TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG

KHOA CÔNG NGHỆ THÔNG TIN

-----🙚🙘 🕮 🙚🙘-----



**TIỂU LUẬN**

**HỌC PHẦN LẬP TRÌNH DI ĐỘNG**

**Đề tài:**

**QUẢN LÝ BÁN MỸ PHẨM SỬ DỤNG CƠ SỞ DỮ LIỆU SQLITE**

|  |  |
| --- | --- |
| Nhóm Sinh viên thực hiện | :Nguyễn Văn Duy - 2151220178 |
|  | Võ Đào Nguyên Giáp - 2151220196 |
|  | Trần Minh Nghĩa - 2151220193 |
|  | Dư Thị Ánh Tuyết - 2151220197 |
|  | Kiều Phước Vinh - 2151220204 |
| Lớp | : 21CT4 |
| Giáo viên hướng dẫn | : ThS.Nguyễn Văn Khương |

Đà Nẵng – 2024

# PHÂN CÔNG NHIỆM VỤ THÀNH VIÊN TRONG NHÓM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Họ và tên sinh viên** | **Lớp** | **Công việc hoàn thành** | **Điện thoại** | **Trưởng nhóm** |
| 1 | Nguyễn Văn Duy | 21CT4 |  | 0779743659 |  |
| 2 | Võ Đào Nguyên Giáp | 21CT4 | -Làm chức năng đăng nhập  - Vẽ sơ đồ DFD | 0387885366 |  |
| 3 | Trần Minh Nghĩa | 21CT4 | -Làm báo cáo word -Vẽ sơ đồ BFD, ERD | 0942586010 |  |
| 4 | Dư Thị Ánh Tuyết | 21CT4 | +)User :Giao diện User, Mua hàng  +)Admin :Giao diện admin, Quản lý sản phẩm (thêm , sửa , xóa, hiển thị), Quản lý khách hàng (thêm , sửa, xóa, hiển thị), Giỏ hàng  - Làm báo cáo word | 0362631552 | X |
| 5 | Kiều Phước Vinh | 21CT4 | - Làm chức năng đăng ký  - Vẽ sơ đồ IFD , UseCase | 0961964144 |  |

**LỜI CẢM ƠN**

Trước hết, tôi xin gửi lời cảm ơn chân thành và sâu sắc đến thầy giáo ThS. Nguyễn Văn Khương, người đã luôn dành sự quan tâm, hỗ trợ và hướng dẫn tận tình trong suốt quá trình tôi thực hiện tiểu luận này. Chính sự giúp đỡ của thầy đã giúp tôi hoàn thành đề tài "Ứng dụng Android bán mỹ phẩm" một cách tốt nhất. Tôi rất trân trọng và biết ơn những chỉ bảo quý báu từ thầy.

Tôi xin chân thành cảm ơn Ban giám hiệu và quý thầy cô giáo khoa Công nghệ thông tin, Trường Đại học Kiến trúc Đà Nẵng, những người đã truyền đạt cho tôi những kiến thức quan trọng trong suốt thời gian học tập. Những bài học này không chỉ hỗ trợ tôi trong quá trình nghiên cứu mà còn là nền tảng vững chắc cho sự nghiệp tương lai.

Tôi cũng xin gửi lời cảm ơn đến gia đình và bạn bè đã luôn bên cạnh, ủng hộ và động viên tôi trong suốt quá trình học tập và nghiên cứu, giúp tôi vượt qua mọi khó khăn và thử thách.

Cuối cùng, tôi xin chân thành cảm ơn tất cả những ai đã góp phần vào sự thành công của tiểu luận này. Trong quá trình thực hiện, tôi cũng không tránh khỏi những thiếu sót và sai sót. Vì vậy, tôi rất mong nhận được những ý kiến đóng góp để có thể cải thiện và hoàn thiện bản thân hơn nữa.

Xin chân thành cảm ơn!

Đà Nẵng, ngày 16 tháng 12 năm 2024

Nguyễn Văn Duy

Võ Đào Nguyên Giáp

Trần Minh Nghĩa

Dư Thị Ánh Tuyết

Kiều Phước Vinh

# PHẦN 1: MỞ ĐẦU

## 1. LÝ DO CHỌN ĐỀ TÀI

Trong bối cảnh công nghệ thông tin ngày càng phát triển, thương mại điện tử đã trở thành xu hướng tất yếu, thay đổi cách thức mua sắm và tiêu dùng của con người. Ngành mỹ phẩm, với sự gia tăng không ngừng của nhu cầu làm đẹp, đã chứng kiến sự bùng nổ trong thị trường trực tuyến. Tuy nhiên, nhiều khách hàng vẫn gặp khó khăn trong việc tìm kiếm sản phẩm chất lượng, đáng tin cậy và thông tin rõ ràng.

Việc xây dựng một ứng dụng bán mỹ phẩm chuyên biệt trên nền tảng Android không chỉ đáp ứng nhu cầu mua sắm của người tiêu dùng mà còn tạo cơ hội để các nhà cung cấp dễ dàng tiếp cận khách hàng hơn. Ứng dụng này sẽ là cầu nối, cung cấp thông tin chi tiết, đánh giá khách quan từ người dùng và hỗ trợ mua sắm an toàn, tiện lợi, giúp nâng cao trải nghiệm khách hàng và hiệu quả kinh doanh.

## 2. MỤC TIÊU CỦA ĐỀ TÀI

Mục tiêu của đề tài là phát triển một ứng dụng Android chuyên dụng cho việc bán mỹ phẩm với các đặc điểm:

* Tối ưu trải nghiệm người dùng: Giao diện trực quan, thân thiện và dễ sử dụng, giúp người dùng dễ dàng tìm kiếm, so sánh và mua sản phẩm.
* Cung cấp thông tin chi tiết: Hiển thị thông tin sản phẩm rõ ràng, có hình ảnh minh họa, thành phần, giá cả, và đánh giá từ khách hàng.
* Hỗ trợ quá trình mua sắm: Bao gồm tính năng quản lý giỏ hàng, thanh toán an toàn và quản lý đơn hàng.
* Hỗ trợ nhà cung cấp: Cho phép đăng tải sản phẩm, quản lý tồn kho và xử lý đơn hàng dễ dàng.  
  Ứng dụng không chỉ hướng tới việc phục vụ khách hàng cá nhân mà còn góp phần nâng cao năng lực cạnh tranh cho các nhà bán lẻ mỹ phẩm trên nền tảng thương mại điện tử.

## ĐỐI TƯỢNG PHẠM VI NGHIÊN CỨU

 Đối tượng nghiên cứu:

* Nghiên cứu các công nghệ và công cụ phát triển ứng dụng Android như Java, Android Studio, Firebase và SQLite.
* Tìm hiểu về hành vi người dùng trong mua sắm trực tuyến, đặc biệt trong lĩnh vực mỹ phẩm.
* Phân tích các mô hình thương mại điện tử thành công để áp dụng vào thiết kế và phát triển ứng dụng.

 Phạm vi nghiên cứu:

* Phát triển ứng dụng bán mỹ phẩm trên nền tảng Android với các tính năng chính: đăng ký/đăng nhập tài khoản, tìm kiếm sản phẩm, quản lý giỏ hàng và thanh toán.
* Tập trung vào thị trường Việt Nam, hướng tới nhóm khách hàng tiềm năng và phù hợp với nhu cầu sử dụng tại địa phương.
* Đảm bảo ứng dụng hoạt động ổn định trên các thiết bị Android phổ biến, với hiệu năng và tính bảo mật cao.

## PHƯƠNG PHÁP NGHIÊN CỨU

Để hoàn thành đề tài "Ứng dụng bán mỹ phẩm trên nền tảng Android", các phương pháp nghiên cứu chính được áp dụng bao gồm:

### 4.1. Nghiên cứu tài liệu

* Thu thập và tìm hiểu các tài liệu liên quan đến phát triển ứng dụng Android, bao gồm ngôn ngữ lập trình Java, nền tảng Android Studio và các công nghệ hỗ trợ khác như Firebase, SQLite.
* Nghiên cứu về các nguyên tắc thiết kế giao diện người dùng (UI/UX) để tạo ra một ứng dụng thân thiện, dễ sử dụng và tối ưu trải nghiệm người dùng.
* Tìm hiểu về các mô hình và giải pháp thương mại điện tử hiện có để áp dụng vào việc xây dựng chức năng của ứng dụng.

### 4.2. Phân tích yêu cầu

* Thu thập và phân tích các yêu cầu của khách hàng, bao gồm tính năng quản lý tài khoản, quản lý sản phẩm, giỏ hàng, thanh toán, và quản lý kho hàng.
* Định nghĩa phạm vi và chức năng của ứng dụng, xác định các tính năng cần thiết để đáp ứng nhu cầu người dùng mục tiêu.

### 4.3. Thiết kế hệ thống

* Sử dụng phương pháp phân tích và thiết kế hệ thống để xây dựng kiến trúc ứng dụng, bao gồm:
  + Thiết kế giao diện người dùng (UI).
  + Thiết kế cơ sở dữ liệu để lưu trữ thông tin sản phẩm, tài khoản, đơn hàng.
  + Thiết kế luồng dữ liệu và các chức năng chính của ứng dụng.

### 4.4. Phát triển ứng dụng

* Xây dựng ứng dụng trên nền tảng Android bằng cách sử dụng Java và Android Studio.
* Tích hợp các thư viện và công cụ như Retrofit cho kết nối API, Firebase cho xác thực và lưu trữ dữ liệu thời gian thực, và các thư viện UI để nâng cao hiệu quả thiết kế giao diện.
* Kiểm tra và tối ưu hiệu suất ứng dụng để đảm bảo hoạt động mượt mà trên các thiết bị Android.

### 4.5. Kiểm thử

* Thực hiện kiểm thử phần mềm để đảm bảo chất lượng ứng dụng, bao gồm:
  + Kiểm thử chức năng: Đảm bảo các tính năng chính của ứng dụng hoạt động đúng như yêu cầu.
  + Kiểm thử hiệu suất: Kiểm tra ứng dụng hoạt động ổn định với nhiều loại dữ liệu khác nhau.
  + Kiểm thử giao diện: Đảm bảo giao diện hiển thị chính xác trên các thiết bị Android có độ phân giải khác nhau.

