TRƯỜNG ĐẠI HỌC THỦY LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated with low confidence

**BÁO CÁO MÔN HỌC**

PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG

**ĐỀ TÀI: APP QUẢN LÝ CHI TIÊU**

Giáo viên hướng dẫn: Phạm Huy Thông

Sinh viên thực hiện: Nguyễn Hoàng Sơn - 2051063870

Nguyễn Thành Hưng - 2051063589

Trần Thanh Thảo - 2051063511

Trần Việt Phương - 2051063533

Vũ Thanh Sơn - 2051063469

Lớp học phần: 62TH-NB

Nhóm thực hiện: Nhóm 9

**Hà Nội, tháng 6 năm 2023**

**MỤC LỤC**

[**GIỚI THIỆU** 3](#_Toc138885005)

[**CHƯƠNG I: TỔNG QUAN** 4](#_Toc138885006)

[1. Tổng quan về Android 4](#_Toc138885007)

[1.1. Android là gì? 4](#_Toc138885008)

[1.2. Đặc tính mở của android 4](#_Toc138885009)

[1.3.Kiến trúc của Android 4](#_Toc138885010)

[2.Ngôn ngữ lập trình và môi trường lập trình Android 6](#_Toc138885011)

[3. Tìm hiểu về công cụ lập trình trên Android Studio 7](#_Toc138885012)

[3.1. Bộ phát triển ngôn ngữ Java : Java Development Kit (JDK) 7](#_Toc138885013)

[3.2. Bộ Công cụ phát triển phần mềm : Software Development Kit (SDK): 8](#_Toc138885014)

[**CHƯƠNG II: MÃ NGUỒN VÀ MÔ PHỎNG CHƯƠNG TRÌNH** 10](#_Toc138885015)

[1. Giới thiệu về chương trình 10](#_Toc138885016)

[2. Biểu đồ usecase 10](#_Toc138885017)

[3. Xây dựng đặc tả 11](#_Toc138885018)

[4. Thiết kế cơ sở dữ liệu 19](#_Toc138885019)

[5. Giao diện 20](#_Toc138885020)

[**Chương 3: Kết luận** 24](#_Toc138885021)

[**3.1 Kết luận** 24](#_Toc138885022)

[3.2. Hướng phát triển 24](#_Toc138885023)

[Tài liệu tham khảo 24](#_Toc138885024)

# **GIỚI THIỆU**

Trong thế giới ngày nay, quản lý chi tiêu cá nhân là một nhiệm vụ quan trọng để đảm bảo tài chính cá nhân ổn định và đạt được mục tiêu tài chính. Tuy nhiên, việc theo dõi và kiểm soát chi tiêu hàng ngày có thể trở nên phức tạp và đòi hỏi sự tỉ mỉ và đầu tư thời gian. May mắn thay, sự phát triển của công nghệ đã mang lại những giải pháp hiệu quả và tiện lợi cho việc quản lý chi tiêu cá nhân thông qua các ứng dụng di động.

Trong báo cáo này, nhóm chúng em xin giới thiệu một ứng dụng quản lý chi tiêu tuyệt vời có tên "Quản lý chi tiêu cá nhân". Ứng dụng này không chỉ giúp người dùng theo dõi chi tiêu hàng ngày một cách thuận tiện, mà còn cung cấp nhiều tính năng hữu ích để phân loại, phân tích và quản lý ngân sách cá nhân. Với giao diện người dùng thân thiện và tính năng linh hoạt, Quản lý chi tiêu cá nhân đã trở thành một công cụ quan trọng để giúp người dùng duy trì sự cân đối tài chính và đạt được mục tiêu tài chính cá nhân.

Trong phần tiếp theo của báo cáo, nhóm chúng em sẽ trình bày chi tiết về tính năng và lợi ích của ứng dụng, bao gồm việc theo dõi chi tiêu hàng ngày, quản lý khoản chi/thu, thống kê. Ngoài ra cũng sẽ nhấn mạnh về sự dễ sử dụng và hiệu quả của ứng dụng này trong việc quản lý chi tiêu cá nhân.

**Phân chia công việc**

|  |  |
| --- | --- |
| Tên thành viên | Công việc |
| Trần Thanh Thảo | - Chức năng đăng kí, đăng nhập, đăng xuất  - Đặc tả hệ thống |
| Nguyễn Thành Hưng | - Thêm, sửa, xóa khoản thu  - Vẽ biểu đồ usecase |
| Trần Việt Phương | - Thêm, sửa, xóa khoản chi  - Thiết kế CSDL |
| Nguyễn Hoàng Sơn | Chức năng thống kê |
| Vũ Thanh Sơn | Thêm, sửa, xóa loại thu và loại chi |

# **CHƯƠNG I: TỔNG QUAN**

## **1. Tổng quan về Android**

### **1.1. Android là gì?**

Android là hệ điều hành trên điện thoại di động (và hiện nay là cả trên một số đầu phát HD, HD Player, TV) dựa trên nền tảng Linux kernel và các phần mềm mã nguồn mở. Lập trình android là một lập trình ứng dụng di động phổ biến. Trước đây, Android được phát triển bởi công ty liên hợp Android (sau đó được Google mua lại vào năm 2005).

