**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**ĐỒ ÁN**

**XÂY DỰNG ỨNG DỤNG GIỚI THIỆU CỬA HÀNG TIỆN LỢI**

**HÀ NỘI**

SINH VIÊN:

MÃ LỚP:

HƯỚNG DẪN:

**– 2020**

NHẬN XÉT

**Nhận xét của giảng viên hướng dẫn:**

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**LỜI CAM ĐOAN**

Em xin cam đoan đồ án “Xây dựng ứng dụng giới thiệu cửa hàng tiện lợi Hà Nội” là kết quả thực hiện của bản thân em dưới sự hướng dẫn của cô

Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong đồ án và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiện.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

*, ngày 20 tháng 12 năm 2021*

Sinh viên

**LỜI CẢM ƠN**

Để có thể hoàn thành đồ án này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn cô đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện đồ án vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được đồ án này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

**MỤC LỤC**

[NHẬN XÉT 2](#_Toc92145896)

[DANH MỤC CÁC THUẬT NGỮ 7](#_Toc92145897)

[DANH MỤC CÁC BẢNG 8](#_Toc92145898)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ 9](#_Toc92145899)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 10](#_Toc92145900)

[1.1. Lý do chọn đề tài 10](#_Toc92145901)

[1.2. Mục tiêu của đề tài 10](#_Toc92145902)

[1.2.1 Mục tiêu tổng quát 10](#_Toc92145903)

[1.2.2 Mục tiêu cụ thể 10](#_Toc92145904)

[1.3. Giới hạn và phạm vi của đề tài 11](#_Toc92145905)

[1.3.1 Đối tượng nghiên cứu 11](#_Toc92145906)

[1.3.2 Phạm vi nghiên cứu 11](#_Toc92145907)

[1.4. Nội dung thực hiện 11](#_Toc92145908)

[1.5. Phương pháp tiếp cận 11](#_Toc92145909)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 12](#_Toc92145910)

[2.1. Quy trình phát triển phần mềm 12](#_Toc92145911)

[2.2. Phương pháp phân tích thiết kế hướng đối tượng 12](#_Toc92145912)

[2.3. Giới thiệu tổng quan về hệ quản trị Cơ sở dữ liệu SQLite 18](#_Toc92145913)

[2.4. Giới thiệu về drawble và animation 20](#_Toc92145914)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 23](#_Toc92145915)

[3.1 Đặc tả yêu cầu phần mềm 23](#_Toc92145916)

[3.1.1 Các yêu cầu chức năng 23](#_Toc92145917)

[3.1.2 Biểu đồ lớp thực thể 32](#_Toc92145918)

[3.1.3 Các yêu cầu phi chức năng 32](#_Toc92145919)

[3.2 Thiết kế hệ thống 33](#_Toc92145920)

[3.2.1 Thiết kế lớp đối tượng 33](#_Toc92145921)

[3.2.2 Thiết kế cơ sở dữ liệu 38](#_Toc92145922)

[3.2.3 Thiết kế giao diện 40](#_Toc92145923)

[CHƯƠNG 4: TRIỂN KHAI ỨNG DỤNG 47](#_Toc92145924)

[4.1 Triển khai các chức năng của hệ thống 47](#_Toc92145925)

[4.1.1 Chức năng xem sản phẩm theo danh mục 47](#_Toc92145926)

[4.1.2 Chức năng chọn và xem chi tiết sản phẩm 54](#_Toc92145927)

[4.1.3 Chức năng thanh toán 59](#_Toc92145928)

[4.1.4 Chức năng lưu thông tin khách hàng 64](#_Toc92145929)

[4.1.5 Chức năng liên hệ 69](#_Toc92145930)

[4.2 Kiểm thử và triển khai ứng dụng 71](#_Toc92145931)

[4.2.1 Kiểm thử 71](#_Toc92145932)

[4.2.2 Đóng gói ứng dụng 73](#_Toc92145933)

[4.2.3 Triển khai ứng dụng 73](#_Toc92145934)

[KẾT LUẬN 74](#_Toc92145935)

[TÀI LIỆU THAM KHẢO 75](#_Toc92145936)

DANH MỤC CÁC THUẬT NGỮ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Từ viết tắt | Cụm từ tiếng anh | Diễn giải |
| 1 | CSDL | Data base | Cơ sở dữ liệu |
| 2 | OOAD | Object Oriented Analysis and Design | Phân tích và thiết kế hướng đối tượng |
| 3 | UML | Unified Modeling Language | Ngôn ngữ mô hình hóa thống nhất |
| 4 | SQL | Structured Query Language | Ngôn ngữ truy vấn dữ liệu |

DANH MỤC CÁC BẢNG

* + - * + [Bảng 1. 1 Ngôn ngữ định nghĩa dữ liệu 20](#_Toc91365561)
        + [Bảng 1. 2 Ngôn ngữ thao tác dữ liệu 20](#_Toc91365562)
        + [Bảng 1. 3 Ngôn ngữ truy vấn dữ liệu 20](#_Toc91365563)
        + [Bảng 1. 4 Dòng sự kiện quản lí thông tin cá nhân 24](#_Toc91365564)
        + [Bảng 1. 5 Dòng sự kiện quản lí danh mục sản phẩm 25](#_Toc91365565)
        + [Bảng 1. 6 Dòng sự kiện quản lí thông tin sản phẩm 26](#_Toc91365566)
        + [Bảng 1. 7 Dòng sự kiện quản lí đặt hàng và thanh toán hóa đơn 27](#_Toc91365567)
        + [Bảng 1. 8 Dòng sự kiện quản lí khách hàng 28](#_Toc91365568)
        + [Bảng 1. 9 Dòng sự kiện quản lí thông tin cá nhân 29](#_Toc91365569)
        + [Bảng 1. 10 Chức năng của phân hệ người dùng 29](#_Toc91365570)
        + [Bảng 1. 11 Phân tích yêu cầu chức năng 30](#_Toc91365571)
        + [Bảng 1. 12 Các yêu cầu phi chức năng 31](#_Toc91365572)
        + [Bảng 1. 13 Danh sách thuộc tính lớp khách hàng 33](#_Toc91365573)
        + [Bảng 1. 14 Danh sách phương thức lớp khách hàng 33](#_Toc91365574)
        + [Bảng 1. 15 Danh sách phương thức lớp sản phẩm 34](#_Toc91365575)
        + [Bảng 1. 16 Danh sách thuộc tính lớp sản phẩm 34](#_Toc91365576)
        + [Bảng 1. 17 Danh sách phương thức lớp hóa đơn 34](#_Toc91365577)
        + [Bảng 1. 18 Danh sách thuộc tính lớp hóa đơn 35](#_Toc91365578)
        + [Bảng 1. 19 Bảng danh sách thuộc tính lớp Admin 35](#_Toc91365579)
        + [Bảng 1. 20 Danh sách phương thức lớp chi tiết hóa đơn 35](#_Toc91365580)
        + [Bảng 1. 21 Danh sách thuộc tính lớp chi tiết hóa đơn 36](#_Toc91365581)
        + [Bảng 1. 22 Danh sách phương thức lớp chi tiết sản phẩm 36](#_Toc91365582)
        + [Bảng 1. 23 Danh sách thuộc tính lớp chi tiết sản phẩm 37](#_Toc91365583)
        + [Bảng 1. 24 Bảng cơ sở dữ liệu khách hàng 37](#_Toc91365584)
        + [Bảng 1. 25 Bảng cơ sở dữ liệu sản phẩm 37](#_Toc91365585)
        + [Bảng 1. 26 Bảng cơ sở dữ liệu đơn hàng 38](#_Toc91365586)
        + [Bảng 1. 27 Bảng cơ sở dữ liệu chi tiết sản phẩm 38](#_Toc91365587)
        + [Bảng 1. 28 Bảng cơ sở dữ liệu chi tiết đơn hàng 38](#_Toc91365588)
        + [Bảng 1. 29 Bảng kiểm thử chức năng mua hàng 70](#_Toc91365589)
        + [Bảng 1. 30 Bảng kiểm thử chức năng thanh toán 70](#_Toc91365590)
        + [Bảng 1. 31 Bảng kiểm thử chức năng lưu thông tin khách hàng 70](#_Toc91365591)

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

* + - * + [Hình 1. 1 Các View trong OOAD sử dụng UML 15](#_Toc91366524)
        + [Hình 1. 2 Các bản vẽ trong OOAD sử dụng UML 16](#_Toc91366525)
        + [Hình 1. 3 Kí hiệu về Use Case 17](#_Toc91366526)
        + [Hình 1. 4 Kí hiệu về Class 17](#_Toc91366527)
        + [Hình 1. 5 Kí hiệu về Actor 18](#_Toc91366528)
        + [Hình 3. 1 Biểu đồ user case tổng quát 23](#_Toc92145961)
        + [Hình 3. 2 Biểu đồ use case chọn theo danh mục sản phẩm 23](#_Toc92145962)
        + [Hình 3. 3 Biểu đồ user case xem thông tin sản phẩm 24](#_Toc92145963)
        + [Hình 3. 4 Biểu đồ use case thanh toán 26](#_Toc92145964)
        + [Hình 3. 5 Biểu đồ usecase liên hệ 27](#_Toc92145965)
        + [Hình 3. 6 Biểu đồ use case quản lí khach hàng 28](#_Toc92145966)
        + [Hình 3. 7 Biểu đồ use case quản lí sản phẩm 29](#_Toc92145967)
        + [Hình 3. 9 Biểu đồ lớp thực thể 32](#_Toc92145968)
        + [Hình 3. 10 Biều đồ tuần tự danh mục sản phẩm 33](#_Toc92145969)
        + [Hình 3. 11 Biểu đồ tuần tự thông tin sản phẩm 34](#_Toc92145970)
        + [Hình 3. 12 Biểu đồ tuần tự thanh toán 34](#_Toc92145971)
        + [Hình 3. 13 Giao diện chính 41](#_Toc92145972)
        + [Hình 3. 14 Giao diện danh mục 42](#_Toc92145973)
        + [Hình 3. 15 Giao diện sản phẩm 43](#_Toc92145974)
        + [Hình 3. 16 Giao diện chi tiết sản phẩm 44](#_Toc92145975)
        + [Hình 3. 17 Giao diện giỏ hàng 45](#_Toc92145976)
        + [Hình 3. 18 Giao diện thông tin khách hàng 46](#_Toc92145977)
        + [Hình 3. 19 Đóng gói ứng dụng 73](#_Toc92145978)

# TỔNG QUAN VỀ ĐỀ TÀI

1.1. Lý do chọn đề tài

* Tầm quan trọng, vai trò của đề tài:

Hiện nay ở Hà Nội, khi làm kinh doanh bán hàng để có lượng khách hàng nhiều nhất có thể là vấn đề mà bất kỳ một người chủ rất đau đầu. Từ việc tìm nguồn hàng chất lượng, vận chuyển, tìm kiếm khách hàng đến quảng cáo thương hiệu đến các công việc bán hàng, cân đối thu-chi và tính toán chi phí để có lãi và tái đầu tư tiếp. Thật vậy, khi mức sống ngày càng nâng cao, nhu cầu của người dùng Hà Nội ngày càng tăng cao, hàng loạt các cửa hàng được mở ra giúp khách hàng tiện lợi hơn trong việc mua sắm. Chính vì vậy, nên em đã đã xây dựng ứng dụng giới thiệu cửa hàng tiện lợi Hà Nội để giúp các cửa hàng giải quyết được các vấn đề trên một cách tự động, nhanh chóng, chính xác mà lại có tính bảo mật cao.

* Tính cấp thiết của đề tài:

+ Giúp khách hàng có thể tiếp cận được các sản phẩm của cửa hàng

+ Khách hàng có thể thoải mái lựa chọn đặt mua tất cả các sản phẩm của cửa hàng ở bất cứ mọi nơi

+ Giúp cửa hàng phát triển mạnh

+ Giúp khách hàng có cái nhìn mới trong cuộc sống hiện đại ngày nay

* Những bất cập, hạn chế của hệ thống cũ:

+ Giao diện chưa được ưa nhìn

1.2. Mục tiêu của đề tài

1.2.1 Mục tiêu tổng quát

Xây dựng ứng dụng giới thiệu cửa hàng tiện lợi Hà Nội

1.2.2 Mục tiêu cụ thể

Phát triển phần mềm gồm các chức năng:

- Lựa chọn, mua sản phẩm

- Thanh toán chi phí

- Nhập thông tin khách hàng

1.3. Giới hạn và phạm vi của đề tài

1.3.1 Đối tượng nghiên cứu

- Đối tượng nghiên cứu: Khách hàng

- Khách thể nghiên cứu: Khách hàng và quản lý

1.3.2 Phạm vi nghiên cứu

* Phạm vi không gian: Tại Hà Nội
* Phạm vi thời gian: Giờ hành chính
* Ý nghĩa khoa học và thực tiễn của đề tài: Phát triển ứng dụng một cách mạch lạc, giúp người khách hàng có thể xử lý nhanh gọn, các thủ tục trở nên đơn giản và nhanh hơn.

1.4. Nội dung thực hiện

- Nghiên cứu các sản phẩm mà khách hàng sử dụng hiện nay

- Giới thiệu các sản phẩm đảm bảo chất lượng

- Phân tích thiết kế ứng dụng giới thiệu cửa hàng tiện lợi

- Lập trình ứng dụng

- Kiểm thử ứng dụng

1.5. Phương pháp tiếp cận

- Tìm hiểu các sản phẩm của các cửa hàng ở 1 số địa điểm lớn trong nước

- Quản lý ứng dụng: Thiết kế phù hợp, dễ thao tác

- Thông tin ứng dụng: chứa các tin tức liên quan đến phần mềm

# CƠ SỞ LÝ THUYẾT

* 1. Quy trình phát triển phần mềm

Quy trình phát triển phần mềm là một cấu trúc bao gồm tập hợp các thao tác và các kết quả tương quan sử dụng trong việc phát triển để sản xuất ra một phần mềm.

