## BỘ GIÁO DỤC VÀ ĐÀO TẠO



**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

ĐỒ ÁN 4

**XÂY DỰNG ỨNG DỤNG BÁN QUẦN ÁO ONLINE**

NGÀNH: CÔNG NGHỆ THÔNG TIN CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

SINH VIÊN: **ĐÀO VĂN ĐẠT**

MÃ SINH VIÊN: **10120208**

MÃ LỚP: 125205

HƯỚNG DẪN: **BÙI ĐỨC THỌ**

## HƯNG YÊN – 2023

# 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 bán quần áo online” là kết quả thực hiện của bản thân em dưới sự hướng dẫn của thầy Bùi Đức Thọ.

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.

*Hưng Yên, ngày tháng năm 2023*

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 – Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn thầy Bùi Đức Thọ 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**

[CHƯƠNG 1: MỞ ĐẦU 8](#_Toc22870)

[1.1. Lý do chọn đề tài 8](#_Toc24833)

[1.2. Mục tiêu của đề tài 9](#_Toc18889)

[1.2.1 Mục tiêu tổng quát 9](#_Toc26374)

[1.2.2 Mục tiêu cụ thể 9](#_Toc17343)

[1.3. Giới hạn và phạm vi của đề tài 10](#_Toc7858)

[1.3.1 Đối tượng nghiên cứu 10](#_Toc20274)

[1.3.2 Phạm vi nghiên cứu 10](#_Toc31643)

[1.4. Nội dung thực hiện 11](#_Toc30421)

[1.5. Phương pháp tiếp cận 11](#_Toc5429)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 12](#_Toc5529)

[2.1 Quy trình phát triển phần mềm 12](#_Toc7895)

[2.1.1 Giới thiệu Quy trình phát triển phần mềm 12](#_Toc19731)

[2.1.2 Một số mô hình quy trình phát triển phần mềm 13](#_Toc26858)

[2.2 Phương pháp phân tích thiết kế hướng đối tượng 17](#_Toc2955)

[2.3 Tổng quan về Web API 18](#_Toc12137)

[2.4 Giới thiệu về Retrofit và Volley trong Android studio 20](#_Toc11825)

[2.5. REST và RESTful API 21](#_Toc1405)

[2.6 APS.NET Web API 22](#_Toc25878)

[2.7 Tổng quan về Drawable và Animation 23](#_Toc2430)

[2.7.1 Giới thiệu về Drawable 23](#_Toc21110)

[2.7.2 Một số Drawable 23](#_Toc30307)

[2.7.3 Giới thiệu về Animation 24](#_Toc27438)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 24](#_Toc9207)

[3.1 Đặc tả yêu cầu ứng dụng 25](#_Toc13283)

[3.1.1 Các yêu cầu chức năng của hệ thống 26](#_Toc7932)

[3.1.2 Biểu đồ lớp thực thể 42](#_Toc19338)

[3.1.3 Các yêu cầu phi chức năng 43](#_Toc258)

[3.2 Thiết kế hệ thống 44](#_Toc24863)

[3.2.1 Thiết kế cơ sở dữ liệu 44](#_Toc26985)

[3.2.2 Thiết kế các lớp và các đối tượng 46](#_Toc5336)

[3.2.3 Thiết kế giao diện ứng dụng 55](#_Toc2807)

[CHƯƠNG 4: TRIỂN KHAI ỨNG DỤNG 58](#_Toc27235)

[4.1 Triển khai các chức năng của ứng dụng 58](#_Toc19960)

[4.1.1 Các chức năng nghiệp vụ 58](#_Toc2080)

[4.1.2 Các chức năng hệ thống 60](#_Toc11166)

[4.2 Kiểm thử và triển khai ứng dụng 60](#_Toc2352)

[KẾT LUẬN 63](#_Toc7816)

[TÀI LIỆU THAM KHẢO 64](#_Toc458)

# DANH SÁCH CÁC THUẬT NGỮ

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Cụm từ đầy đủ** |
| 1 | API | Application Programming Interface |
| 2 | DB | DataBase |
| 3 | CSS | Cascading Style Sheets |
| 4 | GCM | Google Cloud Messaging |
| 5 | GPS | Global Positioning Service |
| 6 | GUI | Graphical User Interface |
| 7 | HTML | Hyper Text Markup Language |
| 8 | HTTP | Hyper Text Transfer Protocol |
| 9 | HTTPS | Secure Hyper Text Transfer Protocol |
| 10 | IP | Internet Protocol |
| 11 | JDBC | Java Database Connectivity |
| 12 | JSON | JavaScript Object Notation |
| 13 | LINQ | Language Integrated Query |
| 14 | REST | Representational State Transfer |
| 15 | RESTFul | Representational State Transfer |
| 16 | RPC | Remote Procedure Calls |
| 17 | RSS | Really Simple Syndication |
| 18 | SOAP | Simple Object Access Protocol |
| 19 | SQL | Structured Query Language |
| 20 | SSL | Secure Sockets Layer |
| 21 | TCP | Transmission Control Protocol |
| 22 | TCP/IP | Transmission Control Protocol/Internet Protocol |
| 23 | URI | Uniform Resource Identifier |
| 24 | URL | Uniform Resource Locator |
| 25 | WCF | Windows Communication Foundation |
| 26 | WSDL | Web Service Description Language |
| 27 | XML | Extensible Markup Language |

# CHƯƠNG 1: MỞ ĐẦU

## Lý do chọn đề tài

Trong những năm trở lại gần đây, nền tảng di động phát triển cực kì mạnh mẽ. Nó là công cụ thuận tiện nhỏ bé mà con người tạo ra cho đến ngày nay. Đặc biệt là những chiếc điện thoại Android có số lượng người dùng vô cùng lớn, Android ở khắp mọi nơi trên toàn cầu, nhắc đến nền tảng di động không thể thiếu hệ điều hành Android, nó vẫn là hệ điều hành nắm giữ vị trí số 1 cho đến ngày nay chiếm giữ 85% số lượng thiết bị di động . Vì vậy việc xây dựng ứng dụng trên nền tảng này ngày cảng phổ biến và phát triển cực kì mạnh mẽ đáp ứng được ngày càng nhiều yêu cầu thực tế của người dùng.

Các ứng dụng trên android có thể làm được rất nhiều thứ phục vụ con người, nó phục vụ cho công việc, tính toán, học tập, liên lạc. Xã hội thì ngày càng phát triển, việc phục vụ giải trí cho con người là rất cần thiết. Vì vậy các nhà phát triển đã hướng tới xây dựng ứng dụng đa phương tiện nhằm phục vụ nhu cầu thiết yếu cho con người như: xem phim, nghe nhạc, mạng xã hội, chụp ảnh, chơi game, mua sắm. Một trong những nhu cầu đó thì ứng dụng mua sắm được rất nhiều người lựa chọn là biện pháp tiện lợi nhất.

Việc xây dựng ứng dụng bán quần áo trên thiết bị di động Android không chỉ dễ dùng mà còn mang lại nhiều lợi ích: không cần phải đi đến các cửa hàng bán quần áo và không chắc chắn rằng có tìm được thứ mình mong muốn hay không, có thể mua sắm ở mọi nơi.

- Tăng doanh thu: Phần mềm bán quần áo online giúp bạn tiếp cận được nhiều khách hàng hơn thông qua các kênh online như website, app, ví điện tử hay các đơn vị vận chuyển. Bạn có thể tạo ra các chương trình khuyến mãi, combo hay tặng kèm để kích thích khách hàng đặt hàng nhiều hơn.

- Tiết kiệm chi phí: Phần mềm bán quần áo online giúp bạn tiết kiệm chi phí nhân sự, thiết bị và vật tư. Bạn không cần thuê quá nhiều nhân viên để quản lý cửa hàng hay giao hàng. Bạn cũng không cần đầu tư vào các thiết bị chuyên dụng như máy POS hay máy in hóa đơn. Bạn chỉ cần sử dụng các thiết bị có sẵn như điện thoại hay máy tính để quản lý và thanh toán. Phần mềm cũng giúp bạn định số lượng và quản lý kho chặt chẽ để hạn chế thất thoát và lãng phí.

Trên cơ sở kiến thức đã được học ở trường và quá trình tìm hiểu của bản thân cùng với sự tìm tòi khám phá công nghệ. Vì vậy, em quyết định xây dựng một ứng dụng bán quần áo mang tên King of Fashion trên thiết bị di động Android nhằm phục vụ nhu cầu mua sắm của con người.

## Mục tiêu của đề tài

### Mục tiêu tổng quát

Mục tiêu tổng quát của ứng dụng bán quần áo online là cung cấp cho khách hàng một trải nghiệm mua sắm tiện lợi, nhanh chóng và an toàn. Ứng dụng phải có các tính năng sau:

- Cho phép khách hàng xem danh sách các loại quần áo, giá cả và thông tin sản phẩm.

- Cho phép khách hàng đặt hàng và thanh toán trực tuyến qua các phương thức như thẻ tín dụng, ví điện tử hoặc COD.

- Cho phép khách hàng theo dõi trạng thái đơn hàng và nhận thông báo khi đơn hàng được giao.

- Cho phép khách hàng đánh giá chất lượng sản phẩm và dịch vụ sau khi nhận hàng.

- Cho phép khách hàng tích lũy điểm thưởng và nhận các ưu đãi hấp dẫn từ chương trình khách hàng thân thiết.

Ngoài ra, ứng dụng cũng phải đảm bảo về mặt kỹ thuật, bảo mật và vận hành. Ứng dụng phải có giao diện thân thiện, tương thích với nhiều loại thiết bị và hệ điều hành. Ứng dụng cũng phải bảo vệ thông tin cá nhân của khách hàng. Cuối cùng, ứng dụng cũng phải có hỗ trợ kỹ thuật và chăm sóc khách hàng tốt để giải quyết các vấn đề phát sinh.

### Mục tiêu cụ thể

Ứng dụng bán quần áo online là một giải pháp hiện đại và tiện lợi cho những người yêu thích mua sắm. Để tạo ra một ứng dụng thành công và thu hút khách hàng, ứng dụng phải có những mục tiêu cụ thể sau:

- Thiết kế giao diện đẹp mắt, thân thiện và dễ sử dụng. Giao diện là yếu tố quan trọng để tạo ấn tượng đầu tiên cho người dùng và khuyến khích họ trải nghiệm ứng dụng.

- Cung cấp thông tin chi tiết và chính xác về các loại sản phẩm, giá cả, khuyến mãi và chính sách giao hàng. Thông tin này giúp người dùng lựa chọn sản phẩm phù hợp với nhu cầu và túi tiền của họ, cũng như tăng sự tin tưởng và uy tín của ứng dụng.

- Tích hợp các tính năng thanh toán an toàn, nhanh chóng và linh hoạt. Người dùng có thể thanh toán qua các phương thức như thẻ tín dụng, ví điện tử, COD hay QR code. Đây là yếu tố quan trọng để nâng cao trải nghiệm của người dùng và giảm thiểu rủi ro khi giao dịch online.

- Tạo ra một cộng đồng người yêu mua sắm trên ứng dụng. Người dùng có thể chia sẻ ý kiến, đánh giá, góp ý về các sản phẩm và dịch vụ của ứng dụng. Đây là cách để thu thập phản hồi từ khách hàng, cũng như tăng sự gắn kết và trung thành của họ với ứng dụng.

Những mục tiêu trên là những điều cần thiết để xây dựng một ứng dụng bán quần áo online chất lượng và hiệu quả.

## Giới hạn và phạm vi của đề tài

### Đối tượng nghiên cứu

Đối tượng nghiên cứu của ứng dụng bán quần áo online là những người tiêu dùng có nhu cầu mua sắm trực tuyến và sử dụng ví điện tử để thanh toán. Để xác định được đối tượng nghiên cứu, ta cần phân tích các yếu tố ảnh hưởng đến hành vi sử dụng ứng dụng bán quần áo online của người tiêu dùng, như thái độ, ý thức, kinh nghiệm, thu nhập, giới tính, tuổi tác và văn hóa. Ngoài ra, ta cũng cần khảo sát các xu hướng nghiên cứu marketing xanh trong lĩnh vực thương mại điện tử để tìm ra các giải pháp bền vững cho ứng dụng bán quần áo online.

### Phạm vi nghiên cứu

Phạm vi nghiên cứu của ứng dụng bán quần áo online bao gồm các khía cạnh sau: thị trường tiềm năng, nhu cầu và hành vi của khách hàng, đối thủ cạnh tranh, chiến lược tiếp thị và phân phối, mô hình kinh doanh và tài chính. Mục tiêu của nghiên cứu là xác định các yếu tố ảnh hưởng đến thành công của ứng dụng và đề xuất các giải pháp để tối ưu hóa hiệu quả hoạt động. Nghiên cứu sử dụng các phương pháp thu thập dữ liệu như khảo sát, phỏng vấn, quan sát và phân tích số liệu.

Nghiên cứu được thực hiện trong vòng 6 tháng từ tháng 1 đến tháng 6 năm 2023.

## Nội dung thực hiện

Chức năng hệ thống:

* Cập nhật thông tin, cung cấp đầy đủ thông tin về các loại mặt hàng, thông tin đơn hàng của các khách hàng.
* Tìm kiếm các loại thông tin về thời trang nam và nữ, thông tin đơn hàng và các hóa đơn bán xuất và nhập.
* Báo cáo và thống kê đầy đủ các bài toán thực tế của ứng dụng.
* Nghiệp vụ đầy đủ chức năng thực tiễn.

## Phương pháp tiếp cận

Khảo sát thực tế hệ thống kết hợp nghiên cứu lý thuyết để xây dựng cơ sở dữ liệu.

Lựa chọn công cụ lập trình và tiến hành xây dựng các module xử lý.

Android Studio: Dùng để lập trình.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Quy trình phát triển phần mềm

### Giới thiệu Quy trình phát triển phần mềm

Quy trình xây dựng phần mềm bao gồm tập hợp các thao tác và kết quả tương quan sử dụng trong việc phát triển để sản xuất ra một sản phẩm phần mềm. Mỗi một phần mềm lại có đặc điểm và yêu cầu khác nhau, tuy nhiên tất cả đều cần phải trải qua các bước sau:

**− Khảo sát:**

+ Xác định yêu cầu là bước đầu tiên của xây dựng phần mềm. Các yêu cầu phải có tính đo lường, kiểm thử được, có liên quan đến các nhu cầu hoặc cơ hội doanh nghiệp đã được xác định, và phải được định nghĩa ở mức độ chi tiết đủ cho việc thiết kế hệ thống.

**− Thiết kế:**

+ Là một quá trình giải quyết vấn đề và lập kế hoạch cho một giải pháp phần mềm. Sau khi các mục đích và các đặc điểm kỹ thuật của phần mềm được giải quyết, lập trình viên sẽ thiết kế hoặc thuê người thiết kế để phát triển một kế hoạch cho giải pháp phần mềm. Nó bao gồm các thành phần cấp thấp, các vấn đề thuật toán cũng như một khung nhìn kiến trúc. Thiết kế chức năng, cơ sở dữ liệu và giao diện.

**− Kiểm thử:**

+ Kiểm thử phần mềm là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng của sản phẩm hoặc dịch vụ sau khi đã được kiểm tra. Kiểm thử có thể cung cấp cho doanh nghiệp một chất lượng đảm bảo, một cách nhìn khách quan về phần mềm để từ đó đánh giá và thấu hiểu được những rủi ro trong quá trình triển khai phần mềm.

**− Triển khai, bàn giao:**

Sau khi phần mềm được kiểm thử và khắc phục những sai sót sẽ được triển khai đưa vào sử dụng trong thực tế. Đối với những yêu cầu mới được phát sinh từ phía khách hàng sẽ được chỉnh sửa. Trong quá trình triển khai cũng luôn đánh giá hiệu quả sử dụng của phần mềm, xem xét những nhược điểm để lên kế hoạch thiết kế phần mềm hiệu quả hơn.