Android là hệ điều hành mở mã nguồn chính duy nhất với 12 triệu dòng mã bao gồm 3 triệu dòng mã XML, 2,8 triệu mã C, 2,1 triệu dòng Java và 1,75 triệu dòng C++.

Các nhà phát triển viết ứng dụng cho Android dựa trên ngôn ngữ Java. Sự ra mắt của Android vào ngày 5 tháng 11 năm 2007 gắn với sự thành lập của liên minh thiết bị cầm tay mã nguồn mở, bao gồm 78 công ty phần cứng, phần mềm và viễn thông nhằm mục đích tạo nên một chuẩn mở cho điện thoại di động trong tương lai.

* Được xây dựng trên nền tảng của mã nguồn mở.
* Không chỉ là hệ điều hành cho các thiết bị di động mà còn là HĐH cho cả các thiết bị khác có sử dụng Internet.
* Là HĐH dễ sử dụng và cho phép người dùng dễ dàng publish các ứng dụng.

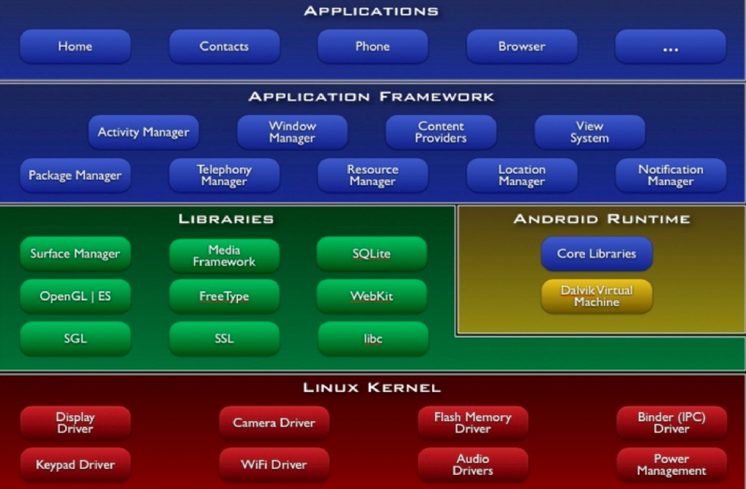
### **1.2. Đặc tính mở của android**

Android được xây dựng để cho phép các nhà phát triển tạo ra các ứng dụng di động hấp dẫn, tận dụng tất cả các tính năng của một chiếc điện thoại đã cung cấp. Android được xây dựng trên mã nguồn mở Linux Kernel. Hơn nữa, nó sử dụng một máy ảo tùy chỉnh được thiết kế để tối ưu hóa bộ nhớ và tài nguyên phần cứng trong môi trường di động.

Android cung cấp truy cập đến một loạt các thư viện công cụ hữu ích và có thể sử dụng để xây dựng các ứng dụng phong phú. Ngoài ra, Android bao gồm một tập hợp đầy đủ công cụ đã được xây dựng công phu, với việc cung cấp nền tảng phát triển, với năng suất cao và cái nhìn sâu vào các ứng dụng.

### **1.3.Kiến trúc của Android**

Sơ đồ kiến trúc của hệ điều hành Android với năm lớp và bốn tầng cơ bản như sau:



**Tầng Applications**: Đây là tầng mà được tích hợp sẵn một số ứng dụng cần thiết cơ bản như: contact, browser, camera, Phone,… Tất cả các ứng dụng chạy trên hệ điều hành Android đều được viết bằng Java.

**Tầng Application Framework**: Lớp Android Framework cung cấp các dịch vụ cấp độ cao hơn cho các ứng dụng dưới dạng các lớp Java. Các nhà phát triển ứng dụng được phép sử dụng các dịch vụ này trong ứng dụng của họ.

**Android Framework** bao gồm các dịch vụ chính sau:

Activity Manager – Kiểm soát tất cả khía cạnh của vòng đời ứng dụng và ngăn xếp các Activity.

Content Providers – Cho phép các ứng dụng chia sẻ dữ liệu với các ứng dụng khác.

Resource Manager – Cung cấp quyền truy cập vào các tài nguyên như các chuỗi, màu sắc, các layout giao diện người dùng…

Notifications Manager – Cho phép các ứng dụng hiển thị cảnh báo và các thông báo cho người dùng.

View System – Tập các thành phần giao diện (view) được sử dụng để tạo giao diện người dùng.

**Tầng Libraries**: Chứa các thư viện, API gần như là cốt lõi của Android, bao gồm bộ quản lý bề mặt cảm ứng (Surface Manager), OpenGL (phục vụ cho việc dựng đồ họa phức tạp),…

**Tầng Android Runtime**: Chứa các thư viện lõi của Android và máy ảo Dalvik Virtual Machine (từ Android 4 trở lên chúng ta có thêm máy ảo ART).