Một cách đơn giản, chúng ta có thể chia quy trình phát triển phần mềm ra làm 4 giai đoạn:

Đặc tả phần mềm: là tiến trình để hiểu và xác định những dịch vụ nào cần có trong hệ thống, những ràng buộc đối với việc phát triển và chức năng của hệ thống. Tiến trình này sẽ sinh ra các tài liệu yêu cầu (bản đặc tả hệ thống).

Thiết kế và thực thi phần mềm: liên quan đến việc chuyển những yêu cầu phần mềm thành hệ thống có thể thực thi được

- Thiết kế phần mềm: là việc mô tả cấu trúc phần mềm, dữ liệu của hệ thống, giao diện giao tiếp giữa các thành phần, thuật toán được sử dụng, …

- Thực thi phần mềm: Các lập trình viên dùng các ngôn ngữ lập trình để viết lệnh (source) thực thi để tạo ra hệ thống dựa trên các bản đặc tả thiết kế chi tiết, đồng thời tiến hành các thử nghiệm (test case) với dữ liệu giả định.

Kiểm thử phần mềm: là quá trình vận hành chương trình để tìm ra lỗi

Cài đặt và bảo trì phần mềm:

Cài đặt và triển khai hệ thống vừa phát triển để người dùng có thể sử dụng được.

Bảo trì phần mềm: điều chỉnh các lỗi chưa được phát hiện trong các giai đoán trước, nâng cao tính năng sử dụng và an toàn vận hành của phần mềm, đảm bảo cho việc phần mềm được cập nhật khi môi trường và yêu cầu của người sử dụng thay đổi.

* 1. Phương pháp phân tích thiết kế hướng đối tượng

 1. **Khái niệm về Phân tích và thiết kế hướng đối tượng (Object Oriented Analysis and Design: OOAD)**

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD).

**2. Giới thiệu về UML (Unified Modeling Language)**

UML (Unified Modeling Language) là ngôn ngữ dành cho việc đặc tả, hình dung, xây dựng và làm tài liệu của các hệ thống phần mềm.

UML tạo cơ hội để viết thiết kế  hệ thống, bao gồm những khái niệm như tiến trình nghiệp vụ và các chức năng của hệ thống.

Cụ thể, nó hữu dụng cho những ngôn ngữ khai báo, giản đồ cơ sở dữ liệu, thành phần phần mềm có khả năng tái sử dụng.

UML được phát triển bởi Rational Rose và một số nhóm cộng tác, nó nhanh chóng trở thành một trong những ngôn ngữ chuẩn để xây dựng hệ thống phần mềm hướng đối tượng (Object-Oriented).

Đây là ngôn ngữ kế vị xứng đáng cho những ngôn ngữ mô hình hoá như Booch, OOSE/Jacobson, OMT và một số các phương thức khác.

UML cung cấp cho người dùng một ngôn ngữ mô hình hoá trực quan sẵn sàng để dùng và có ý nghĩa:

  Cho phép phát triển và trao đổi những mô hình mang nhiều ý nghĩa.

  Cung cấp khả năng mở rộng và chuyên môn hoá để mở rộng những khái niệm cốt lõi.

  Độc lập với ngôn ngữ lập trình chuyên biệt và các tiến trình phát triển.

  Cung cấp nền tảng về sự hiểu biết ngôn ngữ mô hình hoá.

 Khuyến khích và hỗ trợ sự phát triển của các công cụ hướng đối tượng.

Hỗ trợ những khái niệm phát triển cấp độ cao như collaboration, framework, pattern and component.

  Tích hợp một cách tốt nhất với thực tiễn.

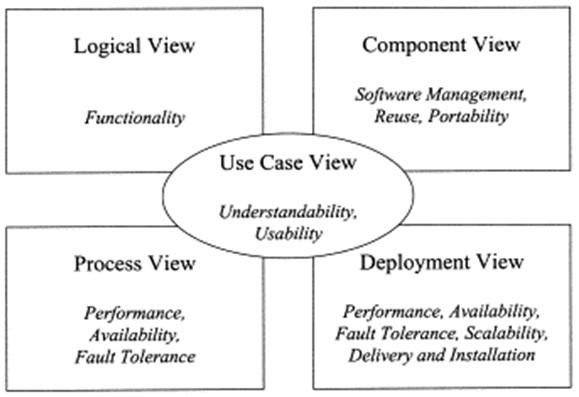
**3.  Tại sao lại là OOAD và UML?**

OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện.  Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.

**4. OOAD sử dụng UML**

UML sử dụng để vẽ cho nhiều lĩnh vực khác nhau như phần mềm, cơ khí, xây dựng v… trong phạm vi các bài viết này chúng ta chỉ nghiên cứu cách sử dụng UML cho phân tích và thiết kế hướng đối tượng trong ngành phần mềm. OOAD sử dụng UML bao gồm các thành phần sau:

View (góc nhìn): Mỗi góc nhìn như thầy bói xem voi, nó không thể hiện hết hệ thống nhưng thể hiện rõ hệ thống ở một khía cạnh. Chính vì thế trong xây dựng có bản vẽ kiến trúc (nhìn về mặt kiến trúc), bản vẽ kết cấu (nhìn về mặt kết cấu), bản vẽ thi công (nhìn về mặt thi công). Trong phần mềm cũng như vậy, OOAD sử dụng UML có các góc nhìn sau:



Hình 1. 1 Các View trong OOAD sử dụng UML

Trong đó:

+ Use Case View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.

+ Logical View: cung cấp góc nhìn về cấu trúc hệ thống, xem nó được tổ chức như thế nào. Bên trong nó có gì.

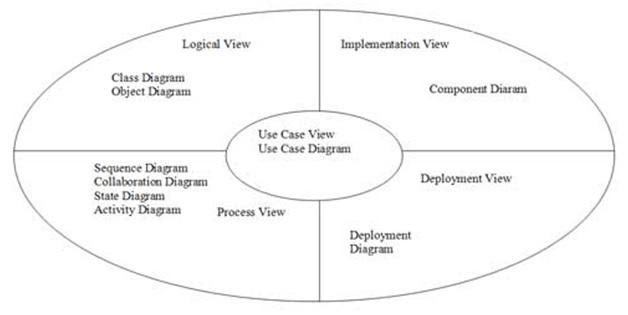
+ Process View: cung cấp góc nhìn động về hệ thống, xem các thành phần trong hệ thống tương tác với nhau như thế nào.

+ Component View: cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.

+ Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Trong hình 1 chúng ta thấy góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của Use Case View.

Diagram (bản vẽ): Diagram các bạn có thể dịch là sơ đồ. Tuy nhiên ở đây chúng ta sử dụng từ bản vẽ cho dễ hình dung. Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.



Hình 1. 2 Các bản vẽ trong OOAD sử dụng UML

Trong đó:

+ Use Case Diagram: bản vẽ mô tả về ca sử dụng của hệ thống. Bản vẽ này sẽ giúp chúng ta biết được ai sử dụng hệ thống, hệ thống có những chức năng gì. Lập được bản vẽ này bạn sẽ hiểu được yêu cầu của hệ thống cần xây dựng.

+ Class Diagram: bản vẽ này mô tả cấu trúc của hệ thống, tức hệ thống được cấu tạo từ những thành phần nào. Nó mô tả khía cạnh tĩnh của hệ thống.

+ Object Diagram: Tương tự như Class Diagram nhưng nó mô tả đến đối tượng thay vì lớp (Class).

+ Sequence Diagram: là bản vẽ mô tả sự tương tác của các đối tượng trong hệ thống với nhau được mô tả tuần tự các bước tương tác theo thời gian.

+ Collaboration Diagram: tương tự như sequence Diagram nhưng nhấn mạnh về sự tương tác thay vì tuần tự theo thời gian.

+ State Diagram: bản vẽ mô tả sự thay đổi trạng thái của một đối tượng. Nó được dùng để theo dõi các đối tượng có trạng thái thay đổi nhiều trong hệ thống.

+ Activity Diagram: bản vẽ mô tả các hoạt động của đối tượng, thường được sử dụng để hiểu về nghiệp vụ của hệ thống.

+ Component Diagram: bản vẽ mô tả về việc bố trí các thành phần của hệ thống cũng như việc sử dụng các thành phần đó.

+ Deployment Diagram: bản vẽ mô tả việc triển khai của hệ thống như việc kết nối, cài đặt, hiệu năng của hệ thống v.v…

+ Notations (ký hiệu): Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Bạn phải biết từ vựng thì mới ghép thành câu, thành bài được. Chúng ta sẽ tìm hiểu kỹ các notations trong từng bản vẽ sau này. Dưới đây là vài ví dụ về notation.



Hình 1. 3 Kí hiệu về Use Case



Hình 1. 4 Kí hiệu về Class



Hình 1. 5 Kí hiệu về Actor

Mechanisms (quy tắc, cơ chế): Mechanisms là các quy tắc để lập nên bản vẽ, mỗi bản vẽ có quy tắc riêng và bạn phải nắm được để tạo nên các bản vẽ thiết kế đúng. Các quy tắc này chúng ta sẽ bàn kỹ trong các bài về các bản vẽ.

* 1. Giới thiệu tổng quan về hệ quản trị Cơ sở dữ liệu SQLite

**1**. **SQLite là gì?**

SQLite là một thư viện phần mềm mà triển khai một SQL Database Engine, không cần máy chủ, không cần cấu hình, khép kín và nhỏ gọn. Nó là một cơ sở dữ liệu, không cần cấu hình, có nghĩa là giống như các cơ sở dữ liệu khác mà bạn không cần phải cấu hình nó trong hệ thống của mình.

SQLite engine không phải là một quy trình độc lập (standalone process) như các cơ sở dữ liệu khác, bạn có thể liên kết nó một cách tĩnh hoặc động tùy theo yêu cầu của bạn với ứng dụng của bạn. SQLite truy cập trực tiếp các file lưu trữ (storage files) của nó.

**2. Một số đặc điểm chính của SQLite**

SQLite không cần mô hình client – server để hoạt động.

SQLite không cần phải cấu hình tức là bạn không cần phải cài đặt.

Với SQLite database được lưu trữ trên một tập tin duy nhất.

SQLite hỗ trợ hầu hết các tính năng của ngôn ngữ truy vấn SQL theo chuẩn SQL92.

SQLite rất nhỏ gọn bản đầy đủ các tính năng nhỏ hơn 500kb, và có thể nhỏ hơn nếu lược bớt một số tính năng.

Các thao tác dữ liệu trên SQLite chạy nhanh hơn so với các hệ quản trị cơ sở dữ liệu theo mô hình client – server.

SQLite rất đơn giản và dễ dàng sử dụng.

SQLite tuân thủ 4 tính chất ACID (là tính nguyên tổ (Atomic), tính nhất quán (Consistent), tính cô lập (Isolated), và tính bền vững (Durable)).

Với đặc tính nhỏ gọn, truy xuất dữ liệu nhanh SQLite thường được sử dụng để nhúng vào các dự án.

**Nhược điểm của SQLite**

* Do sử dụng cơ chế coarse-gained locking nên trong cùng một thời điểm SQLite có thể hỗ trợ nhiều người đọc dữ liệu, nhưng chỉ có một người ghi được dữ liệu.
* SQLite không phải là lựa chọn hoàn hảo để đáo ứng nhu cầu xử lý trên một khối lượng dữ liệu lớn, phát sinh liên tục.

**Truy vấn trên SQlite**

SQLite là một CSDL vì vậy có thể sử dụng tất cả các câu lệnh SQL để thao tác trên CSDL SQLite này: CREATE, SELECT, INSERT, UPDATE, DELETE và DROP.

**DDL - Ngôn ngữ định nghĩa dữ liệu (Data Definition Language)**

|  |  |
| --- | --- |
| Lệnh | Mô tả |
| CREATE | Tạo mới một bảng, view của bảng hoặc đối tượng khác trong cơ sở dữ liệu. |
| ALTER | Sửa đổi một đối tượng cơ sở dữ liệu đang tồn tại, chẳng hạn như bảng. |
| DROP | Xóa toàn bộ bảng, view của bảng hoặc đối tượng khác trong cơ sở dữ liệu. |

Bảng 1. 1 Ngôn ngữ định nghĩa dữ liệu

**DML - Ngôn ngữ thao tác dữ liệu (Data Manipulation Language)**

|  |  |
| --- | --- |
| Lệnh | Mô tả |
| INSERT | Tạo một bản ghi |
| UPDATE | Sửa một bản ghi |
| DELETE | Xóa một bản ghi |

Bảng 1. 2 Ngôn ngữ thao tác dữ liệu

**DQL - Ngôn ngữ truy vấn dữ liệu (Data Query Language)**

|  |  |
| --- | --- |
| Lệnh | Mô tả |
| SELECT | Lấy một số bản ghi nhất định từ một hoặc nhiều bảng. |

Bảng 1. 3 Ngôn ngữ truy vấn dữ liệu

* 1. Giới thiệu về drawble và animation
     + - * Giới thiệu về drawable

Trong Android, drawable được sử dụng khá nhiều ví dụ như background của view, src của ImageView, state (press, normal..) của view và nhiều chức năng các nữa. Nhưng thực sự khi mới bắt đầu không phải ai cũng hiểu được drawable là cái 1

* + - * + Khái niệm

Drawable là khái niệm chung về graphics để chỉ những gì mà bạn có thể vẽ. Đơn giản là hình ảnh (Drawable hình ảnh trong android gọi là BitmapDrawable).