### 4.6. Đánh giá và hoàn thiện

* Thu thập phản hồi từ người dùng thử nghiệm để đánh giá tính hiệu quả và trải nghiệm của ứng dụng.
* Hoàn thiện và sửa lỗi (nếu có) để đảm bảo ứng dụng đạt chất lượng tốt nhất trước khi triển khai thực tế.

## 5. CẤU TRÚC ĐỀ TÀI

Cấu trúc của bài khóa luận này bao gồm các phần: Lời cảm ơn, Mục lục, Phần mở đầu, Phần nội dung nghiên cứu, Phần kết luận và Phần tài liệu tham khảo. Trong đó phần Nội dung nghiên cứu sẽ gồm các nội dung chính như sau:

 **Chương 1: Tổng quan về cơ sở lý thuyết**  
Trình bày các kiến thức nền tảng về lập trình ứng dụng Android, cơ sở dữ liệu SQLite, và các công nghệ liên quan. Tập trung vào các khái niệm cần thiết để phát triển ứng dụng bán hàng trực tuyến.

 **Chương 2: Phân tích và thiết kế hệ thống**

* Mô tả bài toán: Xây dựng ứng dụng bán mỹ phẩm, bao gồm quản lý danh mục sản phẩm, giỏ hàng, đặt hàng, thanh toán và quản lý thông tin khách hàng.
* Phân tích yêu cầu: Chức năng chính, yêu cầu phi chức năng, đối tượng sử dụng.
* Thiết kế hệ thống:
  + Sơ đồ chức năng (DFD).
  + Mô hình thực thể quan hệ (ERD).
  + Giao diện ứng dụng (UI/UX).

 **Chương 3: Xây dựng ứng dụng bán mỹ phẩm trên Android**

* Triển khai các chức năng chính:
  + Hiển thị sản phẩm.
  + Quản lý giỏ hàng và đặt hàng.
  + Tích hợp thanh toán online.
  + Đăng ký, đăng nhập và quản lý thông tin người dùng.
* Mô tả giao diện người dùng và các kết quả minh họa.
* Kiểm thử và đánh giá hiệu năng của ứng dụng.

# PHẦN 2: NỘI DUNG NGHIÊN CỨU

# CHƯƠNG I : CƠ SỞ LÝ THUYẾT

## 1.1. GIỚI THIỆU

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng công nghệ vào hoạt động kinh doanh đã trở thành xu hướng tất yếu nhằm nâng cao hiệu quả quản lý, tối ưu hóa quy trình làm việc và mang lại trải nghiệm tốt nhất cho khách hàng. Đặc biệt, ngành mỹ phẩm – một lĩnh vực đang phát triển vượt bậc với sự đa dạng về sản phẩm và thương hiệu – rất cần đến các giải pháp công nghệ để quản lý hoạt động bán hàng một cách hiệu quả.

Việc xây dựng một ứng dụng quản lý bán hàng mỹ phẩm không chỉ giúp đơn giản hóa việc quản lý sản phẩm, theo dõi tồn kho và xử lý hóa đơn mà còn hỗ trợ doanh nghiệp trong việc phân tích doanh thu, chăm sóc khách hàng, và cải thiện khả năng cạnh tranh. Ứng dụng này sẽ đóng vai trò như một công cụ hỗ trợ đắc lực, giúp các cửa hàng mỹ phẩm tiết kiệm thời gian, giảm thiểu sai sót, và đáp ứng nhanh chóng các yêu cầu của thị trường.

Dựa trên nhu cầu thực tiễn, đề tài "Xây dựng ứng dụng bán hàng mỹ phẩm" được thực hiện với mục tiêu nghiên cứu và phát triển một giải pháp phần mềm quản lý toàn diện. Ứng dụng sẽ tập trung vào các chức năng chính như quản lý sản phẩm, hóa đơn, khách hàng, thống kê doanh thu, và hỗ trợ tìm kiếm nhanh chóng. Đề tài sẽ ứng dụng các công nghệ hiện đại như ngôn ngữ lập trình C#, nền tảng Windows Forms, và hệ quản trị cơ sở dữ liệu SQL Server để đảm bảo tính ổn định, thân thiện và dễ sử dụng.

Thông qua việc xây dựng ứng dụng, đề tài không chỉ mang lại giá trị thiết thực trong việc nâng cao hiệu quả kinh doanh mà còn góp phần thúc đẩy quá trình số hóa trong lĩnh vực bán lẻ mỹ phẩm, tạo tiền đề cho sự phát triển bền vững của doanh nghiệp trong thời kỳ công nghệ 4.0.

## 1.2. CƠ SỞ LÝ THUYẾT

Cơ sở lý thuyết của đề tài "Xây dựng ứng dụng bán hàng mỹ phẩm" liên quan đến việc áp dụng các nguyên lý, khái niệm trong công nghệ thông tin và quản lý bán hàng để phát triển một hệ thống phần mềm hỗ trợ công việc kinh doanh mỹ phẩm. Những lý thuyết nền tảng trong các lĩnh vực này sẽ cung cấp cơ sở vững chắc cho việc thiết kế, phát triển và triển khai ứng dụng. Dưới đây là một số cơ sở lý thuyết quan trọng được áp dụng trong đề tài:

### 1.2.1 Hệ thống thông tin quản lý (MIS)

Hệ thống thông tin quản lý (Management Information System – MIS) là một hệ thống được thiết kế để thu thập, xử lý và phân tích dữ liệu trong doanh nghiệp, từ đó cung cấp các báo cáo và thông tin cần thiết giúp nhà quản lý đưa ra quyết định. Trong bối cảnh bán hàng mỹ phẩm, MIS sẽ giúp doanh nghiệp theo dõi và quản lý các hoạt động như tồn kho, bán hàng, đơn hàng, khách hàng và doanh thu. Việc áp dụng MIS trong xây dựng ứng dụng bán hàng mỹ phẩm giúp tối ưu hóa quy trình làm việc, giảm thiểu sai sót và nâng cao hiệu quả quản lý.

1.2.2Quản lý kho và chuỗi cung ứng

Quản lý kho là một phần quan trọng trong kinh doanh bán lẻ, đặc biệt trong lĩnh vực mỹ phẩm với số lượng sản phẩm lớn và đa dạng. Lý thuyết quản lý kho đề cập đến các phương pháp và chiến lược nhằm duy trì lượng hàng tồn kho hợp lý, tránh tình trạng hết hàng hay dư thừa, qua đó giảm thiểu chi phí lưu kho. Hệ thống bán hàng mỹ phẩm sẽ áp dụng các nguyên lý này để đảm bảo rằng việc quản lý sản phẩm, cập nhật tồn kho, và xử lý đơn hàng được thực hiện một cách chính xác và kịp thời.

### 1.2.3Quản lý quan hệ khách hàng (CRM)

Quản lý quan hệ khách hàng (Customer Relationship Management – CRM) là một chiến lược và hệ thống để doanh nghiệp duy trì và phát triển mối quan hệ với khách hàng. CRM giúp cải thiện sự hài lòng của khách hàng, tăng khả năng giữ chân khách hàng cũ và tìm kiếm khách hàng mới. Ứng dụng bán hàng mỹ phẩm sẽ tích hợp các chức năng CRM như lưu trữ thông tin khách hàng, lịch sử mua hàng, và phản hồi từ khách hàng, giúp doanh nghiệp dễ dàng tương tác và chăm sóc khách hàng một cách hiệu quả.

### 1.2.4 Hệ quản trị cơ sở dữ liệu (DBMS)

Hệ quản trị cơ sở dữ liệu (Database Management System – DBMS) là công cụ quan trọng giúp lưu trữ, truy xuất và quản lý dữ liệu trong các ứng dụng phần mềm. Trong ứng dụng bán hàng mỹ phẩm, DBMS sẽ được sử dụng để lưu trữ các thông tin về sản phẩm, khách hàng, đơn hàng và hóa đơn. SQL Server là một trong những hệ quản trị cơ sở dữ liệu phổ biến, được sử dụng trong đề tài để đảm bảo việc quản lý dữ liệu một cách an toàn và hiệu quả.

### 1.2.5 Thiết kế giao diện người dùng (UI/UX)

Giao diện người dùng (User Interface – UI) và trải nghiệm người dùng (User Experience – UX) đóng vai trò quan trọng trong việc xây dựng phần mềm dễ sử dụng và thân thiện với người dùng. Trong ứng dụng bán hàng mỹ phẩm, giao diện người dùng cần đơn giản, dễ hiểu và có thể truy cập nhanh chóng các chức năng cần thiết như quản lý sản phẩm, xử lý đơn hàng, và thống kê doanh thu. Các nguyên lý thiết kế UI/UX sẽ được áp dụng để đảm bảo người dùng có thể dễ dàng sử dụng ứng dụng mà không gặp phải khó khăn.

### 1.2.6 Nguyên lý phát triển phần mềm

Quá trình phát triển phần mềm sẽ dựa trên các nguyên lý phát triển phần mềm hiện đại, bao gồm các mô hình phát triển như Agile hoặc Waterfall, để đảm bảo ứng dụng được xây dựng một cách hiệu quả và linh hoạt. Các nguyên lý này giúp quản lý quá trình phát triển từ giai đoạn phân tích yêu cầu, thiết kế, lập trình, kiểm thử cho đến triển khai và bảo trì phần mềm.

### 1.2.7 Tối ưu hóa quy trình bán hàng

Việc tối ưu hóa quy trình bán hàng là yếu tố then chốt để tăng trưởng doanh thu và cải thiện hiệu suất làm việc trong một cửa hàng bán lẻ. Việc sử dụng ứng dụng bán hàng giúp tối giản hóa quy trình nhập liệu, tự động hóa việc xử lý đơn hàng và thu thập dữ liệu thống kê, giúp nhân viên bán hàng tiết kiệm thời gian và giảm thiểu sai sót.

Các cơ sở lý thuyết trên không chỉ giúp hiểu rõ về các khái niệm và nguyên lý áp dụng trong việc xây dựng hệ thống quản lý bán hàng mỹ phẩm mà còn là cơ sở để thiết kế, phát triển và triển khai một ứng dụng hoàn chỉnh, đáp ứng được yêu cầu thực tiễn trong ngành bán lẻ mỹ phẩm hiện nay.

## HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

* SQLite là một hệ quản trị cơ sở dữ liệu (DBMS) nhẹ, nhúng, không yêu cầu một server riêng biệt và được tích hợp trực tiếp vào ứng dụng. Điều này làm cho SQLite trở thành một công cụ lý tưởng cho các ứng dụng di động và ứng dụng nhúng, nơi việc sử dụng các hệ quản trị cơ sở dữ liệu phức tạp như MySQL hay PostgreSQL là không cần thiết. SQLite sử dụng các cơ sở dữ liệu dạng tệp tin (file-based), giúp tiết kiệm tài nguyên hệ thống và rất dễ tích hợp vào bất kỳ ứng dụng nào, đặc biệt là trong Android.



**Hình 1. 1 Logo SQLite**

### 1. 3.1 Kiến Trúc Của SQLite

SQLite là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) với đặc điểm là:

* Tệp cơ sở dữ liệu duy nhất: SQLite lưu trữ toàn bộ cơ sở dữ liệu trong một tệp duy nhất, giúp dễ dàng sao lưu, chia sẻ và di chuyển.
* Không cần server: SQLite hoạt động như một thư viện trong ứng dụng, do đó không yêu cầu server cơ sở dữ liệu riêng biệt.
* Đồng bộ dữ liệu đồng thời: SQLite hỗ trợ nhiều thao tác đọc đồng thời nhưng chỉ cho phép một tác vụ ghi duy nhất tại một thời điểm, đảm bảo sự ổn định và tránh xung đột dữ liệu.
* Khả năng tự phục hồi: SQLite hỗ trợ tính năng rollback (quay lại giao dịch trước đó), cho phép phục hồi dữ liệu trong trường hợp hệ thống gặp sự cố.

### 1.3.2 Các Thành Phần Chính Của SQLite

* Database: Một tệp tin lưu trữ cơ sở dữ liệu. Tệp tin này có định dạng .db hoặc .sqlite và chứa tất cả dữ liệu trong cơ sở dữ liệu.
* Table: Bảng là nơi lưu trữ dữ liệu theo dạng hàng (row) và cột (column). Các bảng có thể liên kết với nhau thông qua các khóa ngoại (foreign key).
* Index: Cung cấp cơ chế tìm kiếm nhanh cho các truy vấn, cải thiện hiệu suất khi làm việc với các bảng lớn.
* View: Là kết quả của một truy vấn được lưu trữ như một bảng ảo, giúp tái sử dụng các truy vấn phức tạp mà không cần phải thực hiện lại chúng mỗi khi cần.
* Trigger: Là các thao tác tự động được thực hiện khi có sự kiện thay đổi trên cơ sở dữ liệu, ví dụ như khi chèn, cập nhật hoặc xóa một bản ghi.

### 1.3.3 Các Phương Thức Tương Tác Với SQLite

Trong Android, tương tác với SQLite thông qua lớp SQLiteDatabase. Các phương thức quan trọng bao gồm:

* insert(): Chèn dữ liệu vào bảng.
* update(): Cập nhật dữ liệu trong bảng.
* delete(): Xóa dữ liệu khỏi bảng.
* query(**)**: Truy vấn dữ liệu từ một hoặc nhiều bảng.
* rawQuery(): Thực hiện truy vấn SQL thô, cho phép sử dụng các câu lệnh SQL phức tạp.
* **execSQL()**: Thực hiện các câu lệnh SQL mà không trả về kết quả (chẳng hạn như CREATE TABLE, INSERT, UPDATE, DELETE).

### 1.3.4 Các Đặc Điểm Nổi Bật Của SQLite

* Khả năng lưu trữ dữ liệu cục bộ: SQLite được tích hợp vào ứng dụng và không yêu cầu kết nối mạng để hoạt động. Điều này rất hữu ích cho các ứng dụng di động hoặc các ứng dụng cần lưu trữ tạm thời mà không phụ thuộc vào mạng.
* Tính di động cao: Vì cơ sở dữ liệu SQLite được lưu trữ trong một tệp duy nhất, người dùng có thể sao chép hoặc chuyển đổi tệp này dễ dàng.
* Hiệu suất tốt với cơ sở dữ liệu nhỏ và vừa: SQLite có thể xử lý các cơ sở dữ liệu có kích thước từ vài MB đến vài GB và thực hiện các truy vấn nhanh chóng, mặc dù không phải là hệ quản trị cơ sở dữ liệu tối ưu cho các cơ sở dữ liệu cực lớn hoặc có lượng truy vấn quá cao.
* Hỗ trợ giao dịch: SQLite hỗ trợ tính năng giao dịch (transaction) giúp đảm bảo tính toàn vẹn của dữ liệu. Các thao tác được thực hiện trong phạm vi của một giao dịch, và nếu có lỗi xảy ra, hệ thống sẽ quay lại trạng thái ban đầu, không làm thay đổi dữ liệu.
* Tính bảo mật: SQLite cung cấp một số cơ chế bảo mật như mã hóa cơ sở dữ liệu (có thể sử dụng thư viện mã hóa bên ngoài như SQLCipher) để bảo vệ dữ liệu khỏi truy cập trái phép.

### 1.3.5 Ưu Nhược Điểm của SQLite

**a) Ưu điểm:**

* Nhẹ và dễ sử dụng: SQLite rất dễ tích hợp vào ứng dụng mà không cần cài đặt hoặc cấu hình phức tạp.
* Không yêu cầu server: Đây là một lợi thế lớn đối với các ứng dụng di động hoặc các ứng dụng không có yêu cầu cao về khả năng mở rộng và hiệu suất.
* Tính di động: Dữ liệu được lưu trữ trong một tệp duy nhất có thể sao chép, di chuyển và sao lưu dễ dàng.
* Hỗ trợ ACID: SQLite đảm bảo các giao dịch luôn được thực hiện đúng cách, đảm bảo tính toàn vẹn dữ liệu.

**b) Nhược điểm:**

* Không tối ưu cho ứng dụng quy mô lớn: SQLite có thể gặp khó khăn khi làm việc với các cơ sở dữ liệu có dung lượng lớn hoặc yêu cầu xử lý đồng thời nhiều kết nối ghi.
* Không hỗ trợ các tính năng cao cấp: SQLite thiếu một số tính năng tiên tiến như phân tán cơ sở dữ liệu, khả năng xử lý đồng thời nhiều kết nối ghi (điều này chỉ cho phép một kết nối ghi tại một thời điểm).
* Không có giao diện quản lý đồ họa: Mặc dù SQLite dễ dàng thao tác qua mã nguồn, nhưng thiếu một công cụ quản lý đồ họa tích hợp sẵn để dễ dàng thao tác với cơ sở dữ liệu.

### 1.3.6 Ứng Dụng SQLite trong Android

SQLite trong Android thường được sử dụng để:

* Lưu trữ dữ liệu cục bộ: Ví dụ như danh sách sản phẩm trong ứng dụng thương mại điện tử, các bài viết trong ứng dụng đọc báo, lịch sử giao dịch trong ứng dụng ngân hàng.
* Quản lý các thông tin tạm thời: Các thông tin mà người dùng nhập vào (chẳng hạn như lịch sử tìm kiếm, tùy chọn người dùng) có thể được lưu trữ trong SQLite để sử dụng lại trong các phiên làm việc tiếp theo.
* Thực hiện các tính toán và phân tích dữ liệu: Các ứng dụng có thể lưu trữ dữ liệu lớn để phân tích sau này, ví dụ như trong ứng dụng phân tích dữ liệu hoặc ứng dụng ghi chép tài chính.

### 1.3.7 Kết luận

SQLite là một công cụ lý tưởng để lưu trữ dữ liệu cục bộ trong các ứng dụng di động và nhúng. Nhờ vào tính nhẹ nhàng, dễ tích hợp và không yêu cầu server, SQLite giúp đơn giản hóa việc quản lý cơ sở dữ liệu trong các ứng dụng Android, giúp tối ưu hiệu suất và giảm thiểu tài nguyên hệ thống. Tuy nhiên, các ứng dụng yêu cầu mở rộng quy mô hoặc xử lý khối lượng dữ liệu lớn có thể gặp phải các hạn chế về khả năng xử lý đồng thời và tính năng cao cấp.

## Công cụ phân tích và thiết kế

Trong quá trình phát triển hệ thống ứng dụng bán hàng mỹ phẩm, việc sử dụng các công cụ phân tích và thiết kế là rất quan trọng để đảm bảo hệ thống hoạt động hiệu quả và đáp ứng được yêu cầu người dùng. Các công cụ này giúp các nhà phát triển mô phỏng, lập kế hoạch, và thiết kế các yếu tố cần thiết cho hệ thống trước khi triển khai. Dưới đây là một số công cụ phổ biến trong quá trình phân tích và thiết kế hệ thống ứng dụng bán hàng mỹ phẩm:

### 1.4.1 Công Cụ Phân Tích

Công cụ phân tích giúp xác định các yêu cầu, dòng chảy công việc, và mô hình dữ liệu của hệ thống. Những công cụ này thường được sử dụng trong giai đoạn đầu của quy trình phát triển phần mềm để thu thập thông tin cần thiết và phân tích các yếu tố ảnh hưởng đến việc thiết kế hệ thống.

* **UML (Unified Modeling Language)**: Là một ngôn ngữ chuẩn để mô tả, xây dựng và phân tích hệ thống phần mềm. UML cung cấp nhiều loại sơ đồ, chẳng hạn như sơ đồ use case, sơ đồ lớp, sơ đồ tuần tự, sơ đồ trạng thái, giúp các nhà phát triển hiểu rõ các yêu cầu và mối quan hệ giữa các thành phần trong hệ thống.
  + Sơ đồ Use Case: Phân tích các chức năng mà người dùng sẽ sử dụng trong hệ thống, như mua hàng, xem sản phẩm, thanh toán, và quản lý đơn hàng.
  + Sơ đồ Lớp: Mô tả các đối tượng trong hệ thống và mối quan hệ giữa chúng, giúp xây dựng kiến trúc cơ sở dữ liệu cho ứng dụng.
  + Sơ đồ tuần tự: Mô tả các tương tác giữa các đối tượng trong hệ thống qua các thời gian cụ thể, ví dụ như cách thức xử lý đơn hàng trong ứng dụng.
* **ERD (Entity Relationship Diagram)**: Đây là công cụ giúp mô tả mối quan hệ giữa các thực thể trong cơ sở dữ liệu. ERD rất quan trọng trong việc thiết kế cơ sở dữ liệu của ứng dụng bán mỹ phẩm, như thông tin sản phẩm, khách hàng, đơn hàng và thanh toán.
  + Entities: Các thực thể chính như Khách hàng, Sản phẩm, Đơn hàng.
  + Relationships: Mối quan hệ giữa các thực thể, ví dụ một khách hàng có thể đặt nhiều đơn hàng.
* **Flowchart (Sơ đồ luồng)**: Sử dụng để mô tả luồng công việc trong ứng dụng. Ví dụ, từ khi khách hàng đăng nhập vào hệ thống đến khi họ hoàn tất thanh toán. Sơ đồ này giúp dễ dàng nhìn thấy các bước thực hiện trong quy trình và nhận diện các điểm có thể cải tiến trong giao diện người dùng hoặc quy trình công việc.
* **IDE (Integrated Development Environment)**: Công cụ phát triển phần mềm là môi trường hỗ trợ lập trình viên trong việc viết mã và kiểm thử hệ thống.
  + Android Studio: Đây là IDE chính để phát triển ứng dụng Android. Nó hỗ trợ lập trình Java hoặc Kotlin, đồng thời tích hợp với các công cụ kiểm thử, trình giả lập, và các công cụ phát triển giao diện người dùng (UI).