**Tầng Kernel:** Là nhân lõi của hệ điều hành, chứa các tập lệnh, driver giao tiếp giữa phần cứng và phần mềm của Android.

## **2.Ngôn ngữ lập trình và môi trường lập trình Android**

**● Ngôn ngữ lập trình:**

Một trong những ngôn ngữ lập trình tiêu biểu nhất của lập trình android là Java. Nó là ngôn ngữ lập trình chính thức được sử dụng rộng rãi nhất trên thế giới có ước tính khoảng 9 triệu nhà phát triển.

Java được thiết kế để tương thích với nhiều môi trường phát triển vì thế mà nó linh hoạt hơn so với ngôn ngữ lập trình hướng đối tượng C++ và ngôn ngữ lập trình C. Bên cạnh đó, Java còn có hiệu suất cao nhờ vào trình thu gom rác, trình giải phóng bộ nhớ với các đối tượng không được dùng đến. Ngoài ra, Java nâng cao còn có thể viết ra các chương trình thực thi nhiều các tác vụ cùng một lúc với tính năng đa luồng của mình, đặc biệt đó là ở trong lập trình game. Đặc biệt, ngôn ngữ lập trình Java còn hỗ trợ bảo mật rất tốt bởi các thuật toán mã hóa như public key hoặc mã one way hashing…

Bên cạnh ngoài ngôn ngữ Java, có thể kể đến một số ngôn ngữ lập trình android phổ biến hiện nay thường gặp như: C++, C#, Kotlin, HTML5, CSS, JavaScript, Python, Lua,…

**● Môi trường lập trình Android:**

Môi trường lập trình (IDE) chính thức của Android là Eclipse (từ phiên bản 3.2) với sự hỗ trợ của plugin Android Development Tools (ADT) trước đây. Tuy nhiên, người lập trình có thể sử dụng bất kỳ 1 IDE hay trình soạn thảo văn bản nào để viết code Java và XML rồi biên dịch nên ứng dụng hoàn chỉnh bằng cách sử dụng dòng lệnh (command lines).

Android Developer Tools Bundle (ADT) bao gồm:

* Eclipse + ADT plugin.
* Android SDK Tools.
* Android Platform Tools.
* Phiên bản hệ điều hành Android.
* Tập tin cài đặt hệ điều hành cho máy ảo.

Ứng dụng Android được đóng gói thành các file .apk và được lưu trong thư mục /data/app của hệ điều hành Android.Java Development Kit (JDK) 5.0.

**● Ứng dụng lập trình Android và các thành phần liên quan**

Ứng dụng lập trình Android là các ứng dụng chỉ có thể được viết bằng một vài ngôn ngữ đặc trưng, có thể kể đến như C++, C#, Javascript, LUA, Python,…dành riêng cho nền tảng Android của thiết bị di động. Sau khi viết code xong và chạy biên dịch chương trình, bộ công cụ SDK Android sẽ biên dịch mã của chương trình - cùng với bất kỳ tệp dữ liệu và tài nguyên nào—vào một APK: một gói Android, đó là một tệp lưu trữ có hậu tố .apk. Một tệp APK chứa tất cả nội dung của một ứng dụng Android và là tệp mà các thiết bị dựa trên nền tảng Android sử dụng để cài đặt ứng dụng.

**● Các thành phần (component) cơ bản tạo nên một ứng dụng Android như sau:**

- Activity: là nền của 1 ứng dụng. Khi khởi động 1 ứng dụng Android nào đó thì bao giờ cũng có 1 mainActivity được gọi, hiển thị màn hình giao diện của ứng dụng cho phép người dùng tương tác.

- Service: là thành phần chạy ẩn trong Android. Service sử dụng để update dữ liệu, đưa ra các cảnh báo (Notification) và không bao giờ hiển thị cho người dùng thấy.

· Content Provider: được sử dụng để quản lý và chia sẻ dữ liệu giữa các ứng dụng.

· Intent: là nền tảng để truyền tải các thông báo. Intent được sử dụng để gửi các thông báo đi nhằm khởi tạo 1 Activity hay Service để thực hiện công việc mong muốn.

· Broadcast Receiver: thành phần thu nhận các Intent bên ngoài gửi tới.

· Notification: đưa ra các cảnh báo mà không làm cho các Activity phải ngừng hoạt động.

## **3. Tìm hiểu về công cụ lập trình trên Android Studio**

### **3.1. Bộ phát triển ngôn ngữ Java : Java Development Kit (JDK)**

JDK viết tắt của Java Development Kit là một bộ phần mềm cung cấp môi trường phát triển ứng dụng viết bằng ngôn ngữ Java. JDK dùng để viết những applet Java hay những ứng dụng Java hoặc ứng dụng Android. Bộ công cụ này được phát hành miễn phí gồm có trình biên dịch, trình thông dịch, trình giúp sửa lỗi (debugger, trình chạy applet) và tài liệu nghiên cứu. Nó cũng là một trong ba gói công nghệ cốt lõi được sử dụng trong lập trình Java, cùng với JVM (Máy ảo Java - Java Virtual Machine) và JRE (Java Runtime Environment - Môi trường Java Runtime) được sử dụng phổ biến trong lập trình Java.