Drawable định nghĩa shape, color, gradient, border… mà bạn có thể sử dụng nó apply vào view trong Activity.

* **Các loại drawable trong Android**

Trong Android có khá nhiều loại drawable. Sử dụng nhiều nhất là BitmapDrawable (sử dụng để set src cho ImageView). Và dưới đây là một số drawable trong Android.

+ BimapDrawable

+ ColorDrawable

+ GradientDrawable

+ ShapeDrawable

+ RippleDrawable (Android 5.0)

+ VectorDrawable

+ AnimatedDrawable (Android 5.0)

+ StateListDrawable

+ Paths Drawables

Ngoài những Drawable trên thì bạn cũng có thể tạo Drawable khác kế thừa từ các Drawable trên (Lớp cha của tất cả drawable là lớp Drawable).

* + - * + Tổng quan về animation

Animation là các phép biến đổi màu sắc, vị trí, kích thước hay hướng của một đối tượng trên màn hình theo thời gian. Android cung cấp các API hỗ trợ rất tốt cho việc áp dụng animation cho các thành phần UI và vẽ đồ họa 2D, 3D.

Android 2.3 và các phiên bản trước đó hỗ trợ kiểu View Animation.

Android 3.0 và các phiên bản sau đã tăng cường khả năng Animation cho phép tạo hiệu ứng động cho các thuộc tính của thành phần UI thông qua kiểu Property Animation.

* **Drawable Animation**

Drawable Animation là quá trình đơn giản, hiển thị một chuỗi các hình ảnh liên tiếp trong các khoảng thời gian ngắn để tạo ra hiệu ứng cuối cùng là đối tượng di chuyển hoặc thay đổi.

Trong Android, việc thực hiện kiểu animation này được thông qua lớp AnimationDrawable.

* **View Animation**

View animation cho phép ta tạo hiệu ứng động với view tùy ý bằng việc thao tác với ma trận biến đổi.

Trong Android, việc hiển thị và biểu diễn sự thay đổi của một đối tượng view thông qua phép biến đổi ma trận. Với một tập các tọa độ đầu vào, chúng ta có thể biểu diễn những phép biến đổi đó thông qua phép nhân ma trận.

Android đưa ra ma trận biến đổi của một view bằng việc cho phép bạn đăng ký một đối tượng animation với view đó.

* **Các kiểu Tweening Animation cơ bản:**

+ **Scale animation (co):** làm cho một view nhỏ hơn hoặc lớn hơn dọc theo trục x hoặc trục y. Ta có thể chỉ định animation diễn ra xung quanh một điểm chốt (pivot point).

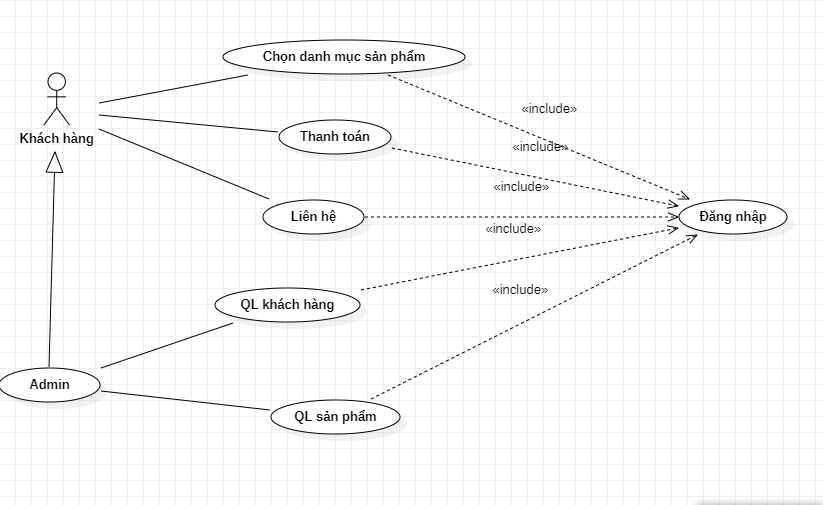
+ **Rotate animation (quay):** quay một view quanh một điểm chốt theo một góc quay xác định.

+ **Translate animation (tịnh tiến):** tịnh tiến một view dọc theo trục x hoặc trục y.

+ **Alpha animation (alpha):** thay đổi độ trong suốt của một view.

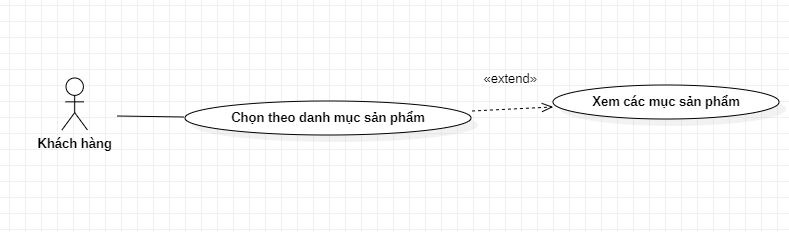
# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

* 1. Đặc tả yêu cầu phần mềm
     1. Các yêu cầu chức năng
        1. Chức năng của phân hệ quản trị nội dung
           + Biểu đồ use case tổng quát



Hình 3. 1 Biểu đồ user case tổng quát

* + - * + Biểu đồ use case chọn danh mục sản phẩm

**

Hình 3. 2 Biểu đồ use case chọn theo danh mục sản phẩm

* Chọn danh mục sản phẩm
* Mục đích

Chọn danh mục sản phẩm

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống

* Điều kiện sau

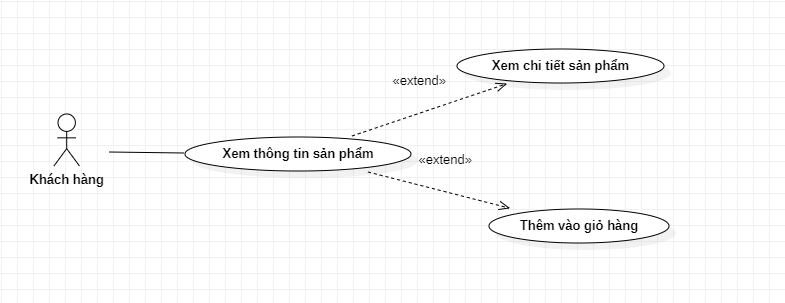
N/A

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Vào giao diện danh mục sản phẩm |
| Sau khi đăng nhập thành công | Hiển thị giao diện danh mục sản phẩm |
| Chọn chức năng quản lý danh mục sản phẩm | Hiển thị giao diện danh mục sản phẩm |
| Tùy chọn các chức năng của hệ thống |  |

Bảng 1. 5 Dòng sự kiện quản lí danh mục sản phẩm

* Biểu đồ xem thông tin sản phẩm



Hình 3. 3 Biểu đồ user case xem thông tin sản phẩm

* Xem thông tin sản phẩm
* Mục đích

Cho khách hàng xem thông tin sản phẩm

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống.

* Điều kiện sau

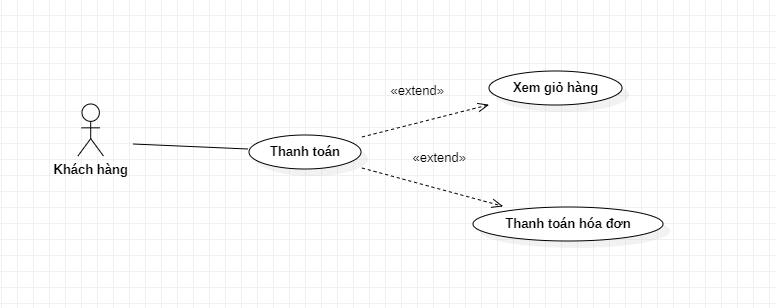
N/A

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Vào giao diện sản phẩm |
| Sau khi đăng nhập thành công | Hiển thị giao diện thông tin sản phẩm |
| Chọn chức năng quản lý thông tin sản phẩm | Hiển thị giao diện thông tin sản phẩm |
| Tùy chọn các chức năng của của hệ thống |  |

Bảng 1. 6 Dòng sự kiện quản lí thông tin sản phẩm

* Biểu đồ thanh toán



Hình 3. 4 Biểu đồ use case thanh toán

* Quản lý đặt hàng & thanh toán hóa đơn
* Mục đích

Quản lý đặt hàng & thanh toán hóa đơn

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống.

* Điều kiện sau

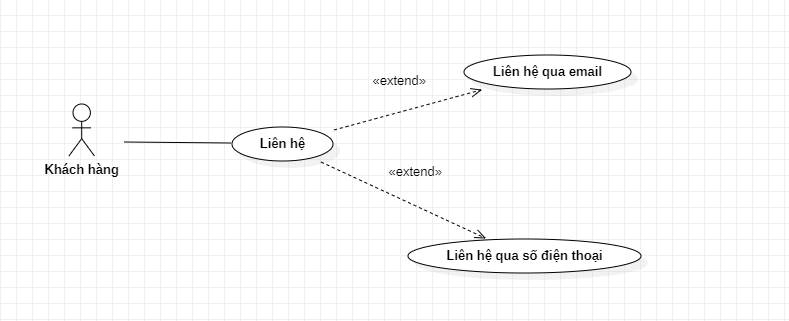
Hiển thị, thêm, sửa, xóa.

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Vào giao diện hóa đơn |
| Sau khi đăng nhập thành công | Hiển thị giao diện hóa đơn |
| Chọn chức năng quản lý hóa đơn | Hiển thị giao diện quản lý hóa đơn |
| Tùy chọn các chức năng của hệ thống |  |

Bảng 1. 7 Dòng sự kiện quản lí đặt hàng và thanh toán hóa đơn

* Liên hệ

**

Hình 3. 5 Biểu đồ usecase liên hệ

* Mục đích

Giúp khách hàng có thể liên hệ với cửa hàng

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống.

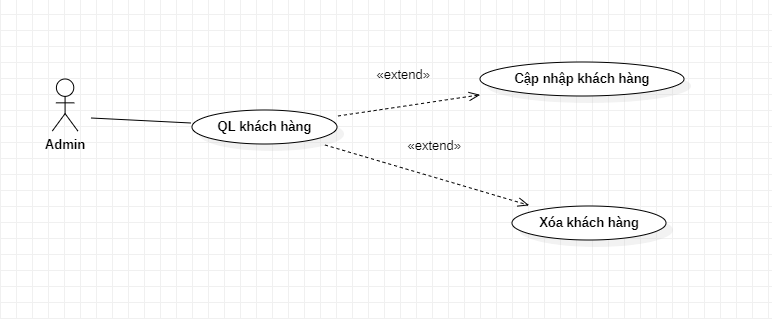
* Điều kiện sau

Liên hệ

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Vào giao diện các danh mục |
| Sau khi đăng nhập thành công | Hiển thị giao diện các danh mục |
| Chọn chức năng quản lý hóa đơn | Hiển thị giao diện liên hệ |
| Tùy chọn các chức năng của hệ thống |  |

* Biểu đồ quản lý khách hàng

**

Hình 3. 6 Biểu đồ use case quản lí khach hàng

* Quản lý khách hàng
* Mục đích

Quản lý khách hàng

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống

* Điều kiện sau

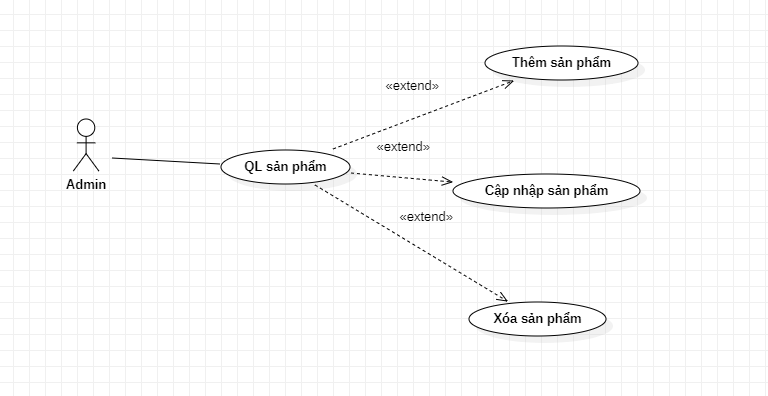
Hiển thị, thêm, sửa, xóa.

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Đăng nhập hệ thống |
| Sau khi đăng nhập thành công | Hiển thị giao diện quản lý khách hàng |
| Chọn chức năng quản lý khách hàng | Hiển thị giao diện quản lý khách hàng |
| Tùy chọn các chức năng của hệ thống |  |

Bảng 1. 8 Dòng sự kiện quản lí khách hàng

* Biểu đồ quản lý sản phẩm

**

Hình 3. 7 Biểu đồ use case quản lí sản phẩm

* Quản lý thông tin sản phẩm
* Mục đích

Quản lý sản phẩm

* Tác nhân

N/A

* Điều kiện trước

Phải đăng nhập được vào hệ thống

* Điều kiện sau

Hiển thị, thêm, sửa, xóa.

