

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI TP. HỒ CHÍ MINH**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

# BÁO CÁO GIỮA KỲ HỌC PHẦN: PROJECT 1

ĐỀ TÀI: TÌM HIỂU VỀ THUẬT TOÁN TABPFN TRONG BÀI TOÁN DỰ ĐOÁN KHẢ NĂNG MUA HÀNG CỦA KHÁCH HÀNG

|  |  |  |
| --- | --- | --- |
| Giảng viên hướng dẫn: ThS. TRẦN PHONG NHÃ  Sinh viên thực hiện: | MSSV: |  |
| VÕ THÀNH HOÀNG PHÚC |  | 6351071056 |
| LÊ HOÀNG PHÚC |  | 6351071057 |
| ĐẶNG THỊ KIM THẢO |  | 6351071066 |
| Lớp : CÔNG NGHỆ THÔNG TIN  Khoá 63 |  |  |

Tp. Hồ Chí Minh, năm 2025

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI PHÂN HIỆU TẠI TP. HỒ CHÍ MINH**



**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

# BÁO CÁO GIỮA KỲ HỌC PHẦN: PROJECT 1

ĐỀ TÀI: TÌM HIỂU VỀ THUẬT TOÁN TABPFN TRONG BÀI TOÁN DỰ ĐOÁN KHẢ NĂNG MUA HÀNG CỦA KHÁCH HÀNG

|  |  |  |
| --- | --- | --- |
| Giảng viên hướng dẫn: ThS. TRẦN PHONG NHÃ  Sinh viên thực hiện: | MSSV: |  |
| VÕ THÀNH HOÀNG PHÚC |  | 6351071056 |
| LÊ HOÀNG PHÚC |  | 6351071057 |
| ĐẶNG THỊ KIM THẢO |  | 6351071066 |
| Lớp : CÔNG NGHỆ THÔNG TIN |  |  |
| Khoá 63 |  |  |

Tp. Hồ Chí Minh, năm 2025

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI **CỘNG HÒA XÃ HỘI CHỦ NGHIÃ VIỆT NAM**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH** Độc lập – Tự do – Hạnh phúc



# NHIỆM VỤ THIẾT KẾ BÁO CÁO GIỮA KỲ

BỘ MÔN: **CÔNG NGHỆ THÔNG TIN**

\*\*\*

### Mã sinh viên: Họ tên SV:

6351071056 Võ Thành Hoàng Phúc

6351071057 Lê Hoàng Phúc

6351071066 Đặng Thị Kim Thảo

**Khóa:** 63 **Lớp:** Công Nghệ Thông Tin

## Tên đề tài báo cáo cuối kì

**ỨNG DỤNG STYLO – QUẢN LÍ VÀ BÁN HÀNG TRỰC TUYẾN CHO CỬA HÀNG THỜI TRANG**

## Mục đích, yêu cầu

* 1. **Mục đích:**
     + Xây dựng nền tảng đa phương thức, phát triển một ứng dụng quản lý bán hàng thời trang đa nền tảng, giúp kết nối hiệu quả giữa khách hàng và quản trị viên.
     + Tự động hóa quy trình, thay thế việc quản lý thủ công (sản phẩm, đơn hàng, khách hàng) bằng hệ thống tự động để giảm thiểu sai sót và tăng hiệu quả kinh doanh.
     + Tích hợp trí tuệ nhân tạo: Ứng dụng các mô hình học máy (Machine Learning) để thông minh hóa hệ thống, giúp cá nhân hóa trải nghiệm người dùng và hỗ trợ ra quyết định.

## Yêu cầu:

* + - Tính đa nền tảng: Ứng dụng phải hoạt động mượt mà, nhất quán trên các thiết bị di động (Android/iOS)
    - Hiệu năng và giao diện: Giao diện thân thiện, thẩm mỹ; tốc độ xử lý dữ liệu nhanh chóng thông qua hệ thống Backend mạnh mẽ.
    - Tính bảo mật và ổn định: Đảm bảo an toàn thông tin người dùng (đăng nhập, mật khẩu) và lưu trữ dữ liệu tập trung, chính xác.
    - Độ chính xác của AI: Các mô hình gợi ý sản phẩm và kích cỡ cần đạt độ chính xác ≥ 80% so với thực tế.