Có thể hiểu về JDK theo 2 cách như sau:

Định nghĩa chuyên ngành: JDK là một hệ tiêu chuẩn trong việc triển khai nền tảng Java, bao gồm các trình thông dịch và thư viện lớp.

Định nghĩa thông thường: JDK là gói phần mềm bạn tải xuống để tạo các ứng dụng dựa trên Java.

Trình biên dịch JDK & Java: Môi trường được sử dụng để chạy các Source code Java, mỗi JDK đều chứa trình biên dịch Java. Trình biên dịch là chương trình phần mềm có khả năng lấy các tệp .java ở dạng thô - văn bản thuần túy - và hiển thị chúng thành các tệp thực thi có đuôi .Class

### **3.2. Bộ Công cụ phát triển phần mềm : Software Development Kit (SDK):**

SDK là một tập hợp công cụ hỗ trợ cho việc phát triển phần mềm thông qua một nền tảng nào đó. Hầu hết các nhà phát triển ứng dụng đều cần dùng tới SDK để thêm vào các tính năng nâng cao, hiển thị quảng cáo, thông báo dạng đẩy (push notification) cho ứng dụng. Chẳng hạn lập trình Android yêu cầu SDK cho Java, với ứng dụng iOS thì SDK cho Swift, còn lập trình ứng dụng Windows yêu cầu có .NET Framework SDK đi kèm với .NET.

SDK có thể chứa các API dưới dạng thư viện để giao tiếp với một ngôn ngữ nào đó hoặc để chứa một hệ thống phần cứng phức tạp có thể giao tiếp với một hệ thống nhúng. Các công cụ phổ biến bao gồm các tiện ích gỡ rối phần mềm (debugging), thường ở dạng môi trường phát triển tích hợp (IDE). SDK cũng thường hay kèm theo các sample code và các ghi chú kỹ thuật hỗ trợ (documentation) hay tài liệu để giúp làm rõ các tài liệu tham khảo chính.

Về phân loại, SDK có ba phân loại chính như sau:

* Windows SDK: bộ công cụ phát triển ứng dụng chạy trên nền tảng Windows.
* Android SDK: bộ công cụ phát triển ứng dụng trên nền tảng android.
* Java SDK: hay nói cách khác là JDK là một tập con mở rộng của SDK hay nói cách khác JDK là SDK for Java hoặc Java SDK. Bộ công cụ này bao gồm các tool hữu ích để xây dựng và phát triển các Applications, Applets và Components viết bằng ngôn ngữ lập trình Java (không dùng ngôn ngữ khác như C, C++, v.v…).

Về lợi ích của việc sử dụng SDK: SDK mang lại nhiều lợi ích với tính năng khác nhau. Người dùng khi sử dụng có thể có được các lợi ích tích cực, đem lại trải nghiệm thú vị cho người dùng. Việc sử dụng SDK sẽ giúp cung cấp một cách chính xác các ứng dụng, giúp người dùng có thể thực hiện tích hợp các công cụ một cách nhanh chóng và dễ dàng. Từ đó tăng cường độ chính xác giúp thực hiện dữ liệu, giúp triển khai và nâng cấp đường dẫn tốt hơn, đồng thời thực hiện loại bỏ đi các thông tin dư thừa, không cần thiết.

Ngoài ra cũng có một số công cụ hỗ trợ lập trình Android tiêu biểu:

* SQLite Manager: Là một addon của Firefox giúp quản lý cơ sở dữ liệu SQLite của Android.
* DroidDraw: Giúp thiết kế file XML giao diện ứng dụng.
* Balsamiq Mockups và AdobeFireworks: Giúp nhanh chóng phác thảo ý tưởng và giao diện sơ bộ của ứng dụng.
* StarUML: Vẽ các lược đồ UML hỗ trợ phân tích thiết kế.

# **CHƯƠNG II: MÃ NGUỒN VÀ MÔ PHỎNG CHƯƠNG TRÌNH**

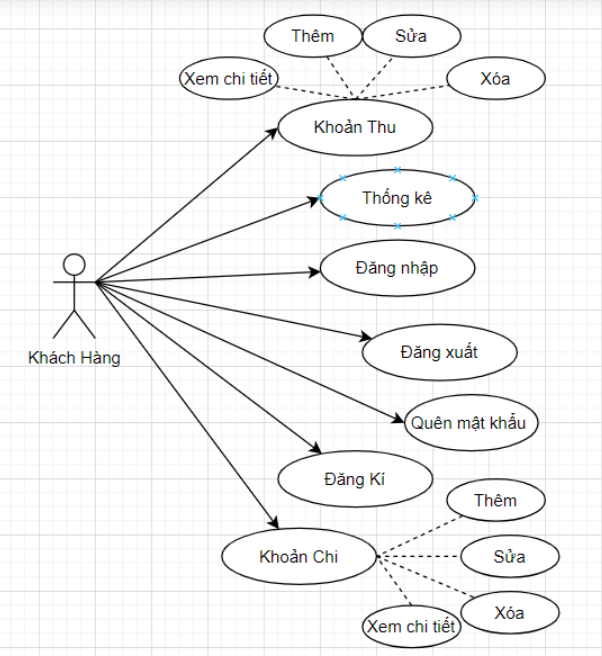
## **1. Giới thiệu về chương trình**

- App quản lý chi tiêu sẽ có 5 chức năng chính:

* Đăng ký, đăng nhập, đăng xuất
* Nhập khoản thu
* Nhập khoản chi
* Nhập loại thu, loại chi
* Thống kê khoản thu, khoản chi

- Tác nhân chính: Người dùng

## **2. Biểu đồ usecase**



### **3. Xây dựng đặc tả**

- Đặc tả Đăng ký

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Đăng ký |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng đăng ký tài khoản |
| Tiền điều kiện | Người dùng chưa có tài khoản |
| Đảm bảo thành công | Người dùng đăng kí tài khoản thành công |
| Điều kiện thất bại | Người dùng đăng kí không thành công |
| Kích hoạt | Khi người dùng nhấn nút ‘đăng ký’ |
| Chuỗi sự kiện chính | 1. Người dùng mở app, nhấn vào nút ‘Tạo mới tài khoản’ 2. Người dùng nhập ‘tên tài khoản’ và ‘mật khẩu’ hợp lệ 3. Người dùng nhấn nút ‘đăng ký’ 4. Hệ thống thông báo đăng ký thành công |

- Đặc tả Đăng nhập

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Đăngnhập |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng đăng nhập vào hệ thống |
| Tiền điều kiện | Người dùng đã có tài khoản |
| Đảm bảo thành công | Người dùng đăng nhập thành công |
| Điều kiện thất bại | Người dùng đăng nhập thất bại |
| Kích hoạt | Khi người dùng nhấn nút ‘đăng nhập’ |
| Chuỗi sự kiện chính | 1. Người dùng mở app 2. Người dùng nhập ‘tên tài khoản’ và ‘mật khẩu’ hợp lệ 3. Người dùng nhấn nút ‘đăng nhập’ 4. Hệ thống thông báo đăng nhập thành công 5. Hệ thống chuyển đến trang chủ app |
| Ngoại lệ | 1. Hệ thống xác minh tài khoản sai 2. Hệ thống thông báo lỗi 3. Hệ thống xác minh tài khoản trống 4. Hệ thống thông báo lỗi 5. Hệ thống xác minh tài khoản không hợp lệ (không đúng định dạng ) 6. Hệ thống hiển thị lỗi |

- Đặc tả đăng xuất

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Đăng xuất |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng đăng xuất khỏi tài khoản |
| Tiền điều kiện | Người dùng đã đăng nhập |
| Đảm bảo thành công | Người dùng đăng xuất tài khoản thành công |
| Điều kiện thất bại | Người dùng đăng xuất thất bại |
| Kích hoạt | Khi người dùng ấn nút thoát |
| Chuỗi sự kiện chính | 1. Người dùng chọn nút ‘thoát’ 2. Hệ thống chuyển về form đăng nhập |

- Đặc tả Quên mật khẩu

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Quên mật khẩu |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng cấp lại mật khẩu mới |
| Tiền điều kiện | Người dùng đã có tài khoản |
| Đảm bảo thành công | Người dùng cập nhật mật khẩu thành công |
| Điều kiện thất bại | Người dùng cập nhật thất bại |
| Kích hoạt | Khi người dùng nhấn nút ‘quên mật khẩu’ |
| Chuỗi sự kiện chính | 1. Người dùng mở app 2. Người dùng nhập ‘tên tài khoản’, ‘mật khẩu cũ’, ‘xác nhật mật khẩu’ 3. Người dùng nhấn nút ‘xác nhận’ 4. Hệ thống thông báo cập nhật mật khẩu thành công 5. Hệ thống chuyển đến trang đăng nhập |

- Đặc tả thêm loại chi / loại thu

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Thêm loại chi – loại thu |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng thêm mới loại chi/loại thu vào hệ thống |
| Tiền điều kiện | Người dùng đã đăng nhập |
| Đảm bảo thành công | Người dùng thêm mới thành công |
| Điều kiện thất bại | Thông báo thêm thất bại |
| Kích hoạt | Khi người dùng nhấn nút ‘Thêm’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Loại chi/ Loại trong mục khoản chi / khoản thu 2. Nhấn icon thêm mới 3. Hệ thống hiển thị form thêm 4. Nhập thông tin loại chi 5. Nhấn nút ‘thêm’ 6. Hệ thống thông báo thành công |
| Ngoại lệ | 1. Người dùng ấn thêm 2. Người dùng không điền thông tin |