* Dòng sự kiện chính:

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Phản ứng của hệ thống** |
| Đăng nhập hệ thống | Vào giao diện chính |
| Sau khi đăng nhập thành công | Hiển thị giao diện quản lý sản phẩm |
| Chọn chức năng quản lý sản phẩm | Hiển thị giao diện quản lý sản phẩm |
| Tùy chọn các chức năng của hệ thống |  |

Bảng 1. 9 Dòng sự kiện quản lí thông tin cá nhân

* + - 1. Chức năng của phân hệ người
         * Khảo sát bài toán

|  |  |
| --- | --- |
| STT | Tên yêu cầu |
| 1 | Hiển thị thông tin các danh mục sản phẩm |
| 2 | Xem chi tiết các sản phẩm |
| 3 | Hóa đơn được thêm vào giỏ hàng |
| 4 | Hiển thị thông tin chi tiết hóa đơn |
| 5 | Thanh toán các hóa đơn |

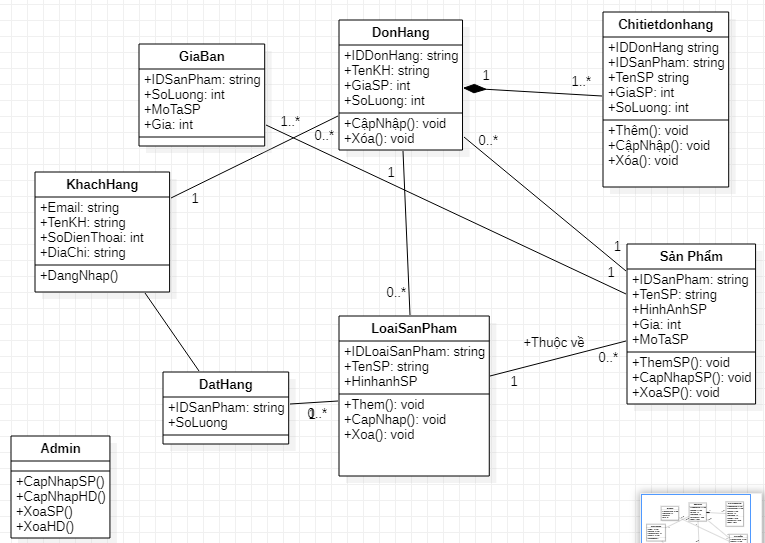
Bảng 1. 10 Chức năng của phân hệ người dùng

* + - * + Phân tích yêu cầu chức năng:

|  |  |  |
| --- | --- | --- |
| STT | Tên yêu cầu | Mô tả yêu cầu |
| **I Các yêu cầu chức năng nghiệp vụ** | | |
| 1 | Chức năng quản lý thông tin | Cập nhập thông tin cá nhân khách hàng |
| 2 | Chức năng chọn theo danh mục sản phẩm | Khách hàng có thể xem, lựa chọn các loại sản phẩm muốn mua ở cửa hàng |
| 3 | Chức năng thanh toán | Khách hàng thanh toán chi phí sản phẩm |
| 4 | Chức năng quản lý khách hàng, sản phẩm dành cho quản trị viên | Thêm, cập nhập chi tiết thông tin khách hàng, sản phẩm |
| **II Yêu cầu chức năng hệ thống** | | |
| 1 | Quản lý ứng dụng | * Quản lý khách hàng * Đăng ký thông tin khách hàng |
| 2 | Quản lý sản phẩm, hóa đơn | * Quản lý sản phẩm * Quản lý hóa đơn |

Bảng 1. 11 Phân tích yêu cầu chức năng

* + 1. Biểu đồ lớp thực thể



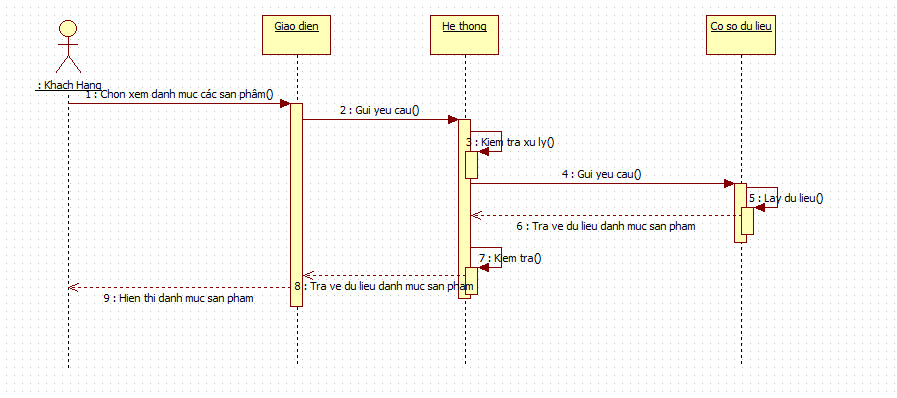
Hình 3. 9 Biểu đồ lớp thực thể

* + 1. Các yêu cầu phi chức năng

|  |  |  |
| --- | --- | --- |
| 1 | Giao diện | Giao diện ứng dụng dễ sử dụng, thân thiện, đáp ứng nhu cầu với khách hàng |
| 2 | Tốc độ xử lý | Tối ưu chương trình nhanh chóng, chính xác |
| 3 | Bảo mật dữ liệu | Tính bảo mật thông tin cao |
| 4 | Yêu cầu riêng tư | Phần mềm không tiết lộ thông tin khách hàng |
| 5 | Khả năng mở rộng | Có khả năng áp dụng cho nhiều đơn vị của cửa hàng |
| 7 | Khả năng phục hồi dữ liệu | Phục hồi và lấy lại thông tin dữ liệu đã bị mất |

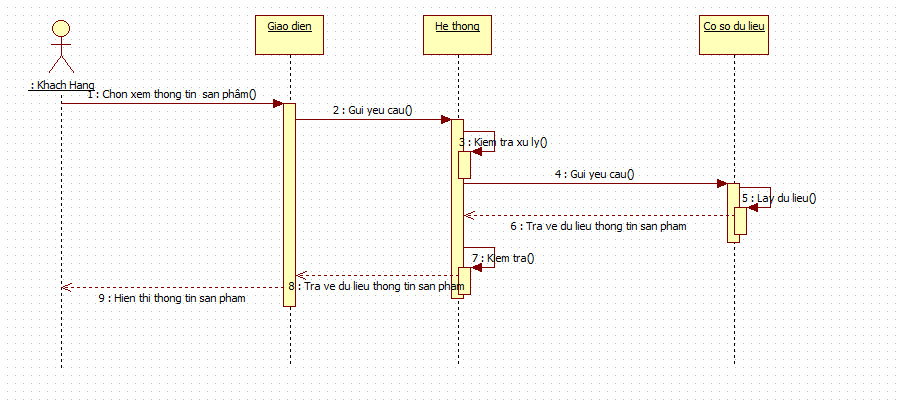
Bảng 1. 12 Các yêu cầu phi chức năng

* 1. Thiết kế hệ thống
     1. Thiết kế lớp đối tượng
        1. Biểu đồ lớp VOPC của các ca sử dụng
        2. Biểu đồ tuần tự
           + Ca danh mục sản phẩm



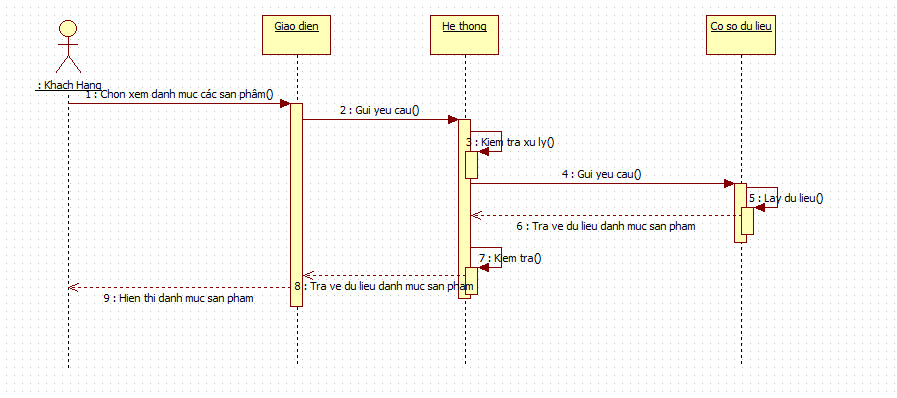
Hình 3. 10 Biều đồ tuần tự danh mục sản phẩm

* + - * + Ca thông tin sản phẩm



Hình 3. 11 Biểu đồ tuần tự thông tin sản phẩm

* + - * + Ca thanh toán



Hình 3. 12 Biểu đồ tuần tự thanh toán

* + - 1. Biểu đồ lớp chi tiết
         * Lớp khách hàng
* Danh sách thuộc tính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | TênKH | String | Khóa chính |  |
| 2 | Id | String |  |  |
| 3 | SoDienThoai | String |  |  |
| 4 | DiaChi | String |  |  |
| 5 | Email | String |  |  |

Bảng 1. 13 Danh sách thuộc tính lớp khách hàng

* Danh sách phương thức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | DangNhap | Void | Public | Đăng nhập |
| 2 | CapNhapThongTin | Void | Public | Cập nhập thông tin |

Bảng 1. 14 Danh sách phương thức lớp khách hàng

* + - * + Lớp sản phẩm
* Danh sách phương thức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | TenSP | String | Khóa chính |  |
| 2 | ID | String |  |  |
| 3 | TênSP | String |  |  |
| 4 | HinhAnh | String |  |  |

Bảng 1. 15 Danh sách phương thức lớp sản phẩm

* Danh sách thuộc tính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | ThemSP | Void | Public | Thêm sản phẩm |
| 2 | CapNhapSp | Void | Public | Cập nhập sản phẩm |
| 3 | XoaSP | Void | Public | Xóa sản phẩm |

Bảng 1. 16 Danh sách thuộc tính lớp sản phẩm

* + - * + Lớp hóa đơn
* Danh sách phương thức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | String | Khóa chính |  |
| 2 | TenKH | String |  |  |
| 3 | GiaSP | String |  |  |
| 4 | SoLuong | String |  |  |

Bảng 1. 17 Danh sách phương thức lớp hóa đơn

* Danh sách thuộc tính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | CapNhap | Void | Public | Cập nhập hóa đơn |
| 2 | Xoa | Void | Public | Xóa hóa đơn |

Bảng 1. 18 Danh sách thuộc tính lớp hóa đơn

* + - * + Lớp quản trị viên
* Danh sách thuộc tính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | DangNhap | Void | Public | Đăng nhập admin |
| 2 | CapNhapSP | Void | Public | Cập nhập sản phẩm |
| 3 | XoaSP | Void | Public | Xóa sản phẩm |
| 4 | CapNhapHD | Void | Public | Cập nhập hóa đơn |
| 5 | XoaHD | Void | Public | Xóa hóa đơn |

Bảng 1. 19 Bảng danh sách thuộc tính lớp Admin

* + - * + Lớp chi tiết hóa đơn
* Danh sách phương thức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | MaSP | String | Khóa chính |  |
| 2 | MaDonHang | String |  |  |
| 3 | TenSP | String |  |  |
| 4 | GiaSP | String |  |  |

Bảng 1. 20 Danh sách phương thức lớp chi tiết hóa đơn

* Danh sách thuộc tính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | ThemHD | Void | Public | Thêm hóa đơn vào giỏ hàng |
| 2 | CapNhapHD | Void | Public | Cập nhập hóa đơn vào giỏ hàng |
| 3 | XoaHD | Void | Public | Xóa hóa đơn ra giỏ hàng |

Bảng 1. 21 Danh sách thuộc tính lớp chi tiết hóa đơn

* + - * + Lớp chi tiết sản phẩm
* Danh sách phương thức

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | String | Khóa chính |  |
| 2 | Tên | String |  |  |
| 3 | Gia | String |  |  |
| 4 | HinhAnh | String |  |  |
| 5 | MoTa | String |  |  |

Bảng 1. 22 Danh sách phương thức lớp chi tiết sản phẩm

* Danh sách thuộc tính:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên phướng thức | Kiểu dữ liệu | Mức truy cập | Mô tả |
| 1 | ThemSP | Void | Pulic | Thêm sản phẩm vào giỏ hàng |
| 2 | CapNhapSP | Void | Pulic | Cập nhập sản phẩm |
| 3 | XoaSP | Void | Pulic | Xóa sản phẩm |

Bảng 1. 23 Danh sách thuộc tính lớp chi tiết sản phẩm

* + 1. Thiết kế cơ sở dữ liệu

- Mô tả cấu trúc bảng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | TenKH | NVARCHAR | Not Null | Tên khách hàng |
| 2 | ID | NVARCHAR |  | Mã khách hàng |
| 3 | SoDienThoai | NVARCHAR |  | Số điện thoại |
| 4 | DiaChi | NVARCHAR |  | Địa chỉ |

Bảng 1. 24 Bảng cơ sở dữ liệu khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | TenSP | NVARCHAR | Not Null | Tên sản phẩm |
| 2 | ID | NVARCHAR |  | Mã sản phẩm |
| 3 | HinhAnhSP | NVARCHAR |  | Hình ảnh sản phẩm |

Bảng 1. 25 Bảng cơ sở dữ liệu sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | NVARCHAR | Not Null | Mã sản phẩm |
| 2 | TenKH | NVARCHAR |  | Tên khách hàng |
| 3 | SĐT | NVARCHAR |  | Số điện thoại khách hàng |
| 4 | DiaChi | NVARCHAR |  | Địa chỉ |
| 5 | Email | NVARCHAR |  | Email |