**− Bảo trì:**

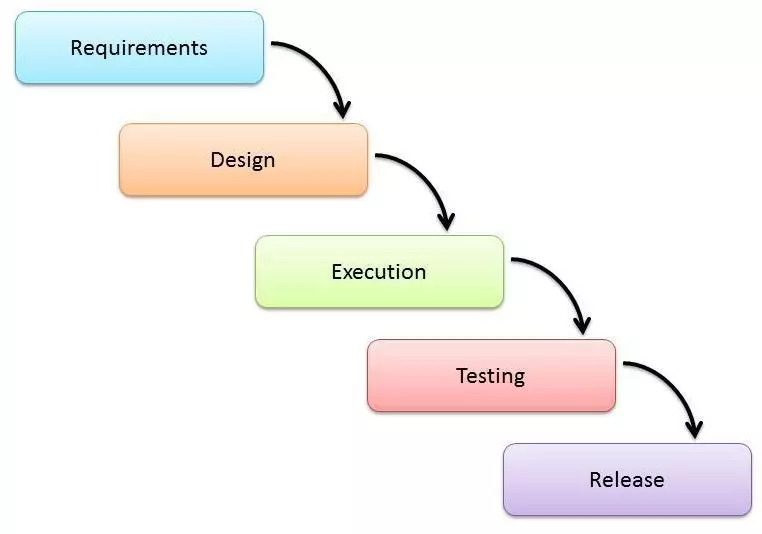
Bảo trì phần mềm bao gồm điều chỉnh các lỗi mà chưa được phát hiện trong các giai đoạn trước chu kỳ sống của phần mềm, nâng cấp tính năng sử dụng và an toàn vận hành của phần mềm. Quá trình phát triển phần mềm bao gồm rất nhiều giai đoạn: thu thập yêu cầu, phân tích, xây dựng, kiểm tra, triển khai và bảo trì phần mềm. Nhiệm vụ của giai đoạn bảo trì phần mềm là giữ cho phần mềm được cập nhật khi môi trường thay đổi và yêu cầu người sử dụng thay đổi. Để xây dựng phần mềm thiết thực, mang lại hiệu quả kinh tế cao.

**− Phát triển tính năng:**

Lắng nghe các góp ý của khách hàng, tìm hiểu các phát sinh trong tình hình mới.

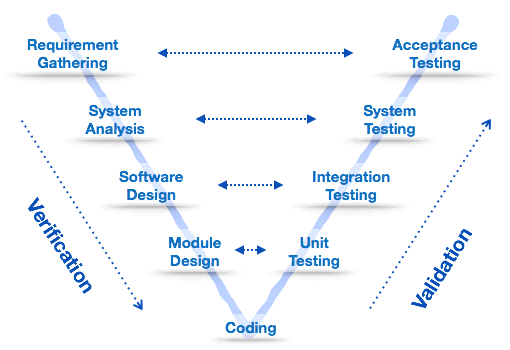
### 2.1.2 Một số mô hình quy trình phát triển phần mềm

* Mô hình thác nước

******

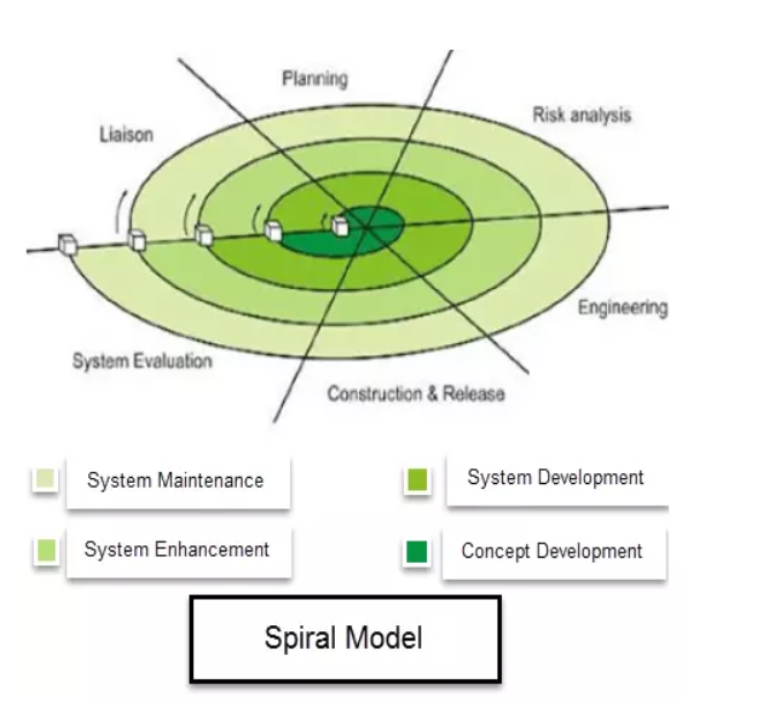
***Hình 2.1 Mô hình thác nước***

* Thu thập yêu cầu (Requirement gathering): Đây là giai đoạn xác định các yêu cầu chức năng và phi chức năng mà hệ thống phần mềm cần có. Kết quả của giai đoạn này là bản tài liệu đặc tả yêu cầu. Tài liệu này sẽ là nền tảng cho những giai đoạn tiếp theo cho đến cuối dự án.
* Phân tích hệ thống (System Analysis): Là giai đoạn định ra làm thế nào để hệ thống phần mềm đáp ứng đúng yêu cầu của khách hàng. Giai đoạn này thực hiện phân tích, thiết kế hệ thống phần mềm.
* Coding: Là giai đoạn thực hiện sản phẩm dựa trên đặc tả yêu cầu và tài liệu thiết kế module
* Testing: Tester sẽ nhận sản phẩm từ developer và thực hiện kiểm thử cho nhóm các thành phần và kiểm thử hệ thống. Khâu kiểm thử cuối cùng sẽ là Kiểm thử chấp nhận, giai đoạn này còn có sự tham gia của khách hàng.
* Implementation: Triển khai hệ thống ra môi trường của khách hàng.
* Operations & Maintenance: Đây là giai đoạn cài đặt, cấu hình và đào tạo cho khách hàng. Giai đoạn này sửa chữa những lỗi của sản phẩm (nếu có) và phát triển những thay đổi mới được khách hàng yêu cầu.
* Mô hình chữ V



***Hình 2.2 Mô hình chữ V***

* Hoạt động tốt với các dự án có quy mô vừa và nhỏ.
* Dễ dàng quản lý vì mỗi giai đoạn có các mục tiêu và mục tiêu được xác định rõ ràng.
* Toàn bộ quy trình được chia thành 2 nhóm giai đoạn tương ứng nhau là phát triển và kiểm thử. Mỗi giai đoạn phát triển sẽ tiến hành song song với một giai đoạn kiểm thử tương ứng. Do đó, các lỗi được phát hiện sớm ngay từ đầu.
* Mô hình xoắn ốc Boehm



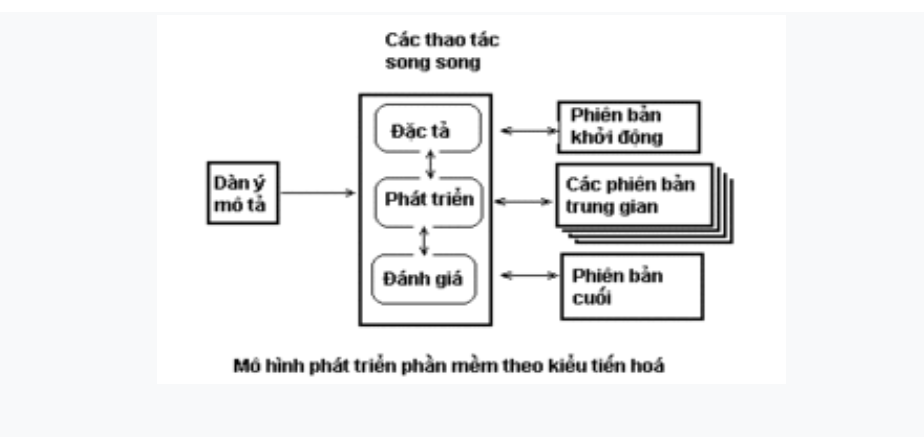
***Hình 2.3 Mô hình xoắn ốc***

Đây là mô hình phát triển từ mô hình thác nước cho thấy mức độ tổng quát hơn của các pha sản xuất của một sản phẩm. Mô hình được đề nghị bởi [Boehm](https://vi.wikipedia.org/w/index.php?title=Boehm&action=edit&redlink=1) vào năm [1988](https://vi.wikipedia.org/wiki/1988). Mô hình này có thể chỉ ra các rủi ro có thể hình thành trên căn bản của mô hình quy trình (sản xuất) tổng quát.

Mô hình Boehm có dạng xoắn ốc. Mỗi vòng lặp đại diện cho một pha của quy trình phần mềm. Vòng trong cùng tập trung về tính khả thi, vòng kế lo về định nghĩa các yêu cầu, kế đến là thiết kế,...

Không có một pha nào được xem là cố định trong vòng xoắn. Mỗi vòng có 4 phần tương ứng với một pha.

* Cài đặt đối tượng: Chỉ ra các đối tượng của pha trong đề án. Những khó khăn hay cưỡng bức của quy trình và của sản phẩm được xác định và được lên kế hoạch chi tiết. Xác định các yếu tố rủi ro của đề án. Các phương án thay thế tùy theo các rủi ro này có thể được dự trù.
* Lượng định và giảm thiểu rủi ro. Tiến hành phân tích mỗi yếu tố rủi ro đã xác định. Các bước đặt ra để giảm thiểu rủi ro.
* Phát triển và đánh giá: Sau khi đánh giá các yếu tố rủi ro, một mô hình phát triển cho hệ thống được chọn.
* Lên kế hoạch: Đề án được xem xét và quyết định có nên hay không tiếp tục pha mới trong vòng lặp.
* Mô hình phát triển tiến hoá



***Hình 2.4 Mô hình tiến hóa***

Phân loại sự phát triển tiến hóa:

1. Lập trình thăm dò: đối tượng của quy trình bằng cách làm việc với khách hàng để thăm dò các yêu cầu và phân phối phần mềm dứt điểm. Sự phát triển nên bắt đầu với những phần nào đã được hiểu rõ. Phần mềm sẽ được thêm vào các chức năng mới khi mà nó được đề nghị cho khách hàng (và nhận về các thông tin).

#### Mẫu thăm dò: đối tượng của phát triển tiến hoá này là nhằm hiểu các yêu cầu của khách hàng và do đó phát triển các định nghĩa yêu cầu tốt hơn cho phần mềm. Các mẫu tập trung trên các thí nghiệm với những phần đòi hỏi nào của khách hàng mà có thể gây sự khó hiểu hay ngộ nhận.

* Phân tích mô hình: Mô hình phát triển tiến hóa này hiệu quả hơn mô hình thác nước. Tuy nhiên, nó vẫn còn các khuyết điểm:

1. Quy trình thì không nhìn thấy rõ được: Các nhà quản lý cần phân phối thường xuyên để đo lường sự tiến bộ. Nó không kinh tế trong việc làm ra các hồ sơ cho phần mềm.
2. Phần mềm thường được cấu trúc nghèo nàn: Sự thay đổi liên tục dễ làm đổ vỡ cấu trúc của phần mềm, tạo ra sự khó khăn và tốn phí.
3. Thường đòi hỏi những kỹ năng đặc biệt: Hầu hết các hệ thống khả dĩ theo cách này được tiến hành bởi các nhóm nhỏ có kỹ năng cao cũng như các cá nhân phải năng động.

* Mô hình này thích hợp với:

1. Phát triển các loại phần mềm tương đối nhỏ.
2. Phát triển các loại phần mềm có đời sống tương đối ngắn.
3. Tiến hành trong các hệ thống lớn hơn ở những chỗ mà không thể biểu thị được các đặc tả chi tiết trong lúc tiến hành. Thí dụ của trường hợp này là các hệ thống thông minh nhân tạo ([AI](https://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o)) và các giao diện cho người dùng.

## Phương pháp phân tích thiết kế hướng đối tượng

### Giới thiệu ngôn ngữ UML

* **Ngôn ngữ UML là gì ?**

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.

### 2.2.2 Các bước phân tích thiết kế hướng đối tượng

Phân tích và thiết kế hướng đối tượng là một kỹ thuật tiếp cận phổ biến dùng để phân tích, thiết kế một ứng dụng, hệ thống. Nó dựa trên bộ các nguyên tắc chung, đó là một tập các hướng dẫn để giúp chúng ta tránh khỏi một thiết kế xấu. 5 nguyên tắc trong thiết kế hướng đối tượng:

* Xác định các đối tượng và nhóm thành các lớp
* Xác định mối quan hệ giữa các lớp
* Tạo sơ đồ mô hình đối tượng người dùng
* Xác định thuộc tính đối tượng người dùng
* Xác định các hoạt động nên được thực hiện trên các lớp

## Tổng quan về Web API

Theo Wikipedia API là “An application programming interface (API) is a computing

interface which defines interactions between multiple software intermediaries.

API là từ viết tắt của “Application Programming Interface” với nghĩa là một giao diện

lập trình ứng dụng. Đây là phần mềm trung gian cho phép kết nối 2 ứng dụng với nhau.

### Giới thiệu về Web API

Web API là một giao diện lập trình ứng dụng (API) cho phép các ứng dụng web giao tiếp với nhau qua mạng. Web API cung cấp các phương thức để truy xuất, tạo, cập nhật hoặc xóa dữ liệu từ một nguồn dữ liệu trên web. Web API thường sử dụng các định dạng dữ liệu như JSON hoặc XML để trao đổi thông tin giữa các bên. Web API có thể được sử dụng cho nhiều mục đích khác nhau, chẳng hạn như tích hợp các dịch vụ bên ngoài, cung cấp chức năng cho các ứng dụng di động hoặc web, hoặc tạo ra các ứng dụng phân tán. Web API là một công nghệ quan trọng trong lĩnh vực phát triển web hiện đại.

### Một số đặc điểm chính của Web API

Một số đặc điểm chính của web API là:

- Web API là một giao diện lập trình ứng dụng (API) cho phép các ứng dụng web giao tiếp với nhau qua giao thức HTTP.

- Web API có thể sử dụng nhiều định dạng dữ liệu khác nhau, nhưng phổ biến nhất là JSON và XML.

- Web API có thể hỗ trợ nhiều phương thức HTTP, như GET, POST, PUT, DELETE, PATCH, HEAD và OPTIONS, để thực hiện các thao tác khác nhau trên tài nguyên web.

- Web API có thể tuân theo các tiêu chuẩn và quy ước thiết kế khác nhau, như RESTful, SOAP, GraphQL và OData, để tăng tính nhất quán, khả năng mở rộng và bảo mật của giao diện.

- Web API có thể được phát triển bằng nhiều ngôn ngữ lập trình và công nghệ khác nhau, như Java, Python, .NET, Node.js và Ruby on Rails.

### 2.3.3 Truy vấn trong Web API

Truy vấn trong web api là một kỹ năng quan trọng cho các nhà phát triển web, bởi vì nó cho phép tương tác với các nguồn dữ liệu online một cách hiệu quả và linh hoạt. Truy vấn trong web api bao gồm các hoạt động như:

- Lấy dữ liệu từ các trang web có yêu cầu đăng nhập, bằng cách sử dụng các công cụ hỗ trợ phân tích cơ chế xác thực và chuyển đổi các thông tin truy vấn từ trình duyệt sang M-Code trong Power Query.

- Gửi và nhận dữ liệu qua các API, bằng cách sử dụng các thư viện hỗ trợ hoặc viết mã M-Code để thiết lập kết nối, gửi yêu cầu và xử lý phản hồi.

- Tìm kiếm và lọc dữ liệu theo các tiêu chí khác nhau, bằng cách sử dụng các tùy chọn truy vấn hệ thống hoặc tự định nghĩa các hàm lọc trong M-Code.

- Sắp xếp và phân trang dữ liệu, bằng cách sử dụng các tham số trong URL hoặc các hàm trong M-Code để chỉ định thứ tự và số lượng bản ghi trả về.

Truy vấn trong web api là một kỹ năng có thể áp dụng cho nhiều nguồn dữ liệu online khác nhau, như chứng khoán, vận đơn, kế toán, tiền kỹ thuật số, cơ sở kiến thức, v.v. Để truy vấn trong web api một cách thành công, bạn cần có kiến thức về cơ chế hoạt động của web api, cách phân tích và chuyển đổi các truy vấn từ trình duyệt sang M-Code, và cách sử dụng các công cụ và thư viện hỗ trợ cho việc tương tác với web api.