- Đặc tả xóa loại thu / loại chi

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Xóa loại thu / loại chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng xóa khoản thu/ khoản chi |
| Tiền điều kiện | 1. Người dùng đã đăng nhập 2. Người dùng đã tạo loại thu / loại chi 3. Người dùng đã tạo khoản thu/ khoản chi |
| Đảm bảo thành công | Hệ thống xóa thành công |
| Điều kiện thất bại | Thông báo xóa thất bại |
| Kích hoạt | Khi người dùng nhấn vào icon ‘xóa’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Loại thu / Loại chi 2. Nhấn vào icon ‘xóa’ 3. Hệ thống thông báo xóa thành công |

- Đặc tả thêm mới Khoản thu / Khoản chi

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Thêm khoản thu / khoản chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng thêm mới khoản thu/ khoản chi vào hệ thống |
| Tiền điều kiện | * + - 1. Người dùng đã đăng nhập       2. Người dùng đã tạo loại thu / loại chi |
| Đảm bảo thành công | Người dùng thêm mới thành công |
| Điều kiện thất bại | Thông báo thêm thất bại |
| Kích hoạt | Khi người dùng nhấn nút ‘Thêm’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Khoản thu / Khoản chi 2. Nhấn icon thêm mới 3. Hệ thống hiển thị form thêm 4. Nhập thông tin khoản thu / khoản chi 5. Nhấn nút ‘thêm’ 6. Hệ thống thông báo thành công |
| Ngoại lệ | 1. Người dùng ấn thêm 2. Người dùng không điền thông tin |

- Đặc tả sửa khoản thu / khoản chi

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Sửa khoản thu / khoản chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng sửa khoản thu/ khoản chi vào hệ thống |
| Tiền điều kiện | 1. Người dùng đã đăng nhập 2. Người dùng đã tạo loại thu / loại chi 3. Người dùng đã tạo khoản thu/ khoản chi |
| Đảm bảo thành công | Người dùng sửa thành công |
| Điều kiện thất bại | Thông báo sửa thất bại |
| Kích hoạt | Khi người dùng nhấn icon ‘Sửa’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Khoản thu / Khoản chi 2. Nhấn icon sửa 3. Hệ thống hiển thị form sửa 4. Cập nhật thông tin khoản thu / khoản chi 5. Nhấn nút ‘cập nhật’ 6. Hệ thống thông báo thành công |

- Đặc tả xem chi tiết Khoản thu / Khoản chi

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Xem chi tiết khoản thu / khoản chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng xem khoản thu/ khoản chi |
| Tiền điều kiện | 1. Người dùng đã đăng nhập 2. Người dùng đã tạo loại thu / loại chi 3. Người dùng đã tạo khoản thu/ khoản chi |
| Đảm bảo thành công | Hệ thống hiển thị chi tiết khoản thu/ khoản chi |
| Điều kiện thất bại |  |
| Kích hoạt | Khi người dùng nhấn vào thông tin đã thêm |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Khoản thu / Khoản chi 2. Nhấn vào thông tin 3. Hệ thống hiển thị form chi tiết |