Bảng 1. 26 Bảng cơ sở dữ liệu đơn hàng

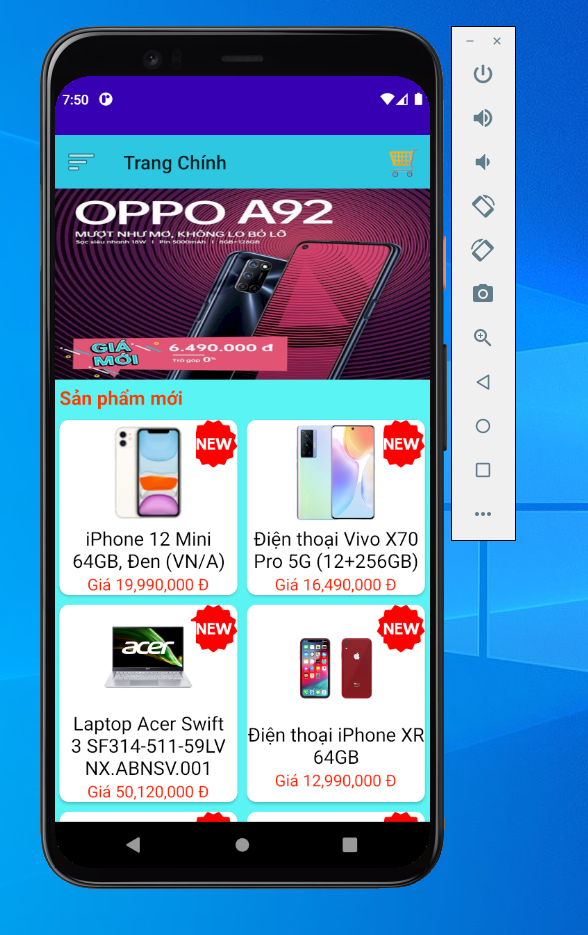
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | TenSP | NVARCHAR |  | Tên sản phẩm |
| 2 | MaDonHang | NVARCHAR |  | Mã đơn hàng |
| 3 | HinhAnhSP | NVARCHAR |  | Hình ảnh sản phẩm |
| 4 | MoTa | NVARCHAR |  | Mô tả sản phẩm |
| 5 | ID | INT | Not Null | Mã sản phẩm |

Bảng 1. 27 Bảng cơ sở dữ liệu chi tiết sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | NoNull | Mã sản phẩm |
| 2 | MaDonHang | INT |  | Mã đơn hàng |
| 3 | MaSP | INT |  | Tên sản phẩm |
| 4 | GiaSP | INT |  | Giá sản phẩm |
| 5 | TenSP | NVARCHAR |  | Tên sản phẩm |
| 6 | SoLuongSP | INT |  | Số lượng sản phẩm |

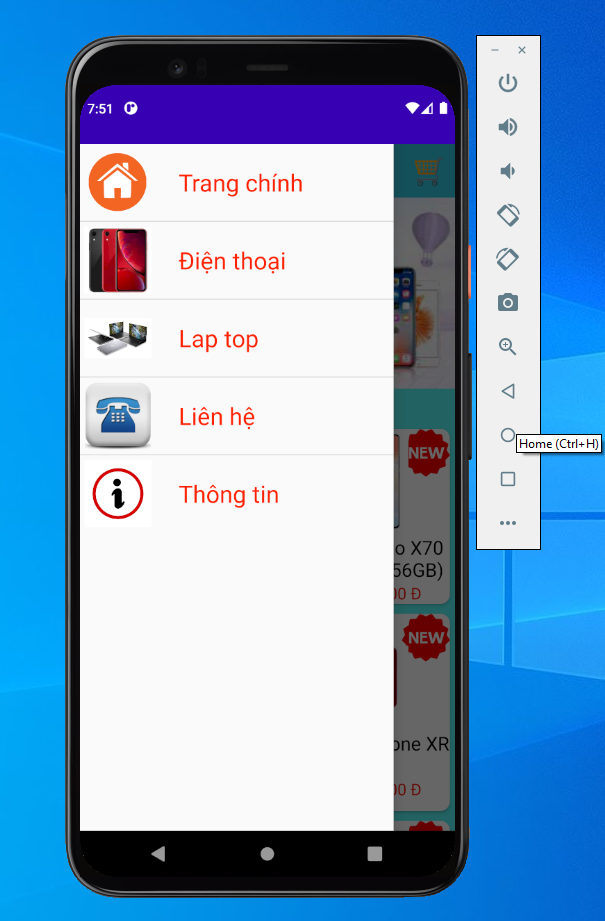
Bảng 1. 28 Bảng cơ sở dữ liệu chi tiết đơn hàng

* + 1. Thiết kế giao diện
* Giao diện chính



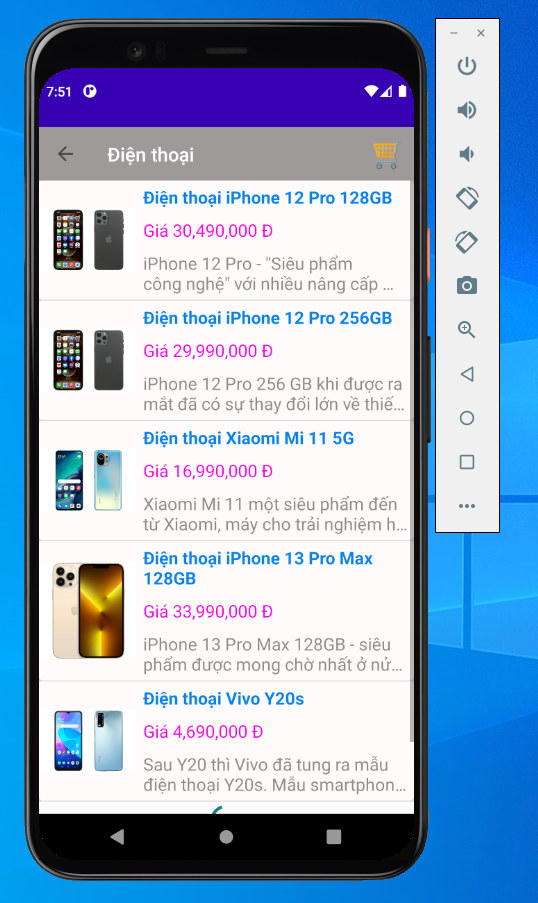
Hình 3. 13 Giao diện chính

* Giao diện danh mục



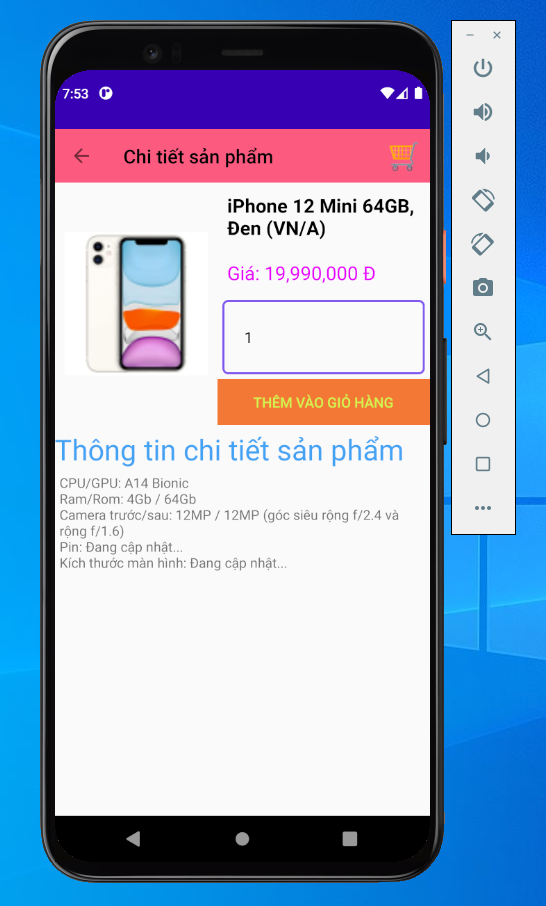
Hình 3. 14 Giao diện danh mục

* Danh mục sản phẩm



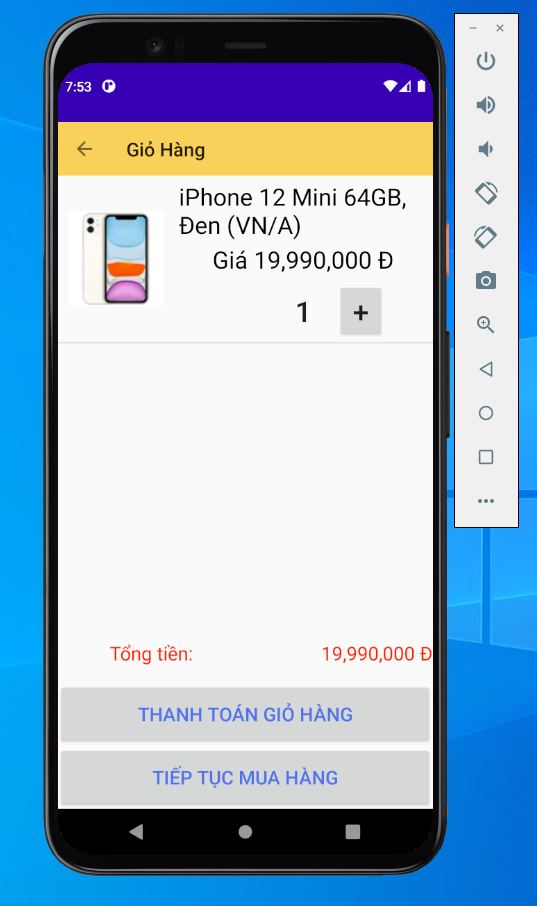
Hình 3. 15 Giao diện sản phẩm

* Giao diện chi tiết sản phẩm



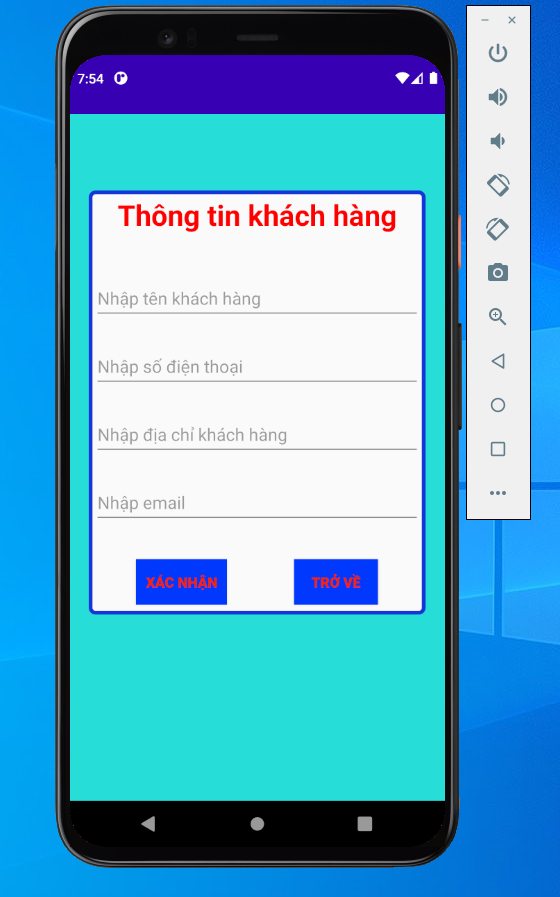
Hình 3. 16 Giao diện chi tiết sản phẩm

* Giao diện giỏ hàng



Hình 3. 17 Giao diện giỏ hàng

* Giao diện thông tin khách hàng



Hình 3. 18 Giao diện thông tin khách hàng

# TRIỂN KHAI ỨNG DỤNG

* 1. Triển khai các chức năng của hệ thống

Ứng dụng có các chức năng chính :

* Chức năng chọn và xem chi tiết sản phẩm
* Chức năng thanh toán
* Chức năng lưu thông tin khách hàng
* Chức năng liên hệ với Admin qua thông tin cung cấp trước(Email, Số điện thoại)
  + 1. Chức năng xem sản phẩm theo danh mục

a) Thiết kế layout

* Xây dựng layout bằng các thẻ XML

**androidx.drawerlayout.widget.DrawerLayout  
 android:id="@+id/drawerlayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**androidx.appcompat.widget.Toolbar  
 android:id="@+id/toolbarmanhinhchinh"  
 app:title="Trang Chính"  
 android:background="#EB1EC2DF"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?actionBarSize"  
 tools:ignore="InvalidId"**>  
  
 </**androidx.appcompat.widget.Toolbar**>  
 <**ViewFlipper  
 android:id="@+id/viewlipper"  
 android:layout\_width="match\_parent"  
 android:layout\_height="200dp"**>  
  
 </**ViewFlipper**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#5AF4F4"  
 android:orientation="vertical"  
 tools:ignore="ExtraText"**>  
 <**TextView  
 android:layout\_margin="5dp"  
 android:textStyle="bold"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Sản phẩm mới"  
 android:textSize="20sp"  
 android:textColor="#F1400A"**/>  
 <**androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/recyclerview"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**/>  
 </**LinearLayout**>  
  
  
 </**LinearLayout**>  
 <**com.google.android.material.navigation.NavigationView  
 android:id="@+id/navigationview"  
 android:layout\_gravity="start"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**ListView  
 android:id="@+id/listviewmanhinhchinh"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 </**com.google.android.material.navigation.NavigationView**>  
  
</**androidx.drawerlayout.widget.DrawerLayout**>

* Kỹ thuật định dạng bằng drawable

*<?***xml version="1.0" encoding="utf-8"***?>*<**set xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shareInterpolator="false"**>  
 <**translate android:duration="2000" android:fromXDelta="-100%" android:toXDelta="0%"**/>  
 <**alpha android:duration="2000" android:fromAlpha="0.0" android:toAlpha="1.0"**/>  
  
</**set**>

b) Code java

**package** Activity;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.view.View;  
**import** android.view.animation.Animation;  
**import** android.view.animation.AnimationUtils;  
**import** android.widget.AdapterView;  
**import** android.widget.ImageView;  
**import** android.widget.ListView;  
**import** android.widget.ViewFlipper;  
  
