

# Chương 1.

## TỔNG QUAN ĐỀ TÀI

### 1. Giới Thiệu

#### Giới thiệu về đề tài

Đối với quá trình hoàn thiện sản phẩm CNTT trong quá trình thực tập, cụ thể là ứng dụng quản lý cửa hàng tạp hóa đã giúp em có những khái niệm cơ bản về công việc quản lý cơ bản trong cửa hàng. Kết hợp xây dựng phần mềm với việc ôn tập để làm khóa luận tốt nghiệp, bổ sung những kiến thức đã có và tìm hiểu thêm những kiến thức mới thông qua ngôn ngữ lập trình Java và PHP.

App bán hàng MEGA Shop. Đây là sàn thương mại điện tử cho phép người bán đăng tải bán và quản lý các sản phẩm miễn phí. Điểm nổi bật của MEGA Shop là app có tính tương tác cao. Người mua và người bán có thể trao đổi với nhau qua ứng dụng. Cùng với đó là nhiều mã giảm giá, chương trình hấp dẫn cho cả người bán và người mua.

### 2. Mục Tiêu

Thiết kế và xây dựng ứng dụng quản lý cửa hàng với các mục tiêu sau :

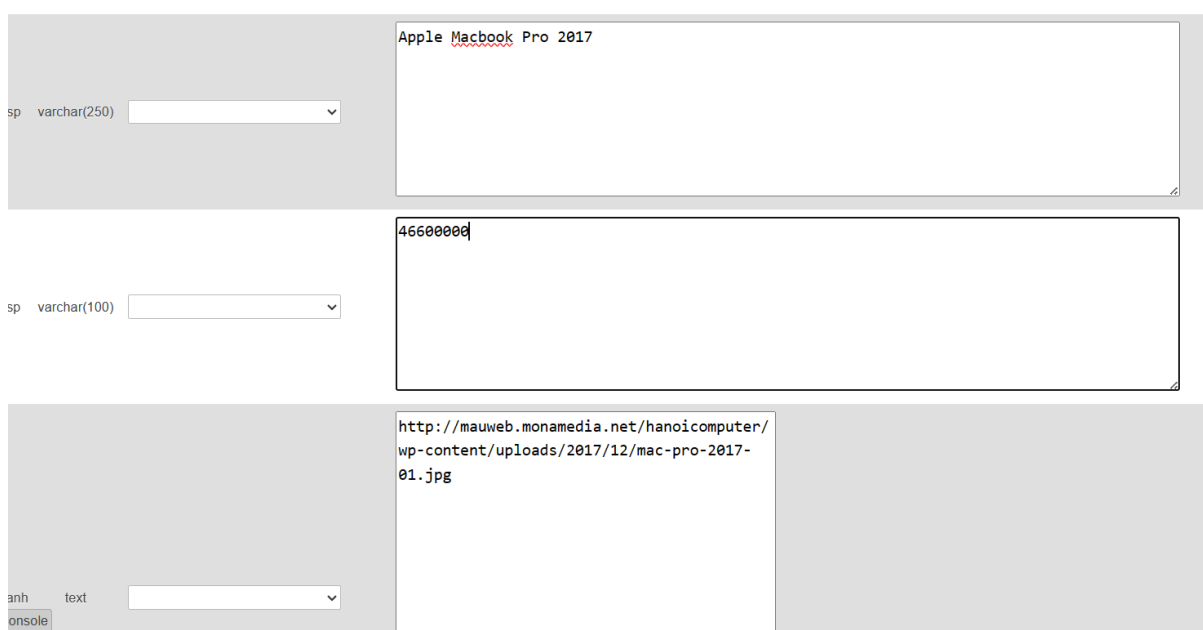
- Phục vụ cho người bán hàng và khách hàng.
- Cải thiện chất lượng dịch vụ khách hàng.
- Có thể xây dựng và quản lý hệ thống CTV với chi phí 0 đồng.
- Điều tiết doanh thu cho doanh nghiệp.
- Kết nối với người tiêu dùng dễ dàng hơn và tạo khả năng trung thành của người dùng với doanh nghiệp.
- Tăng doanh thu bán sản phẩm, dịch vụ.
- Gây dựng lòng tin nơi khách hàng.
- Tăng cường giá trị thương hiệu của doanh nghiệp.
- Tạo điều kiện để khách hàng tiếp cận sản phẩm gần hơn.
- Tăng khả năng truy cập từ lượng khách hàng tiềm năng.
- Tăng khả năng bán hàng trực tuyến.
- Có thêm cơ hội trưng bày sản phẩm trên các ứng dụng di động.
- Liên kết với nhiều khách hàng tiềm năng.
- Dễ dàng tiếp cận tới sản phẩm mọi lúc mọi nơi.

- Nhận được những thông báo về những sự kiện đặc biệt, khuyến mãi, v..v
- Dễ dàng liên hệ chỉ qua 1 bước.
- Không cần phải tốn thời gian đến cửa hàng.
- Có nhiều sự lựa chọn về sản phẩm hơn.
- Dễ tìm được những mặt hàng hiếm.
- Muốn tránh xa các đám đông.
- Không muốn xếp hàng chờ tính tiền.

### 3. Hướng Tiếp Cận

Thực hiện tham gia khảo sát hiện trạng các cửa hàng lớn, nhỏ lẻ tại khu vực thành phố Hồ Chí Minh.

Nguyên cứu hệ điều hành Android, ngôn ngữ lập trình Java, Framework : Java Spring Boot, PHP, dịch vụ Firebase - cơ sở dữ liệu thời gian thực. Xây dựng hướng dịch vụ tận dụng ưu điểm của mạng Internet để tối ưu hóa công việc.



<http://mauweb.monamedia.net/hanoicomputer/wp-content/uploads/2017/12/mac-pro-2017-01.jpg>

## Chương 2.

### CƠ SỞ LÝ THUYẾT

#### 1. Giới thiệu sơ lược về Android, ngôn ngữ lập trình Java

**Android** là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. ...

Chiếc điện thoại đầu tiên chạy **Android** được bán vào năm 2008. **Android** có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache.

Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ GooglePlay.

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "cuộc chiến điện thoại thông minh" giữa các công ty công nghệ.

**Java** có thể xem là ngôn ngữ lập trình phổ biến nhất hiện nay, được sử dụng rộng rãi trên nhiều thiết bị và hệ điều hành. Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun Microsystems.

Java được tạo ra với tiêu chí "*Viết (code) một lần, thực thi khắp nơi*" (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

Và dĩ nhiên, ứng dụng Android cũng được viết dựa trên ngôn ngữ phổ biến nhất này. Tuy từ giữa năm 2017 Google có nói rằng họ sẽ thay thế Java bằng ngôn ngữ mới

toanh có tên *Kotlin*, nhưng theo những gì mình biết, Java, cho đến giờ phút này, vẫn đang là ngôn ngữ thống trị trong các ứng dụng Android.

## 2. Visual Studio Code

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn mở gọn nhẹ nhưng có khả năng vận hành mạnh mẽ trên 3 nền tảng là Windows, Linux và macOS được phát triển bởi Microsoft. Nó hỗ trợ cho JavaScript, Node.js và TypeScript, cũng như cung cấp một hệ sinh thái mở rộng vô cùng phong phú cho nhiều ngôn ngữ lập trình khác.

S Code là sự kết hợp giữa một trình soạn thảo mã nguồn với những công cụ phát triển mạnh mẽ như Git, Debug hay Syntax Highlighter.

VS Code được thiết kế để có thể chạy tốt trên macOS, Linux và Windows. Vì thế, bạn có thể thoải mái sử dụng trình soạn thảo này mà không cần lo lắng về vấn đề lựa chọn nền tảng tương thích.

Tốc độ là một trong những ưu thế vượt trội của VS Code. Nó có thể hỗ trợ rất nhiều các loại ngôn ngữ lập trình khác nhau như Java, Python, C++, C#, Golang hay PHP.

Điều này cho phép bạn có thể ứng dụng rất nhiều tính năng thú vị như đánh dấu cú pháp, tự động thực lè, đối sánh dấu ngoặc,... cho công việc của mình.

Hệ thống phím tắt sinh động, trực quan giúp thao tác trở nên nhanh chóng hơn, từ đó năng suất và hiệu quả lập trình cũng được tăng lên. Ngoài ra, tính năng nhắc mã

IntelliSense chuyên nghiệp trên VS Code sẽ tự động phát hiện đoạn mã không đầy đủ và thông báo cho lập trình viên. Hoặc nó có thể tự động bổ sung cú pháp còn thiếu khi bạn quên không khai báo biến.

VS Code được xây dựng dựa trên Electron, kết hợp các công nghệ web như Node.js và JavaScript với tốc độ và sự linh hoạt của các ứng dụng gốc. Ngoài ra, kiến trúc dịch vụ công cụ của VS Code giúp nó tích hợp được với các công nghệ hỗ trợ Visual Studio như Roslyn cho .NET, TypeScript, công cụ gỡ lỗi Visual Studio,...

## 3. Layout

- Là bản thiết kế giao diện, người dùng để tương tác với hệ thống thông qua giao diện.