## Ngôn ngữ lập trình

Phát triển ứng dụng Android yêu cầu sử dụng một số ngôn ngữ lập trình và công cụ hỗ trợ để xây dựng các tính năng, giao diện và quản lý dữ liệu hiệu quả. Dưới đây là các ngôn ngữ lập trình chính được sử dụng trong phát triển ứng dụng Android, cùng với các đặc điểm chi tiết của từng ngôn ngữ:

**a) Java**

Java là ngôn ngữ lập trình chính và phổ biến nhất trong phát triển ứng dụng Android. Mặc dù từ năm 2017, Google đã giới thiệu Kotlin như một ngôn ngữ chính thức cho Android, Java vẫn được sử dụng rộng rãi và có một cộng đồng người dùng mạnh mẽ. Dưới đây là các lý do Java được sử dụng trong phát triển ứng dụng Android:

* Tính ổn định: Java đã có mặt trên thị trường từ lâu và được tối ưu hóa để phát triển các ứng dụng lớn, bền vững và hiệu quả. Điều này làm cho Java trở thành lựa chọn phổ biến cho các ứng dụng Android phức tạp.
* Hỗ trợ lập trình hướng đối tượng (OOP): Java hỗ trợ lập trình hướng đối tượng, cho phép người lập trình dễ dàng xây dựng các mô-đun, tái sử dụng mã nguồn và quản lý các đối tượng phức tạp trong ứng dụng.
* Tính tương thích cao: Java chạy trên nền tảng Java Virtual Machine (JVM), giúp mã Java có thể chạy trên bất kỳ nền tảng nào hỗ trợ JVM, bao gồm Android.
* Thư viện phong phú: Java có một kho thư viện (libraries) đồ sộ, bao gồm các thư viện cho giao diện người dùng (UI), quản lý cơ sở dữ liệu, kết nối mạng và xử lý đa luồng (multithreading).
* Google hỗ trợ: Google cung cấp rất nhiều tài liệu và công cụ phát triển ứng dụng Android bằng Java, bao gồm Android Studio và SDK (Software Development Kit).

**Cấu trúc ứng dụng Android bằng Java**

Ứng dụng Android thường được tổ chức thành các Activity, Service, Broadcast Receiver, và Content Provider, mỗi thành phần có chức năng riêng biệt. Java sẽ được sử dụng để xử lý các sự kiện và logic của ứng dụng. Ví dụ, khi người dùng nhấn vào một nút, Java sẽ xử lý hành động này và cập nhật giao diện người dùng.



**Hình 1. 2 Lập trình Android**

**b) XML (Extensible Markup Language)**

Trong phát triển ứng dụng Android, XML không phải là ngôn ngữ lập trình, nhưng lại đóng một vai trò vô cùng quan trọng trong việc xây dựng giao diện người dùng (UI). XML được sử dụng để mô tả các thành phần giao diện của ứng dụng, bao gồm các nút bấm, văn bản, hình ảnh, và các layout (bố cục) của màn hình.

* Tách biệt giao diện và logic: XML giúp tách biệt phần giao diện người dùng khỏi phần xử lý logic của ứng dụng, giúp mã nguồn dễ dàng bảo trì và nâng cấp.
* Dễ đọc và dễ hiểu: XML có cú pháp đơn giản và dễ hiểu, giúp các nhà phát triển có thể dễ dàng thiết kế và chỉnh sửa giao diện.
* Đa dạng layout: Android cung cấp nhiều kiểu layout (LinearLayout, RelativeLayout, ConstraintLayout, v.v.) giúp bố trí các thành phần giao diện một cách dễ dàng và linh hoạt.

**c) Gradle**

**Gradle** là công cụ xây dựng (build tool) chính thức trong phát triển ứng dụng Android. Gradle sử dụng các tệp cấu hình bằng ngôn ngữ Groovy hoặc Kotlin DSL để xây dựng các ứng dụng Android.

* Quản lý phụ thuộc: Gradle hỗ trợ việc quản lý các thư viện phụ thuộc mà ứng dụng cần sử dụng, giúp dễ dàng thêm hoặc loại bỏ các thư viện bên ngoài.
* Tích hợp tốt với Android Studio: Gradle được tích hợp chặt chẽ với Android Studio, giúp tự động hóa quá trình biên dịch, kiểm tra và đóng gói ứng dụng.
* Tối ưu hóa quá trình xây dựng: Gradle cho phép tối ưu hóa quá trình xây dựng bằng cách chỉ biên dịch lại các phần mã nguồn đã thay đổi, giảm thiểu thời gian xây dựng ứng dụng.

**d) Công Cụ Hỗ Trợ và Thư Viện**

* Android Studio: Đây là môi trường phát triển chính thức và mạnh mẽ cho Android. Android Studio hỗ trợ đầy đủ các tính năng như code editor, thiết kế giao diện, gỡ lỗi, và thử nghiệm ứng dụng.
* Firebase: Là một nền tảng của Google cung cấp các dịch vụ như xác thực người dùng, cơ sở dữ liệu thời gian thực, và phân tích ứng dụng. Firebase giúp lập trình viên dễ dàng xây dựng các ứng dụng Android có tính năng mạnh mẽ mà không cần phải xây dựng lại từ đầu.
* Retrofit: Là thư viện HTTP phổ biến cho phép Android giao tiếp dễ dàng với các API từ xa. Retrofit giúp gửi và nhận dữ liệu từ các dịch vụ web RESTful.
* Room: Là một thư viện trong Android giúp quản lý cơ sở dữ liệu SQLite với API đơn giản và dễ sử dụng. Room hỗ trợ lưu trữ dữ liệu trong ứng dụng Android một cách nhanh chóng và hiệu quả.
  1. **Ưu điểm – nhược điểm**
* **Ưu điểm**:
* Tính ổn định: Java đã được sử dụng rộng rãi trong phát triển ứng dụng Android từ khi Android ra đời, và có cộng đồng hỗ trợ lớn. Điều này giúp các lập trình viên dễ dàng tìm kiếm tài liệu, ví dụ, và giải pháp cho các vấn đề.
* Tính tương thích cao: Java có thể chạy trên nhiều nền tảng thông qua JVM (Java Virtual Machine), giúp mã Java dễ dàng chuyển sang các hệ thống khác ngoài Android.
* Cộng đồng mạnh mẽ: Java có một cộng đồng lập trình viên rất lớn, giúp giải quyết các vấn đề nhanh chóng và tìm kiếm các thư viện hỗ trợ cho ứng dụng Android.
* Hỗ trợ tốt từ Google: Google cung cấp nhiều công cụ và tài liệu hỗ trợ Java trong phát triển ứng dụng Android, bao gồm Android SDK và Android Studio.
* **Nhược điểm**:
* Cú pháp dài dòng: So với Kotlin, Java có cú pháp dài và phức tạp hơn, khiến mã nguồn dễ bị rối và khó bảo trì, đặc biệt đối với các ứng dụng lớn.
* Thiếu tính năng hiện đại: Java không hỗ trợ nhiều tính năng mới như Kotlin (ví dụ, các tính năng về an toàn với null, mở rộng thư viện, v.v.), dẫn đến việc lập trình viên phải sử dụng nhiều dòng mã hơn và khó tránh được lỗi.
* Khó bảo trì mã nguồn: Các ứng dụng Java lớn với mã nguồn phức tạp có thể gặp khó khăn trong việc bảo trì và cập nhật, nhất là khi có yêu cầu thêm các tính năng mới.

# CHƯƠNG II: PHÂN TÍCH THIẾT KẾ CHƯƠNG TRÌNH

## 2.1 MÔ TẢ BÀI TOÁN

Với sự phát triển mạnh mẽ của thương mại điện tử và nhu cầu làm đẹp ngày càng cao, việc bán mỹ phẩm qua ứng dụng Android trở thành xu hướng phổ biến. Tuy nhiên, nhiều cửa hàng mỹ phẩm vẫn gặp khó khăn trong việc quản lý sản phẩm, đơn hàng, khách hàng, và thanh toán, dẫn đến hạn chế trong việc nâng cao trải nghiệm mua sắm.

**Yêu cầu đặt ra:**  
Xây dựng một ứng dụng Android bán mỹ phẩm với các tính năng quản lý và hỗ trợ kinh doanh hiệu quả, bao gồm:

* Quản lý sản phẩm: Hỗ trợ thêm, sửa, xóa thông tin mỹ phẩm (tên, giá, mô tả, hình ảnh, số lượng tồn kho).
* Quản lý danh mục: Phân loại mỹ phẩm theo từng danh mục như chăm sóc da, trang điểm, nước hoa, v.v.
* Quản lý khách hàng: Lưu trữ thông tin khách hàng (tên, số điện thoại, địa chỉ).
* Xử lý đơn hàng: Hiển thị giỏ hàng, tính toán tổng giá trị, trạng thái giao hàng.
* Thanh toán: Hỗ trợ nhiều phương thức như tiền mặt, thẻ tín dụng, ví điện tử.
* Thống kê: Báo cáo doanh thu, số lượng sản phẩm bán ra, tồn kho.
* Tìm kiếm: Cho phép người dùng tìm kiếm sản phẩm theo tên hoặc danh mục.