**import** androidx.annotation.NonNull;  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** androidx.appcompat.widget.Toolbar;  
**import** androidx.core.view.GravityCompat;  
**import** androidx.drawerlayout.widget.DrawerLayout;  
**import** androidx.recyclerview.widget.GridLayoutManager;  
**import** androidx.recyclerview.widget.RecyclerView;  
  
**import** com.android.volley.RequestQueue;  
**import** com.android.volley.Response;  
**import** com.android.volley.VolleyError;  
**import** com.android.volley.toolbox.JsonArrayRequest;  
**import** com.android.volley.toolbox.Volley;  
**import** com.example.cuahangthietbionline.R;  
**import** com.google.android.material.navigation.NavigationView;  
**import** com.squareup.picasso.Picasso;  
  
**import** org.json.JSONArray;  
**import** org.json.JSONException;  
**import** org.json.JSONObject;  
  
**import** java.util.ArrayList;  
  
**import** adapter.Loaispadapter;  
**import** adapter.Sanphamadapter;  
**import** model.GioHang;  
**import** model.Loaisp;  
**import** model.Sanpham;  
**import** until.Checkconnection;  
**import** until.Server;  
  
**public class** MainActivity **extends** AppCompatActivity {  
 Toolbar **toolbar**;  
 ViewFlipper **viewFlipper**;  
 RecyclerView **recyclerViewmanhinhchinh**;  
 NavigationView **navigationView**;  
 ListView **listView**;  
 DrawerLayout **drawerLayout**;  
 ArrayList<Loaisp> **mangloaisp**;  
 Loaispadapter **loaispadapter**;  
 ArrayList<Sanpham> **mangsanpham**;  
 Sanphamadapter **sanphamadapter**;  
 **int id**;  
 String **tenloaisp**=**""**;  
 String **hinhanhsp**=**""**;  
 **public static** ArrayList<GioHang> *manggiohang*;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Anhxa();  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Actionbar();  
 ActionViewFlipper();  
 Getdulieuloaisp();  
 Getdulieusanphammoinhat();  
 chonItemlistview();  
 }**else** {  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Kiểm tra lại mạng"**);  
 finish();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.***menugiohang***,menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(@NonNull MenuItem item) {  
 **switch** (item.getItemId()){  
 **case** R.id.***menugiohang***:  
 Intent intent=**new** Intent(getApplicationContext(),Giohang.**class**);  
 startActivity(intent);  
 }  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 **private void** chonItemlistview() {  
 **listView**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> adapterView, View view, **int** i, **long** l) {  
 **switch** (i){  
 **case** 0:  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Intent intent=**new** Intent(MainActivity.**this**,MainActivity.**class**);  
 startActivity(intent);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Hay kiem tra mang"**);  
 }  
 **drawerLayout**.closeDrawer(GravityCompat.***START***);  
 **break**;  
 **case** 1:  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Intent intent=**new** Intent(MainActivity.**this**, DienthoaiActivity.**class**);  
 intent.putExtra(**"idloaisp"**,**mangloaisp**.get(i).getId());  
 startActivity(intent);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Hay kiem tra mang"**);  
 }  
 **drawerLayout**.closeDrawer(GravityCompat.***START***);  
 **break**;  
 **case** 2:  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Intent intent=**new** Intent(MainActivity.**this**, LaptopActivity.**class**);  
 intent.putExtra(**"idloaisp"**,**mangloaisp**.get(i).getId());  
 startActivity(intent);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Hay kiem tra mang"**);  
 }  
 **drawerLayout**.closeDrawer(GravityCompat.***START***);  
 **break**;  
 **case** 3:  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Intent intent=**new** Intent(MainActivity.**this**, LienheActivity.**class**);  
 startActivity(intent);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Hay kiem tra mang"**);  
 }  
 **drawerLayout**.closeDrawer(GravityCompat.***START***);  
 **break**;  
 **case** 4:  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Intent intent=**new** Intent(MainActivity.**this**, ThongtinActivity.**class**);  
 startActivity(intent);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Hay kiem tra mang"**);  
 }  
 **drawerLayout**.closeDrawer(GravityCompat.***START***);  
 **break**;  
 }  
 }  
 });  
 }  
  
 **private void** Getdulieusanphammoinhat() {  
 RequestQueue requestQueue=Volley.*newRequestQueue*(getApplicationContext());  
 JsonArrayRequest jsonArrayRequest=**new** JsonArrayRequest(Server.*Đuongấnnphammoinhat*, **new** Response.Listener<JSONArray>() {  
 @Override  
 **public void** onResponse(JSONArray response) {  
 **if**(response!=**null**){  
 **int** ID=0;  
 String Tensp=**""**;  
 Integer Giasp=0;  
 String Hinhanhsp=**""**;  
 String Motasp=**""**;  
 **int** IDsanpham=0;  
 **for**(**int** i=0;i<response.length();i++){  
 **try** {  
 JSONObject jsonObject=response.getJSONObject(i);  
 ID=jsonObject.getInt(**"id"**);  
 Tensp=jsonObject.getString(**"tensp"**);  
 Giasp=jsonObject.getInt(**"giasp"**);  
 Hinhanhsp=jsonObject.getString(**"hinhanhsp"**);  
 Motasp=jsonObject.getString(**"motasp"**);  
 IDsanpham=jsonObject.getInt(**"idsanpham"**);  
 **mangsanpham**.add(**new** Sanpham(ID,Tensp,Giasp,Hinhanhsp,Motasp,IDsanpham));  
 **sanphamadapter**.notifyDataSetChanged();  
 } **catch** (JSONException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
  
 }  
 }, **new** Response.ErrorListener() {  
 @Override  
 **public void** onErrorResponse(VolleyError error) {  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),error.toString());  
 }  
 });  
 requestQueue.add(jsonArrayRequest);  
 }  
  
 **private void** Getdulieuloaisp() {  
 *//hàm của thư viện Volley đọc nội dung đường dẫn Url* RequestQueue requestQueue= Volley.*newRequestQueue*(getApplicationContext());  
 JsonArrayRequest jsonArrayRequest=**new** JsonArrayRequest(Server.*Đuonganloaisp*, **new** Response.Listener<JSONArray>() {  
 @Override  
 **public void** onResponse(JSONArray response) {  
 **if**(response!=**null**){  
 **for** (**int** i=0;i<response.length();i++){  
 **try** {  
 JSONObject jsonObject=response.getJSONObject(i);  
 **id**=jsonObject.getInt(**"id"**);  
 **tenloaisp**=jsonObject.getString(**"tenloaisp"**);  
 **hinhanhsp**=jsonObject.getString(**"hinhanhloaisp"**);  
 **mangloaisp**.add(**new** Loaisp(**id**,**tenloaisp**,**hinhanhsp**));  
 **loaispadapter**.notifyDataSetChanged();  
 } **catch** (JSONException e) {  
 e.printStackTrace();  
 }  
 }  
 **mangloaisp**.add(3,**new** Loaisp(3,**"Liên hệ"**,**"https://i.pinimg.com/originals/57/cf/21/57cf2127a1b9c8fdb334e5860fc22f61.png"**));  
 **mangloaisp**.add(4,**new** Loaisp(4,**"Thông tin"**,**"https://png.pngtree.com/png-vector/20190916/ourlarge/pngtree-info-icon-for-your-project-png-image\_1731084.jpg"**));  
 }  
 }  
 }, **new** Response.ErrorListener() {  
 @Override  
 **public void** onErrorResponse(VolleyError error) {  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),error.toString());  
 }  
 });  
 requestQueue.add(jsonArrayRequest);  
 }  
 *//ViewFlipper chạy hình ảnh quản cáo* **private void** ActionViewFlipper() {  
 *//mảng chứa đường dẫn những tấm hình* ArrayList<String> mangquangcao=**new** ArrayList<>();  
 *//thêm vào đường dẫn Url những tấm hình* mangquangcao.add(**"https://genk.mediacdn.vn/2019/6/22/promotiondoublestorage-1561186618199280355933.jpg"**);  
 mangquangcao.add(**"https://st.quantrimang.com/photos/image/2018/07/14/quang-cao-smartphone-650.jpg"**);  
 mangquangcao.add(**"https://genk.mediacdn.vn/Images/Uploaded/Share/2011/02/28/40iphone.jpg"**);  
 mangquangcao.add(**"https://cafebiz.cafebizcdn.vn/2017/untitled-1509952351674.png"**);  
 **for**(**int** i=0;i<mangquangcao.size();i++){  
 ImageView imageView=**new** ImageView(getApplicationContext());  
 *//hàm thư viện của Picasso để truyền hình ảnh vào Img bằng Url* Picasso.*get*().load(mangquangcao.get(i)).into(imageView);  
 *//thuộc canh vừa với Viewflipper* imageView.setScaleType(ImageView.ScaleType.***FIT\_XY***);  
 **viewFlipper**.addView(imageView);  
 }  
 *//viewflipper tự chạy 5s* **viewFlipper**.setFlipInterval(5000);  
 **viewFlipper**.setAutoStart(**true**);  
 *//khởi tại 2 resoure file anim* Animation animation\_in= AnimationUtils.*loadAnimation*(getApplicationContext(),R.anim.***slide\_in\_right***);  
 Animation animation\_out=AnimationUtils.*loadAnimation*(getApplicationContext(),R.anim.***slide\_out\_right***);  
 **viewFlipper**.setInAnimation(animation\_in);  
 **viewFlipper**.setOutAnimation(animation\_out);  
 }  
  
 **private void** Actionbar() {  
 *//Hàm hỗ trợ toolbar* setSupportActionBar(**toolbar**);  
 *//set núp HOME* getSupportActionBar().setDisplayHomeAsUpEnabled(**true**);  
 *//Tạo núp menu sort\_by\_sie(---)* **toolbar**.setNavigationIcon(android.R.drawable.***ic\_menu\_sort\_by\_size***);  
 *//bắt sự kiện khi clik vào mở ra thanh menu* **toolbar**.setNavigationOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 *//Mở ra từ trái qua giữa* **drawerLayout**.openDrawer(GravityCompat.***START***);  
 }  
 });  
 }  
 *//Ánh xạ* **private void** Anhxa() {  
 **toolbar**=findViewById(R.id.***toolbarmanhinhchinh***);  
 **viewFlipper**=findViewById(R.id.***viewlipper***);  
 **recyclerViewmanhinhchinh**=findViewById(R.id.***recyclerview***);  
 **navigationView**=findViewById(R.id.***navigationview***);  
 **listView**=findViewById(R.id.***listviewmanhinhchinh***);  
 **drawerLayout**=findViewById(R.id.***drawerlayout***);  
 **mangloaisp**=**new** ArrayList<>();  
 **mangloaisp**.add(0,**new** Loaisp(0,**"Trang chính"**,**"https://noithattinnghia.com/wp-content/uploads/2019/03/cropped-icon-home-cam.png"**));  
 **loaispadapter**=**new** Loaispadapter(**mangloaisp**,getApplicationContext());  
 **listView**.setAdapter(**loaispadapter**);  
 **mangsanpham**=**new** ArrayList<>();  
 **sanphamadapter**=**new** Sanphamadapter(getApplicationContext(), **mangsanpham**, **new** Sanphamadapter.ItemClickListener() {  
 @Override  
 **public void** onItemCick(Sanpham details) {  
 onClickrecyclerview(details);  
 }  
 });  
 **recyclerViewmanhinhchinh**.setHasFixedSize(**true**);  
 **recyclerViewmanhinhchinh**.setLayoutManager(**new** GridLayoutManager(getApplicationContext(),2));  
 **recyclerViewmanhinhchinh**.setAdapter(**sanphamadapter**);  
 **if**(*manggiohang*!=**null**){  
  
 }**else**{  
 *manggiohang*=**new** ArrayList<>();  
 }  
 }  
 **private void** onClickrecyclerview(Sanpham details){  
 Intent intent=**new** Intent(MainActivity.**this**, Chitietsanpham.**class**);  
 Bundle bundle=**new** Bundle();  
 bundle.putSerializable(**"thongtinsanpham"**,details);  
 intent.putExtras(bundle);  
 startActivity(intent);  
 }  
  
}

* + 1. Chức năng chọn và xem chi tiết sản phẩm
* **Thiết kế Layout**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="Activity.DienthoaiActivity"**>  
 <**androidx.appcompat.widget.Toolbar  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 app:title="Điện thoại"  
 android:id="@+id/toolbardienthoai"  
 app:titleTextColor="#FFFEFE"  
 android:background="#F2989594"**/>  
 <**ListView  
 android:id="@+id/listviewdienthoai"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**/>  
  
</**LinearLayout**>

* **Code Java**

**package** Activity;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.os.Message;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.view.View;  
**import** android.widget.AbsListView;  
**import** android.widget.AdapterView;  
**import** android.widget.ListView;  
  
**import** androidx.annotation.NonNull;  
**import** androidx.annotation.Nullable;  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** androidx.appcompat.widget.Toolbar;  
  
**import** com.android.volley.AuthFailureError;  
**import** com.android.volley.Request;  
**import** com.android.volley.RequestQueue;  
**import** com.android.volley.Response;  
**import** com.android.volley.VolleyError;  
**import** com.android.volley.toolbox.StringRequest;  
**import** com.android.volley.toolbox.Volley;  
**import** com.example.cuahangthietbionline.R;  
  
**import** org.json.JSONArray;  
**import** org.json.JSONException;  
**import** org.json.JSONObject;  
  
**import** java.util.ArrayList;  
**import** java.util.HashMap;  
**import** java.util.Map;  
  
**import** adapter.DienthoaiAdapter;  
**import** model.Sanpham;  
**import** until.Checkconnection;  
**import** until.Server;  
  