- Tất cả các thành phần cấu tạo nên Layouts đều kế thừa từ đối tượng View và ViewGroup. Trong đó, View thì thể hiện ra ngoài và người dùng tương tác được. Ngược

lại thì ViewGroup được ẩn đi (invisible container) dù ViewGroup là ROOT để định nghĩa View và ViewGroup khác.

#### **4. ViewGroup**

-LinearLayout là view group mà các views (Button, TextView, ... ) được biểu diễn Horizontally (Hoành độ) hoặc Vertically (Tung độ).

-RelativeLayout là ViewGroup mà chúng ta có thể layouts (sắp xếp) các views/widgets dựa trên vị trí của các view/widgets. Nó không phụ thuộc trục hoành và trục tung.

-Frame Layout là các thiết kế các view controls ở dạng khối trong những vùng riêng trên cùng giao diện (screen).

-Table Layout là cách sắp xếp ViewGroup theo hàng rows và cột columns. Table Layout không thể hiện biên (border line) của columns, rows or cells.

-Absolute Layout là thiết kế layout dựa trên tọa độ x và y.

-GridView là view group thể hiện ở dạng mảng 2 chiều - two dimensional.

#### **5. Các layout hay dùng và các thuộc tính đi kèm**

•Ba Views Cơ Bản: TextView, EditText, Button sử dụng RelativeLayout

•Relativelayout

-android:id là định danh layout

-android:gravity là định nghĩa cho các Views

-android:ignoreGravity là bỏ qua định nghĩa các vị trí cho Views.

•GridLayout

-android:id là định danh của layout

-android:columnWidth là độ rộng của mỗi cột

-android:gravity chỉ rõ vị trí mỗi ô (cell) : top, bottom, left, right, center, center\_vertical, center\_horizontal, ...

-android:horizontalSpacing là khoảng cách giữa các cột với đơn vị đo là px, dp, sp, in, or mm.

-android:numColumns là số cột cần hiện thị

-android:verticalSpacing là khoảng cách giữa các dòng

-android:stretchMode dùng để lấp đầy khoảng trống (empty space)

## 6. Activity

-Activity là một trong 4 thành phần quan trọng của ứng dụng Android gồm: Activity, Service, Content Provider, Broadcast receivers.

-Activity là một thành phần của ứng dụng Android.

-Android Activity là nơi để ứng dụng tương tác trực tiếp với người dùng thông qua giao diện.

-Một ứng dụng có thể sẽ có nhiều màn hình và mỗi màn hình có thể là một Activity (nếu không sử dụng Fragment).

-Mỗi Activity thường hoạt động độc lập với nhau nhưng có thể tương tác và truyền dữ liệu qua nhau thông qua Intent.

-Chính vì Activity hoạt động độc lập nên sẽ có vòng đời riêng từ lúc được khởi tạo cho đến lúc được huỷ đi.

•Vòng đời Activity

-**onCreate()**: Được gọi bởi OS khi Activity được tạo (chỉ gọi duy nhất một lần). Dùng để khởi tạo giao diện hoặc dữ liệu để dùng cho sau này.

-**onStart()**: Ngay trước khi Activity được chạy và tương tác với người dùng thì hàm được gọi. Thông thường, tại hàm này chúng ta sẽ khởi tạo animation cho UI (nếu có), Audio, hay bất kì đối tượng nào cần thiết để Activity có thể hiển thị cho người dùng.

-**onResume()**: Hàm này được gọi khi Activity chuẩn bị chạy hoặc Activity được kích hoạt trở lại từ trạng thái stop.

-**onPause()**: Hàm này được gọi khi ứng dụng chuyển sang chạy nền.

-**onStop()**: Hàm này được gọi ngay sau onPause(), khi Activity không còn tương tác với người dùng.

-**onRestart()**: Được gọi sau khi Activity bị tạm dừng và giờ được kích hoạt trở lại.

-**onDestroy()**: Đây là hàm cuối cùng trong vòng đời của Activity. Có 2 cách để hàm được gọi: một là bạn gọi hàm finish() để chủ động tắt một Activity, hai là được gọi tự động bởi OS khi tài nguyên cạn kiệt để tránh bị lỗi Memory Leak.

## 7. Các kiểu lập trình sự kiện trong Android

- Xử lý sự kiện (Event Handling) trong Android là sự tương tác của người dùng với các thành phần tương tác của ứng dụng
- Event Listeners – là một Interface trong lớp View chứa một phương thức callback. Các phương thức này sẽ có thể được gọi bởi AndroidFramework khi Listener đã đăng ký được kích hoạt bởi tương tác của người dùng với item trong giao diện UI.
- Event Listeners Registration – là tiến trình mà bởi đó một Event Handler được đăng ký với một Event Listener để mà Handler này được gọi khi Event Listener kích hoạt sự kiện.
- Event Handlers – Khi một Event xảy ra, và chúng ta đã đăng ký một Event Listener cho sự kiện, thì Event Listener gọi Event Handler, là phương thức thực sự xử lý sự kiện đó.

## 8. Giới thiệu Intent

- Intent là một đối tượng message dùng để yêu cầu (request) một hành động từ một vài component trong ứng dụng
- Intent có 2 loại chính là Explicit Intent và Implicit Intent
- Explicit Intent là intent xác định rõ và cụ thể các thành phần tham gia hoạt động.
- Implicit Intent dùng để chỉ ra hành động cần được thực hiện (action) và dữ liệu cho hành động đó (data)

## 9. Toast Notification

- Toast là một message, có dạng như một popup, xuất hiện trên màn hình trong 1 khoảng thời gian ngắn và tự biến mất sau đó.
- Toast có thể được tạo và hiển thị trong Activity hoặc trong Service.
- Không cho phép người sử dụng tương tác
- Có 2 giá trị mặc định: hằng số
  - +Toast.LENGTH\_SHORT hiển thị trong 2 giây
  - +Toast.LENGTH\_LONG hiển thị trong 3.5 giây

## 10. ListView, GridView và Spinner

- ListView hiển thị dữ liệu dạng danh sách trong Android ListView là phần tử View được dùng để hiển thị dữ liệu là một danh sách (mảng) từ một nguồn cấp gọi là Adapter
- Các bước để tạo và sử dụng ListView gồm có:

- Các bước để tạo và sử dụng ListView gồm có:
  - + Khai báo ListView trong Layout
  - + Gán cho ListView một Adapter là nguồn cấp dữ liệu cho nó
  - + Xây dựng Adapter cho ListView
- GridView trong Android hiển thị các item trong mảng lưới hai chiều có thể scroll và các item này không cần thiết phải được định nghĩa trước.
- GridView hoàn toàn tương tự như ListView, dữ liệu được đưa vào GridView thông qua các mảng 1 chiều và dựa vào số cột của GridView để ngắt số hàng và cột nhưng dựa vào số cột ta thiết lập, mà nó tự động ngắt hàng.
- GridView có thể hiển thị Text hoặc hình ảnh.
- Spinner trong Android được sử dụng để hiển thị các item theo dạng danh sách. Spinner chỉ được chọn một item trong danh sách. Sau khi chọn thì danh sách item sẽ co lại, item được chọn sẽ hiển thị ra bên ngoài Spinner.

## 11. XML Parser trong Android

- XML (Extensible Markup Language) là ngôn ngữ đánh dấu mở rộng. Đây là một dạng ngôn ngữ đánh dấu, có chức năng truyền dữ liệu và mô tả nhiều loại dữ liệu khác nhau.
- Ưu điểm lớn nhất của XML chính là sự độc lập. Điều này cũng rất dễ hiểu, vì XML được sử dụng để mô tả dữ liệu dưới dạng text, nên hầu hết các phần mềm hay các chương trình bình thường đều có thể đọc được chúng. XML có thể đọc và phân tích nguồn dữ liệu khá dễ dàng nên nó được sử dụng với mục đích chính là trao đổi dữ liệu giữa các chương trình, các hệ thống khác nhau. XML được sử dụng cho Remote Procedure Calls nhằm phục vụ cho các dịch vụ của website.
- Nhược điểm của XML: Tỷ lệ sai sót khi sử dụng XML để truyền dữ liệu khoảng từ 5-7%. Con số này tuy không quá cao, nhưng trên thực tế, người ta vẫn cần cân nhắc trước khi sử dụng nó để trao đổi thông tin.
- Kỹ thuật dùng DOM
- DOM (Document Object Model ): Cache all – cơ chế của nó là đọc toàn bộ nội dung tập tin XML vào bộ nhớ
- Do đó nếu XML lớn thì làm chậm chương trình và có thể phung phí bộ nhớ vì không phải lúc nào ta cũng muốn đọc hết nội dung XML
- Kỹ thuật dùng SAX
- Đỡ tốn bộ nhớ , Chạy nhanh , viết phức tạp hơn DOM

- Duyệt XML bằng DOM và SAX
- Với một tài liệu XML, mô hình DOM sẽ duyệt và chuyển nó thành một mô hình cây của các Object
- SAX là một API thuộc JAXP, nó cho phép xây dựng các ứng dụng truy xuất, phân tích các tài liệu XML theo cơ chế tuần tự và hướng sự kiện.
- Không như DOM, SAX phân tích dựa vào khái niệm sự kiện và phản hồi sự kiện, SAX không tạo ra một hình thức đại diện của tài liệu XML trong bộ nhớ, và cũng không sử dụng đối tượng để mô tả các thành phần trong file XML như DOM, thay vào đó, nó sử dụng các phương thức callback của đối tượng.

