dự án

# 

# **TÀI LIỆU THAM KHẢO**

Tài liệu:

1. Professional Android 2 Application Development 2010 (ebook).
2. Lê Hoàng Sử - Hồ Trung Thành – Trần Duy Thanh, Giáo trình phát triển ứng dụng di động cơ bản, NXB ĐHQG\_HCM, 2018.
3. Giáo trình phát triển ứng dụng cho thiết bị di động – ĐH Công nghiệp Hà Nội

Website:

1. <https://developer.android.com/>
2. <https://www.geeksforgeeks.org/>
3. <https://www.vogella.com/tutorials/Android/article.html>