**public class** DienthoaiActivity **extends** AppCompatActivity {  
 Toolbar **toolbar**;  
 ListView **listViewdt**;  
 DienthoaiAdapter **dienthoaiAdapter**;  
 ArrayList<Sanpham> **sanphamArrayList**;  
 **int idsp**=0;  
 **int page**=1;  
 View **footerview**;  
 **boolean isLoading**=**false**;  
 mHandler **mHandler**;  
 **boolean limitdata**= **false**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_dienthoai***);  
 Anhxa();  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())){  
 Getidsp();  
 Actiontoolbar();  
 Getdata(**page**);  
 LoadMoreData();  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Kiem tra mang"**);  
 finish();  
 }  
  
 }  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.***menugiohang***,menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(@NonNull MenuItem item) {  
 **switch** (item.getItemId()){  
 **case** R.id.***menugiohang***:  
 Intent intent=**new** Intent(getApplicationContext(),Giohang.**class**);  
 startActivity(intent);  
 }  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 **private void** LoadMoreData() {  
 **listViewdt**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> adapterView, View view, **int** i, **long** l) {  
 Intent intent=**new** Intent(getApplicationContext(),Chitietsanpham.**class**);  
 intent.putExtra(**"thongtinsanpham"**,**sanphamArrayList**.get(i));  
 startActivity(intent);  
 }  
 });  
 **listViewdt**.setOnScrollListener(**new** AbsListView.OnScrollListener() {  
 @Override  
 **public void** onScrollStateChanged(AbsListView absListView, **int** i) {  
  
 }  
  
 @Override  
 **public void** onScroll(AbsListView absListView, **int** FisrtItem, **int** VisibleItem, **int** TotalItem) {  
 **if**(FisrtItem+VisibleItem==TotalItem && TotalItem!=0 && **isLoading**==**false** && **limitdata**==**false**){  
 **isLoading**=**true**;  
 ThreadData threadData=**new** ThreadData();  
 threadData.start();  
 }  
  
 }  
 });  
 }  
  
 **private void** Getdata(**int** Page) {  
 RequestQueue requestQueue= Volley.*newRequestQueue*(getApplicationContext());  
 String duongdan= Server.*Đuongandienthoai*+String.*valueOf*(**page**);  
 StringRequest stringRequest=**new** StringRequest(Request.Method.***POST***, duongdan, **new** Response.Listener<String>() {  
 @Override  
 **public void** onResponse(String response) {  
 **int** id=0;  
 String Tendt=**""**;  
 **int** Giadt=0;  
 String Hinhanhdt=**""**;  
 String Mota=**""**;  
 **int** Idspdt=0;  
 **if**(response!=**null** && response.length()!=2){  
 **listViewdt**.removeFooterView(**footerview**);  
 **try** {  
 JSONArray jsonArray=**new** JSONArray(response);  
 **for** (**int** i=0;i<jsonArray.length();i++){  
 JSONObject jsonObject=jsonArray.getJSONObject(i);  
 id=jsonObject.getInt(**"id"**);  
 Tendt=jsonObject.getString(**"tensp"**);  
 Giadt=jsonObject.getInt(**"giasp"**);  
 Hinhanhdt=jsonObject.getString(**"hinhanhsp"**);  
 Mota=jsonObject.getString(**"motasp"**);  
 Idspdt=jsonObject.getInt(**"idsanpham"**);  
 **sanphamArrayList**.add(**new** Sanpham(id,Tendt,Giadt,Hinhanhdt,Mota,Idspdt));  
 **dienthoaiAdapter**.notifyDataSetChanged();  
 }  
 } **catch** (JSONException e) {  
 e.printStackTrace();  
 }  
 }**else**{  
 **limitdata**=**true**;  
 **listViewdt**.removeFooterView(**footerview**);  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Het du lieu"**);  
 }  
 }  
 }, **new** Response.ErrorListener() {  
 @Override  
 **public void** onErrorResponse(VolleyError error) {  
  
 }  
 }){  
 @Nullable  
 @Override  
 **protected** Map<String, String> getParams() **throws** AuthFailureError {  
 HashMap<String,String> param=**new** HashMap<String,String>();  
 param.put(**"idsanpham"**,String.*valueOf*(**idsp**));  
 **return** param;  
 }  
 };  
 requestQueue.add(stringRequest);  
 }  
  
 **private void** Actiontoolbar() {  
 setSupportActionBar(**toolbar**);  
 getSupportActionBar().setDisplayHomeAsUpEnabled(**true**);  
 **toolbar**.setNavigationOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 finish();;  
 }  
 });  
 }  
  
 **private void** Getidsp() {  
 **idsp**=getIntent().getIntExtra(**"idloaisp"**,-1);  
 Log.*d*(**"Gia tri laoi san pham"**,**idsp**+**""**);  
 }  
  
 **private void** Anhxa() {  
 **toolbar**=(Toolbar) findViewById(R.id.***toolbardienthoai***);  
 **listViewdt**=(ListView) findViewById(R.id.***listviewdienthoai***);  
 **sanphamArrayList**=**new** ArrayList<>();  
 **dienthoaiAdapter**=**new** DienthoaiAdapter(getApplicationContext(),**sanphamArrayList**);  
 **listViewdt**.setAdapter(**dienthoaiAdapter**);  
 LayoutInflater inflater= (LayoutInflater) getSystemService(***LAYOUT\_INFLATER\_SERVICE***);  
 **footerview**=inflater.inflate(R.layout.***processbar***,**null**);  
 **mHandler**=**new** mHandler();  
 }  
 **public class** mHandler **extends** Handler{  
 @Override  
 **public void** handleMessage(@NonNull Message msg) {  
 **switch** (msg.**what**){  
 **case** 0:  
 **listViewdt**.addFooterView(**footerview**);  
 **break**;  
 **case** 1:  
 Getdata(++**page**);  
 **isLoading**= **false**;  
 **break**;  
 }  
 **super**.handleMessage(msg);  
 }  
 }  
 **public class** ThreadData **extends** Thread{  
 @Override  
 **public void** run() {  
 **mHandler**.sendEmptyMessage(0);  
 **try** {  
 Thread.*sleep*(3000);  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }  
 Message message=**mHandler**.obtainMessage(1);  
 **mHandler**.sendMessage(message);  
 **super**.run();  
 }  
 }  
}

* + 1. Chức năng thanh toán
* **Thiết kế Layout**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="Activity.Giohang"  
 android:orientation="vertical"  
 android:weightSum="10"**>  
 <**androidx.appcompat.widget.Toolbar  
 app:title="Giỏ hàng"  
 android:background="#D7F8C93C"  
 android:id="@+id/toolbargiohang"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 <**LinearLayout  
 android:layout\_weight="7"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"**>  
 <**FrameLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**TextView  
 android:layout\_gravity="center"  
 android:id="@+id/textviewthongbao"  
 android:textSize="30sp"  
 android:text="Giỏ hàng bạn trống"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 <**ListView  
 android:id="@+id/listviewgiohang"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**/>  
 </**FrameLayout**>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:layout\_weight="1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"**>  
 <**LinearLayout  
 android:layout\_marginTop="5dp"  
 android:orientation="horizontal"  
 android:weightSum="2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**LinearLayout  
 android:gravity="center"  
 android:layout\_weight="1"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"**>  
 <**TextView  
 android:textColor="#f20"  
 android:text="Tổng tiền:"  
 android:textSize="20sp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:gravity="right|center"  
 android:layout\_weight="1"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"**>  
 <**TextView  
 android:id="@+id/textviewtongtien"  
 android:textColor="#f20"  
 android:textSize="20sp"  
 android:text="$$$$$$$$$"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
 </**LinearLayout**>  
 </**LinearLayout**>  
 <**Button  
 android:textColor="#DA3B62F1"  
 android:textSize="20sp"  
 android:text="Thanh toán giỏ hàng"  
 android:id="@+id/btnthanhtoangiohang"  
 android:layout\_weight="1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"**/>  
 <**Button  
 android:textColor="#DA3B62F1"  
 android:textSize="20sp"  
 android:text="Tiếp tục mua hàng"  
 android:id="@+id/btntieptucmuahang"  
 android:layout\_weight="1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"**/>  
  
</**LinearLayout**>

**Code Java**

**package** Activity;  
  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.Button;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
  
**import** androidx.appcompat.app.AlertDialog;  
**import** androidx.appcompat.widget.Toolbar;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** com.example.cuahangthietbionline.R;  
  
**import** java.text.DecimalFormat;  
  
**import** adapter.GiohangAdapter;  
**import** until.Checkconnection;  
  
**public class** Giohang **extends** AppCompatActivity {  
 Toolbar **Toolbargiohang**;  
 TextView **txtThongbao**;  
 **static** TextView *txtTongtien*;  
 ListView **listViewgiohang**;  
 Button **btnthanhtoan**, **btntieptucmua**;  
 GiohangAdapter **giohangAdapter**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_giohang***);  
 Anhxa();  
 ActionTollbar();  
 CheckData();  
 *EventUtil*();  
 CacthOnItem();  
 EventButton();  
 }  
  
 **private void** EventButton() {  
 **btntieptucmua**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 CheckData();  
 Intent intent=**new** Intent(Giohang.**this**,MainActivity.**class**);  
 **giohangAdapter**.notifyDataSetChanged();  
 startActivity(intent);  
 }  
 });  
 **btnthanhtoan**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **if**(MainActivity.*manggiohang*.size()<=0){  
 Checkconnection.*Showtoast\_short*(Giohang.**this**,**"Giỏ hàng của bạn không có sản phẩm để thanh toán"**);  
 }**else**{  
 Intent intent=**new** Intent(getApplicationContext(),Thongtinnguoidung.**class**);  
 startActivity(intent);  
  
 }  
 }  
 });  
 }  
  
 **private void** CacthOnItem() {  
 **listViewgiohang**.setOnItemLongClickListener(**new** AdapterView.OnItemLongClickListener() {  
 @Override  
 **public boolean** onItemLongClick(AdapterView<?> adapterView, View view, **int** position, **long** l) {  
 AlertDialog.Builder builder=**new** AlertDialog.Builder(Giohang.**this**);  
 builder.setTitle(**"Xác nhận xóa sản phẩm"**);  
 builder.setMessage(**"Bạn có chắc muốn xóa sản phẩm này?"**);  
 builder.setPositiveButton(**"Có"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 **if**(MainActivity.*manggiohang*.size()<=0){  
 **txtThongbao**.setVisibility(View.***VISIBLE***);  
 }**else**{  
 MainActivity.*manggiohang*.remove(position);  
 **giohangAdapter**.notifyDataSetChanged();  
 *EventUtil*();  
 **if**(MainActivity.*manggiohang*.size()<=0) {  
 **txtThongbao**.setVisibility(View.***VISIBLE***);  
 }**else**{  
 **txtThongbao**.setVisibility(View.***INVISIBLE***);  
 **giohangAdapter**.notifyDataSetChanged();  
 *EventUtil*();  
 }  
 }  
 }  
 });  
 builder.setNegativeButton(**"Không"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 **giohangAdapter**.notifyDataSetChanged();  
 *EventUtil*();  
 }  
 });  
 builder.show();  
 **return true**;  
 }  
 });  
 }  
  
 **public static void** EventUtil() {  
 **long** tongtien=0;  
 **for** (**int** i=0;i<MainActivity.*manggiohang*.size();i++){  
 tongtien+=MainActivity.*manggiohang*.get(i).getGiasp();  
 }  
 DecimalFormat decimalFormat=**new** DecimalFormat(**"###,###,###"**);  
 *txtTongtien*.setText(decimalFormat.format(tongtien)+**" Đ"**);  
 }  
  
 **private void** CheckData() {  
 **if**(MainActivity.*manggiohang*.size()<=0)  
 {  
 **giohangAdapter**.notifyDataSetChanged();  
 **txtThongbao**.setVisibility(View.***VISIBLE***);  
 **listViewgiohang**.setVisibility(View.***INVISIBLE***);  
 }**else**{  
 **giohangAdapter**.notifyDataSetChanged();  
 **txtThongbao**.setVisibility(View.***INVISIBLE***);  
 **listViewgiohang**.setVisibility(View.***VISIBLE***);  
 }  
 }  
  
 **private void** ActionTollbar() {  
 setSupportActionBar(**Toolbargiohang**);  
 getSupportActionBar().setDisplayHomeAsUpEnabled(**true**);  
 **Toolbargiohang**.setNavigationOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 finish();  
 }  
 });  
  
 }  
  
 **private void** Anhxa() {  
 **txtThongbao**=(TextView) findViewById(R.id.***textviewthongbao***);  
 *txtTongtien*=(TextView) findViewById(R.id.***textviewtongtien***);  
 **Toolbargiohang**=(Toolbar) findViewById(R.id.***toolbargiohang***);  
 **listViewgiohang**=(ListView) findViewById(R.id.***listviewgiohang***);  
 **btnthanhtoan**=(Button) findViewById(R.id.***btnthanhtoangiohang***);  
 **btntieptucmua**=(Button) findViewById(R.id.***btntieptucmuahang***);  
 **giohangAdapter**=**new** GiohangAdapter(Giohang.**this**,MainActivity.*manggiohang*);  
 **listViewgiohang**.setAdapter(**giohangAdapter**);  
 }  
}

* + 1. Chức năng lưu thông tin khách hàng
* **Thiết kế Layout**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#25DDD8"  
 android:orientation="vertical"  
 tools:context="Activity.Thongtinnguoidung"**>  
  
 <**LinearLayout  
 android:orientation="vertical"  
 android:gravity="center"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="20dp"  
 android:layout\_marginTop="80dp"  
 android:layout\_marginRight="20dp"  
 android:layout\_marginBottom="80dp"  
 android:background="@drawable/boder"**>  
  