## 12. Cơ sở dữ liệu và kết nối cơ sở dữ liệu Firebase

**Firestore** là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

Cụ thể là những giao diện lập trình ứng dụng API đơn giản. Mục đích nhằm tăng số lượng người dùng và thu lại nhiều lợi nhuận hơn.

Đặc biệt, còn là dịch vụ đa năng và bảo mật cực tốt. Firestore hỗ trợ cả hai nền tảng Android và IOS. Không có gì khó hiểu khi nhiều lập trình viên chọn Firestore làm nền tảng đầu tiên để xây dựng ứng dụng cho hàng triệu người dùng trên toàn thế giới.

Firestore ra đời với sản phẩm cung cấp là dịch vụ Backend-as-a-Service. Tiếp đến, vào năm 2014, Google mua lại Firestore và phát triển nó thành một dịch vụ đa chức năng được hàng triệu người sử dụng cho đến hiện nay.

Ưu điểm nổi bật

- Tạo tài khoản và sử dụng dễ dàng
- Tốc độ phát triển nhanh
- Nhiều dịch vụ trong một nền tảng
- Được cung cấp bởi Google
- Tập trung vào phát triển giao diện người dùng
- Firestore không có máy chủ
- Học máy (Machine Learning)
- Tạo lưu lượng truy cập
- Theo dõi lỗi
- Sao lưu

## 13. Ngôn ngữ PHP, cơ sở dữ liệu MySQL và XAMP

### PHP

PHP là viết tắt của cụm từ Personal Home Page này đã được chuyển thành Hypertext Preprocessor. Hiểu đơn giản thì PHP là một ngôn ngữ lập trình kịch bản (scripting language) đa mục đích.

Ngôn ngữ PHP được biết tới là một mã nguồn mở. Vì vậy việc liên tục cập nhật và phát triển thêm mới các ứng dụng từ PHP cũng được xem như một công việc rất cần thiết.

Ứng dụng chủ yếu của PHP đó chính là việc xây dựng nên các chương trình dành cho các server máy chủ.

## **MySQL**

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (gọi tắt là RDBMS) hoạt động theo mô hình client-server. Với RDBMS là viết tắt của Relational Database Management System. MySQL được tích hợp apache, PHP. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL. MySQL được phát hành từ thập niên 90s.

## **XAMP**

XAMPP hoạt động dựa trên sự tích hợp của 5 phần mềm chính là Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) và Perl (P), nên tên gọi XAMPP cũng là viết tắt từ chữ cái đầu của 5 phần mềm này:

Chữ X đầu tiên là viết tắt của hệ điều hành mà nó hoạt động với: Linux, Windows và Mac OS X.

Apache: Web Server mã nguồn mở Apache là máy chủ được sử dụng rộng rãi nhất trên toàn thế giới để phân phối nội dung Web. Ứng dụng được cung cấp dưới dạng phần mềm miễn phí bởi Apache Software Foundation.

MySQL / MariaDB: Trong MySQL, XAMPP chứa một trong những hệ quản trị cơ sở dữ liệu quan hệ phổ biến nhất trên thế giới. Kết hợp với Web Server Apache và ngôn ngữ lập trình PHP, MySQL cung cấp khả năng lưu trữ dữ liệu cho các dịch vụ Web. Các phiên bản XAMPP hiện tại đã thay thế MySQL bằng MariaDB (một nhánh của dự án MySQL do cộng đồng phát triển, được thực hiện bởi các nhà phát triển ban đầu).

PHP: Ngôn ngữ lập trình phía máy chủ PHP cho phép người dùng tạo các trang Web hoặc ứng dụng động. PHP có thể được cài đặt trên tất cả các nền tảng và hỗ trợ một số hệ thống cơ sở dữ liệu đa dạng.

Perl: ngôn ngữ kịch bản Perl được sử dụng trong quản trị hệ thống, phát triển Web và lập trình mạng. Giống như PHP, Perl cũng cho phép người dùng lập trình các ứng dụng Web động.

Ưu điểm lớn nhất của XAMPP là không phải trả phí bản quyền và sử dụng mã nguồn mở, bên cạnh đó cấu hình của web server này tương đối đơn giản, gọn nhẹ nên được sử dụng ngày càng phổ biến hiện nay.

## Chương 3.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 1. Phân tích nghiệp vụ người dùng

Người chủ cửa hàng có quy trình nghiệp vụ cụ thể như sau:

Quản lý nhân viên, quản lý phát lương cho nhân viên, quản lý và kiểm tra tất cả các chứng từ hóa đơn của cửa hàng, phân tích thị trường nhờ vào các hóa đơn chứng từ đó, và từ thông kê của thủ kho để biết được mặt hàng nào bán chạy, mặt hàng nào gần hết hạn mà vẫn còn tồn nhiều,... rồi từ đó sẽ lên kế hoạch bán hàng/nhập hàng cho cửa hàng.

Quản lý sẽ là người tìm hiểu và liên hệ với nhà cung cấp/nhà sản xuất để thương lượng nhập hàng.

Yêu cầu quản lý việc bán hàng, nhập hàng, hồ sơ khách hàng, nhà cung cấp, hồ sơ nhân viên, các quy trình bảo hành, thông kê, các nhật ký về mọi hoạt động của cửa hàng.

Nhưng ở trong ứng dụng lần này, chỉ thiết kế và xây dựng nghiệp vụ quản lý sản phẩm và báo cáo doanh thu của cửa hàng.

### 2. Xác định các tác nhân của hệ thống

Sau khi khảo sát ta nhận thấy các tác nhân hệ thống gồm có :

Những người muốn mua hàng vào ứng dụng để xem thông tin. Những người này là Khách hàng tiềm năng (Guest).

Những người đã đặt hàng vào kiểm tra đơn hàng, thanh toán v.v.. gọi là Khách hàng (Customer).

Về phía đơn vị bán hàng, có những người sau đây tham gia vào hệ thống:

Người quản lý bán hàng: quyết định nhập hàng, giá bán, quản lý tồn kho, doanh thu, chính sách khuyến mãi.

Người bán hàng: Tư vấn cho khách hàng, theo dõi đơn hàng, thu tiền, theo dõi chuyển hàng cho khách.

Quản lý kho: xuất, nhập hàng, quản lý tồn kho

Quản trị hệ thống: Tạo người dùng, Phân quyền, Tạo cửa hàng

### 3. Các chức năng của hệ thống

Hệ thống bao gồm các chức năng chính sau :

- Xác thực Email & Mật khẩu
- Lấy lại mật khẩu
- Trạng thái xác thực tồn tại
- Tìm kiếm sản phẩm
- Thông tin chi tiết sản phẩm
- Nhận ưu đãi trong ngày
- Xem đơn đặt hàng của tôi
- Xem chi tiết & trạng thái đơn hàng
- Đặt hàng
- Thông báo
- Thông tin
- Quản lý
- Chat
- Đăng xuất
- Bảng quản trị
- Xem tất cả các sản phẩm
- Thêm sản phẩm
- Xóa sản phẩm
- Xem đơn đặt hàng
- Thay đổi trạng thái đơn hàng
- Thông báo
- Xem tổng thu nhập
- Xem thu nhập trên Biểu đồ

Trong đó gồm có

- Tìm kiếm sản phẩm theo văn bản gõ vào
- Xem chi tiết sản phẩm được chọn
- Xem khuyến mãi
- Xem so sánh
- Mua hàng
- Quản lý giỏ hàng

- Chat với người bán hàng
- Đăng ký tài khoản để trở thành khách hàng

Tiếp theo, Actor “Khách hàng” và nhận thấy họ sử dụng chức năng:

- Đăng nhập
- Xem đơn hàng
- Thanh toán

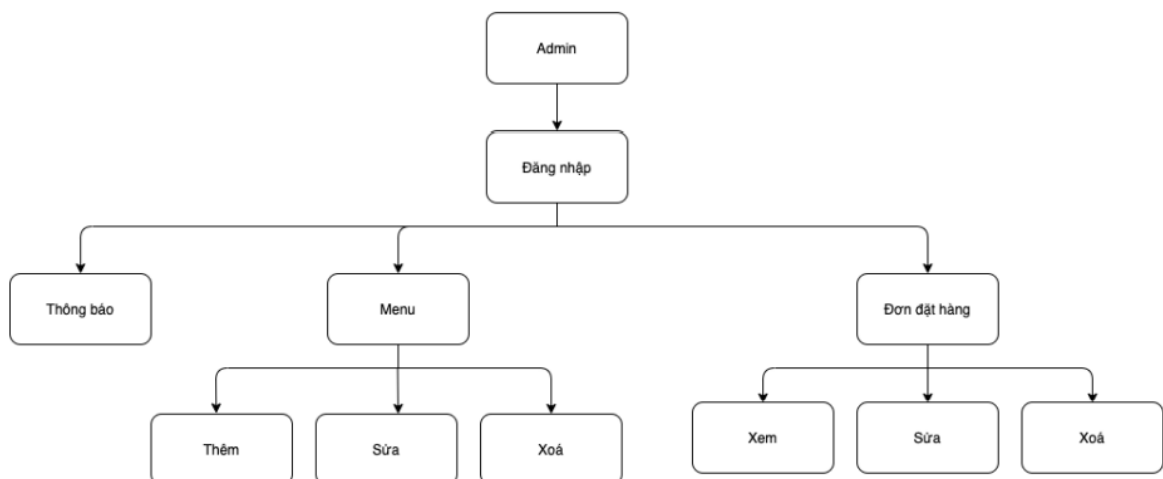
Tiếp theo, Actor “Người bán hàng” và họ có thể sử dụng các chức năng:

- Đăng nhập
- Chat với khách hàng
- Theo dõi đơn hàng
- Thu tiền
- Theo dõi chuyển hàng

Chức năng xem sản phẩm có 2 cách là chọn loại sản phẩm, nhà sản xuất để xem và gõ vào ô tìm kiếm. Nên chúng ta tách ra làm 2 là Xem sản phẩm và Tìm kiếm.