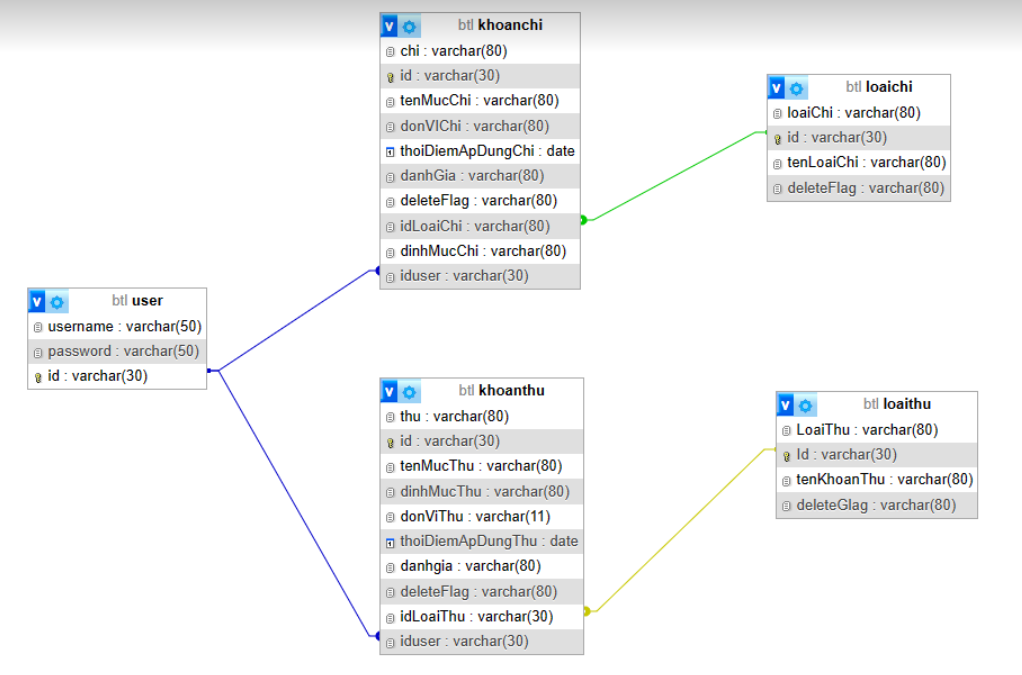
- Đặc tả xóa Khoản thu / Khoản chi

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Xóa khoản thu / khoản chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng xóa khoản thu/ khoản chi |
| Tiền điều kiện | 1. Người dùng đã đăng nhập 2. Người dùng đã tạo loại thu / loại chi 3. Người dùng đã tạo khoản thu/ khoản chi |
| Đảm bảo thành công | Hệ thống xóa thành công |
| Điều kiện thất bại | Thông báo xóa thất bại |
| Kích hoạt | Khi người dùng nhấn vào icon ‘xóa’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Khoản thu / Khoản chi 2. Nhấn vào icon ‘xóa’ 3. Hệ thống thông báo xóa thành công |

- Đặc tả thống kê

|  |  |
| --- | --- |
| **Tiêu chí** | **Ý Nghĩa** |
| Tên Use case | Thống kê khoản thu và khoản chi |
| Tác nhân chính | Khách hàng |
| Mức | 0 |
| Mục đích | Cho phép người dùng hiển thị tổng thu và tổng chi |
| Tiền điều kiện | 1. Người dùng đã đăng nhập 2. Người dùng đã tạo loại thu / loại chi 3. Người dùng đã tạo khoản thu/ khoản chi |
| Đảm bảo thành công | Hệ thống hiển thị tổng khoản thu/ khoản chi |
| Điều kiện thất bại |  |
| Kích hoạt | Khi người dùng nhấn vào tab ‘Thống kê’ |
| Chuỗi sự kiện chính | 1. Người dùng chọn tab Thống kê 2. Hệ thống hiển thị tổng chi và tổng thu theo ngày / tháng/ năm |