 <**TextView  
 android:layout\_marginTop="30dp"  
 android:layout\_margin="5dp"  
 android:layout\_marginRight="5dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="Thông tin khách hàng"  
 android:textAlignment="center"  
 android:textSize="30sp"  
 android:textStyle="bold"  
 android:textColor="#FF0303"**/>  
 <**com.google.android.material.textfield.TextInputLayout  
 android:layout\_marginTop="30dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**EditText  
 android:layout\_margin="5dp"  
 android:id="@+id/edittextnhaptenkh"  
 android:inputType="text"  
 android:hint="Nhập tên khách hàng"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 </**com.google.android.material.textfield.TextInputLayout**>  
 <**com.google.android.material.textfield.TextInputLayout  
 android:layout\_marginTop="10dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**EditText  
 android:layout\_margin="5dp"  
 android:id="@+id/edittextnhapsodienthoai"  
 android:inputType="phone"  
 android:hint="Nhập số điện thoại"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 </**com.google.android.material.textfield.TextInputLayout**>  
 <**com.google.android.material.textfield.TextInputLayout  
 android:layout\_marginTop="10dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**EditText  
 android:layout\_margin="5dp"  
 android:id="@+id/edittextnhapdiachikh"  
 android:inputType="text"  
 android:hint="Nhập địa chỉ khách hàng"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 </**com.google.android.material.textfield.TextInputLayout**>  
 <**com.google.android.material.textfield.TextInputLayout  
 android:layout\_marginTop="10dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**EditText  
 android:layout\_margin="5dp"  
 android:id="@+id/edittextnhapemail"  
 android:inputType="textEmailAddress"  
 android:hint="Nhập email"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**/>  
 </**com.google.android.material.textfield.TextInputLayout**>  
 <**LinearLayout  
 android:gravity="center"  
 android:layout\_marginBottom="10dp"  
 android:layout\_marginTop="30dp"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**Button  
 android:layout\_marginRight="70dp"  
 android:textSize="15sp"  
 android:textStyle="bold"  
 android:padding="10dp"  
 android:background="#0338FD"  
 android:textColor="#f20"  
 android:text="Xác nhận"  
 android:id="@+id/btnxacnhapthongtin"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 <**Button  
  
 android:textSize="15sp"  
 android:textStyle="bold"  
 android:padding="10dp"  
 android:background="#0338FD"  
 android:textColor="#f20"  
 android:text="Trở về"  
 android:id="@+id/btntrove"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
  
 </**LinearLayout**>  
  
</**LinearLayout**>

* **Code Java**

**package** Activity;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.util.Log;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
  
**import** androidx.annotation.Nullable;  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** com.android.volley.AuthFailureError;  
**import** com.android.volley.Request;  
**import** com.android.volley.RequestQueue;  
**import** com.android.volley.Response;  
**import** com.android.volley.VolleyError;  
**import** com.android.volley.toolbox.StringRequest;  
**import** com.android.volley.toolbox.Volley;  
**import** com.example.cuahangthietbionline.R;  
  
**import** org.json.JSONArray;  
**import** org.json.JSONException;  
**import** org.json.JSONObject;  
  
**import** java.util.HashMap;  
**import** java.util.Map;  
  
**import** until.Checkconnection;  
**import** until.Server;  
  
**public class** Thongtinnguoidung **extends** AppCompatActivity {  
 EditText **edttenkhachhang**,**edtsodienthoai**,**edtdiachi**,**edtemail**;  
 Button **btnxacnhanthongtin**, **btntrove**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_thongtinnguoidung***);  
 Anhxa();  
 **btntrove**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 finish();  
 }  
 });  
 **if**(Checkconnection.*HaveNetworkConnection*(getApplicationContext())) {  
 EventButton();  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Kiểm tra lại kết nối mạng"**);  
 }  
 }  
  
 **private void** EventButton() {  
 {  
 **btnxacnhanthongtin**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 String ten=**edttenkhachhang**.getText().toString().trim();  
 String sdt=**edtsodienthoai**.getText().toString().trim();  
 String diachi=**edtdiachi**.getText().toString().trim();  
 String email=**edtemail**.getText().toString().trim();  
 **if**(ten.length()>0 && sdt.length()>0 && diachi.length()>0 && email.length()>0){  
 RequestQueue requestQueue= Volley.*newRequestQueue*(getApplicationContext());  
 StringRequest stringRequest=**new** StringRequest(Request.Method.***POST***, Server.*Duongdandonhang*, **new** Response.Listener<String>() {  
 @Override  
 **public void** onResponse(String madonhang) {  
 Log.*d*(**"onResponse: "**,madonhang);  
 **if**(Integer.*parseInt*(madonhang)>0){  
 RequestQueue queue= Volley.*newRequestQueue*(getApplicationContext());  
 StringRequest request = **new** StringRequest(Request.Method.***POST***, Server.*Duongdanchitietdonhang*, **new** Response.Listener<String>() {  
 @Override  
 **public void** onResponse(String response) {  
 **if**(response.equals(**"1"**)){  
 MainActivity.*manggiohang*.clear();  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Bạn đã thêm dữ liệu giỏ hàng thành công"**);  
 Intent intent=**new** Intent(getApplicationContext(),MainActivity.**class**);  
 startActivity(intent);  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Mời bạn tiếp tục mua hàng"**);  
 }**else**{  
 Checkconnection.*Showtoast\_short*(getApplicationContext(),**"Dữ liệu giỏ hàng của bạn bị lỗi"**);  
 }  
 }  
 }, **new** Response.ErrorListener() {  
 @Override  
 **public void** onErrorResponse(VolleyError error) {  
  
 }  
 }){  
 @Nullable  
 @Override  
 **protected** Map<String, String> getParams() **throws** AuthFailureError {  
 JSONArray jsonArray=**new** JSONArray();  
 **for** (**int** i=0;i<MainActivity.*manggiohang*.size();i++){  
 JSONObject jsonObject=**new** JSONObject();  
 **try** {  
 jsonObject.put(**"madonhang"**,madonhang);  
 jsonObject.put(**"masanpham"**,MainActivity.*manggiohang*.get(i).getIdsp());  
 jsonObject.put(**"tensanpham"**,MainActivity.*manggiohang*.get(i).getTensp());  
 jsonObject.put(**"giasanpham"**,MainActivity.*manggiohang*.get(i).getGiasp());  
 jsonObject.put(**"soluongsanpham"**,MainActivity.*manggiohang*.get(i).getSoluong());  
 } **catch** (JSONException e) {  
 e.printStackTrace();  
 }  
 jsonArray.put(jsonObject);  
 }  
 HashMap<String,String> hashMap=**new** HashMap<>();  
 hashMap.put(**"json"**,jsonArray.toString());  
 **return** hashMap;  
 }  
 };  
 queue.add(request);  
 }  
 }  
 }, **new** Response.ErrorListener() {  
 @Override  
 **public void** onErrorResponse(VolleyError error) {  
  
 }  
 }){  
 @Nullable  
 @Override  
 **protected** Map<String, String> getParams() **throws** AuthFailureError {  
 HashMap<String,String> hashMap=**new** HashMap<>();  
 hashMap.put(**"tenkhachhang"**,ten);  
 hashMap.put(**"sodienthoai"**,sdt);  
 hashMap.put(**"diachi"**,diachi);  
 hashMap.put(**"email"**,email);  
 **return** hashMap;  
 }  
 };  
 requestQueue.add(stringRequest);  
 }**else**{  
 Checkconnection.Showtoast\_short(getApplicationContext(),**"Bạn hãy điền đầy đủ thông tin"**);  
 }  
 }  
 });  
 }  
 }  
  
 **private void** Anhxa() {  
 edttenkhachhang=(EditText) findViewById(R.id.edittextnhaptenkh);  
 edtsodienthoai=(EditText) findViewById(R.id.edittextnhapsodienthoai);  
 edtdiachi=(EditText) findViewById(R.id.edittextnhapdiachikh);  
 edtemail=(EditText) findViewById(R.id.edittextnhapemail);  
 btnxacnhanthongtin=(Button) findViewById(R.id.btnxacnhapthongtin);  
 btntrove=(Button) findViewById(R.id.btntrove);  
 }  
}

* + 1. Chức năng liên hệ
* Thiết kế Layout

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="Activity.LienheActivity"  
 android:background="@drawable/bgrou"  
 android:orientation="vertical"**>  
 <**LinearLayout  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**TextView  
 android:layout\_marginTop="20dp"  
 android:textStyle="bold"  
 android:textColor="#FF8400"  
 android:textSize="40sp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Liên hệ với chúng tôi"**/>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:layout\_marginTop="100dp"  
 android:layout\_gravity="center"  
 android:gravity="center"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**LinearLayout  
 android:background="@drawable/boder"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**ImageView  
 android:padding="5dp"  
 android:layout\_width="80dp"  
 android:layout\_height="80dp"  
 android:src="@drawable/human"** />  
 <**TextView  
 android:layout\_margin="10dp"  
 android:layout\_gravity="center"  
 android:textSize="25sp"  
 android:text="Họ và tên: Lương Trung Đức"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:layout\_marginTop="50dp"  
 android:background="@drawable/boder"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**ImageView  
 android:padding="5dp"  
 android:layout\_width="80dp"  
 android:layout\_height="80dp"  
 android:src="@drawable/iconemail"** />  
 <**TextView  
 android:layout\_margin="10dp"  
 android:layout\_gravity="center"  
 android:textSize="24sp"  
 android:text="Email: ducduc27112k1@gmail.com"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:layout\_marginTop="50dp"  
 android:background="@drawable/boder"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**ImageView  
 android:padding="5dp"  
 android:layout\_width="80dp"  
 android:layout\_height="80dp"  
 android:src="@drawable/iconphone"** />  
 <**TextView  
 android:layout\_margin="10dp"  
 android:layout\_gravity="center"  
 android:textSize="25sp"  
 android:text="Phone: 0325776082"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**/>  
 </**LinearLayout**>  
  
 </**LinearLayout**>  
  
  
  
</**LinearLayout**>

* Code Java

**package** Activity;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** android.os.Bundle;  
  
**import** com.example.cuahangthietbionline.R;  
  
**public class** LienheActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_lienhe***);  
 }  
}

* 1. Kiểm thử và triển khai ứng dụng
     1. Kiểm thử

Chức năng mua hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên testcase | Mục đích | Thực hiện | Kết quả |
| 1 | Thực hiện mua hàng | Kiểm tra chức năng mua và xem sản phẩm | Nhấn vào sản phẩm muốn mua | Ứng dụng hiển thị chi tiết sản phẩm |

Bảng 1. 29 Bảng kiểm thử chức năng mua hàng

Chức năng thanh toán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên testcase | Mục đích | Thực hiện | Kết quả |
| 1 | Thực hiện thanh toán | Thanh toán sản phẩm muốn mua | Nhấn vào nút thanh toán ở giao diện | Sản phẩm thanh toán thành công |
| 2 | Tiếp tục mua hàng | Khi khách hàng muốn mua thêm | Nhấn vào nút tiếp tục mua hàng | Quay lại giao diện sản phẩm |

Bảng 1. 30 Bảng kiểm thử chức năng thanh toán

Chức năng lưu thông tin khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên testcase | Mục đích | Thực hiện | Kết quả |
| 1 | Kiểm tra thông tin khách hàng | Lưu lại thông tin khách hàng vào cơ sở dữ liệu | Nhập đầy đủ thông tin ở giao diện | Thông tin được lưu lại |

Bảng 1. 31 Bảng kiểm thử chức năng lưu thông tin khách hàng

* + 1. Đóng gói ứng dụng

Đóng gói ứng dụng:



Hình 3. 19 Đóng gói ứng dụng

* + 1. Triển khai ứng dụng
* Môi trường chạy: Andorid
* Yêu cầu:
  + - * + Android 6.0 Mashmallow trở lên
        + CPU 2 nhân trở lên
        + Ram 2GB trở lên
        + Bộ nhớ trong trống: 200mb

# KẾT LUẬN

**Kết quả đạt được**

* Hiểu được quy trình làm ra một chương trình
* Học được các kiến thức cơ bản để tạo ra một ứng dụng
* Tạo ra được sản phẩm đầu tiên có giá trị
* Rèn luyện được tính tư duy lập trình
* Hiểu hơn về ngôn ngữ java và ứng dụng nó vào trong công việc

**Hạn chế của đề tài**

* Một số chức năng chưa tối ưu
* Phần phân tích thiết kế còn lủng củng chưa rõ ràng
* Giao diện chưa được thẩm mỹ

**Hướng phát triển của đề tài**

* Vẫn trên ý tưởng này nhưng em sẽ tối ưu tốt hơn để phù hợp với nhiều mày dùng hơn..
* Chú trọng vào tối ưu tốc độ xử lý của hệ thống để đem lại một trải nghiệm tốt nhất cho người dùng.
* Nâng cao  bảo mật, thiết kế cở sở dữu liệu sử dụng mạng để liên kết được nhiều   người sử dụng

TÀI LIỆU THAM KHẢO

1. Ed Burnette, *Hello, Android Introducing Google’s Mobile Development Platform*, The Pragmatic Bookshelf, 2010.
2. Marziah Karch, *Android for Work Productivity for Professional*, Press, 2010.
3. J.F. DiMarzio, *Android A Programmer’s Guide*, The McGraw-Hill Companies, 2008.
4. Chris Haseman, *Android Essentials*, Apress, 2008.
5. Sams, *Android Application Development*, Lauren Darcey and Shane Conder, 2010.
6. Stackoverflow.com, <https://stackoverflow.com>