Chức năng mua hàng, thực chất là thêm vào giỏ hàng nên có thể xem là chức năng con của quản lý giỏ hàng.

Ứng dụng gồm có 2 phần chính: ứng dụng trên android và server lưu dữ liệu: firebase.



Cây phân cấp chức năng của người quản trị

## Mô hình hóa cơ sở dữ liệu

Hệ thống gồm các bảng sau :

| Tên cột  | Ý nghĩa          | Kiểu dữ liệu | Ghi chú     |
|----------|------------------|--------------|-------------|
| ID       | ID người dùng    | ObjectId     | PRIMARY KEY |
| Email    | Email người dùng | required     | FOREIGN KEY |
| Password | Mật khẩu         | required     |             |
| Username | Tên người dùng   | String       |             |
| Mobile   | Số điện thoại    |              |             |

Bảng 3. 1 Bảng user

| Tên cột    | Ý nghĩa           | Kiểu dữ liệu | Ghi chú       |
|------------|-------------------|--------------|---------------|
| ID         | ID sản phẩm       | ObjectId     | PRIMARY KEY   |
| tensanpham | Tên sản phẩm      | required     | FOREIGN KEY   |
| hinhanh    | Hình ảnh sản phẩm | required     | Default: null |

Bảng 3. 2 Bảng sản phẩm

| Tên cột | Ý nghĩa           | Kiểu dữ liệu | Ghi chú       |
|---------|-------------------|--------------|---------------|
| ID      | ID người dùng     | ObjectId     | PRIMARY KEY   |
| tensp   | Tên sản phẩm      | required     | FOREIGN KEY   |
| giasp   | Giá sản phẩm      | required     |               |
| Hinhanh | Hình ảnh sản phẩm | null         | Default: null |
| Mota    | Mô tả sản phẩm    | String       |               |
| Loai    | Loại sản phẩm     | Boolean      |               |

Bảng 3. 3 Bảng chi tiết sản phẩm

| Tên cột | Ý nghĩa                  | Kiểu dữ liệu | Ghi chú     |
|---------|--------------------------|--------------|-------------|
| _id     |                          | ObjectId     |             |
| mobile  | Số điện thoại người dùng | Number       | FOREIGN KEY |

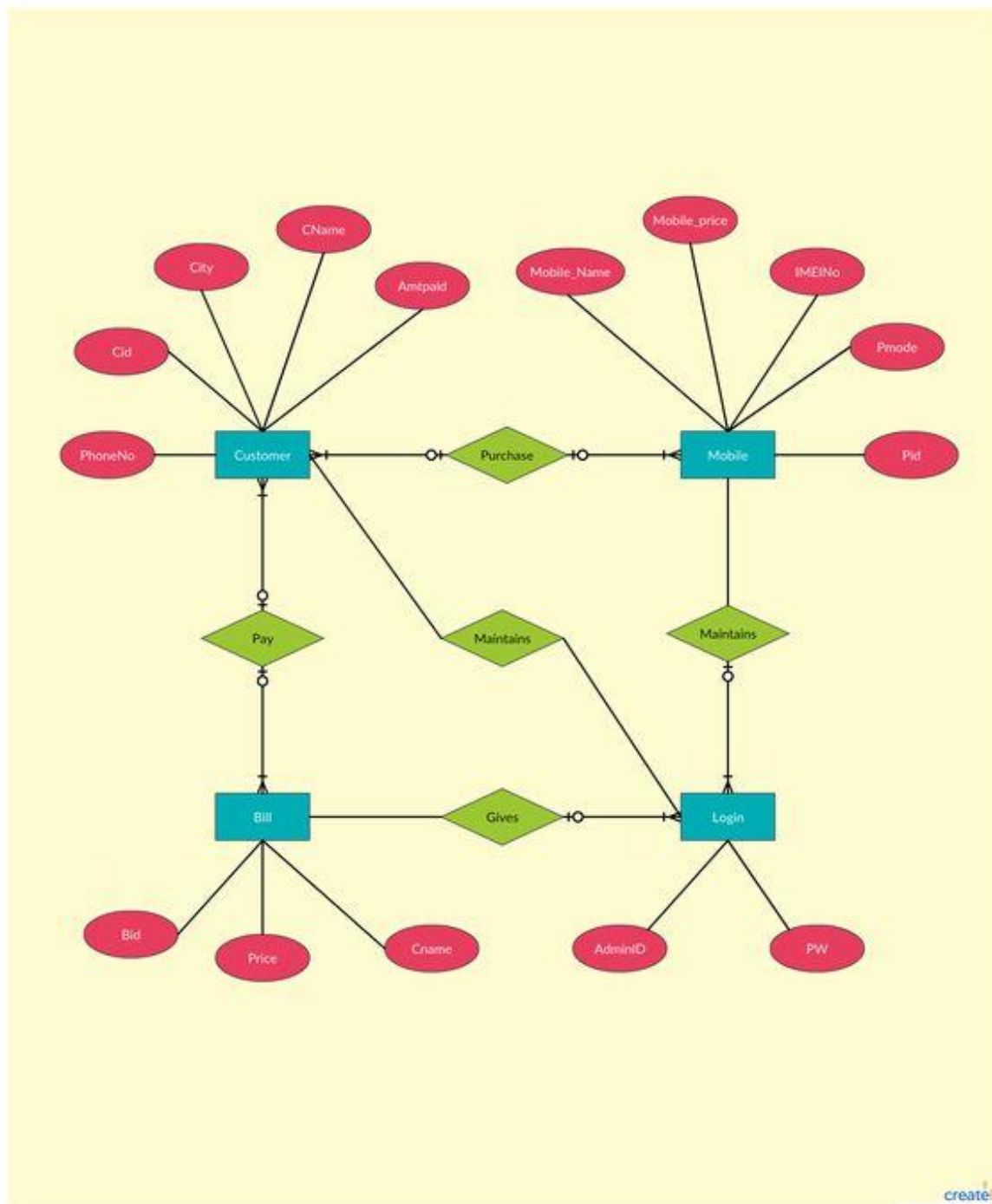
|          |                     |        |               |
|----------|---------------------|--------|---------------|
| userName | Tên người dùng      | String | FOREIGN KEY   |
| address  | Địa chỉ             | String | Default: ""   |
| total    | Tổng giá            | Number |               |
| tensp    | Các hàng            | String |               |
| status   | Tình trạng đơn hàng | String | Default: true |

Bảng 3. 4 Bảng đặt hàng (Request)

| Tên cột   | Ý Nghĩa     | Kiểu dữ liệu | Ghi chú     |
|-----------|-------------|--------------|-------------|
| Iddonhang | Id đơn hàng | ObjectId     | PRIMARY KEY |
| Idsanpham | Id sản phẩm | ObjectId     | PRIMARY KEY |
| Soluong   | Số lượng    | Number       |             |
| Gia       | Giá         | Number       |             |