## 2.4 Giới thiệu về Retrofit và Volley trong Android studio

### 2.4.1 Retrofit

Retrofit là một thư viện HTTP Client cho Android và Java được phát triển bởi Square. Retrofit giúp bạn dễ dàng kết nối tới một REST web service bằng cách dịch API thành các Java interface. Retrofit hỗ trợ bạn xử lý dữ liệu JSON hoặc XML và chuyển đổi chúng thành các đối tượng Java (POJO). Retrofit cũng hỗ trợ bạn thực hiện các yêu cầu HTTP như GET, POST, PUT, DELETE, PATCH với các annotation tương ứng. Retrofit sử dụng OkHttp để gửi và nhận các yêu cầu mạng một cách hiệu quả.

Retrofit có nhiều ưu điểm so với các thư viện khác như:

- Retrofit sử dụng các annotation để định nghĩa các phương thức HTTP, các tham số và các header cho mỗi request. Điều này giúp cho việc viết code trở nên gọn gàng và rõ ràng hơn.

- Retrofit hỗ trợ nhiều loại converter để chuyển đổi dữ liệu từ dạng thô (JSON, XML) sang dạng đối tượng Java và ngược lại. Các converter phổ biến nhất là Gson, Moshi và Jackson.

- Retrofit cũng hỗ trợ nhiều loại adapter để xử lý các yêu cầu bất đồng bộ hoặc đồng bộ. Các adapter phổ biến nhất là RxJava, Coroutines và LiveData.

- Retrofit có hiệu năng cao và thời gian thực thi nhanh hơn so với AsyncTask hay Volley. Retrofit cũng sử dụng thư viện OkHttp để quản lý các kết nối HTTP, lưu trữ cache và xử lý lỗi.

### 2.4.2 Volley

Volley là một thư viện networking cho Android được Google giới thiệu vào năm 2013. Volley giúp cho việc kết nối và xử lý dữ liệu từ server trở nên đơn giản và hiệu quả hơn. Volley cung cấp các lớp và phương thức để tạo và quản lý các network request, hỗ trợ cache, huỷ request, thiết lập độ ưu tiên và nhiều tính năng khác. Volley cũng hỗ trợ nhiều loại kết quả trả về như String, JSONObject, JSONArray, Bitmap...

Để sử dụng Volley trong Android Studio, bạn cần thêm dependency vào file build.gradle của module app và cấp quyền Internet cho ứng dụng

Volley có nhiều ưu điểm cho việc lập trình ứng dụng Android, chẳng hạn như:

- Tự động lập lịch cho các yêu cầu mạng và hỗ trợ đặt độ ưu tiên cho chúng.

- Bộ nhớ đệm trong bộ nhớ và trên đĩa cho các phản hồi mạng với sự phù hợp với bộ nhớ đệm HTTP chuẩn.

- Hỗ trợ nhiều loại kết quả trả về, như chuỗi, đối tượng JSON, mảng JSON, bitmap...

- Có thể hủy yêu cầu mạng theo API hoặc theo khối hoặc phạm vi yêu cầu.

- Dễ dàng tùy biến, ví dụ như cho thử lại và lùi lại.

- Thứ tự rõ ràng giúp dễ dàng điền đúng dữ liệu vào giao diện người dùng với dữ liệu được tải xuống không đồng bộ từ mạng.

- Công cụ gỡ lỗi và theo dõi.

Volley rất phù hợp cho các hoạt động kiểu RPC được sử dụng để điền giao diện người dùng, chẳng hạn như tìm kiếm một trang kết quả dưới dạng dữ liệu có cấu trúc. Nó tích hợp dễ dàng với bất kỳ giao thức nào và có sẵn hỗ trợ cho chuỗi thô, hình ảnh và JSON. Bằng cách cung cấp hỗ trợ tích hợp cho các tính năng bạn cần, Volley giúp bạn thoát khỏi việc viết mã lặp đi lặp lại và cho phép bạn tập trung vào logic riêng biệt của ứng dụng của bạn.

# 2.5. REST và RESTful API

# REST (REpresentational State Transfer) về cơ bản là một kiểu kiến trúc để phát triển ứng dụng dịch vụ web. REST phổ biến do tính đơn giản của nó và thực tế là nó được xây dựng dựa trên các hệ thống và tính năng hiện có của Giao thức truyền siêu văn bản trên internet (HTTP) để đạt được các mục tiêu của nó, thay vì tạo ra các tiêu chuẩn, khuôn khổ và công nghệ mới. RESTful API là một kiểu kiến trúc gọi REST sử dụng các yêu cầu GET, PUT, POST và DELETE tương ứng với thao tác với dữ liệu là Read, Update, Create và Delete. REST có một số nguyên tắc như sau:

# REST cho phép các máy khách truy cập máy chủ API và thực hiện các lệnh lấy về, chỉnh sửa hay xóa dữ liệu từ external server. Các lập trình viên có thể thoải mái truy xuất, chỉnh sửa dữ liệu từ máy chủ mà không cần biết hệ thống hoạt động như thế nào.

# Giao thức chính của REST sử dụng là HTTP, một giao thức phổ biến với hầu hết các ứng dụng hay dịch vụ web hiện nay. Nó đem tới hiệu qu nhanh chóng trong bối cảnh đường truyền mạnh mẽ và khiến cho REST kiến trúc tốc độ nhanh hơn.

# REST cũng có ưu điểm khi sử dụng giao thức stateless (không trạng thái). Hệ thống này không sử dụng session, cookie, không cần biết những thông tin đó trong mỗi lần request đến máy chủ ngoài. Điều này giúp REST giảm tải cho máy chủ ngoài, nâng cao hiệu suất làm việc.

# REST ban đầu được thiết kế để sử dụng cho các dịch vụ web. Tuy nhiên, bất cứ phần mềm nào cũng có thể ứng dụng REST làm cầu nối giao tiếp với các ứng dụng khác tốt và hiệu quả hơn. Đó cũng là lý do giúp cho REST trở thành tiêu chuẩn mặc định cho hầu hết các giao thức. Và những API đượ thiết kế theo cấu trúc REST được gọi là RESTful API.

# Nó sẽ chỉ trả về kết quả dưới dạng JSON hoặc XML, nguyên tử, OData, v.v. (dữ liệu nhẹ). Trong đó:

# REST based services Thực hiện theo một số nguyên tắc trên và không phải tất cả

# RESTFUL services có nghĩa là nó tuân theo tất cả các nguyên tắc trên.

# **2.6 APS.NET Web API**

# The ASP.NET Web API là một framework có khả năng mở rộng để xây dựng các dịch vụ dựa trên HTTP, nó có thể thể truy cập trong các ứng dụng khác nhau trên các nền tảng khác nhau như web, windows, mobile, v.v.

# Cách thức hoạt động của ASP.NET Web API hầu như giống như cách thức ứng dụng web ASP.NET MVC, ASP.NET Web API hoạt động khác ASP.NET MVC ở chỗ nó gửi dữ liệu dưới dạng các respond (phản hồi) thay vì trả lại các html view.

# ASP.NET Web API giống như một webservice hoặc WCF service nhưng nó khác là nó chỉ hỗ trợ giao thức HTTP. Các dịch vụ Web API có thể được sử dụng bởi nhiều loại clients khác nhau thông qua các request. Ví dụ: Browsers, Mobile applications, Desktop applications, IOTs, ...

# **2.7 Tổng quan về Drawable và Animation**

# **2.7.1 Giới thiệu về Drawable**

# Drawable là một khái niệm chung ám chỉ đối tượng đồ họa nào đó có thể được vẽ ra trên màn hình. Trong dự án ứng dụng Android với Android Studio thường các tài nguyên đồ họa này đặt trong thư mục res/drawable, sau đó nó được sử dụng bằng cách lấy ra từ trong code bằng các hàm API như ContextCompat.getDrawable, RC.getDrawable hay gán vào các thuộc tính phù hợp trong XML như android:drawable, android: icon,....

# **2.7.2 Một số Drawable**

***Bảng 2.1 giới thiệu về Drawble***

|  |  |
| --- | --- |
| ShapeDrawable | Là một Drawable bằng phần tử shape trong XML, nó cho phép vẽ một số hình cơ bản. |
| GradientDrawable | Tạo Gradient bên trong shape |
| VectorDrawable | Là một Drawable bằng phần tử vector trong XML, nó cho phép vẽ ảnh vector như SVG |
| BitmapDrawable | Biểu diễn ảnh Bitmap, hỗ trợ các định dạng png, jpg, gif, các tài nguyên ảnh png, jpg đặt trong drawable tự động được phân tích thành BitmapDrawable |
| NinePatchDrawable | Các ảnh png với phần tên có dạng .9.png sẽ tự động phân tích thành dạng Nine-Patch, loại Drawable cho phép thu phóng phần nội dung |
| LayerDrawable | Một loại Drawable, chứa nhiều Drawable bên trong, các Drawable bên trong được vẽ theo thứ tự. |
| StateListDrawable | Một Drawable có chứa nhiều Drawable bên trong để cho đối tượng sử dụng tùy trạng thái mà vẽ Drawable trong nó |
| LevelListDrawable | Một Drawable có chứa nhiều Drawable bên trong, mỗi Drawable được giá trị Level (min và max). Sau đó đối tượng sử dụng tùy giá trị Level mà sẽ sử dụng Drawable nào bên trong để vẽ. |
| TransitionDrawable | Một Drawable có chứa 2 Drawable bên trong, nó cung cấp khả năng chuyển đổi qua lại giữa 2 Drawable (hiệu ứng fade). |
| InsetDrawable | Một Drawable tạo ra từ cách chèn một Drawable khác vào và chỉ ra khoảng cách các mép. |
| ClipDrawable | Một Drawable dựa trên Level, nó cho phép thiết lập cắt chiều rộng / cao dưa trên giá trị level |
| ScaleDrawable | Là một Drawable tạo ra từ Drawable khác bằng cách điều chỉnh tỷ lệ thu phóng- Download & sử dụng miễn phí |

# **2.7.3 Giới thiệu về Animation**

# Animation là những chuyển động hay tập hợp những chuyển động trên các đối

# tượng giúp chúng chuyển động mượt mà hơn.

# Thực chất Animation trong Android chỉ là sự thay đổi những thuộc tính của

# View. Khi có sự thay đổi, Android sẽ tiến hành vẽ lại View (gọi lại phương thức onDraw của class View). Animation là sự chuyển động của 1 đối tượng trong 1 khoảng thời gian.CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

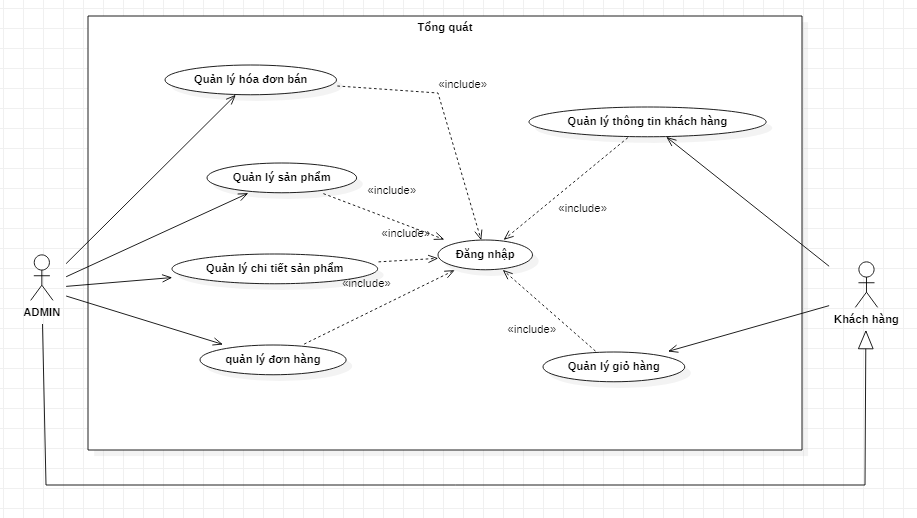
## Đặc tả yêu cầu ứng dụng

***Bảng 3.1. Bảng yêu cầu chức năng***

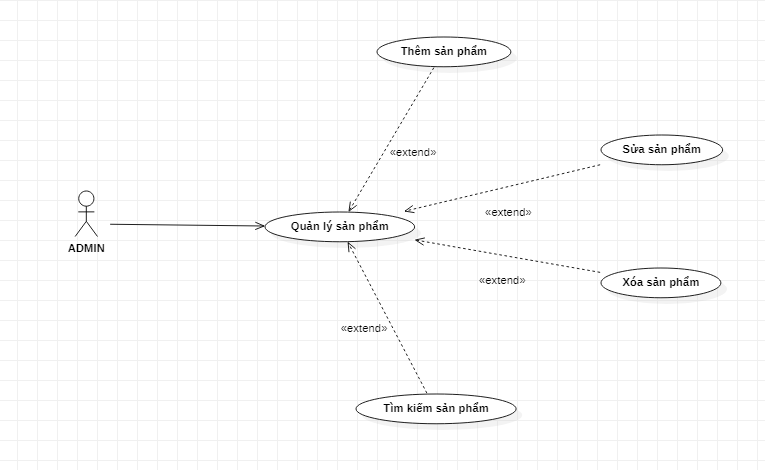
|  |  |  |
| --- | --- | --- |
| **Mục** | **Tên yêu cầu** | **Mô tả yêu cầu** |
| **I** | **Các yêu cầu chức năng nghiệp vụ** | Là các chức năng của phần mềm tương ứng với các công việc trong thế giới thực. (Là một ứng dụng bán quần áo online cho 1 cửa hàng lên các mặt hàng không chỉ có quần áo mà còn có giày, dép, nước hoa, túi xách và các loại phụ kiện để khách hàng sử dụng dùng). |
| 1 | Module quản lý | Hiển thị các mục cần quản lý để cho người sử dụng có thể lựa chọn trong quá trình quản lý:  + Quản lý sản phẩm (quần áo, nước hoa và các loại giày, dép, phụ kiện)  + Quản lý tài khoản  + Quản lý giỏ hàng  + Quản lý hóa đơn bán  + Quản lý sản phẩm  + Quản lý chi tiết sản phẩm  + Quản lý đơn hàng  + Quản lý khách hàng |
| 2 | Module tìm kiếm, hiển thị, update, xóa | + Cho phép người dùng tìm kiếm mục mình cần quản lý và tìm kiếm ra thông tin của các mặt hàng, giỏ hàng, thông báo hoặc hóa đơn trong mục quản lý đó.  + Cho phép người dùng chỉnh sửa thông tin quản lý nếu có sai sót.  + Cho phép xóa nếu thông tin đó không còn tồn tại hoặc không cần thiết. |
| 3 | Module quản lý hóa đơn | Người sử dụng có thể thống kê ra được chi phí trong ngày hoặc trong tháng, cập nhật kịp thời và chính xác. |
| 4 | Quản lý tài khoản | Cho phép người dùng kiểm tra, tìm kiếm, thống kê thông tin của các khách hàng. Lọc được các loại khách hàng để có chính sách kinh doanh hiệu quả hơn. |
| **II** | **Các yêu cầu chức năng hệ thống** | Là các chức năng của phần mềm được phát sinh thêm khi thực hiện một công việc trên máy tính thay vì trong thế thới thực hoặc không tương ứng với bất kỳ công việc nào trong thế giới thực. |
| 1 | Quản lý hệ thống | + Quản lý thông tin tài khoản  + Đổi mật khẩu  + Thêm tài khoản |
| 2 | Quản lý thông tin | + Quản lý các lớp đối tượng lưu trữ  + Quản lý các quyền của đối tượng |
| 3 | Quản lý tài khoản | + Quản lý người dùng  + Quản lý liên hệ |

### Các yêu cầu chức năng của hệ thống

***Hình 3.1: UC tổng quát của hệ thống bán quần áo online***



**a. Use-case phân rã của quản lý sản phẩm.**



***Hình 3.2 Use-case phân rã của quản lý sản phẩm***

* **Kịch bản use-case quản lý sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng quản lý sản phẩm |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Admin yêu cầu chức năng | * 1. Hệ thống lấy thông tin về sản phẩm   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của sản phẩm. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

|  |  |
| --- | --- |
| Kích hoạt | Admin chức năng quản lý sản phẩm |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Admin yêu cầu chức năng | * 1. Hệ thống lấy thông tin về sản phẩm   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của sản phẩm. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

* **Kịch bản Use-case nhập thông tin sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chức năng thêm thông tin sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng nhập thông tin sản phẩm | 1.1 Hệ thống hiển thị giao diện nhập thông tin sản phẩm |
| 1. Admin nhập thông tin về sản phẩm mới. | 2.1 Hệ thống sinh mã tự động cho sản phẩm.  2.2 Hệ thống hợp lệ hóa thông tin  2.3 Hệ thống cập nhật thông tin của các sản phẩm lên hệ thống. |