**Lợi ích của ứng dụng:**

* Đơn giản hóa quy trình bán hàng, tiết kiệm thời gian và công sức quản lý.
* Tăng trải nghiệm mua sắm tiện lợi cho khách hàng, từ đó thu hút nhiều người dùng hơn.
* Hỗ trợ đưa ra quyết định kinh doanh thông qua các báo cáo thống kê chi tiết.
* Nâng cao tính cạnh tranh của cửa hàng trong bối cảnh số hóa hiện nay.

**Mục tiêu:**  
Ứng dụng không chỉ đáp ứng nhu cầu hiện tại mà còn có khả năng mở rộng để tích hợp các tính năng nâng cao như khuyến mãi, gợi ý sản phẩm theo sở thích, hoặc đánh giá và xếp hạng sản phẩm.

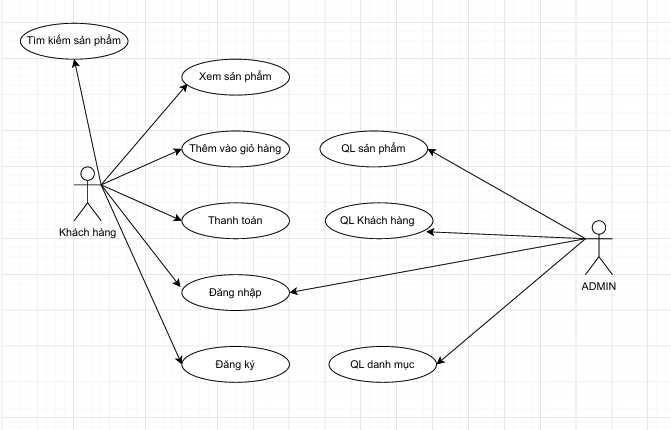
## 2.2 TRÌNH BÀY CÁC MÔ HÌNH, SƠ ĐỒ (UC,BFD, DFD, ERD)

### 2.2.1 Mô hình UseCase – Sơ đồ ngữ cảnh

- Tác nhân: + Khách hàng + Tìm kiếm sản phẩm + Xem sản phẩm + Thêm vào giỏ hàng + Thanh toán + Đăng nhập + Đăng ký

- Admin: + Quản lý sản phẩm + Quản lý khách hàng + Quản lý danh mục

- Mối quan hệ: + Khách hàng thực hiện các chức năng mua sắm và tài khoản. + Admin quản lý dữ liệu hệ thống. USE CASE



**Hình 1. 3 Sơ đồ ngữ cảnh UseCase**

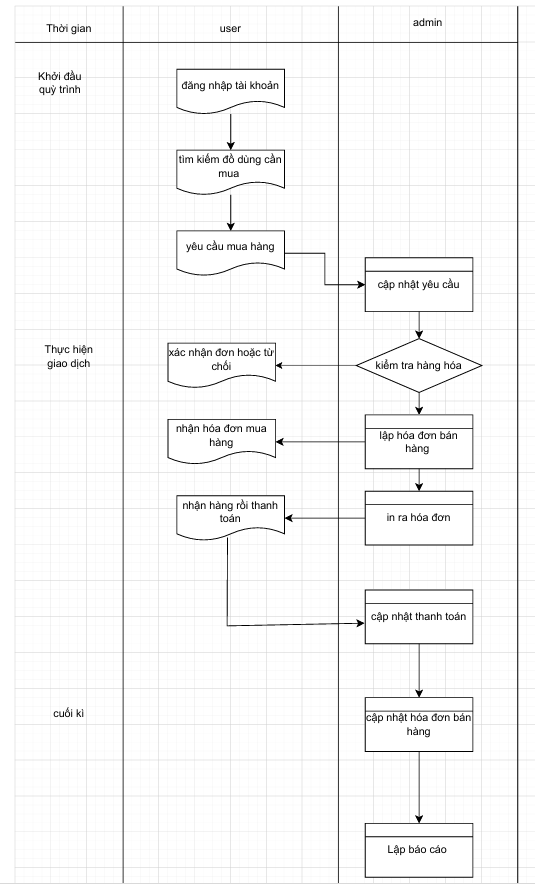
2.2.2 Mô hình IFD – Sơ đồ **luồng thông tin**

**-** Sơ đồ mô tả quy trình mua hàng giữa User và Admin như sau:

Khởi đầu quy trình: User đăng nhập tài khoản → tìm kiếm đồ cần mua → yêu cầu mua hàng. Admin nhận yêu cầu và cập nhật yêu cầu.

Thực hiện giao dịch: Admin kiểm tra hàng hóa → lập hóa đơn và in hóa đơn. User xác nhận đơn → nhận hóa đơn → nhận hàng và thanh toán. Admin cập nhật thanh toán và hóa đơn bán hàng.

Cuối kỳ: Admin lập báo cáo.



**Hình 1. 4 Sơ đồ luồng thông tin - IFD**

### 2.2.3 Mô hình BFD – Sơ đồ phân rã chức năng

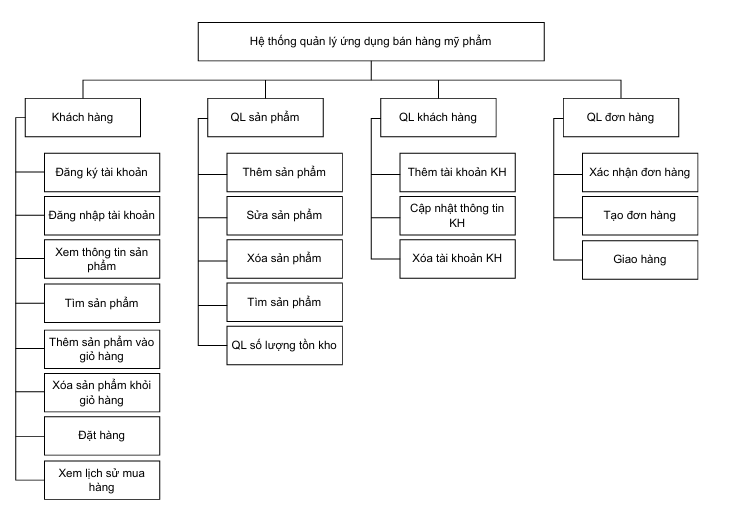
- Hệ thống quản lý ứng dụng bán hàng mỹ phẩm

1. Chức Năng Khách hàng: • Đăng ký/đăng nhập tài khoản. • Xem thông tin, tìm sản phẩm. • Thêm/xóa sản phẩm khỏi giỏ hàng. • Đặt hàng và xem lịch sử mua hàng.

2. Chức Năng Quản lý sản phẩm (QL sản phẩm): • Thêm, sửa, xóa sản phẩm. • Tìm kiếm sản phẩm. • Quản lý số lượng tồn kho.

3. Chức Năng Quản lý khách hàng: • Thêm. • Cập nhật. • Xóa tài khoản khách hàng.

4. Chức Năng Quản lý đơn hàng: • Xác nhận. • Tạo đơn hàng. • Giao hàng.



**Hình 1. 5 Sơ đồ phân rã chức năng**

### 2.2.4 Mô hình DFD – Sơ đồ luồng dữ liệu

- Mức đỉnh :

1. Tác nhân (External Entities):

• Admin: Quản lý hệ thống.

• Khách hàng: Thực hiện thao tác đặt hàng.

2. Chức năng chính (Processes):

• Quản lý khách hàng: Admin quản lý danh sách khách hàng (DS khách hàng).

• Quản lý sản phẩm: Admin quản lý danh sách sản phẩm (DS khách hàng).

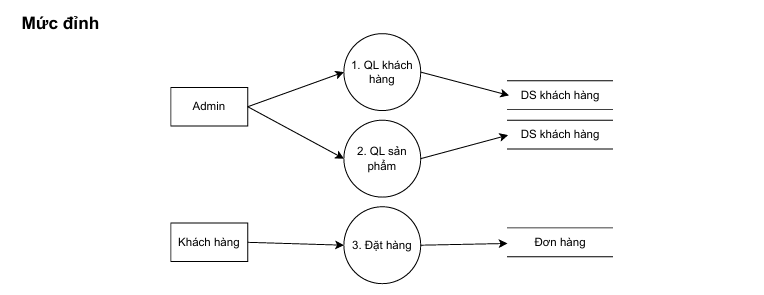
• Đặt hàng: Khách hàng thực hiện đặt hàng và kết quả là tạo ra đơn hàng.

3. Luồng dữ liệu (Data Flows):

• Admin gửi dữ liệu đến: o Quản lý khách hàng → Xuất ra DS khách hàng.

o Quản lý sản phẩm → Xuất ra DS khách hàng.

• Khách hàng gửi yêu cầu đến chức năng Đặt hàng, kết quả tạo ra Đơn hàng.



**Hình 1. 6 Sơ đồ luồng dữ liệu - DFD mức đỉnh**

-Mức dưới đỉnh chức năng khách hàng:

1. Tác nhân (External Entity):

• Admin: Người thao tác và quản lý thông tin khách hàng.

2. Chức năng chính (Processes):

• Xác nhận thông tin KH: Xác nhận thông tin khách hàng.

• Thêm tài khoản KH: Thêm mới tài khoản khách hàng và lưu thông tin vào danh sách khách hàng (Lưu TK).

• Tìm kiếm thông tin KH: Tìm kiếm thông tin khách hàng từ danh sách khách hàng.

• Cập nhật thông tin KH: Cập nhật thông tin tài khoản khách hàng (Cập nhật TK).

• Xóa tài khoản KH: Xóa tài khoản khách hàng khỏi danh sách (Xóa TK).

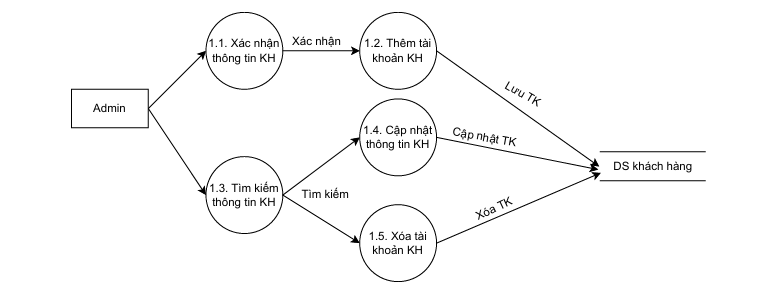
3. Kho dữ liệu (Data Store):

• DS khách hàng: Nơi lưu trữ danh sách khách hàng, bao gồm thông tin đã được thêm, cập nhật hoặc xóa.