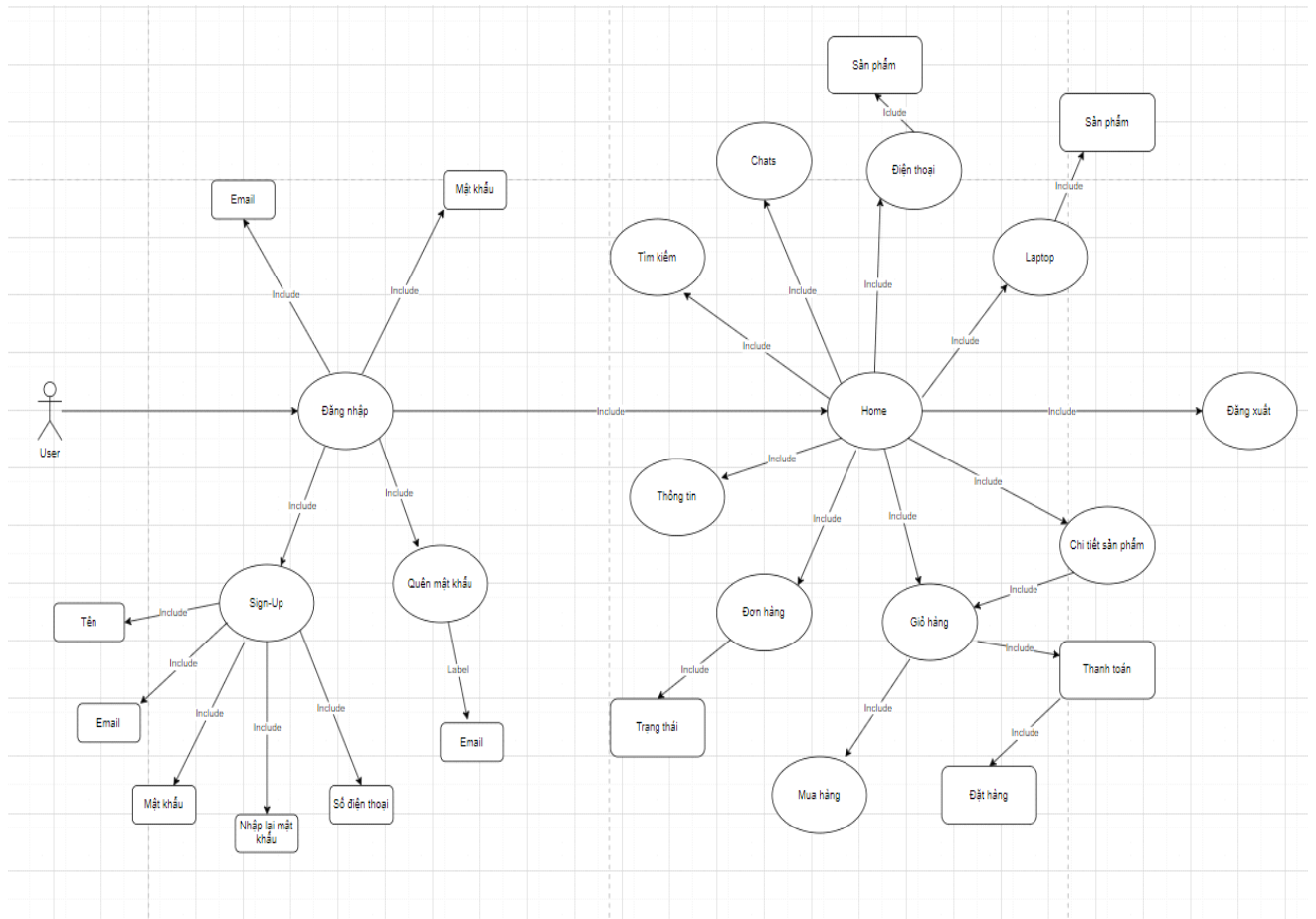
Bảng 3. 5 Bảng chi tiết đơn hàng

### Mức khái niệm ERD



Hình 3. 1 Mô hình ERD

## Sơ đồ Use case



Hình 3. 2 Sơ đồ Use case tổng quát

|                        | <b>Đăng nhập</b>                                    |
|------------------------|---|
| <b>Mục đích</b>        | Đăng nhập vào hệ thống để quản lý các danh mục      |
| <b>Tác nhân</b>        | Quản trị viên, khách hàng                           |
| <b>Điều kiện trước</b> | Tài khoản có chức năng đăng nhập                    |
| <b>Mô tả</b>           | Người sử dụng dùng tài khoản đăng nhập vào hệ thống |

Bảng 3. 6 Đặc tả use case đăng nhập

|                        | <b>Thông Kê</b>   |
|------------------------|---|
| <b>Mục đích</b>        | Quản trị viên giám sát được hoạt động của cửa hàng  |
| <b>Tác nhân</b>        | Quản trị viên   |
| <b>Điều kiện trước</b> | Tài khoản được cấp quyền quản trị.  |
| <b>Mô tả</b>           | Người quản trị xem trạng thái thống kê của số lượng đơn hàng trong ngày, tài khoản đăng ký trong ngày, tổng số sản phẩm |

|  |                                 |
|--|---------------------------------|
|  | hiện có và doanh thu trong ngày |
|--|---------------------------------|

Bảng 3. 7 Đặc tả use case thống kê

| <b>Quản lý tài khoản</b> |   |
|--------------------------|---|
| <b>Mục đích</b>          | Quản lý các tài khoản quản trị và khách hàng  |
| <b>Tác nhân</b>          | Quản trị viên   |
| <b>Điều kiện trước</b>   | Tài khoản được cấp quyền quản trị.  |
| <b>Mô tả</b>             | Người quản lý sẽ tạo ra các tài khoản mới, thay đổi quyền hạn của các tài khoản hiện có, xóa một tài khoản nào đó ra khỏi hệ thống. |

Bảng 3. 8 Đặc tả use case quản lý tài khoản

| <b>Quản lý sản phẩm</b> |   |
|-------------------------|---|
| <b>Mục đích</b>         | Quản lý danh sách các sản phẩm  |
| <b>Tác nhân</b>         | Quản trị viên   |
| <b>Điều kiện trước</b>  | Tài khoản được cấp quyền quản trị.  |
| <b>Mô tả</b>            | Người quản lý sau khi đăng nhập vào hệ thống có thể chuyển vào phần quản lý sản phẩm. Tại đây có thể tiến hành thao tác thêm, sửa, xóa với 1 sản phẩm trong cơ sở dữ liệu bằng cách chọn từ danh sách hoặc nhập thông tin theo mẫu yêu cầu. |

Bảng 3. 9 Đặc tả use case quản lý sản phẩm

| <b>Quản lý Đơn hàng</b> |  |
|-------------------------|--|
| <b>Mục đích</b>         | Quản lý danh sách Đơn hàng   |
| <b>Tác nhân</b>         | Quản trị viên  |
| <b>Điều kiện trước</b>  | Tài khoản được cấp quyền quản trị.   |
| <b>Mô tả</b>            | Người quản lý sau khi đăng nhập vào hệ thống có thể chuyển vào phần quản lý đơn hàng. Tại đây có thể tiến hành thao tác sửa, xóa với 1 đơn hàng bất kỳ |

Bảng 3. 10 Đặc tả use case quản lý đơn hàng

#### 4. Danh sách các đối tượng của hệ thống

Toàn bộ hệ thống bao gồm 5 usecase với chức năng cụ thể như sau:

- **Đăng nhập:** những người có tài khoản trên hệ thống có thể đăng nhập vào hệ thống để thực hiện các chức năng nằm trong phạm vi cho phép của mình.
- **Thống kê:** giúp người quản lý có thể thấy được các thông tin cần thiết về khách hàng sản phẩm, đơn hàng...
- **Quản lý tài khoản:** là chức năng giành cho người quản trị bao gồm các hoạt động tạo mới, chỉnh sửa – cập nhật, thay đổi quyền cũng như xóa cho toàn bộ các tài khoản trên hệ thống.
- **Quản lý sản phẩm:** người quản lý có thể cập nhật thông tin sản phẩm qua các thao tác thêm, sửa, xóa...
- **Quản lý đơn hàng:** tất cả các đơn hàng được khách hàng đặt sẽ hiển thị tại đây. Người quản lý có thể xem thông tin về các đơn hàng cũng như cập nhật tình trạng (đã chuyển -chưa chuyển) của các đơn hàng hiện có.

## Chương 4.

# THỰC NGHIỆM VÀ TRIỂN KHAI

### 1. Thiết kế giao diện chương trình

Giao diện chương trình gồm giao diện dành cho chủ quản lí (người bán hàng) và giao diện dành cho người dùng (người mua hàng).

### 2. Giao diện của Admin ( chủ cửa hàng, người quản lí )

#### 1.1 Giao diện màn hình chờ



Hình 4. 1 Giao diện màn hình chờ

## 1.2 Giao diện màn hình đăng kí

The image shows a mobile application registration screen with a dark blue background. At the top, the status bar displays the time 00:14, signal strength, Wi-Fi, and battery level at 97%. The title "Đăng kí" is centered in yellow. Below it are five white input fields, each with a corresponding icon: a person for "Tên", an envelope for "Mail", a key for "Mật khẩu", a key for "Nhập Lại mật khẩu", and a checkmark for "Số điện thoại". A blue button labeled "XÁC NHẬN" is positioned below the fields. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Hình 4. 2 Giao diện màn hình đăng kí