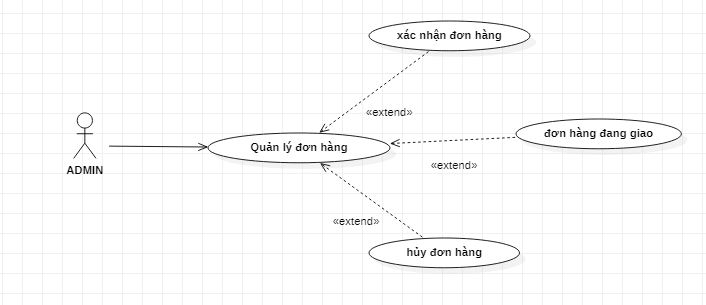
* **Kịch bản Use-case sửa thông tin sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng sửa thông tin sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng sửa thông tin sản phẩm | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Admin sửa thông tin sản phẩm. | 2.1 Hiển thị giao diện sửa. |
| 1. Nhân viên yêu cầu sửa thông tin admin. | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống cập nhật thông tin sản phẩm. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các sản phẩm liên quan đến sản phẩm. |
| Luồng phụ 2 | Xóa đi các sản phẩm chênh lệch. |

* **Kịch bản Use-case xóa thông tin sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng xóa thông tin sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng xóa thông tin sản phẩm. | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Admin chọn thông tin sản phẩm cần xóa |  |
| 1. Admin yêu cầu xóa thông tin sản phẩm | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống xóa thông tin sản phẩm tương ứng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các sản phẩm. |
| Luồng phụ 2 | Xóa đi các sản phẩm chênh lệch. |

**b) Use-case phân rã của quản lý đơn hàng:**



***Hình 3.3 Use-case phân rã của quản lý đơn hàng***

* **Kịch bản use-cae quản lý đơn hàng:**

|  |  |
| --- | --- |
| Tên use case | Quản lý đơn hàng |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng quản lý đơn hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Admin yêu cầu chức năng | * 1. Hệ thống lấy thông tin về đơn hàng   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của đơn hàng. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

* **Kịch bản Use-case xác nhận thông tin đơn hàng:**

|  |  |
| --- | --- |
| Tên use case | Quản lý đơn hàng |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chức năng thêm thông tin đơn hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng nhập thông tin loại sản phẩm | 1.1 Hệ thống hiển thị giao diện nhập thông tin đơn hàng |
| 1. Admin nhập thông tin về loại sản phẩm mới. | 2.1 Hệ thống sinh mã tự động cho đơn hàng.  2.2 Hệ thống hợp lệ hóa thông tin  2.3 Hệ thống cập nhật thông tin của các đơn hàng lên hệ thống. |

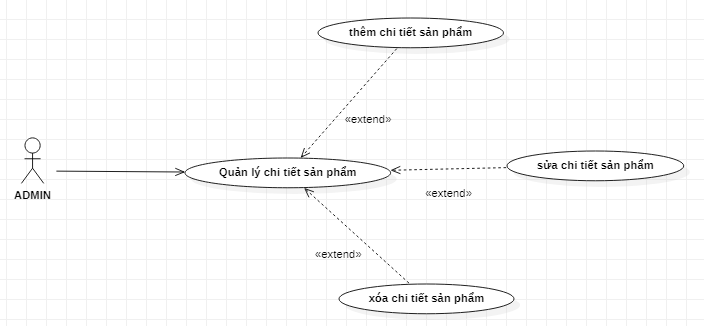
* **Kịch bản Use-case sửa thông tin loại đơn hàng:**

|  |  |
| --- | --- |
| Tên use case | Quản lý đơn hàng |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng sửa thông tin đơn hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng sửa thông tin loại sản phẩm | 1.1 Hệ thống hiển thị thông tin các đơn hàng. |
| 1. Admin sửa thông tin loại sản phẩm. | 2.1 Hiển thị giao diện sửa. |
| 1. Nhân viên yêu cầu sửa thông tin admin. | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống cập nhật thông tin đơn hàng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các sản phẩm liên quan đến đơn hàng. |
| Luồng phụ 2 | Xóa đi các đơn hàng chênh lệch. |

* **Kịch bản Use-case hủy dơn hàng:**

|  |  |
| --- | --- |
| Tên use case | Quản lý đơn hàng |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng xóa thông tin đơn hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng xóa thông tin loại sản phẩm. | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Admin chọn thông tin sản phẩm cần xóa |  |
| 1. Admin yêu cầu xóa thông tin loại sản phẩm | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống xóa thông tin loại sản phẩm tương ứng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các loại sản phẩm. |
| Luồng phụ 2 | Xóa đi các sản phẩm chênh lệch. |

**c) Use-case phân rã của quản lý chi tiết sản phẩm:**



***Hình 3.4 Use-case phân rã của quản lý chi tiết sản phẩm***

* **Kịch bản use-cae quản lý chi tiết sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý chi tiết sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng quản lý chi tiết sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Admin yêu cầu chức năng | * 1. Hệ thống lấy thông tin về chi tiết sản phẩm   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của chi tiết sản phẩm. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

* **Kịch bản Use-case nhập thông tin chi tiết sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý chi tiết sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chức năng thêm thông tin chi tiết sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng nhập thông tin loại sản phẩm | 1.1 Hệ thống hiển thị giao diện nhập thông tin chi tiết sản phẩm |
| 1. Admin nhập thông tin về chi tiết sản phẩm mới. | 2.1 Hệ thống sinh mã tự động cho chi tiết sản phẩm.  2.2 Hệ thống hợp lệ hóa thông tin  2.3 Hệ thống cập nhật thông tin của các chi tiết sản phẩm lên hệ thống. |

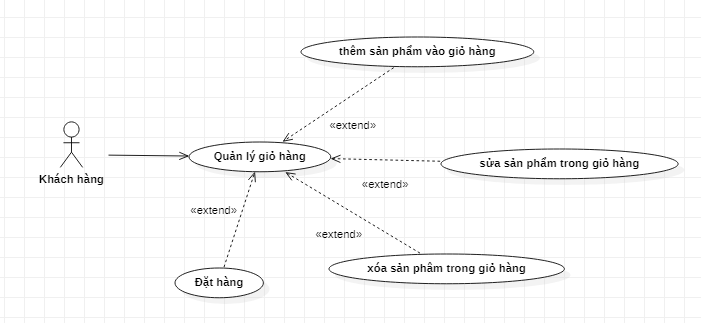
* **Kịch bản Use-case sửa thông tin chi tiết sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý chi tiết sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng sửa thông tin loại sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng sửa thông tin loại sản phẩm | 1.1 Hệ thống hiển thị thông tin các loại sản phẩm. |
| 1. Admin sửa thông tin loại sản phẩm. | 2.1 Hiển thị giao diện sửa. |
| 1. Nhân viên yêu cầu sửa thông tin admin. | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống cập nhật thông tin loại sản phẩm. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các sản phẩm liên quan đến chi tiết sản phẩm. |
| Luồng phụ 2 | Xóa đi các chi tiết sản phẩm chênh lệch. |

* **Kịch bản Use-case xóa thông tin chi tiết sản phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quản lý chi tiết sản phẩm |
| Tên Actor | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Admin chức năng xóa thông tin chi tiết sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng xóa thông tin chi tiết sản phẩm. | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Admin chọn thông tin sản phẩm cần xóa |  |
| 1. Admin yêu cầu xóa thông tin loại sản phẩm | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống xóa thông tin chi tiết sản phẩm tương ứng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Sửa các thông tin các chi tiết sản phẩm. |
| Luồng phụ 2 | Xóa đi các chi tiết sản phẩm chênh lệch. |

**d) Use-case phân rã của quản lý giỏ hàng :**



***Hình 3.5 Use-case phân rã của quản lý giỏ hàng***

* **Kịch bản UC quản lý giỏ hàng:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiohang |
| Tên Actor | Khách hàng |
| Tiền điều kiện | Khách hàng phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Khách hàng chọn chức năng quản lý quản lý giỏ hàng |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Khách hàng yêu cầu chức năng | * 1. Hệ thống lấy thông tin về giỏ hàng của khách hàng.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của giỏ hàng. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

* **Kịch bản UC thêm sản phẩm vào giỏ:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiohang |
| Tên Actor | Khách hàng |
| Tiền điều kiện | Khách hàng phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Khách hàng chức năng thêm sản phẩm vào giỏ hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Khách hàng chọn sản phẩm cho vào giỏ | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Khách hàng chọn sản phẩm | 2.1 Hệ thống tự động thêm thông tin sản phẩm vào giỏ hàng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Kiểm tra số lượng sản phẩm còn lại trong hệ thống. |

* **Kịch bản UC sửa số lượng phẩm:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiohang |
| Tên Actor | Khách hàng |
| Tiền điều kiện | Khách hàng phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Khách hàng chọn thêm số lượng sản phẩm vào giỏ hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Khách hàng chọn sản phẩm trong giỏ | 1.1 Hệ thống hiển thị thông tin các sản phẩm trong giỏ. |
| 1. Khách hàng chọn số lượng sản phẩm | 2.1 Hệ thống kiểm tra số lượng sản phẩm trong hệ thống.  2.2 Hệ thống cập nhật số sản phẩm trong giỏ hàng |
| **Ngoại lệ** | |
| Luồng phụ 1 | Kiểm tra số lượng sản phẩm còn lại trong hệ thống. |
| Luồng phụ 2 | Xóa sản phẩm trong giỏ hàng nếu sản phẩm hết. |

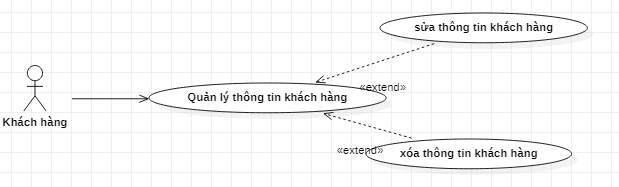
* **Kịch bản UC xóa sản phẩm trong giỏ hàng:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiohang |
| Tên Actor | Khách hàng |
| Tiền điều kiện | Khách hàng phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Khách hàng chọn xóa sản phẩm trong giỏ hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Khách hàng chọn sản phẩm trong giỏ | 1.1 Hệ thống hiển thị thông tin các sản phẩm trong giỏ. |
| 1. Khách hàng chọn sản phẩm |  |
| 1. Khách hàng chọn xóa | 3.1 Hệ thống xóa sản phẩm trong hệ thống.  3.2 Hệ thống cập nhật lại số sản phẩm trong giỏ hàng |

* **Kịch bản UC đặt hàng :**

|  |  |
| --- | --- |
| Tên use case | Quanlygiohang |
| Tên Actor | Khách hàng |
| Tiền điều kiện | Khách hàng phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Khách hàng chọn đặt sản phẩm. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Khách hàng chọn đặt hàng sản phẩm trong giỏ. | 1.1 Hệ thống hiển thị thông tin các sản phẩm. |
| 1. Khách hàng chọn xác nhận đặt hàng sản phẩm. | 2.1 Hệ thống kiểm tra số lượng sản phẩm trong hệ thống.  2.2 Hệ thống kiểm tra thông tin khách hàng.  2.3 Hệ thống gửi yêu cầu hóa đơn bán. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Không cho đặt sản phẩm trong giỏ hàng nếu số lượng sản phẩm không đủ. |
| Luồng phụ 2 | Cập nhật số sản phẩm trong hệ thống. |
| Luồng phụ 3 | Thêm hóa đơn bán. |

**e) Use-case phân rã quản lý thông tin khách hàng**



***Hình 3.6 Use-case phân rã quản lý thông tin khách hàng***

* **Kịch bản UC quản lý thông tin khách hàng:**

|  |  |
| --- | --- |
| Tên use case | Quanlythongtinkhachhang |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng quản lý thông tin khách hàng |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng | * 1. Hệ thống lấy thông tin của khách hàng.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin của các khách hàng. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

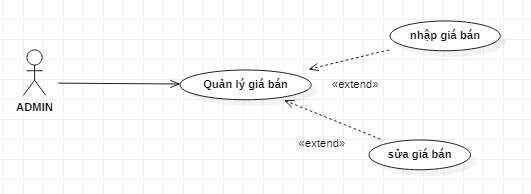
* **Kịch bản UC sửa thông tin khách hàng:**

|  |  |
| --- | --- |
| Tên use case | Quanlythongtinkhachhang |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chỉnh sửa thông tin khách hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn thông tin khách hàng | 1.1 Hệ thống hiển thị thông tin khách hàng. |
| 1. Nhân viên sửa thông tin khách hàng | 2.1 Hệ thống kiểm tra giữ liệu nhập |
| 1. Nhân viên chọn lưu thông tin | 3.1 Hệ thống cập nhật thông tin khách hàng |
| **Ngoại lệ** | |
| Luồng phụ 1 | Không cho phép sửa các thông tin cá nhân của khách |

* **Kịch bản UC xóa thông tin khách hàng:**

|  |  |
| --- | --- |
| Tên use case | Quanlythongtinkhachhang |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chức năng xóa thông tin khách hàng. |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên chọn chức năng xóa thông tin khách hàng | 1.1 Hệ thống hiển thị thông tin các khách hàng. |
| 1. Nhân viên chọn thông tin khách hàng cần xóa |  |
| 1. Nhân viên yêu cầu xóa thông tin khách hàng | * 1. Hệ thống kiểm tra hợp lệ thông tin nhập.   2. Hệ thống xóa thông tin khách hàng tương ứng. |
| **Ngoại lệ** | |
| Luồng phụ 1 | Không được xóa thông tin khách hàng còn hoạt động hoặc đang đặt đơn hàng. |

**f) UC phân rã quản lý giá bán:**



***Hình 3.7 UC phân rã quản lý giá bán***

* **Kịch bản UC quản lý giá bán:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiaban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng quản lý giá bán |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng | * 1. Hệ thống lấy thông tin giá bán của các sản phẩm.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin giá bán của các sản phẩm. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

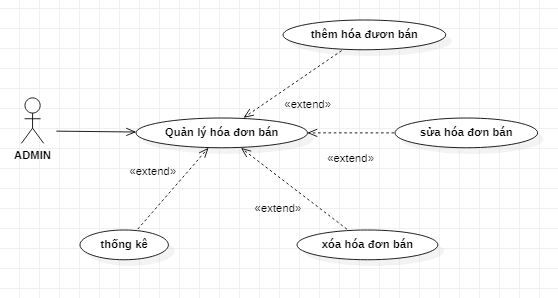
* **Kịch bản UC nhập giá bán:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiaban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng nhập giá |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng nhập giá | * 1. Hệ thống lấy thông tin giá bán của các sản phẩm.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin giá bán của các sản phẩm. |
| 1. Nhân viên chọn sản phẩm cần nhập giá |  |
| 1. Nhân viên chọn lưu | * 1. Hệ thống kiểm tra thông tin nhập.   2. Hệ thống cập nhật thông tin |

* **Kịch bản UC sửa giá bán:**

|  |  |
| --- | --- |
| Tên use case | Quanlygiaban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng sửa giá |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng sửa giá | * 1. Hệ thống lấy thông tin giá bán của các sản phẩm.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin giá bán của các sản phẩm. |
| 1. Nhân viên chọn sản phẩm cần thay đổi giá |  |
| 1. Nhân viên chọn lưu | * 1. Hệ thống kiểm tra thông tin nhập.   2. Hệ thống cập nhật thông tin |

**g) UC phân dã hóa đơn bán:**



***Hình 3.8 UC phân dã hóa đơn bán***

* **Kịch bản UC quản lý hóa đơn bán:**

|  |  |
| --- | --- |
| Tên use case | Quanlyhoadonban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng quản lý hóa đơn bán |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng | * 1. Hệ thống lấy thông tin hóa đơn bán.   2. Lấy về thông tin ngày giờ hệ thống.   3. Hiển thị các thông tin các hóa đơn bán. |
| 1. Thực hiện các chức năng theo yêu cầu |  |