4. Luồng dữ liệu (Data Flows):

• Admin thực hiện các chức năng xác nhận, thêm, tìm kiếm, cập nhật, và xóa tài khoản khách hàng.

• Các chức năng này tương tác trực tiếp với DS khách hàng để lưu, cập nhật, tìm kiếm hoặc xóa dữ liệu.



**Hình 1. 7 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng khách hàng**

* Mức dưới đỉnh chức năng sản phẩm:

1) Actor: Admin • Người thực hiện các thao tác trên hệ thống quản lý sản phẩm.

2) Các hoạt động chính:

• Nhập thông tin SP: Admin nhập thông tin sản phẩm mới.

• Thêm sản phẩm: Sau khi nhập thông tin, Admin thực hiện thêm sản phẩm vào danh sách sản phẩm (DS sản phẩm).

• Tìm kiếm SP: Admin tìm kiếm sản phẩm có trong danh sách.

• Cập nhật thông tin SP: Admin cập nhật thông tin của sản phẩm hiện có.

• Xóa SP: Admin xóa sản phẩm khỏi danh sách.

3)Dữ liệu chung: DS sản phẩm

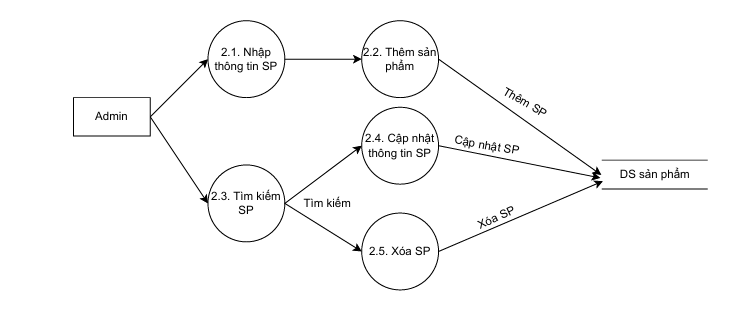
• DS sản phẩm (Danh sách sản phẩm) là nơi lưu trữ thông tin về sản phẩm.

• Các hoạt động như "Thêm SP", "Cập nhật SP", "Xóa SP" đều tác động đến DS sản phẩm.

4)Luồng hoạt động:

• Nhập thông tin SP dẫn đến Thêm sản phẩm.

• Tìm kiếm SP có thể thực hiện độc lập và không ảnh hưởng đến các bước khác. • Cập nhật thông tin SP và Xóa SP được thực hiện dựa trên kết quả tìm kiếm sản phẩm hoặc lựa chọn từ DS sản phẩm.



**Hình 1. 8 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng sản phẩm**

* Mức dưới đỉnh chức năng đặt mua hàng:

1 Actor: Admin • Người thực hiện các thao tác trên hệ thống quản lý sản phẩm.

2 Các hoạt động chính:

• Nhập thông tin SP: Admin nhập thông tin sản phẩm mới.

• Thêm sản phẩm: Sau khi nhập thông tin, Admin thực hiện thêm sản phẩm vào danh sách sản phẩm (DS sản phẩm).

• Tìm kiếm SP: Admin tìm kiếm sản phẩm có trong danh sách.

• Cập nhật thông tin SP: Admin cập nhật thông tin của sản phẩm hiện có.

• Xóa SP: Admin xóa sản phẩm khỏi danh sách.

* + 1. Dữ liệu chung: DS sản phẩm

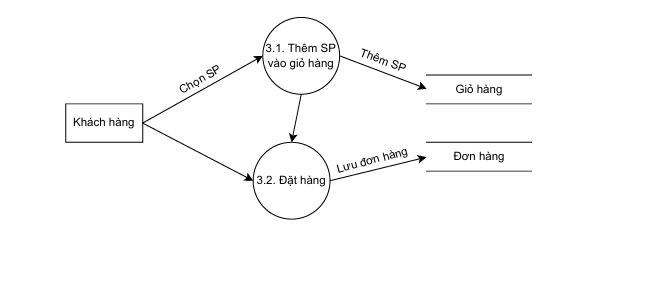
• DS sản phẩm (Danh sách sản phẩm) là nơi lưu trữ thông tin về sản phẩm.

• Các hoạt động như "Thêm SP", "Cập nhật SP", "Xóa SP" đều tác động đến DS sản phẩm.

4 Luồng hoạt động:

• Nhập thông tin SP dẫn đến Thêm sản phẩm.

• Tìm kiếm SP có thể thực hiện độc lập và không ảnh hưởng đến các bước khác. • Cập nhật thông tin SP và Xóa SP được thực hiện dựa trên kết quả tìm kiếm sản phẩm hoặc lựa chọn từ DS sản phẩm.



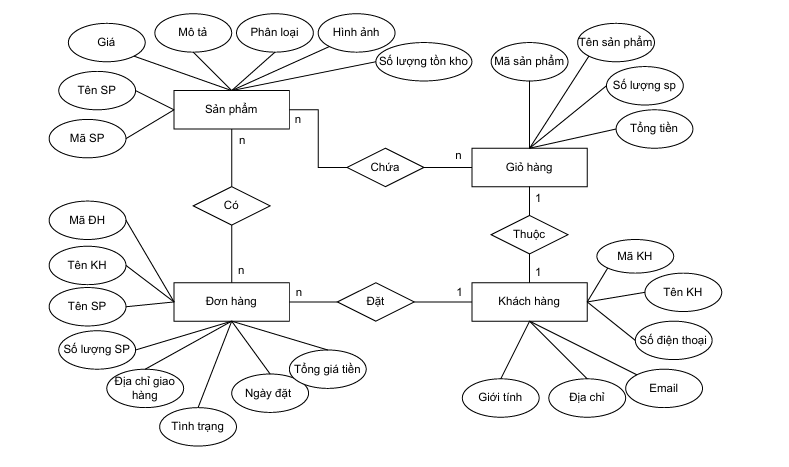
**Hình 1. 9 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng mua hàng**

### 2.2.5 Mô hình ERD – Biểu đồ quan hệ thực thể

- Sơ đồ thực thể (ERD) được sử dụng để biểu diễn mối quan hệ giữa các thực thể (entities) trong cơ sở dữ liệu của hệ thống. Nó tập trung vào việc làm rõ các bảng dữ liệu, thuộc tính (attributes), và các mối quan hệ (relationships) giữa chúng.

**Các thành phần chính trong sơ đồ thực thể:**

* **Thực thể (Entities):** Đại diện cho các đối tượng cần quản lý trong hệ thống, ví dụ: *Khách hàng, Sản phẩm, Đơn hàng*.
* **Thuộc tính (Attributes):** Mô tả các đặc điểm hoặc thông tin của thực thể, ví dụ: *Tên khách hàng, Giá sản phẩm, Số lượng*.
* **Mối quan hệ (Relationships):** Biểu diễn cách các thực thể liên kết với nhau, ví dụ: *Khách hàng đặt Đơn hàng*, *Đơn hàng chứa Sản phẩm*.



**Hình 1. 10 Biểu đồ quan hệ thực thể - ERD**

# CHƯƠNG III: DEMO CHƯƠNG TRÌNH

## 3.1 MÔI TRƯỜNG THỰC NGHIỆM

### 3.1.1 Công cụ phát triển

- Tải và cài đặt Android Studio: Đây là IDE chính thức được Google hỗ trợ cho lập trình Android. Android Studio cung cấp các công cụ cần thiết như trình giả lập, kiểm tra hiệu năng, và tích hợp Git. Cấu hình Android Studio:

- Chọn SDK phiên bản tương thích (thường là từ Android 7.0 - API 24 trở lên).

- Cài đặt AVD (Android Virtual Device) để chạy thử nghiệm ứng dụng trên giả lập. Android Studio yêu cầu JDK 11 hoặc JDK 17 (Java SE Development Kit). Bạn cần tải và cài đặt JDK trước khi cài đặt Android Studio.

### 3.1.2 Cơ sở dữ liệu SQLite (Local Database)

- Tích hợp sẵn trong Android, phù hợp cho ứng dụng nhỏ và chạy offline.

- Sử dụng thư viện Room (ORM) để quản lý cơ sở dữ liệu dễ dàng hơn.

### 3.1.3 Thiết bị thử nghiệm Giả lập Android (Emulator)

- Sử dụng AVD của Android Studio để tạo các cấu hình giả lập.

- Kiểm tra trên các phiên bản Android khác nhau (tối thiểu từ Android 7.0 trở lên). Thiết bị thật (Physical Device)

- Dùng điện thoại Android để kiểm tra ứng dụng: - Bật chế độ Developer Options và cho phép USB Debugging.

- Kết nối qua USB hoặc WiFi để chạy thử ứng dụng trực tiếp.

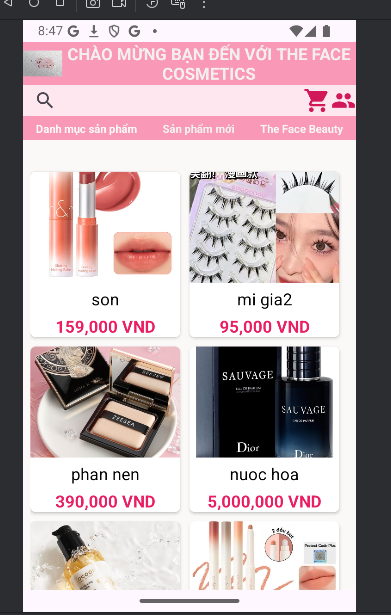
3.1.4 Quản lý mã nguồn Dùng GitHub để quản lý mã nguồn.

- Thiết lập nhánh (branch) để làm việc nhóm.