### 1.3 Giao diện hình đăng nhập



Hình 4. 3 Giao diện màn hình đăng nhập

## 1.4 Giao diện màn hình chính



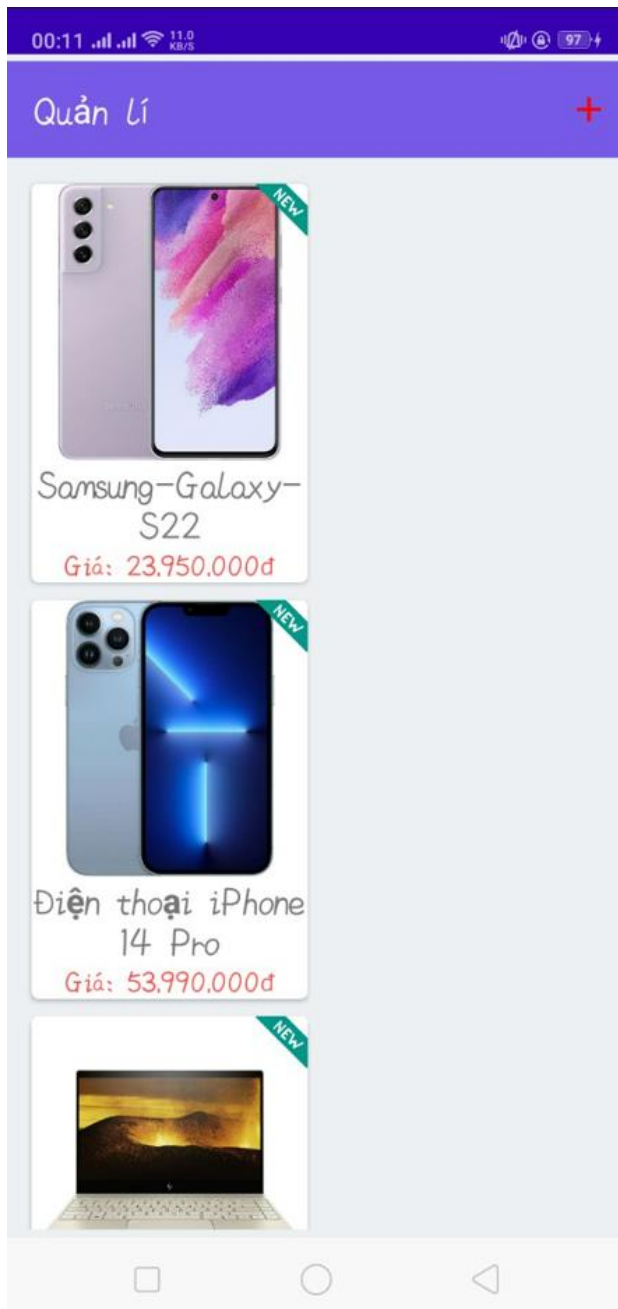
Hình 4. 4 Giao diện màn hình chính

### 1.5 Giao diện màn hình quên mật khẩu



Hình 4. 5 Giao diện màn hình quên mật khẩu

## 1.6 Giao diện màn hình quản lý sản phẩm



Hình 4. 6 Giao diện màn hình quản lý sản phẩm

## 1.7 Giao diện màn hình thêm sản phẩm

00:11 1.00 KB/s 97%

Thêm sản phẩm

Tên sản phẩm

Giá sản phẩm

Hình ảnh sản phẩm

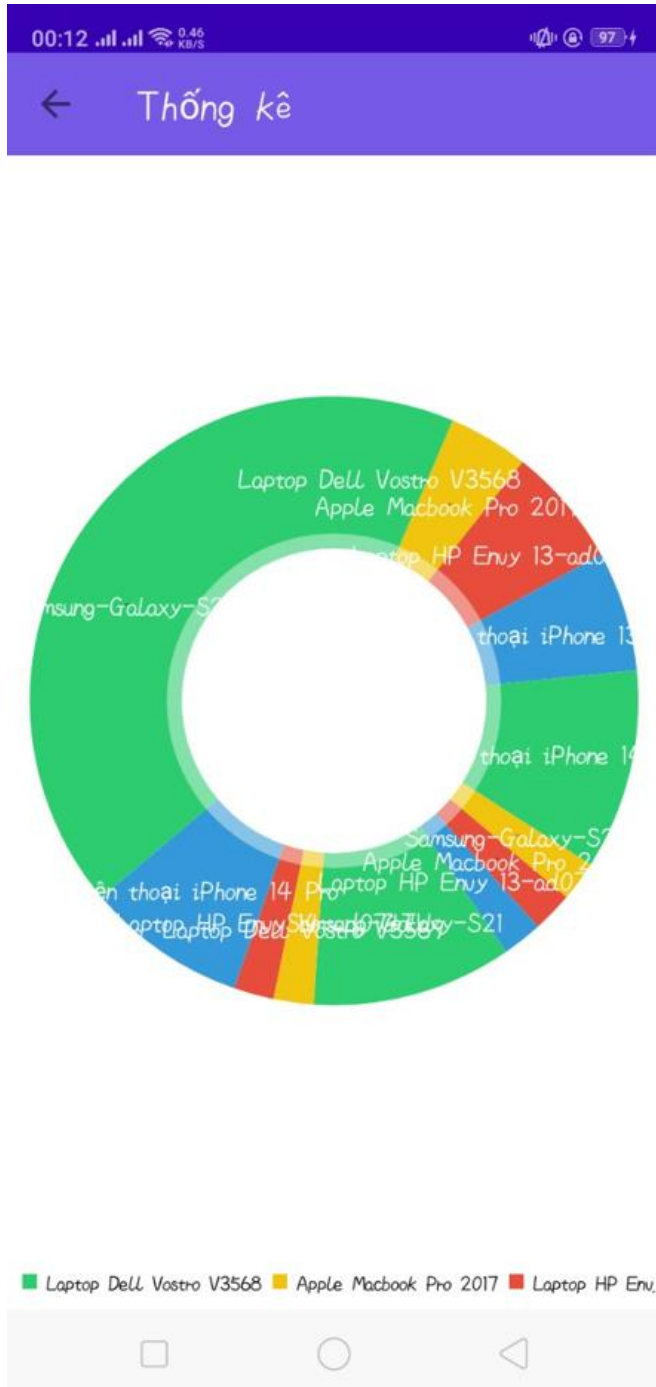
Mô tả sản phẩm

Vui lòng chọn loại

Thêm sản phẩm

Hình 4. 7 Giao diện màn hình thêm sản phẩm

## 1.8 Giao diện màn hình thống kê



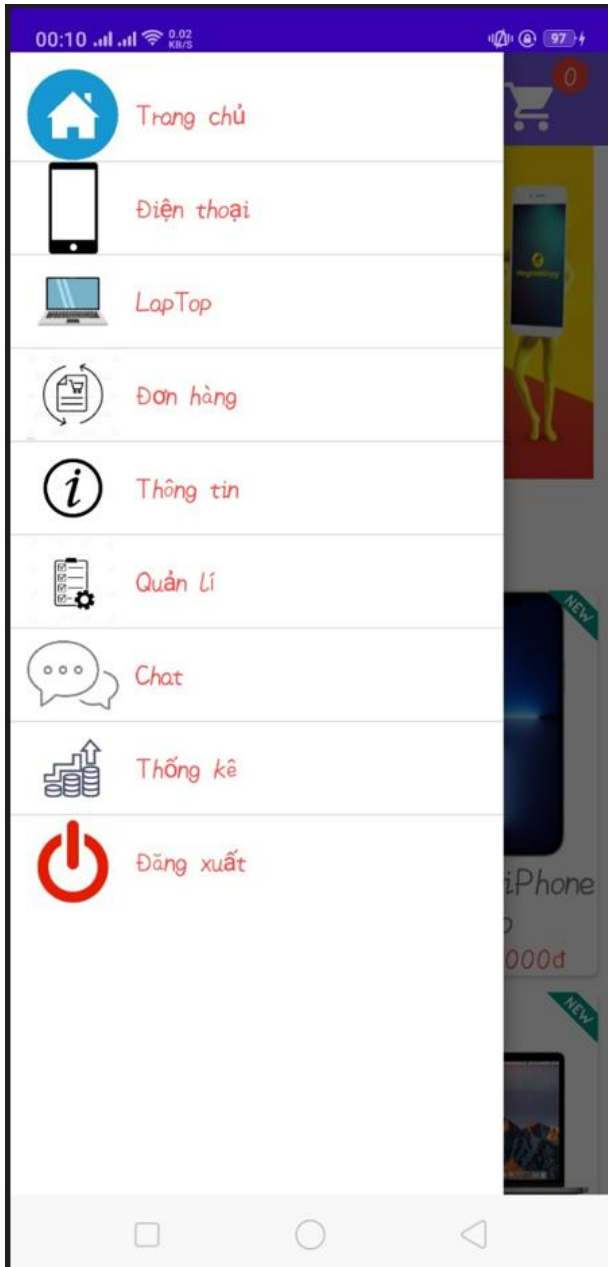
Hình 4. 8 Giao diện màn hình thống kê

## 1.9 Giao diện màn hình tìm kiếm sản phẩm



Hình 4. 9 Giao diện màn hình tìm kiếm

### 1.10 Giao diện màn hình menu



Hình 4. 10 Giao diện màn hình menu

### 3. Giao diện của User( người mua hàng)

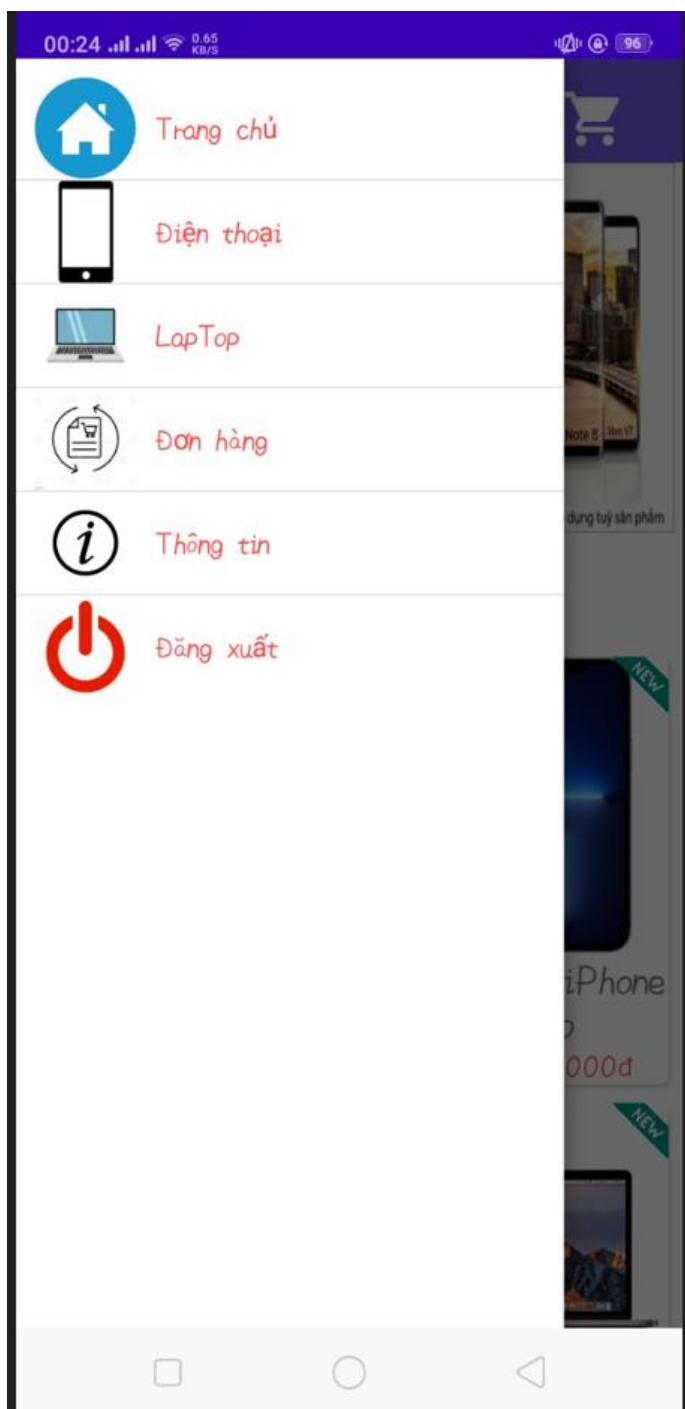
Giao diện của người dùng cũng giống như của người bán hàng, nhưng người dùng sẽ không có chức năng quản lí . Mà chỉ có các chức năng xem sản phẩm, thêm vào giỏ hàng và thanh toán.