* **Kịch bản UC thêm hóa đơn bán:**

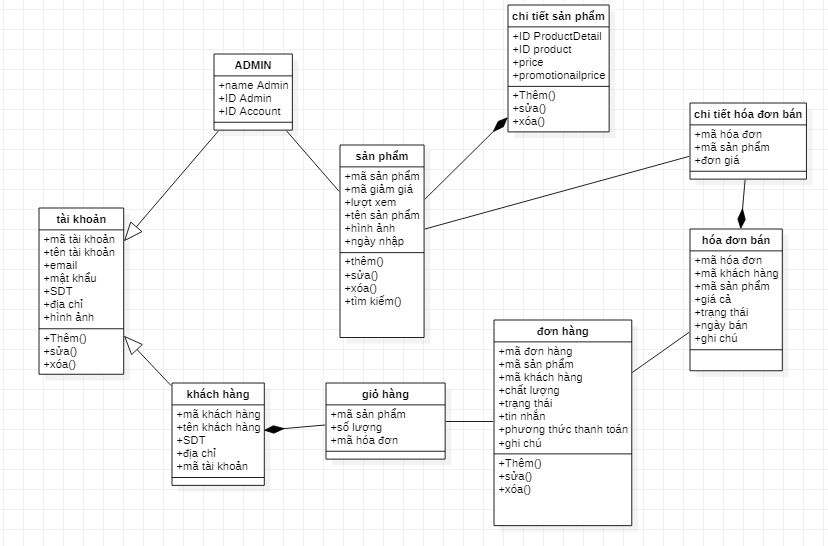
|  |  |
| --- | --- |
| Tên use case | Quanlyhoadonban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng hủy hóa đơn bán |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng hủy hóa đơn bán. | * 1. Lấy về thông tin ngày giờ hệ thống.   2. Hiển thị giao diện nhập hóa đơn bán. |
| 1. Nhân viên hủy thông tin hóa đơn bán |  |
| 1. Nhân viên xác nhận xóa | * 1. Hệ thống kiểm tra thông tin nhập.   2. Hệ thống xóa hóa đơn bán |
| **Ngoại lệ** | |
| Luồng 1 | Số lượng sản phẩm trong đơn lớn hơn số sản phẩm còn thì hủy đơn. |

* **Kịch bản UC thống kê hóa đơn bán:**

|  |  |
| --- | --- |
| Tên use case | Quanlyhoadonban |
| Tên Actor | Nhân viên |
| Tiền điều kiện | Nhân viên phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu |  |
| Đảm bảo thành công |  |
| Kích hoạt | Nhân viên chọn chức năng thống kê hóa đơn bán |
| **Hành động tác nhân** | **Phản ứng hệ thống** |
| 1. Nhân viên yêu cầu chức năng thống kê hóa đơn bán. | * 1. Hệ thống lấy thông tin các hóa đơn bán   2. Hệ thống tổng hợp dữ liệu   3. Hiển thị dữ liệu |
| **Ngoại lệ** | |
| Luồng 1 | Không có hóa đơn nào thì không thống kê |

+) Nguyên tắc: thống kê theo mã hàng hóa , thống kê theo thời gian hóa đơn bán,...

### Biểu đồ lớp thực thể



**Hình 3.9: Biểu đồ lớp thực thể**

### Các yêu cầu phi chức năng

***Bảng 3.2 các yêu cầu phi chức năng***

|  |  |  |
| --- | --- | --- |
| Mục | Tên yêu cầu | Mô tả yêu cầu |
| 1 | Giao diện dễ nhìn | Là giao diện cho người dùng phải dễ nhìn, bắt mắt và thân thiện. |
| 2 | Phân quyền chức năng | Là phân hóa các quyền cho các người dùng khác nhau. |
| 3 | Năng cấp và update | Là chính xác cải tiến, năng cấp trong 1 thời gian dài về sau. |
| 4 | Bảo hành và chịu trắc nghiệm khắc phục các nỗi | Là chịu trắc nghiệm giải thích cho khách hàng về các chức nâng, các thắc mắc và vấn đề khi vận hành sử dụng phần mềm. |

## 3.2 Thiết kế hệ thống

### 3.2.1 Thiết kế cơ sở dữ liệu

**A) Danh sách các thực thể của hệ thống**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thực thể** | **Mô tả** |
| **1** | ADMIN | Lớp này lưu trữ thông tin ADMIN |
| **2** | Khách hàng | Lớp này lưu trữ thông tin khách hàng |
| **3** | Loại sản phẩm | Lớp này lưu trữ thông tin của các loại mặt hàng |
| **4** | Sản phẩm | Lớp này lưu trữ thông tin của các sản phẩm |
| **5** | Hóa đơn bán | Lớp này lưu trữ thông tin hóa đơn bán |
| **6** | Chi tiết hóa đơn bán | Lớp này lưu trữ thông tin chi tiết hóa đơn bán |
| **7** | Giá bán | Lớp này lưu trữ thông tin giá bán của các sản phẩm |
| **8** | Hình ảnh | Lớp này lưu trữ thông tin hình ảnh của các sản phẩm |
| **9** | Giá bán | Lớp này lưu trữ thông tin giá bán của các sản phẩm |

**B) Chi tiết các thực thể trong hệ thống**

- ADMIN

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã admin | Nvarchar(50) | Not null | x |  |
| 2 | Tên admin | Nvarchar(50) | Not null |  |  |
| 3 | Mã tài khoản | Nvarchar(30) | Not null |  |  |

- Khách hàng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã khách hàng | Nvarchar(50) | Not null | x |  |
| 2 | Tên khách hàng | Nvarchar(50) | Not null |  |  |
| 3 | Giới tính | Nvarchar(30) | Not null |  |  |
| 4 | Năm sinh | Datetime | Null |  |  |
| 5 | Địa chỉ | Nvarchar(200) | Not null |  |  |
| 6 | Số điện thoại | Nvarchar(10) | Not null |  |  |

- Loại hàng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã loại hàng | Nvarchar(50) | Not null | x |  |
| 2 | Tên loại hàng | Nvarchar(50) | Not null |  |  |
| 3 | Giá tiền | Nvarchar(50) | Not null |  |  |
| 4 | Khuyến mãi | Nvarchar(50) | Not null |  |  |

- Sản phẩm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã sản phẩm | Nvarchar(50) | Not null | x |  |
| 2 | Tên sản phẩm | Nvarchar(50) | Not null |  |  |
| 3 | Đơn vị tính | Nvarchar(30) | Not null |  |  |
| 4 | Mô tả | Nvarchar | Null |  |  |
| 5 | Hình ảnh | Nvarchar(200) | Not null |  |  |
| 6 | Mã loại hàng | Nvarchar(50) | Not null |  | x |

- Hóa đơn bán

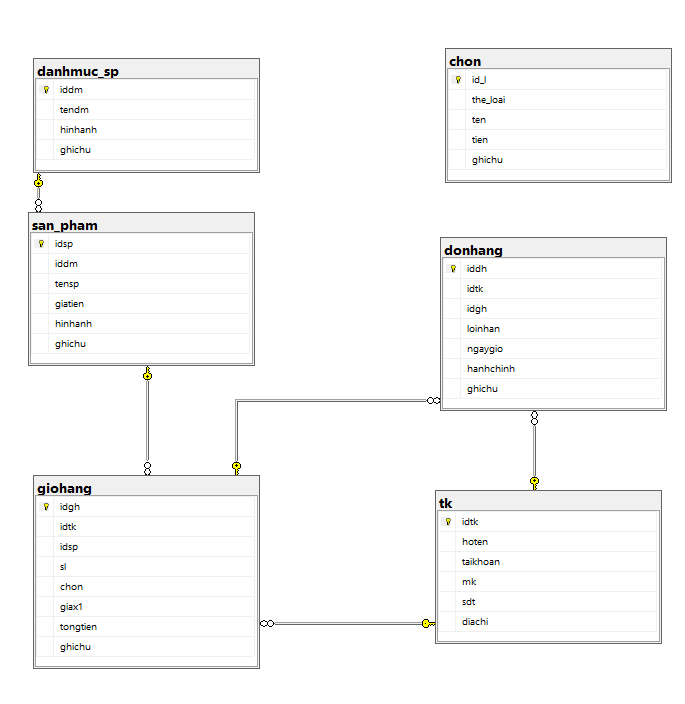
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã hóa đơn bán | Nvarchar(50) | Not null | x |  |
| 2 | Mã khách hàng | Nvarchar(50) | Not null |  | x |
| 3 | Tình trạng | Nvarchar(30) | Not null |  |  |
| 4 | Ghi chú | Nvarchar(40) | Null |  |  |
| 5 | Ngày đặt | Datetime | Not null |  |  |

- Chi tiết hóa đơn bán

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã hóa đơn bán | Nvarchar(50) | Not null | x | x |
| 2 | Mã sản phẩm | Nvarchar(50) | Not null |  | x |
| 3 | Đơn giá | Int | Not null |  |  |
| 4 | Số lượng sản phẩm | Int | Not null |  |  |
| 5 | Thành tiền | Int | Not null |  |  |

- Giá bán

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Not null/Null | Khóa chính | Khóa ngoại |
| 1 | Mã sản phẩm | Nvarchar(50) | Not null | x | x |
| 2 | Giá bán sản phẩm | Int | Not null |  |  |
| 3 | Ngày bắt đầu | Datetime | Null |  |  |
| 4 | Ngày kết thúc | Datetime | Null |  |  |

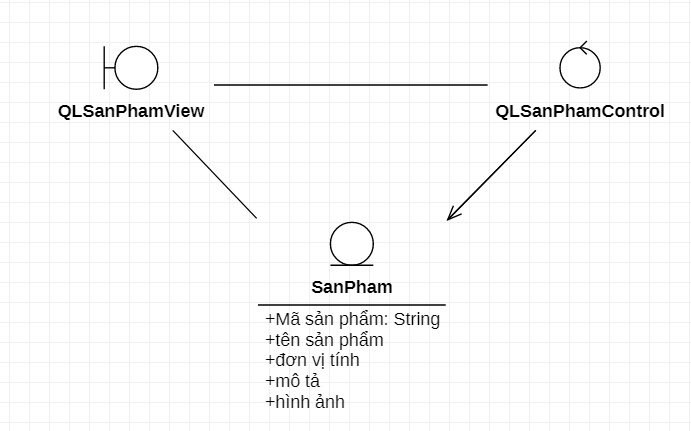


***Hình 3.10: Hình ảnh lược dồ cơ sở dữ liệu***

### 3.2.2 Thiết kế các lớp và các đối tượng

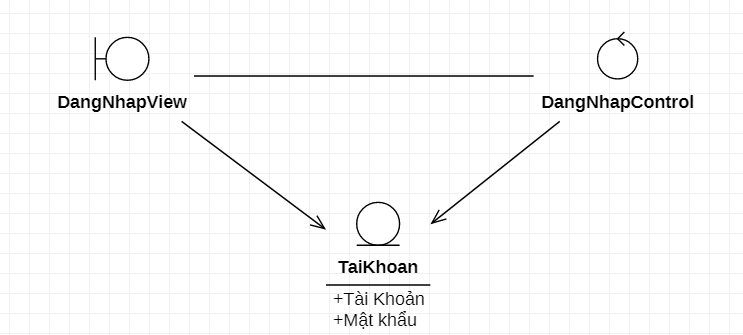
**A) Biểu đồ lớp VOPC của các ca sử dụng**

* Biểu đồ phân tích VOPC Quản lý sản phẩm:



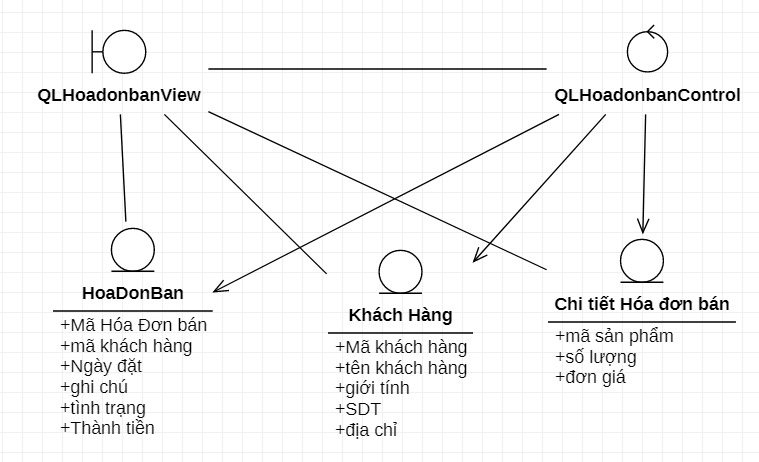
***Hình 3.11 Biểu đồ VOPC Quản lý sản phẩm***

* Biểu đồ phân tích VOPC quản lý đăng nhập:



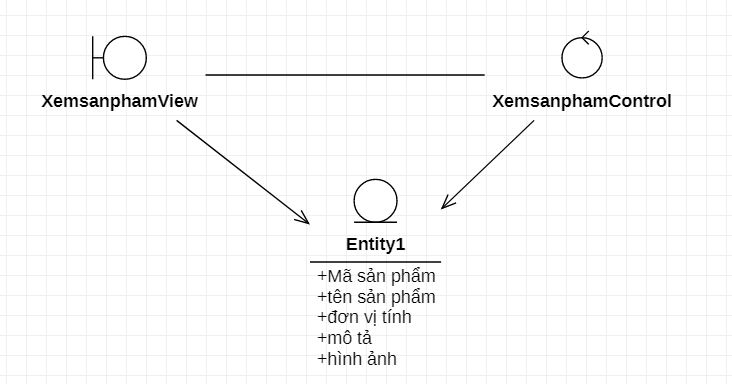
***Hình 3.12 Biểu đồ VOPC quản lý đăng nhập***

* Biểu đồ phân tích VOPC quản lý hóa đơn bán:



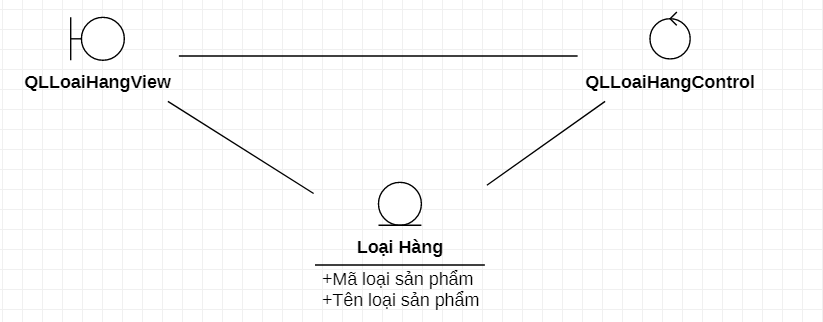
***Hình 3.13 Biểu đồ VOPC quản lý hóa đơn bán***

* Biểu đồ phân tích VOPC xem sản phẩm:



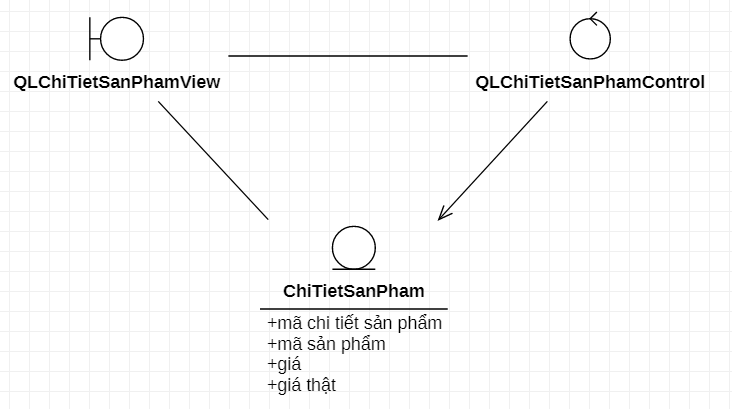
***Hình 3.14 Biểu đồ VOPC xem sản phẩm***

* Biểu đồ phân tích VOPC quản lý theo loại hàng:



***Hình 3.15 Biểu đồ VOPC quản lý theo loại hàng***

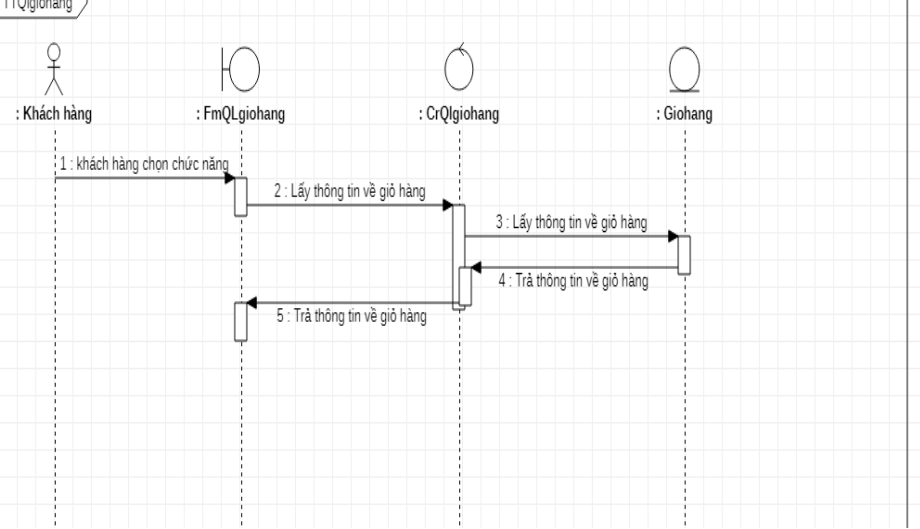
* Biểu đồ phân tích VOPC chi tiết sản phẩm:



***Hình 3.16 Biểu đồ VOPC chi tiết sản phẩm***

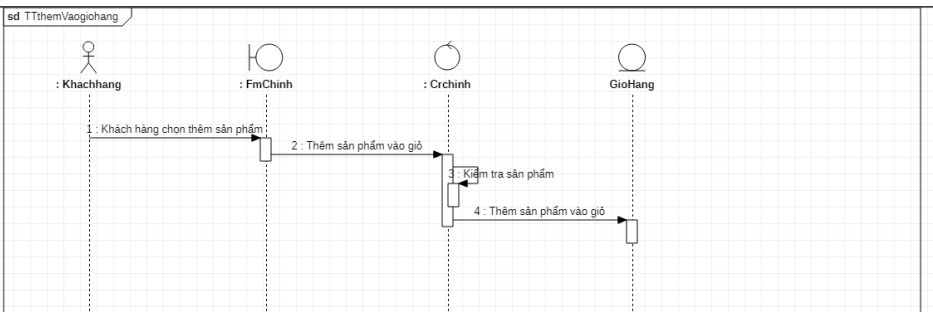
**B) Biểu đồ tuần tự**

* Quản lý giỏ hàng:



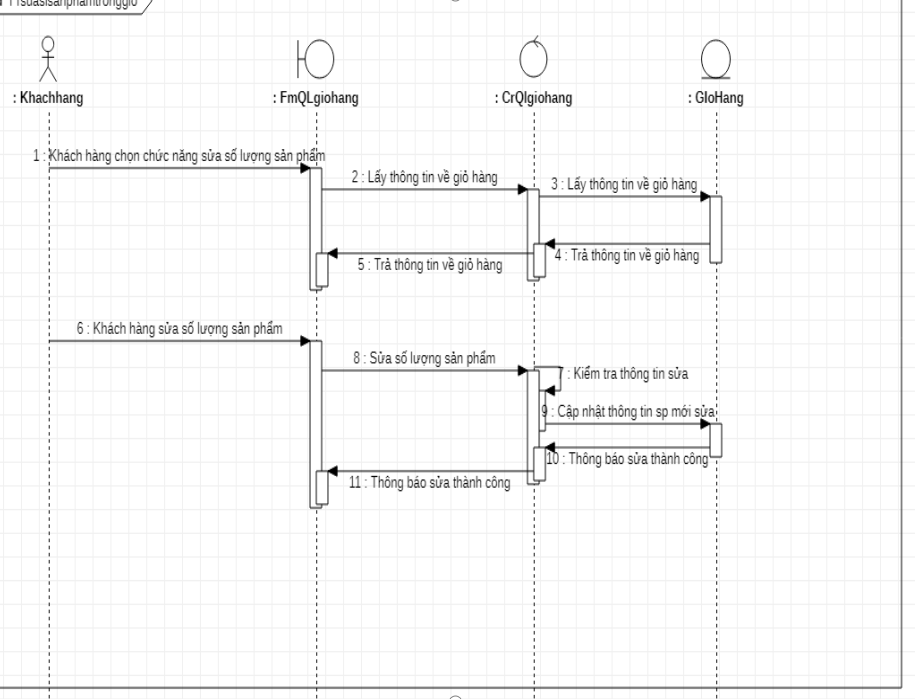
***Hình 3.17 biểu đồ tuần tự quản lý giỏ hàng***

* Thêm sản phẩm vào giỏ:



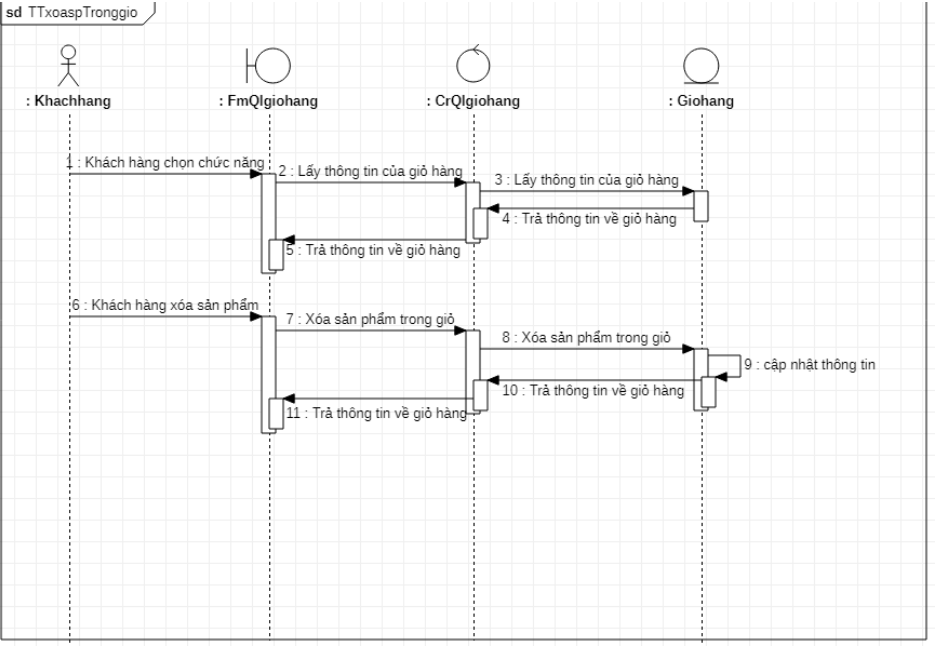
***Hình 3.18 biểu đồ tuần tự thêm sản phẩm vào giỏ***

* Sửa số lượng sản phẩm trong giỏ:



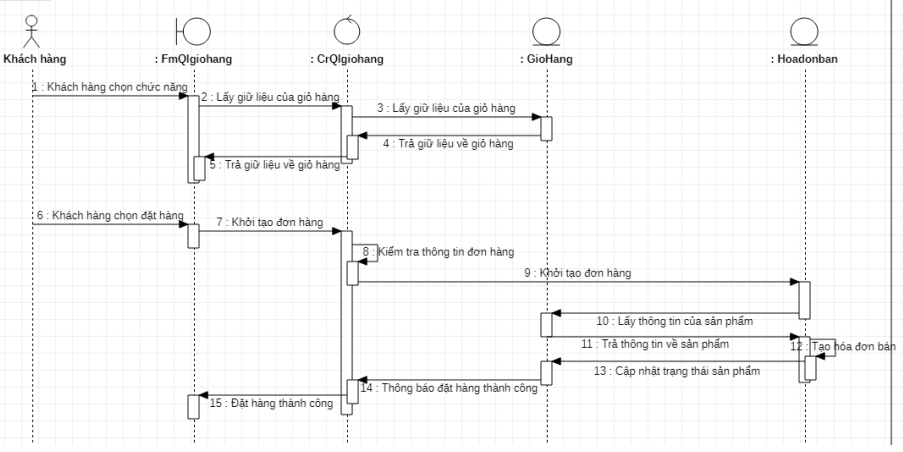
***Hình 3.19 biểu đồ tuần tự sửa số lượng sản phẩm trong giỏ***

* Xóa sản phẩm trong giỏ:



***Hình 3.20 biểu đồ tuần tự xóa sản phẩm trong giỏ***

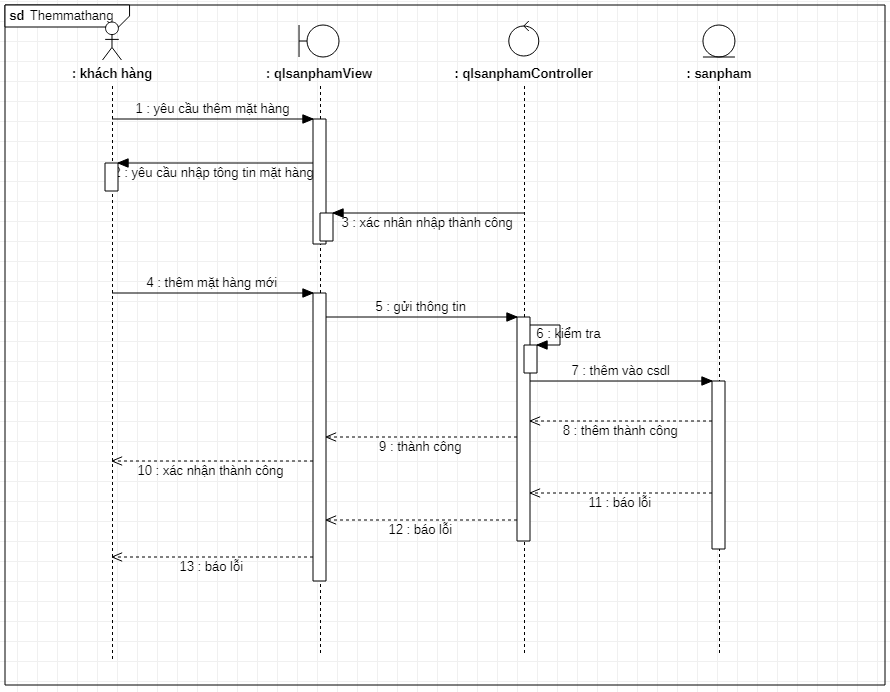
* Đặt hàng trong giỏ:



***Hình 3.21 biểu đồ tuần tự đặt hàng trong giỏ***

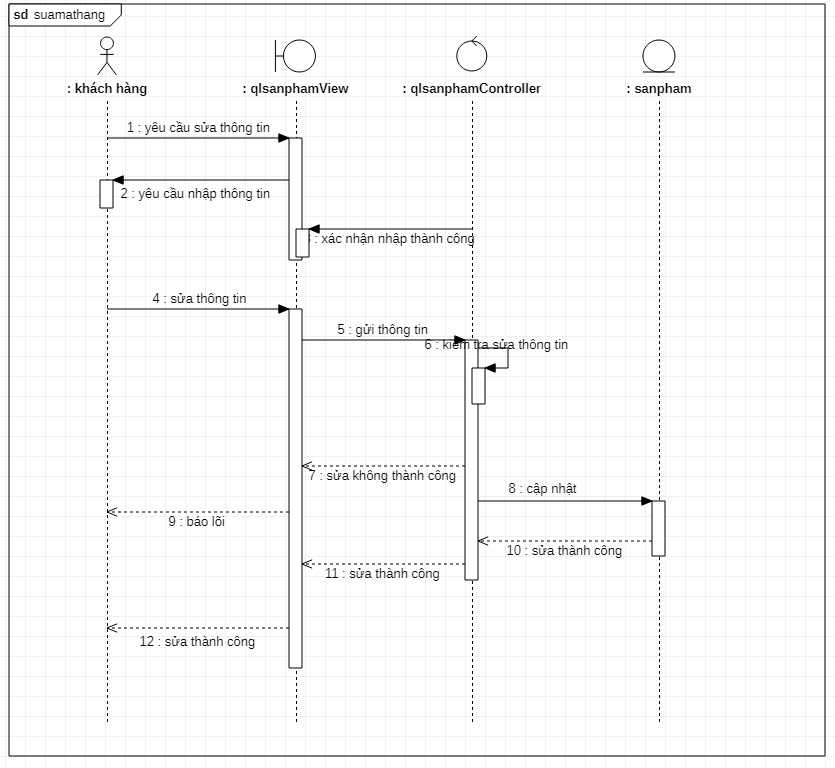
* Quản lý sản phẩm:

- Thêm sản phẩm:



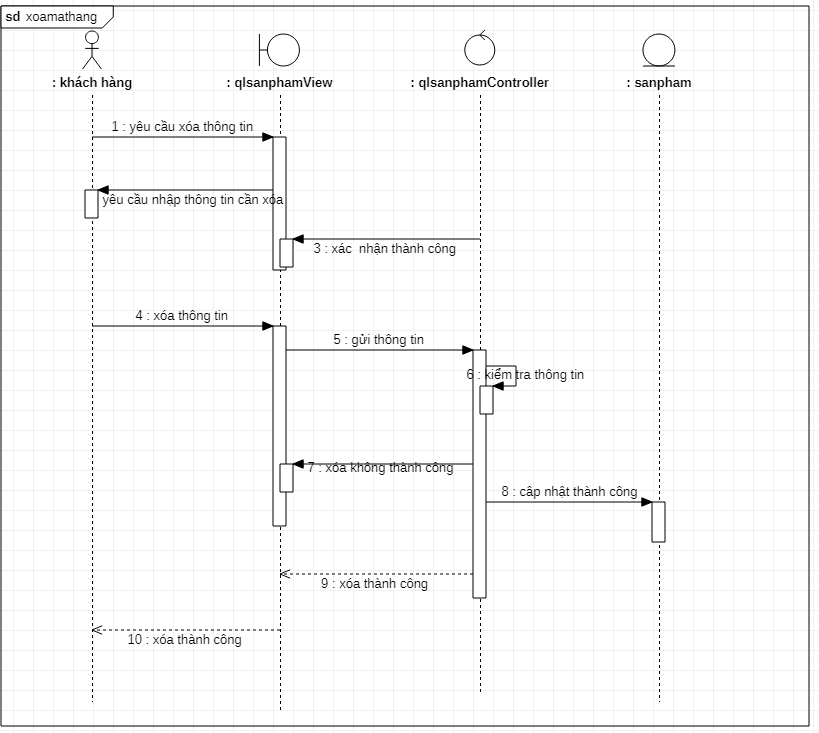
***Hình 3.22 biểu đồ tuần tự thêm sản phẩm***

* Sửa sản phẩm:



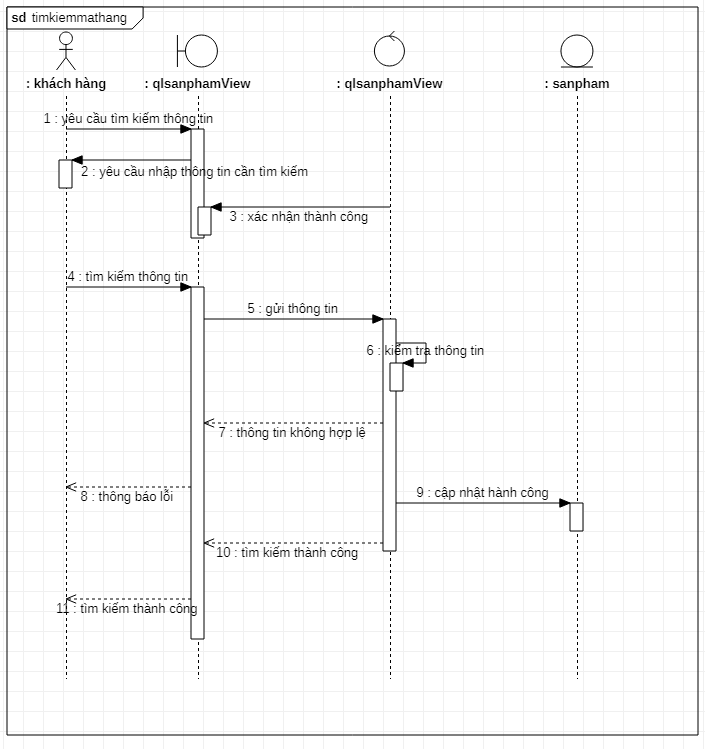
***Hình 3.23 biểu đồ tuần tự sửa sản phẩm***

- Xóa sản phẩm:



***Hình 3.24 biểu đồ tuần tự xóa sản phẩm***

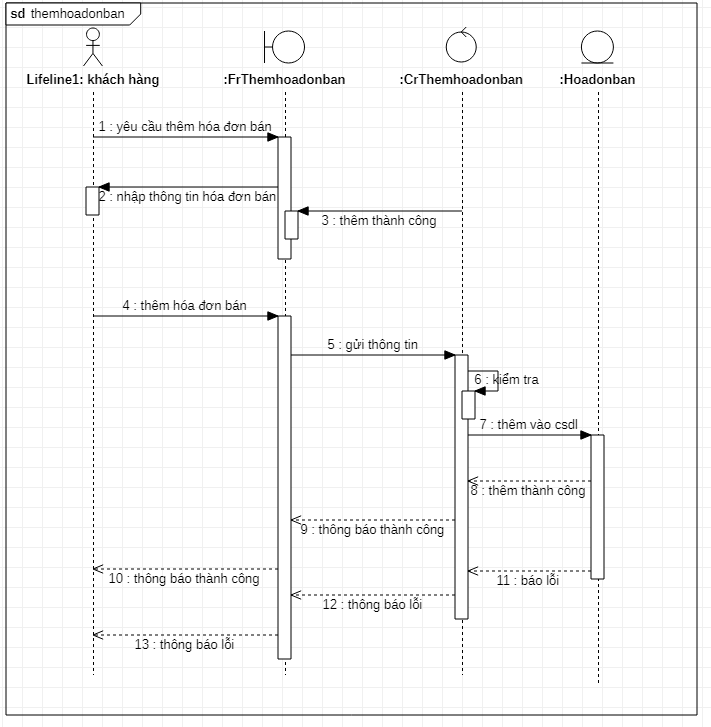
- Tìm kiếm sản phẩm:



***Hình 3.25 biểu đồ tuần tự tìm kiếm sản phẩm***

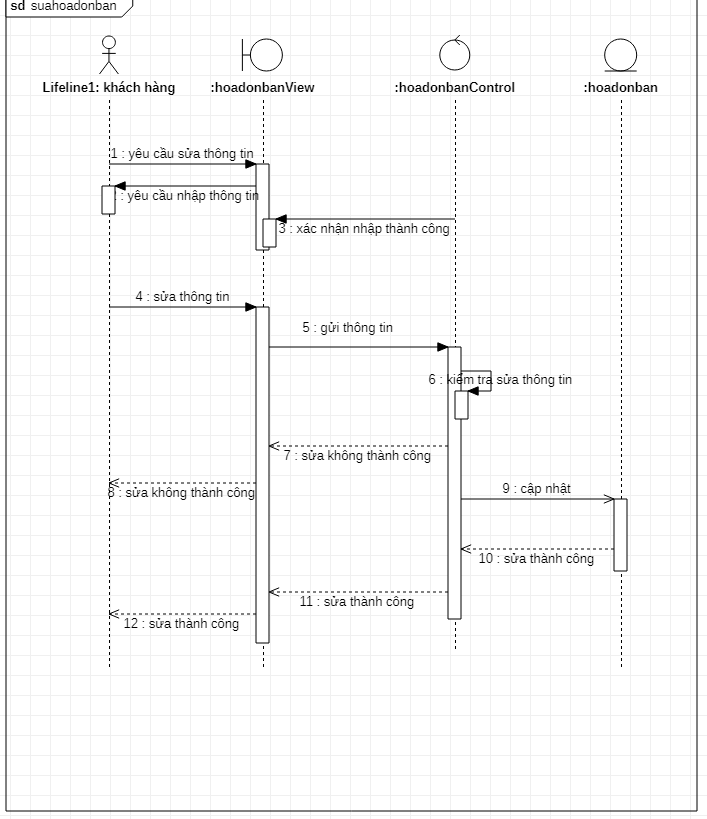
* Quản Lý hóa đơn bán:

- thêm hóa đơn bán:



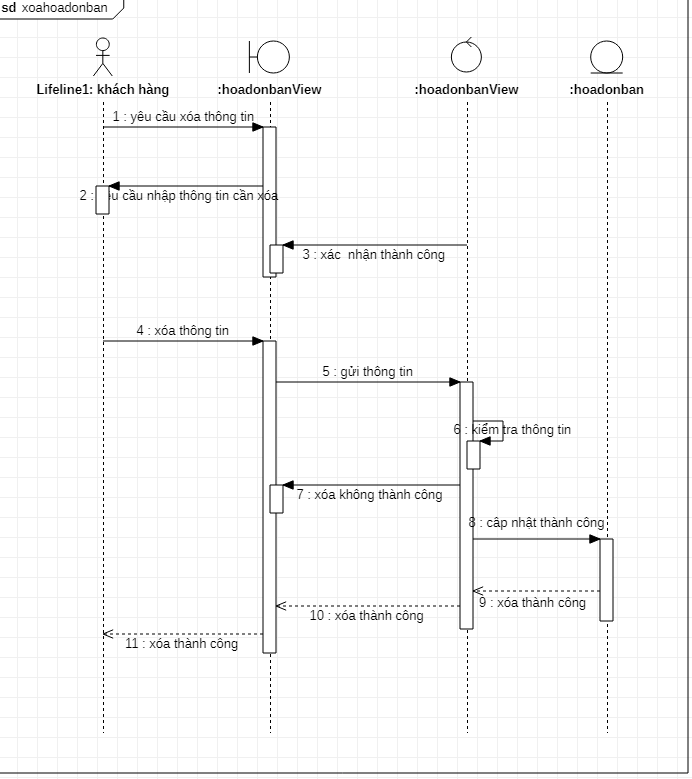
***Hình 3.26 biểu đồ tuần tự thêm hóa đơn bán***

- sửa hóa đơn bán:



***Hình 3.27 biểu đồ tuần tự sửa hóa đơn bán***

- xóa hóa đơn bán:



***Hình 3.28 biểu đồ tuần tự xóa hóa đơn bán***

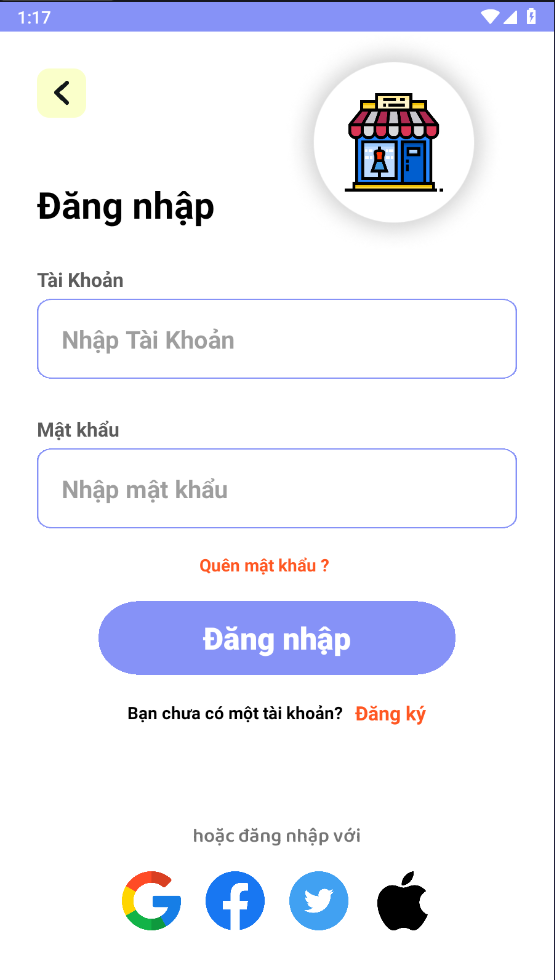
**3.2.3 Thiết kế giao diện ứng dụng**

- Hình ảnh welcome:



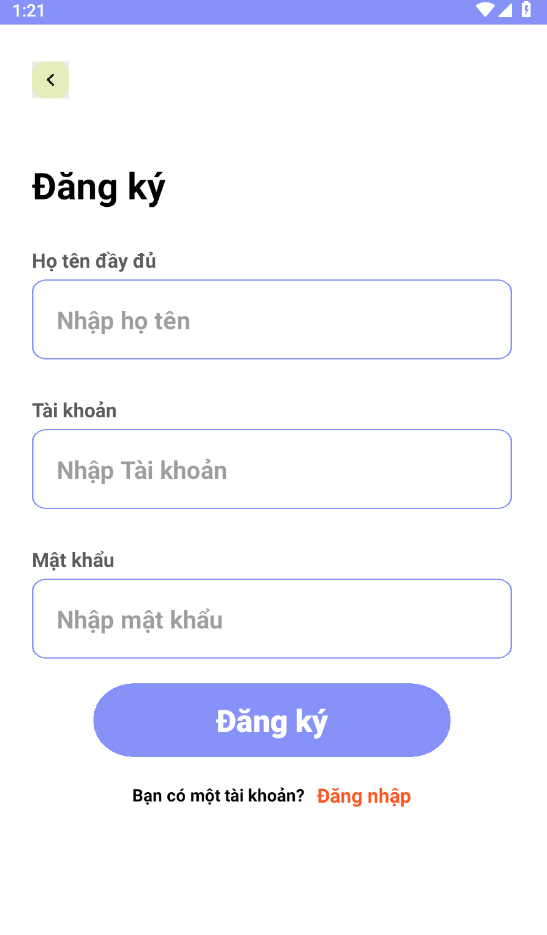
***Hình 3.29 giao diện welcome***

- Hình ảnh giao diện đăng nhập:



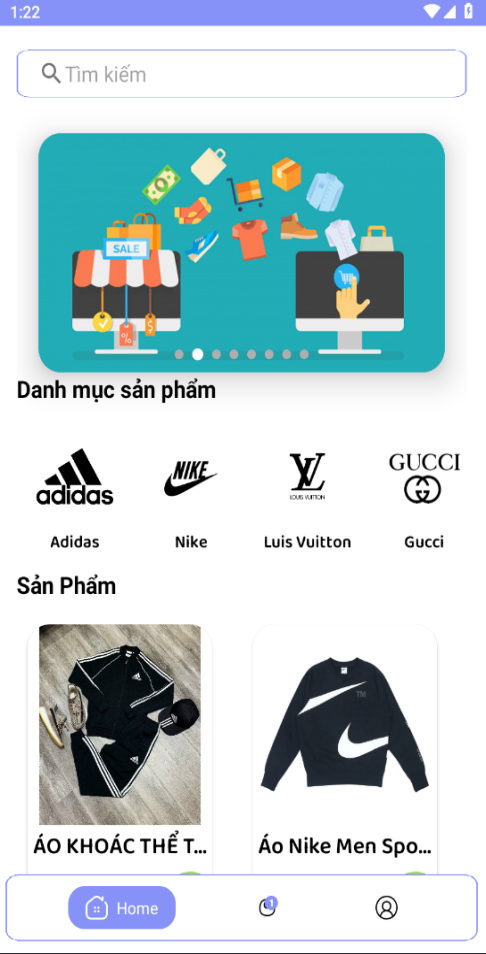
***Hình 3.30 giao diện đăng nhập***

- Hình ảnh giao diện đăng ký:



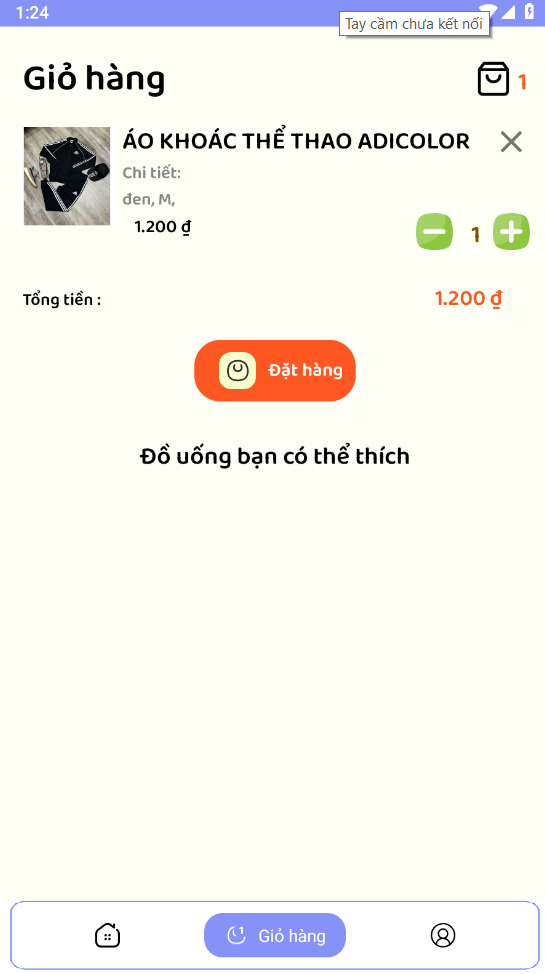
***Hình 3.31 giao diện đăng ký***

- Hình ảnh giao diện chính:



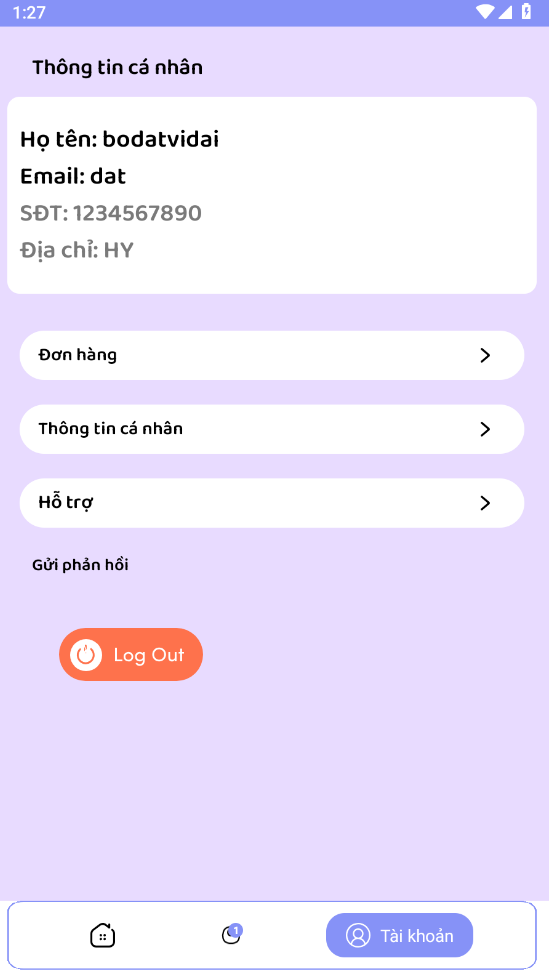
***Hình 3.32 giao diện chính***

- Hình ảnh giao diện giỏ hàng:



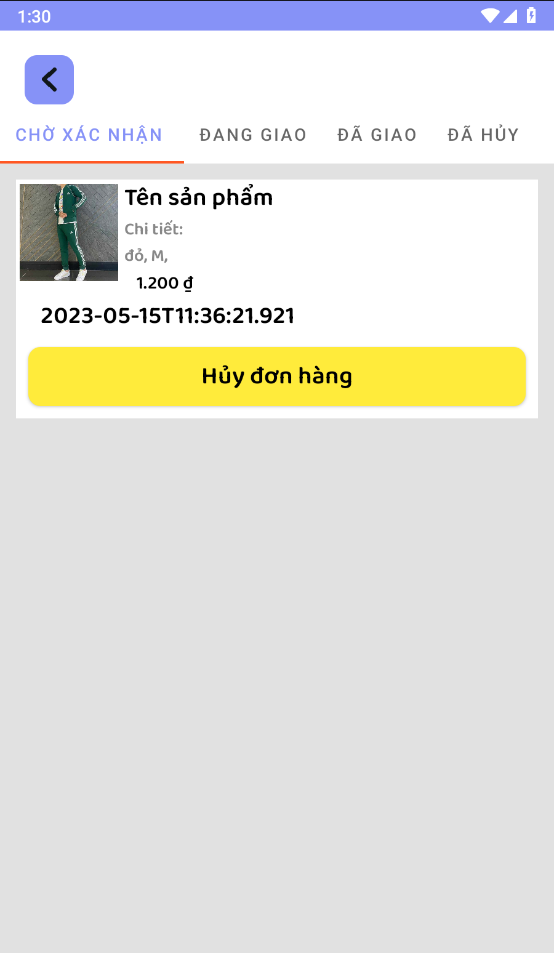
***Hình 3.33 giao diện giỏ hàng***

- Hình ảnh giao diện tài khoản:



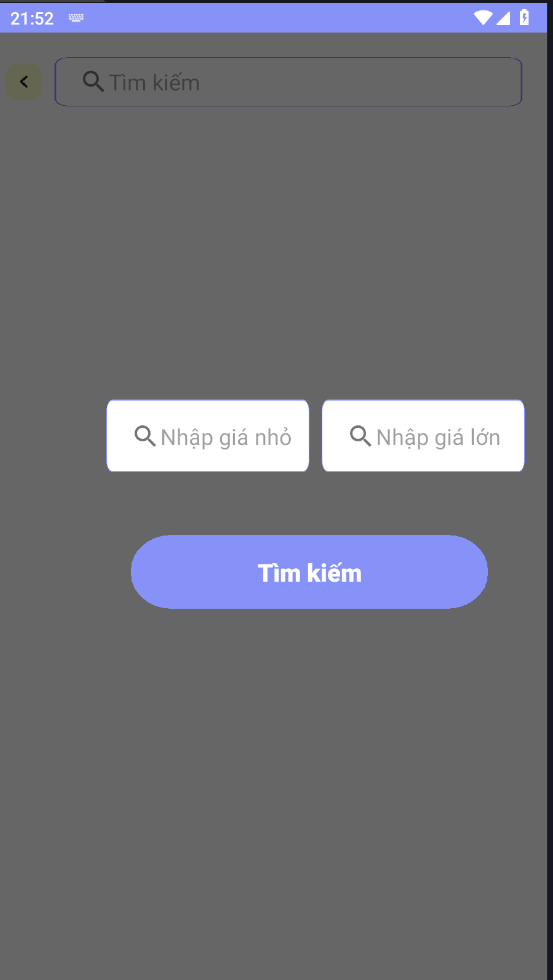
***Hình 3.34 giao diện tài khoản***

- Hình ảnh giao diện đơn hàng:



***Hình 3.34 giao diện đơn hàng***

- Hình ảnh giao diện tìm kiếm theo giá sản phẩm:



***Hình 3.35 giao diện tìm kiếm theo giá sản phẩm***

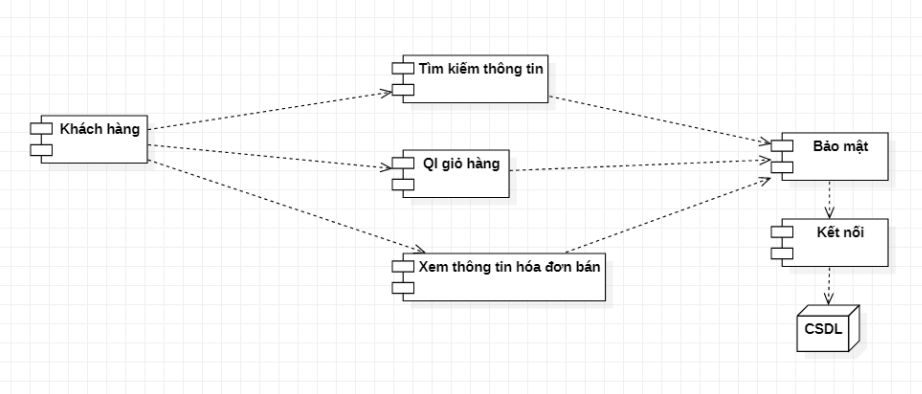
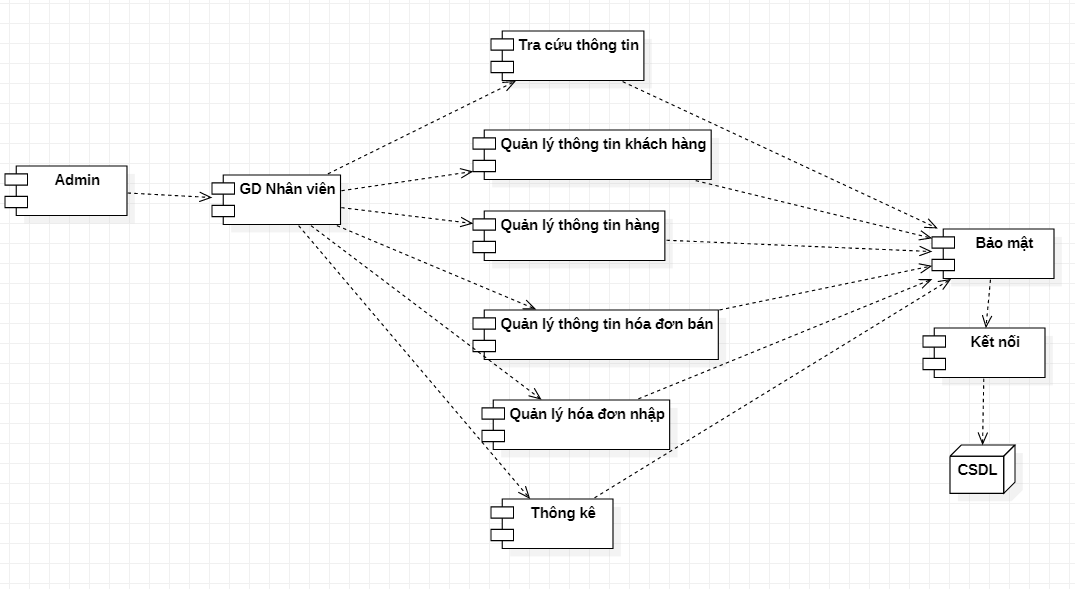
# CHƯƠNG 4: TRIỂN KHAI ỨNG DỤNG

## 4.1 Triển khai các chức năng của ứng dụng

### 4.1.1 Các chức năng nghiệp vụ

***Bảng 4.1 các chức năng nghiệp vụ***

|  |  |  |
| --- | --- | --- |
| Mục | Tên chức năng | Mô tả |
| 1 | Hiển thị thông tin | Là phần hiển thị từ dữ liệu lưu trữ ra đầy đủ các thông tin mà người dùng muốn xem.  - Bao gồm các hiển thị:  + Thông tin khách hàng  + Thông tin sản phẩm  + Thông tin chi tiết sản phẩm  + Thông tin hóa đơn bán  + Thông tin giỏ hàng |
| 2 | Nhập thông tin | Là đưa thông tin chính xác và đầy đủ của đối tượng vào hệ thống để quản lý được tốt nhất.  - Bao gồm nhập các thông tin về:  + Thông tin khách hàng  + Thông tin sản phẩm  + Thông tin hóa đơn bán  + Thông tin đơn hàng |
| 3 | Tìm kiếm thông tin | Là đưa ra thông tin với dữ liệu thông tin chính xác, cập nhật mới nhất. Yêu cầu chính xác và nhanh chóng.  -Bao gồm tìm kiếm các thông tin liên quan đến các đối tượng. Vd như: mã sản phẩm, thông tin hóa đơn bán,... |
| 4 | Xóa thông tin | Là loại bỏ các bản ghi thông tin đối tượng có trong cơ sở dữ liệu khi cần thiết.  - Bao gồm xóa các thông tin về:  + Thông tin sản phẩm  + Thông tin về giỏ hàng của khách hàng |
| 5 | Thống kê thông tin | Là đưa ra thông tin được tổng hợp và chiết xuất. Thể hiện đúng dữ liệu thông tin chiết xuất có trong cơ sở dữ liệu, cụ thể rõ ràng từng hợp đồng của người sử dụng.  - Thống kê bao gồm:  + Số lượng các sản phẩm  + Hóa đơn bán theo các mốc thời gian |
| 6 | Tính toán các khoản tự động dựa vào thông tin cập nhật | Là tính các khoản thu trong chương trình của khách hàng 1 cách chính xác nhanh chóng, hiển thị cụ thể mạch lạc, rõ dành và thực hiện 1 cách khoa học đúng chuyên môn.  - Tính các khoản thu gồm:  + Tính tiền hóa đơn  + Tính số lượng sản phẩm |

**

### 4.1.2 Các chức năng hệ thống

|  |  |  |
| --- | --- | --- |
| Mục | Tên chức năng | Mô tả |
| 1 | Đăng nhập | Kiểm tra thông tin tài khoản và mật khẩu dựa vào đó phân biệt người dùng. |
| 2 | Đăng ký | Cho phép người dùng đăng ký các tài khoản mới để đăng nhập vào hệ thống. |
| 3 | Quản lý người dùng | Phân quyền cho các người dùng và các đối tượng sử dụng. Với các đối tượng khác nhau có các quyền khác nhau.  - Chia thành 2 thành phần người dùng:  + Khách hàng  +Người quản lý app. |
| 4 | Quản lý cơ sở dữ liệu | Lưu trữ và quản lý các dữ liệu có trong hệ thống quản bán quần áo online. |

*Bảng 4.2 các chức năng của hệ thống*

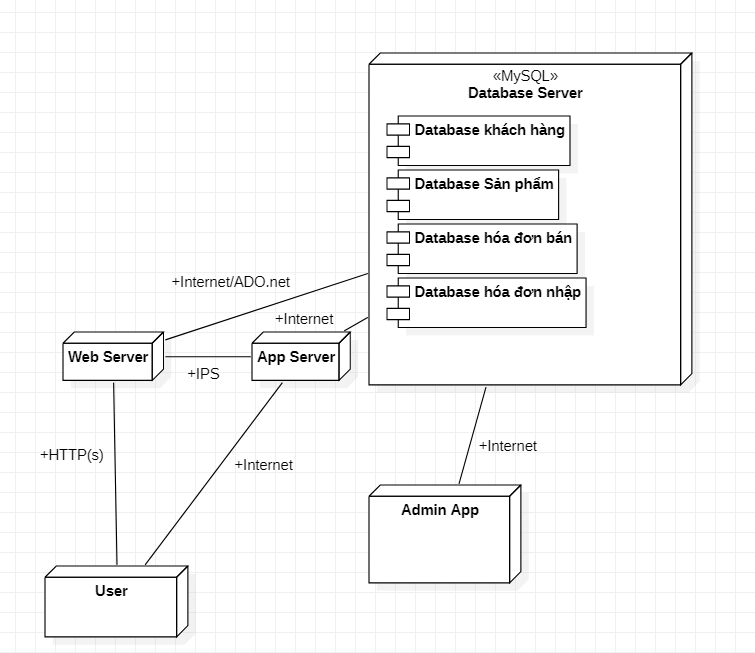
## 4.2 Kiểm thử và triển khai ứng dụng

* Phần cứng:

- Các điện thoại có thông số tối thiểu: 1GB bộ nhớ, 2gb ram, CPU trên 1 GHZ.

* Phần mềm:

- Các điện thoại di động chạy hệ điều hành Android.

Biểu đồ triển khai hệ thống

* Kiểm thử tính khả dụng:

+ Giao diện hợp lý, dễ sử dụng

+ Ổn định

* Kiểm thử chức năng:

+ Hệ thống hiển thị chức năng người dùng chọn.

* Chức năng tìm kiếm thông tin:

+ Tìm kiếm thông tin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Thông tin đã tồn tại** | **Thông tin chưa tồn tại** | **Thông tin rỗng** | **Output** |
| Nhập thông tin tìm kiếm | Yes |  |  | Hiển thị thông tin vừa tìm. |
| Nhập thông tin tìm kiếm |  | Yes |  | Không tìm  thấy thông tin muốn tìm. |
| Nhập thông tin tìm kiếm |  |  | Yes | Mời nhập lại. |

* Chức năng thêm:

+ Thêm thông tin dữ liệu mới

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Thông tin đã tồn tại** | **Thông tin chưa tồn tại** | **Thông tin rỗng** | **Output** |
| Nhập dữ liệu mới | Yes |  |  | Dữ liệu đã tồn tại! mời nhập lại. |
| Nhập dữ liệu mới |  | Yes |  | Thêm dữ liệu mới vào danh sách thông tin. |
| Nhập dữ liệu mới |  |  | Yes | Mời nhập lại, thông tin không được  để trống. |

* Chức năng xóa:

+ Xóa thông tin dữ liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Thông tin đã tồn tại** | **Thông tin chưa tồn tại** | **Output** |
| Nhập thông tin cần xóa | Yes |  | Xóa thành công thông tin. |
| Nhập thông tin cần xóa |  | Yes | Không tìm thấy thông tin muốn xóa. |

* Chức năng sửa

+ Update thông tin

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Thông tin đã tồn tại** | **Thông tin chưa tồn tại** | **Output** |
| Nhập thông tin muốn sửa | Yes |  | Cập nhật lại  thông tin vừa sửa |
| Nhập thông tin muốn sửa |  | Yes | Không tìm thấy thông tin muốn sửa. |

# KẾT LUẬN

Sau thời gian nghiên cứu và thực hiện đề tài, các yêu cầu chính của đề tài cơ bản đã hoàn tất với những nội dung chủ yếu sau:

* **Ưu điểm:**

Chương trình có giao diện thân thiện, thuận tiện cho người dùng và đã hoàn thành được các chức năng đã đề ra.

+) Chức năng cập nhật (thêm, sửa, xóa, tìm kiếm theo giá và theo tên) được thực hiện nhanh chóng.

+) Dễ dàng sử dụng thanh toán, quản lý được thông tin đơn hàng nhanh chóng

+) Đề tài đã xây dựng được các chức năng cần thiết cho hệ thống bán hàng online như: tìm hàng, đặt hàng, thanh toán.

+) Truy xuất dữ liệu nhanh chóng.

+) Người dùng (người quản lý) dễ dàng tiếp nhận thông tin phản hồi của khách hàng.

* **Nhược điểm:**

Mặc dù đã rất cố gắng nhưng do thời gian, kinh nghiệm còn hạn chế nên kết quả vẫn còn một số thiếu sót cần tiếp tục được hoàn thiện để hệ thống có thể ứng dụng trong thực tiễn đạt hiệu quả tốt hơn như:

+) Khả năng xử lý nghiệp vụ chưa linh hoạt nhạy bén.

+) Chưa giải quyết trọn vẹn các vấn đề nếu phát sinh trong quá trình quản lý thực tế.

+) Còn nhiều ứng dụng chưa được nghiên cứu.

+) Một số chức năng của hệ thống chưa hoàn thiện.

+) Chưa thể phân quyền quản lý giữa tài khoản người dùng và tài khoản ADMIN.

+) Chưa có số liệu thống kê sản phẩm đã bán và doanh thu theo thời gian.

+) Chương trình còn dài và chưa được tối ưu.

* **Hướng khắc phục**

Nghiên cứu sâu và nhiều hơn các ứng dụng quản lý và ngôn ngữ Android để hoàn thiện chương trình với giao diện đẹp, chương trình tối ưu hơn. Bổ xung các vấn đề phát sinh khi đi vào thực tiễn.

* **Hướng phát triển:**
* Bổ xung thêm các lựa chọn thanh toán cho người sử dụng.
* Thêm các chức năng thống kê cho người quản lý của ứng dụng.
* Theo dõi được hành trình của đơn hàng, chi tiết đơn hàng rõ ràng.

# TÀI LIỆU THAM KHẢO

1. Tên tác giả (năm XB), Tên sách, NXB, Nơi XB
2. *Giáo trình môn Chuyên đề 1: lập trình Android cơ bản,* Bộ môn CNPM, Trường ĐHSPKTHY.
3. *Giáo trình môn cơ sở kỹ thuật lập trình*, Bộ môn CNPM, Trường ĐHSPKTHY.
4. *Giáo trình môn cấu trúc dữ liệu giải thuật,* Bộ môn CNPM, Trường ĐHSPKTHY.
5. *Giáo trình môn lập trình hướng đối tượng với Java,* Bộ môn CNPM, Trường ĐHSPKTHY.
6. *Giáo trình môn Cơ sở dữ liệu*, Bộ môn CNPM, Trường ĐHSPKTHY.
7. Ed Burnette, *hello, Android Introducing Google’s Mobile Development Platform*, The Pragmatic Bookshelf, 2010.
8. Marziah Karch, *Android for Work Productivity for Professional*, Press, 2010.
9. J.F. DiMarzio, *Android A Programmer’s Guide*, The McGraw-Hill Companies, 2008.
10. Chris Haseman, *Android Essentials*, Apress, 2008.
11. Sams, *Android Application Development*, Lauren Darcey and Shane Conder, 2010.
12. stackoverflow.com, <https://stackoverflow.com>