## 3.2 KẾT QUẢ APP

### 3.2.1 Giao diện trang chủ

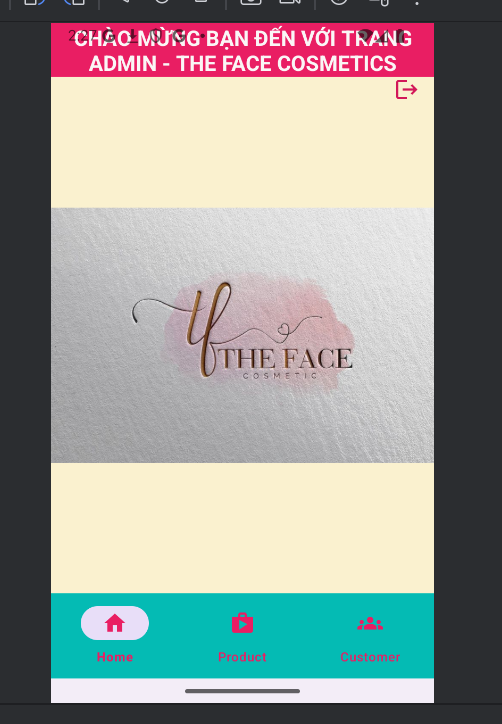
- Trang chủ hiển thị các sản phẩm nổi bật kèm hình ảnh, tên sản phẩm, giá cả giúp người dùng dễ dàng lựa chọn.



**Hình 1. 11 Giao diện trang chủ user**

### 3.2.2 Trang admin

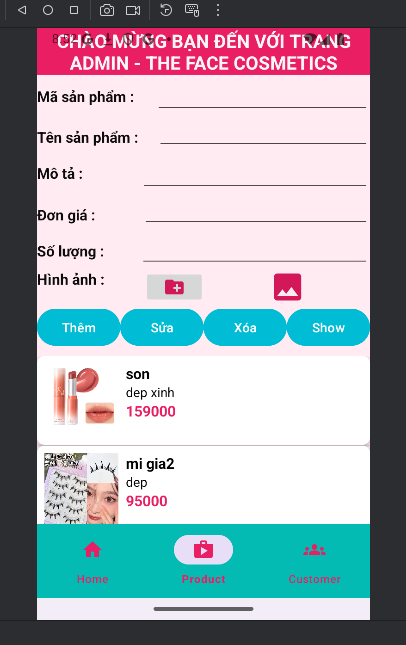
- Trang admin giúp người quản trị quản lý và điều hành hệ thống một cách hiệu quả.



**Hình 1. 12 Giao diện trang admin**

### 3.2.3 Trang quản lý sản phẩm

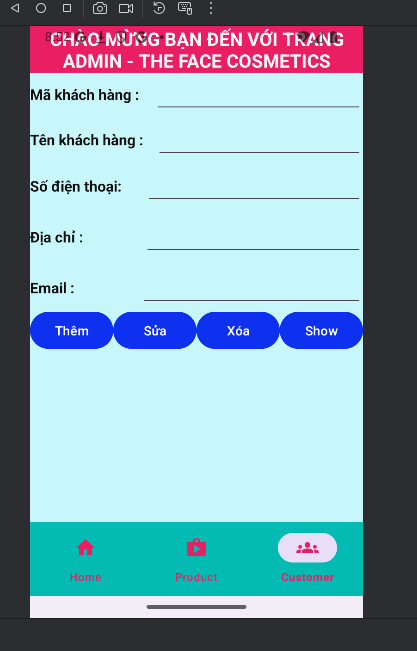
- Trang quản lý sản phẩm giúp người quản trị dễ dàng thao tác và quản lý thông tin sản phẩm trong hệ thống: - Thêm sản phẩm mới: Admin có thể thêm sản phẩm bằng cách nhập thông tin sản phẩm như mã sản phẩm, tên, mô tả, đơn giá, số lượng và hình ảnh, sau đó lưu vào hệ thống. - Chỉnh sửa sản phẩm: Admin có thể chọn sản phẩm muốn sửa và cập nhật thông tin sản phẩm đó. - Xóa sản phẩm: Admin có thể xóa các sản phẩm không cần thiết khỏi hệ thống. - Hiển thị sản phẩm: Admin có thể xem danh sách sản phẩm hiện có trong hệ thống.



**Hình 1. 13 Giao diện trang quản lý sản phẩm**

### 3.2.4 Trang quản lý khách hàng

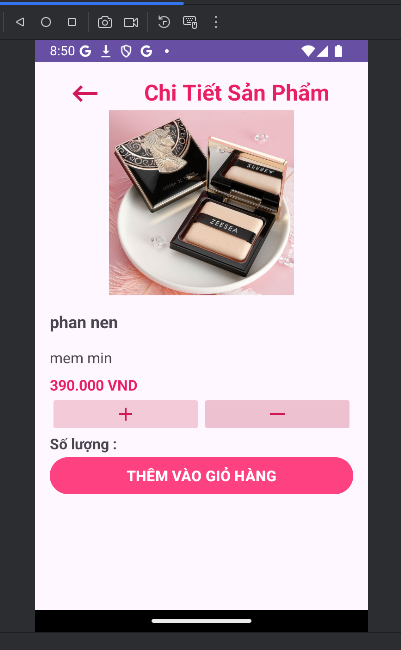
- Trang quản lý khách hàng giúp người quản trị dễ dàng quản lý thông tin của khách hàng: - Thêm khách hàng mới: Admin có thể thêm khách hàng bằng cách nhập thông tin khách hàng như mã khách hàng, tên khách hàng, số điện thoại, địa chỉ, email sau đó lưu vào hệ thống. - Chỉnh sửa khách hàng: Admin có thể chọn khách hàng muốn sửa và cập nhật thông tin khách hàng đó. - Xóa khách hàng: Admin có thể xóa khách hàng khỏi hệ thống. - Hiển thị khách hàng: Admin có thể xem danh sách khách hàng hiện có trong hệ thống.



**Hình 1. 14 Giao diện trang quản lý khách hàng**

### 3.2.5 Trang chi tiết sản phẩm

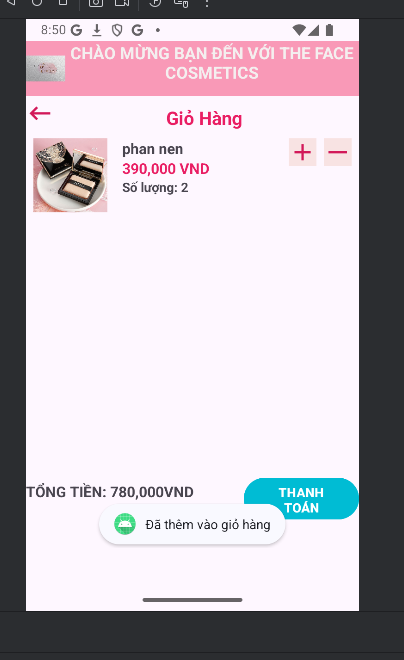
- Trang chi tiết sản phẩm hiển thị thông tin chi tiết của sản phẩm như hình ảnh, tên sản phẩm, mô tả, giá tiền khi người dùng click vào sản phẩm đó. Người dùng có thể tăng giảm số lượng sản phẩm và thêm vào giỏ hàng.



**Hình 1. 15 Giao diện trang chi tiết sản phẩm**

### 3.2.6 Trang giỏ hàng

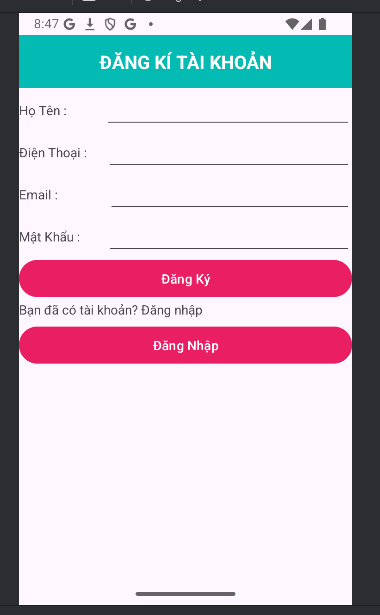
- Trang giỏ hàng hiển thị các sản phẩm mà người dùng đã thêm vào giỏ hàng, người dùng có thể điều chỉnh số lượng sản phẩm và hệ thống sẽ tự động tính tổng tiền các sản phẩm hiện có trong giỏ hàng. Nút "THANH TOÁN" cho phép người dùng chuyển sang bước tiếp theo để hoàn tất đơn hàng.



**Hình 1. 16 Giao diện trang giỏ hàng**

### 3.2.7 Trang đăng ký

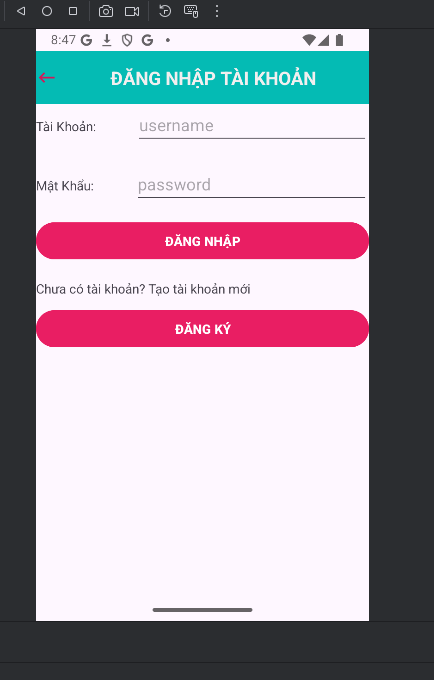
- Trang đăng ký giúp người dùng đăng ký tài khoản mới bao gồm các thông tin: Họ tên, số điện thoại, email, mật khẩu. Sau khi điền đầy đủ thông tin, người dùng nhấn nút "Đăng Ký" để tạo tài khoản mới. Nếu đã có tài khoản, họ có thể chọn tùy chọn "Đăng Nhập" để chuyển sang trang đăng nhập.



**Hình 1. 17 Giao diện trang đăng ký người dùng**

### 3.2.8 Trang đăng nhập

- Trang đăng nhập phép người dùng truy cập vào tài khoản đã có trên hệ thống. Người dùng cần nhập: Tài khoản và mật khẩu, sau đó nhấn nút "Đăng Nhập" để xác thực thông tin và vào hệ thống. Nếu chưa có tài khoản, người dùng có thể nhấn "Đăng Ký" để chuyển sang trang tạo tài khoản mới.