### 1.11 Giao diện màn hình Splash.



Hình 4. 11 Giao diện Spalsh

### 1.12 Giao diện màn hình menu.



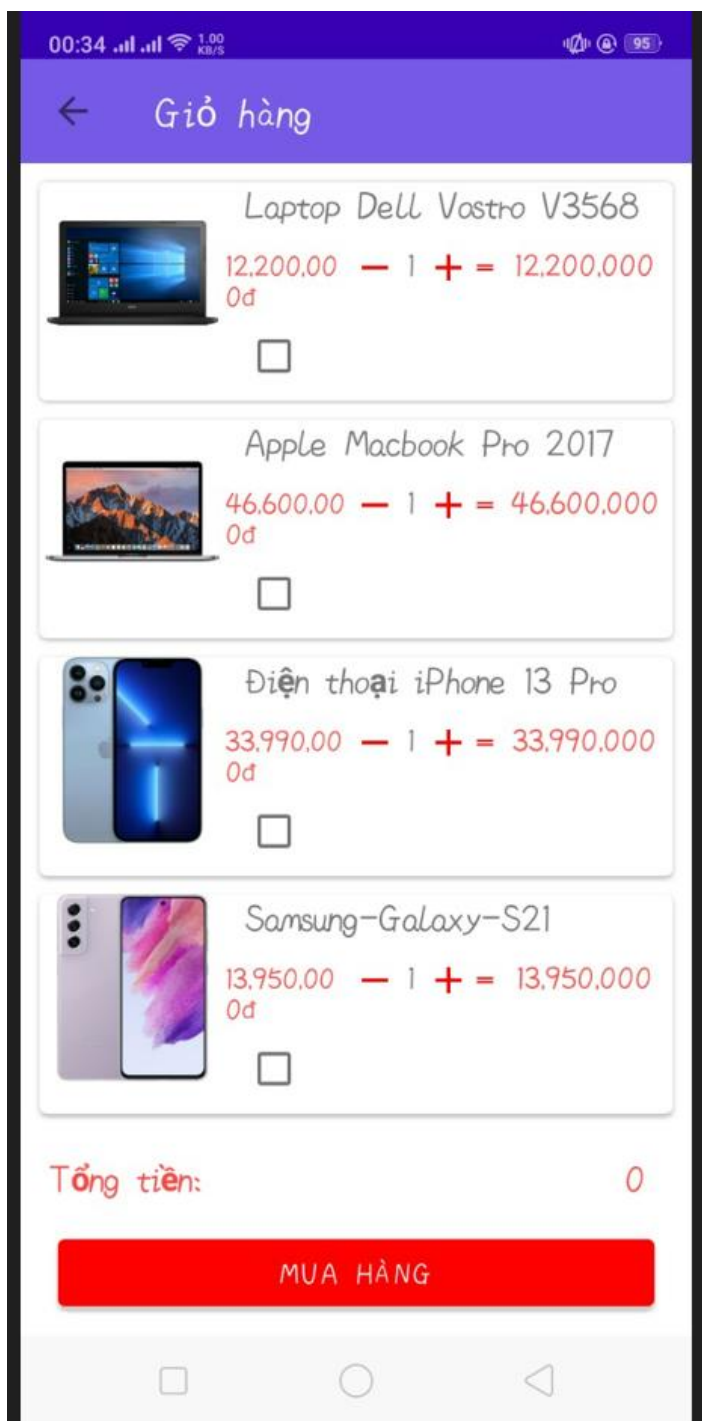
Hình 4. 12 Giao diện menu

### 1.13 Giao diện màn hình giỏ hàng.



Hình 4. 13 Giao diện giỏ hàng

### 1.14 Giao diện màn hình giỏ hàng



Hình 4. 14 Giao diện màn hình giỏ hàng

### 1.15 Giao diện màn hình thanh toán



Hình 4. 15 Giao diện màn hình giỏ hàng

### 1.16 Giao diện màn hình chat



Hình 4. 16 Giao diện chat

## KẾT QUẢ ĐẠT ĐƯỢC

- Hiểu được các nghiệp vụ cơ bản của hệ thống quản lý cửa hàng tạp hóa.
- Xây dựng được hệ thống quản lý với đầy đủ các chức năng cần thiết.
- Giao diện đơn giản, dễ thao tác với người dùng lần đầu.
- Viết bằng ngôn ngữ Java tương đối phổ biến nên dễ chỉnh sửa về sau.
- Lập trình android, MySql và PHP.
- Các chức năng đã thực hiện được:

### + Đối với hệ thống:

- Tìm kiếm, xem sản phẩm
- Thêm, sửa, xóa sản phẩm ( hình ảnh, chi tiết sản phẩm)
- Kiểm tra hàng hóa
- Đăng nhập và Đăng ký
- Lấy lại mật khẩu
- Báo cáo doanh thu
- Chat với User
- Xây dựng được các chức năng quản lý các sản phẩm.
- Hệ thống hỗ trợ trong việc xem thông tin hàng hóa dễ dàng.