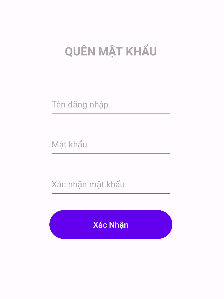
## **4. Thiết kế cơ sở dữ liệu**



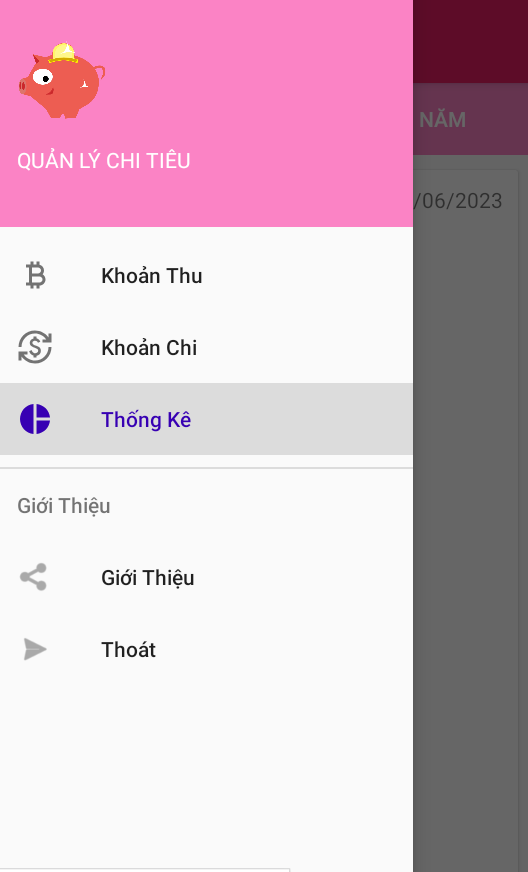
## **5. Giao diện**

|  |  |
| --- | --- |
| Đăng nhập | Đăng ký |

- Quên mật khẩu

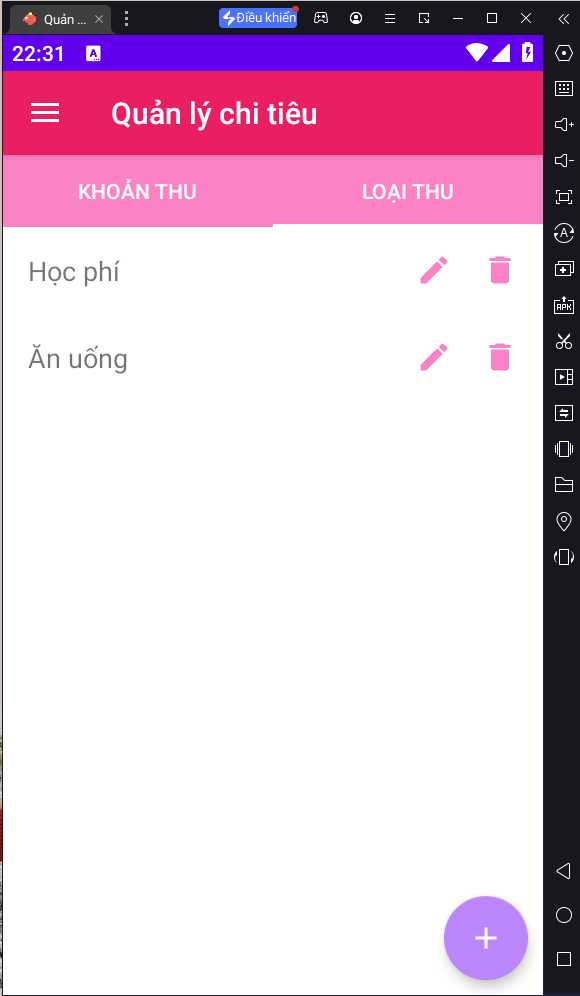


- Trang chủ



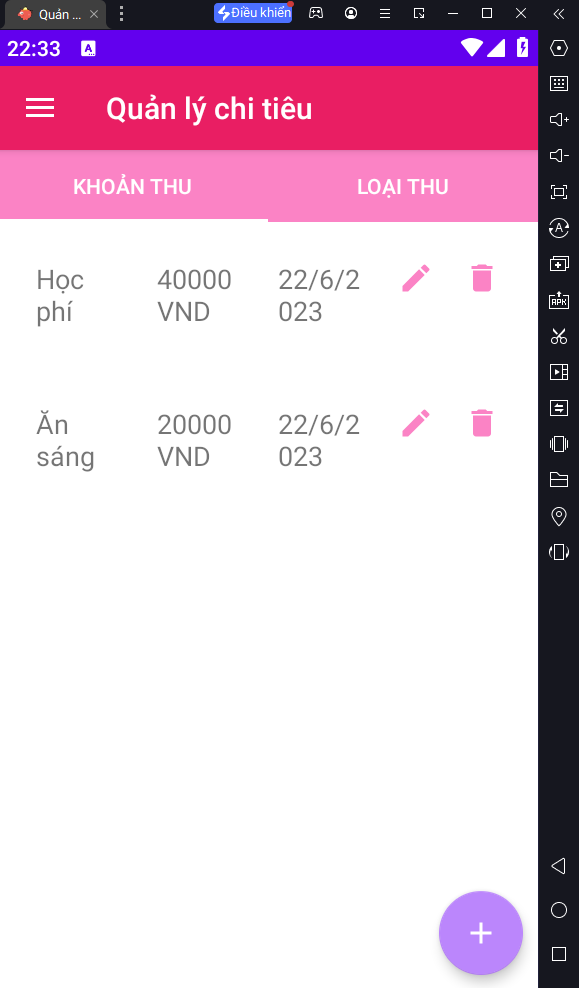
- Khoản thu:

+ Tab Loại thu



|  |  |
| --- | --- |
|  |  |

+ Tab Khoản thu:



|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

# **Chương 3: Kết luận**

# **3.1 Kết luận**

Sau thời gian học tập và xây dựng ứng dụng, nhóm em đã đạt đƣợc kết quả sau:

- Nắm được quy trình để xây dụng ứng dụng android, từ bước khảo sát tới thiết kế và xây dụng ứng dụng hoàn chỉnh.

- Tìm hiểu, sử dụng các công cụ cần thiết để xây dựng ứng dụng cửa hàng bán đồ ăn

- Xây dựng được ứng dụng hoàn thành các mục tiêu sau:

* Xây dựng chức năng đăng kí, đăng nhập/ đăng xuất cho người dùng
* Xây đựng được chức năng Thêm, sửa, xóa các khoản thu chi
* Xây dựng được chức năng Thống kê

- Mặc dù đã có nhiều cố gắng, tuy nhiêm trong quá trình thực hiện, do kiến thực còn chưa vững, chưa có nhiều kinh nghiệm nên không thể tránh được thiếu sót, cũng như thời gian không cho phép nên một số chức năng của nhóm còn chưa thực hiện được. Vì vậy, nhóm em rất mong nhận được sự chỉ bảo góp ý từ cô giáo để ứng dụng của chúng em đƣợc hoàn thiện hơn. Nhóm em xin chân thành cảm ơn thầy nhiều.

## **3.2. Hướng phát triển**

Sử dụng nhiều chức năng hiện đại hơn để xây dựng các ứng dụng hỗ trợ ngƣời dùng tiện ích. Tiếp tục hoàn thiện các chức năng khác còn chưa xong, thêm mới các chức năng trong tương lai để ứng dựng sớm đi vào thực tế.

# Tài liệu tham khảo

* Slide bài giảng
* <https://www.youtube.com/watch?v=dKX2V992pWI>
* <https://viblo.asia/p/ban-biet-gi-ve-intent-trong-android-Do7544AB5M6>
* <https://viblo.asia/p/recycleview-la-gi-so-sanh-voi-listview-WAyK8xWoKxX>
* <https://developer.android.com/studio/write/firebase>