**Hình 1. 18 Giao diện trang đăng nhập người dùng**

# PHẦN 3: KẾT LUẬN

## NHỮNG VẤN ĐỀ ĐẠT ĐƯỢC

Hệ thống đã xây dựng được các thành phần giao diện chính, bao gồm:

* **Trang chủ**: Hiển thị các sản phẩm nổi bật, danh mục sản phẩm và các thông tin khuyến mãi.
* **Giỏ hàng**: Chức năng thêm sản phẩm vào giỏ, chỉnh sửa số lượng, và tính tổng giá trị đơn hàng.
* **Admin**: Giao diện quản lý dành cho quản trị viên để thực hiện các thao tác như thêm, sửa, xóa sản phẩm hoặc quản lý tài khoản.
* **Quản lý sản phẩm**: Hiển thị danh sách sản phẩm, cho phép quản lý thông tin sản phẩm như tên, giá, số lượng, và hình ảnh.
* **Quản lý khách hàng**: Quản lý danh sách khách hàng, thông tin chi tiết, và theo dõi lịch sử mua hàng.
* **Mua hàng**: Hỗ trợ quy trình đặt hàng từ việc chọn sản phẩm đến thanh toán.
* **Chi tiết sản phẩm**: Hiển thị thông tin cụ thể của từng sản phẩm, bao gồm hình ảnh, giá cả, mô tả, và các tùy chọn mua hàng.
* **Đăng ký, Đăng nhập**:
  + **Đăng ký tài khoản**: Cho phép người dùng tạo tài khoản mới bằng cách nhập thông tin cơ bản như tên, email, mật khẩu.
  + **Đăng nhập**: Hỗ trợ người dùng đăng nhập vào hệ thống bằng tài khoản đã đăng ký. Xác thực thông tin đăng nhập, đảm bảo tính bảo mật và quyền truy cập hệ thống.

## NHỮNG VẤN ĐỀ CÒN HẠN CHẾ

Bên cạnh các chức năng đã hoàn thành còn có một số vấn đề cần cải thiện ứng dụng :

* **Tính năng tìm kiếm**: Hệ thống hiện tại chưa tối ưu chức năng tìm kiếm hoặc lọc sản phẩm theo nhiều tiêu chí như giá cả, danh mục, hoặc đánh giá của khách hàng.
* **Tích hợp thanh toán trực tuyến**: Chưa có tính năng kết nối với các cổng thanh toán như thẻ ngân hàng, ví điện tử, hoặc mã QR.
* **Quản lý kho hàng**: Thiếu chức năng quản lý số lượng tồn kho chi tiết khi phát sinh đơn hàng.
* **Chức năng bảo mật**: Cần tăng cường bảo mật thông tin khách hàng và dữ liệu nhạy cảm (như mật khẩu, thông tin thanh toán).
* **Tối ưu giao diện người dùng**: Một số giao diện có thể cần cải thiện về mặt trải nghiệm người dùng (UX) để giúp người dùng dễ dàng thao tác hơn.
* **Thông báo thời gian thực**: Hệ thống chưa có tính năng gửi thông báo cho khách hàng về tình trạng đơn hàng hoặc cập nhật khuyến mãi.
* **Hỗ trợ đa nền tảng**: Chưa tối ưu để hoạt động tốt trên các thiết bị di động hoặc tablet

**MỤC LỤC**

[PHÂN CÔNG NHIỆM VỤ THÀNH VIÊN TRONG NHÓM 1](#_Toc185268537)

[PHẦN 1: MỞ ĐẦU 3](#_Toc185268538)

[1. LÝ DO CHỌN ĐỀ TÀI 3](#_Toc185268539)

[2. MỤC TIÊU CỦA ĐỀ TÀI 3](#_Toc185268540)

[3. ĐỐI TƯỢNG PHẠM VI NGHIÊN CỨU 3](#_Toc185268541)

[4. PHƯƠNG PHÁP NGHIÊN CỨU 4](#_Toc185268542)

[4.1. Nghiên cứu tài liệu 4](#_Toc185268543)

[4.2. Phân tích yêu cầu 4](#_Toc185268544)

[4.3. Thiết kế hệ thống 4](#_Toc185268545)

[4.4. Phát triển ứng dụng 5](#_Toc185268546)

[4.5. Kiểm thử 5](#_Toc185268547)

[4.6. Đánh giá và hoàn thiện 5](#_Toc185268548)

[5. CẤU TRÚC ĐỀ TÀI 5](#_Toc185268549)

[PHẦN 2: NỘI DUNG NGHIÊN CỨU 7](#_Toc185268550)

[CHƯƠNG I : CƠ SỞ LÝ THUYẾT 7](#_Toc185268551)

[1.1. GIỚI THIỆU 7](#_Toc185268552)

[1.2. CƠ SỞ LÝ THUYẾT 7](#_Toc185268553)

[1.2.1 Hệ thống thông tin quản lý (MIS) 8](#_Toc185268554)

[1.2.2Quản lý kho và chuỗi cung ứng 8](#_Toc185268555)

[1.2.3Quản lý quan hệ khách hàng (CRM) 8](#_Toc185268556)

[1.2.4 Hệ quản trị cơ sở dữ liệu (DBMS) 8](#_Toc185268557)

[1.2.5 Thiết kế giao diện người dùng (UI/UX) 8](#_Toc185268558)

[1.2.6 Nguyên lý phát triển phần mềm 9](#_Toc185268559)

[1.2.7 Tối ưu hóa quy trình bán hàng 9](#_Toc185268560)

[1.3 HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU 9](#_Toc185268561)

[1. 3.1 Kiến Trúc Của SQLite 10](#_Toc185268562)

[1.3.2 Các Thành Phần Chính Của SQLite 10](#_Toc185268563)

[1.3.3 Các Phương Thức Tương Tác Với SQLite 11](#_Toc185268564)

[1.3.4 Các Đặc Điểm Nổi Bật Của SQLite 11](#_Toc185268565)

[1.3.5 Ưu Nhược Điểm của SQLite 11](#_Toc185268566)

[1.3.6 Ứng Dụng SQLite trong Android 12](#_Toc185268567)

[1.3.7 Kết luận 12](#_Toc185268568)

[1.4 Công cụ phân tích và thiết kế 12](#_Toc185268569)

[1.4.1 Công Cụ Phân Tích 12](#_Toc185268570)

[1.5 Ngôn ngữ lập trình 13](#_Toc185268571)

[CHƯƠNG II: PHÂN TÍCH THIẾT KẾ CHƯƠNG TRÌNH 17](#_Toc185268572)

[2.1 MÔ TẢ BÀI TOÁN 17](#_Toc185268573)

[2.2 TRÌNH BÀY CÁC MÔ HÌNH, SƠ ĐỒ (UC,BFD, DFD, ERD) 18](#_Toc185268574)

[2.2.1 Mô hình UseCase – Sơ đồ ngữ cảnh 18](#_Toc185268575)

[2.2.2 Mô hình IFD – Sơ đồ 18](#_Toc185268576)

[2.2.3 Mô hình BFD – Sơ đồ phân rã chức năng 20](#_Toc185268577)

[2.2.4 Mô hình DFD – Sơ đồ luồng dữ liệu 20](#_Toc185268578)

[2.2.5 Mô hình ERD – Biểu đồ quan hệ thực thể 24](#_Toc185268579)

[CHƯƠNG III: DEMO CHƯƠNG TRÌNH 26](#_Toc185268580)

[3.1 MÔI TRƯỜNG THỰC NGHIỆM 26](#_Toc185268581)

[3.1.1 Công cụ phát triển 26](#_Toc185268582)

[3.1.2 Cơ sở dữ liệu SQLite (Local Database) 26](#_Toc185268583)

[3.1.3 Thiết bị thử nghiệm Giả lập Android (Emulator) 26](#_Toc185268584)

[3.1.4 Quản lý mã nguồn Dùng GitHub để quản lý mã nguồn. 26](#_Toc185268585)

[3.2 KẾT QUẢ APP 26](#_Toc185268586)

[3.2.1 Giao diện trang chủ 26](#_Toc185268587)

[3.2.2 Trang admin 27](#_Toc185268588)

[3.2.3 Trang quản lý sản phẩm 28](#_Toc185268589)

[3.2.4 Trang quản lý khách hàng 29](#_Toc185268590)

[3.2.5 Trang chi tiết sản phẩm 31](#_Toc185268591)

[3.2.6 Trang giỏ hàng 31](#_Toc185268592)

[3.2.7 Trang đăng ký 32](#_Toc185268593)

[3.2.8 Trang đăng nhập 33](#_Toc185268594)

[PHẦN 3: KẾT LUẬN 35](#_Toc185268595)

[1. NHỮNG VẤN ĐỀ ĐẠT ĐƯỢC 35](#_Toc185268596)

[2. NHỮNG VẤN ĐỀ CÒN HẠN CHẾ 35](#_Toc185268597)

[DANH MỤC HÌNH ẢNH 40](#_Toc185268598)

[NHẬN XÉT CỦA GIẢNG VIÊN 41](#_Toc185268599)

# DANH MỤC HÌNH ẢNH

[Hình 1. 1 Logo SQLite 10](#_Toc185254820)

[Hình 1. 2 Lập trình Android 14](#_Toc185254821)

[Hình 1. 3 Sơ đồ ngữ cảnh UseCase 18](#_Toc185254822)

[Hình 1. 4 Sơ đồ luồng thông tin - IFD 19](#_Toc185254823)

[Hình 1. 5 Sơ đồ phân rã chức năng 20](#_Toc185254824)

[Hình 1. 6 Sơ đồ luồng dữ liệu - DFD mức đỉnh 21](#_Toc185254825)

[Hình 1. 7 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng khách hàng 22](#_Toc185254826)

[Hình 1. 8 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng sản phẩm 23](#_Toc185254827)

[Hình 1. 9 Sơ đồ luồng dữ liệu - DFD dưới mức đỉnh chức năng mua hàng 24](#_Toc185254828)

[Hình 1. 10 Biểu đồ quan hệ thực thể - ERD 25](#_Toc185254829)

[Hình 1. 11 Giao diện trang chủ user 27](#_Toc185254830)

[Hình 1. 12 Giao diện trang admin 28](#_Toc185254831)

[Hình 1. 13 Giao diện trang quản lý sản phẩm 29](#_Toc185254832)

[Hình 1. 14 Giao diện trang quản lý khách hàng 30](#_Toc185254833)

[Hình 1. 15 Giao diện trang chi tiết sản phẩm 31](#_Toc185254834)

[Hình 1. 16 Giao diện trang giỏ hàng 32](#_Toc185254835)

[Hình 1. 17 Giao diện trang đăng ký người dùng 33](#_Toc185254836)

[Hình 1. 18 Giao diện trang đăng nhập người dùng 34](#_Toc185254837)

# NHẬN XÉT CỦA GIẢNG VIÊN