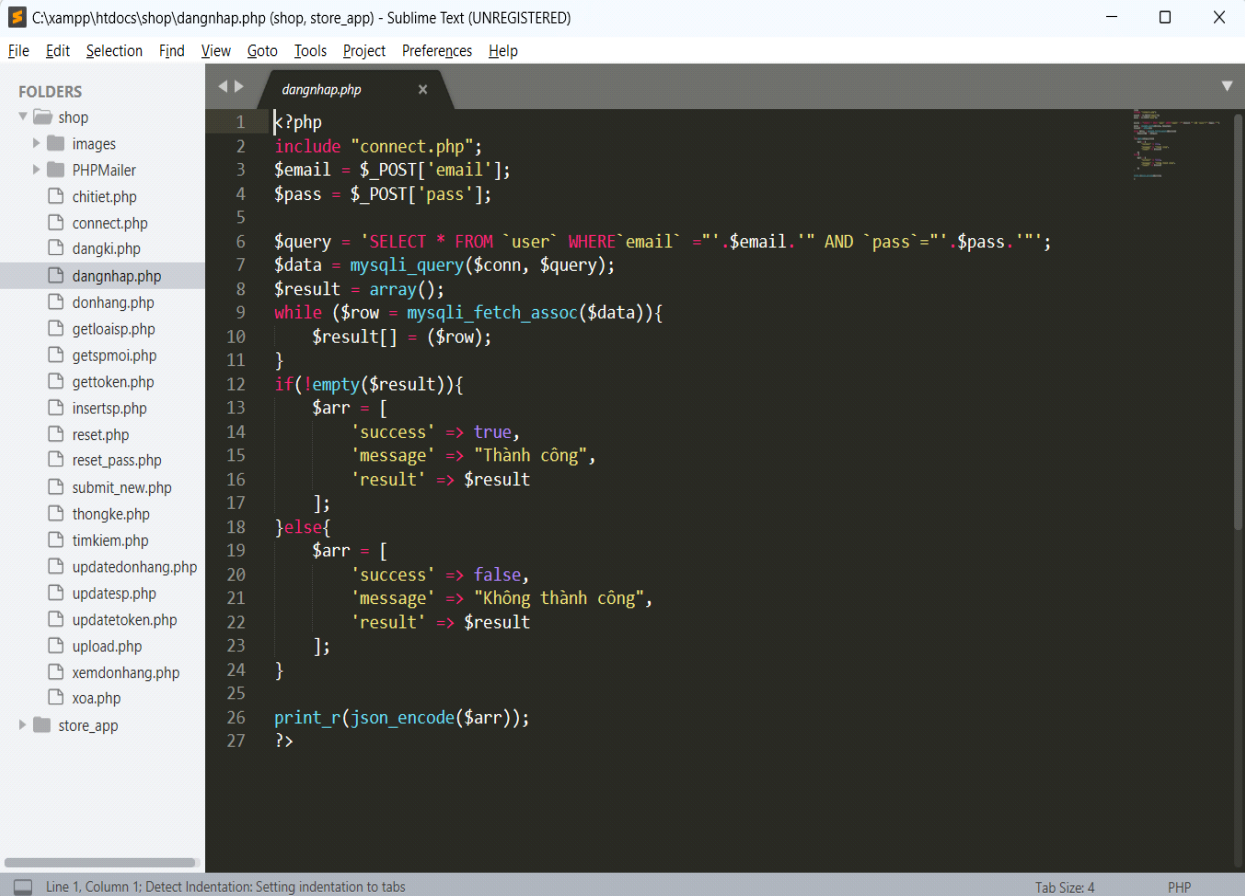
Mỗi tài khoản có thể tự chỉnh sửa thông tin cá nhân và đổi mật khẩu mà không cần đến quản lý.

## HƯỚNG PHÁT TRIỂN

- Nâng cao tốc độ thực thi và hiệu suất của hệ thống.
- Tối ưu hiện đại hóa giao diện đẹp mắt, dễ sử dụng hơn.
- Nâng cao tính năng bảo mật của hệ thống.
- Làm thêm chức năng như đăng nhập bằng vân tay.
- Nâng cấp chức năng nhập sản phẩm bằng cách quét mã QR
- Phân quyền chức năng cho mỗi tài khoản
- Nâng cao cơ chế bảo mật cho hệ thống.

# PHỤ LỤC

Một số hình ảnh source code mẫu



```
1 <?php
2 include "connect.php";
3 $email = $_POST['email'];
4 $pass = $_POST['pass'];
5
6 $query = 'SELECT * FROM `user` WHERE `email` = "'. $email. '" AND `pass` = "'. $pass. "'';
7 $data = mysqli_query($conn, $query);
8 $result = array();
9 while ($row = mysqli_fetch_assoc($data)){
10     $result[] = ($row);
11 }
12 if(!empty($result)){
13     $arr = [
14         'success' => true,
15         'message' => "Thành công",
16         'result' => $result
17     ];
18 }else{
19     $arr = [
20         'success' => false,
21         'message' => "Không thành công",
22         'result' => $result
23     ];
24 }
25
26 print_r(json_encode($arr));
27 ?>
```

```
C:\xampp\htdocs\shop\dangki.php (shop, store_app) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS
shop
  images
  PHPMailer
  chitiet.php
  connect.php
  dangki.php
  dangnhap.php
  donhang.php
  getloaisp.php
  getspmoi.php
  gettoken.php
  insertsp.php
  reset.php
  reset_pass.php
  submit_new.php
  thongke.php
  timkiem.php
  updatedonhang.php
  updatesp.php
  updatetoken.php
  upload.php
  xemdonhang.php
  xoa.php
store_app

dangki.php
1 <?php
2 include "connect.php";
3 $email = $_POST['email'];
4 $pass = $_POST['pass'];
5 $username = $_POST['username'];
6 $mobile = $_POST['mobile'];
7 $uid = $_POST['uid'];
8
9 // Check coi email co chua
10 $query = 'SELECT * FROM `user` WHERE `email` = "'. $email ."'";
11 $data = mysqli_query($conn, $query);
12 $numrow = mysqli_num_rows($data);
13
14 if($numrow > 0) {
15     $arr = [
16         'success' => false,
17         'message' => "Email đã tồn tại"
18     ];
19 }else{
20     $query = 'INSERT INTO `user`(`email`, `pass`, `username`, `mobile`, `uid`) VALUES ("'.
21         $email ."', "'. $pass ."', "'. $username ."', "'. $mobile ."', "'. $uid ."'");
22     $data = mysqli_query($conn, $query);
23
24     if($data == true){
25         $arr = [
26             'success' => true,
27             'message' => "Thành công"
28         ];
29     }else{
30         $arr = [
31             'success' => false,
32             'message' => "Không thành công"
33         ];
34     }
35     print_r(json_encode($arr));
36 }>
```

```
C:\xampp\htdocs\shop\xoa.php (shop, store_app) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS
shop
  images
  PHPMailer
  chitiet.php
  connect.php
  dangki.php
  dangnhap.php
  donhang.php
  getloaisp.php
  getspmoi.php
  gettoken.php
  insertsp.php
  reset.php
  reset_pass.php
  submit_new.php
  thongke.php
  timkiem.php
  updatedonhang.php
  updatesp.php
  updatetoken.php
  upload.php
  xemdonhang.php
  xoa.php
store_app

xoa.php
1 <?php
2 include "connect.php";
3 $id = $_POST['id'];
4 $query = 'DELETE FROM `sanphammoi` WHERE `id` = '.$id;
5 $data = mysqli_query($conn, $query);
6
7 if($data == true){
8     $arr = [
9         'success' => true,
10        'message' => "Xóa thành công"
11    ];
12 }else{
13     $arr = [
14         'success' => false,
15         'message' => "Xóa không thành công"
16    ];
17 }
18 print_r(json_encode($arr));
19 ?>
```

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:background="@color/white"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".activity.SplashActivity">
<ImageView
    app:layout_constraintTop_toTopOf="parent"
    android:src="@drawable/svg"
    android:adjustViewBounds="true"
    android:layout_width="match_parent"
    android:layout_height="match_parent"/>
<com.airbnb.lottie.LottieAnimationView
    android:id="@+id/animation_view"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:lottie_autoPlay="true"
    app:lottie_loop="true"
    app:lottie_rawRes="@raw/loading"
    android:layout_width="150dp"
    android:layout_height="150dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>

```

```

1 package com.example.shop_java.utils;
2
3 import ...
4
5
6
7
8
9 public class Utils {
10     public static final String BASE_URL="http://192.168.1.8/shop/";
11
12     public static List<GioHang> manggiohang;
13
14     public static List<GioHang> mangmuahang = new ArrayList<>();
15
16     public static User user_current = new User();
17
18     public static String ID_RECEIVED;
19
20     public static final String SENDID = "idsend";
21     public static final String RECEIVEDID = "idreceived";
22     public static final String MESS = "message";
23     public static final String DATETIME = "datetime";
24     public static final String PATH_CHAT = "chat";
25 }
26 |

```

```

public class SplashActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash);
        Paper.init(context: this);
        Thread thread = new Thread(){
            public void run(){
                try {
                    sleep(millis: 2000);
                } catch (Exception ex) {

                }

                Intent home = new Intent(getApplicationContext(), MainActivity.class);
                startActivity(home);
                finish();
            }
        };
        thread.start();
    }
}

```

```

72
73
74 @Override
75 protected void onCreate(Bundle savedInstanceState) {
76     super.onCreate(savedInstanceState);
77     setContentView(R.layout.activity_main);
78     apiShop = RetrofitClient.getInstance(Utils.BASE_URL).create(ApiShop.class);
79     Paper.init(context: this);
80     if(Paper.book().read(key: "user") != null){
81         User user = Paper.book().read(key: "user");
82         Utils.user_current = user;
83     }
84     getToken();
85     App();
86     ActionBar();
87     if(isConnected(context: this)){
88         ActionViewFlipper();
89         getLoaiSanPham();
90         getSpMoi();
91         getEventClick();
92     }else{
93         Toast.makeText(getApplicationContext(), text: "Không có Internet, vui lòng kết nối Internet", Toast.LENGTH_LONG).show();
94     }
95 }

```

## TÀI LIỆU THAM KHẢO

[1] Thầy Bùi Tiên Đức, bài giảng môn Lập trình thiết bị di động, Khoa Công nghệ thông tin, trường Đại học Nguyễn Tất Thành.

[2] Android – UI Layouts,

[https://www.tutorialspoint.com/android/android\\_user\\_interface\\_layouts.htm](https://www.tutorialspoint.com/android/android_user_interface_layouts.htm),

ngày tham khảo 08/2023

[3] Android Tutorial, <https://www.geeksforgeeks.org/android-tutorial/>, ngày tham

khảo 08/2023

[4] Cách tạo Splash Screen thế nào cho đúng và hiệu quả :

<https://viblo.asia/p/cach-cao-splash-screen-the-nao-cho-dung-va-hieu-qua-63vKjJBAK2R>, ngày tham khảo 07/2023