## Nội dung và phạm vi đề tài

* 1. **Nội dung:**
     + Dành cho khách hàng: Đăng ký/đăng nhập, tìm kiếm sản phẩm, giỏ hàng, đặt hàng, đánh giá và cập nhật thông tin cá nhân.
     + Dành cho quản trị viên: Quản lý danh mục sản phẩm, quản lý đơn hàng, quản lý tài khoản người dùng và xem báo cáo thống kê doanh thu.
     + Gợi ý sản phẩm tương tự.
     + Tìm kiếm sản phẩm bằng hình ảnh.

## Phạm vi đề tài:

* + - Đối tượng nghiên cứu: Hệ thống quản lý bán hàng bao gồm các thực thể: khách hàng, sản phẩm, đơn hàng và quản trị viên.
    - Giới hạn ứng dụng: Tập trung giải quyết nhu cầu bán hàng trực tuyến cho các cửa hàng thời trang vừa và nhỏ.
    - Công nghệ tích hợp: Chỉ tập trung vào các mô hình học máy hỗ trợ mua sắm (gợi ý, tìm kiếm ảnh, dự đoán kích cỡ).

## Công nghệ, công cụ và ngôn ngữ lập trình

* 1. **Công nghệ:**

**-** Framework Frontend: Flutter (phát triển ứng dụng đa nền tảng).

**-** Framework Backend: ASP.NET Core Web API để xử lý nghiệp vụ.

**-** Hệ quản trị cơ sở dữ liệu: Microsoft SQL Server.

**-** Machine Learning: Sử dụng các thư viện Scikit-learn, PyTorch hoặc TensorFlow.

**-** Mô hình quản lý: Agile/Scrum.

* 1. **Công cụ:**

- Lập trình: Visual Studio, Visual Studio Code, Android Studio.

- Huấn luyện AI: Google Colab.

- Quản lý cơ sở dữ liệu: SQL Server Management Studio.

- Quản lý mã nguồn: GitHub.

* 1. **Ngôn ngữ lập trình:** Python, C#, Dart

## Các kết quả chính dự kiến sẽ đạt được và ứng dụng

* Kết quả dự kiến:
  + Hoàn thiện ứng dụng Stylo chạy trên Android/iOS với đầy đủ tính năng mua sắm và quản lý.
  + Tích hợp thành công AI hỗ trợ tìm kiếm bằng ảnh, gợi ý sản phẩm và phân tích cảm xúc khách hàng.
  + Dashboard thống kê trực quan giúp chủ cửa hàng nắm bắt doanh thu và hành vi khách hàng nhanh chóng.
  + Cung cấp giải pháp chuyển đổi số cho các shop thời trang vừa và nhỏ.
* Ứng dụng:
  + Giúp doanh nghiệp phân tích hành vi khách hàng, từ đó dự đoán xác suất mua hàng để đưa ra chiến lược marketing, khuyến mãi hoặc chăm sóc khách hàng phù hợp.

## Giáo viên và cán bộ hướng dẫn

Họ tên: TRẦN PHONG NHÃ

Đơn vị công tác: Bộ môn Công Nghệ Thông Tin – Trường Đại học Giao thông Vận tải phân hiệu tại TP. Hồ Chí Minh.

Điện thoại: Email:

|  |  |
| --- | --- |
| **Ngày 10 tháng 09 năm 2025**  **Trưởng BM Công nghệ Thông tin** | **Đã giao nhiệm vụ TKBCGK**  **Giáo viên hướng dẫn** |

|  |  |
| --- | --- |
| **ThS. Trần Phong Nhã** | **ThS. Trần Phong Nhã** |

Đã nhận nhiệm vụ TKBCGK

Sinh viên: Ký tên:

VÕ THÀNH HOÀNG PHÚC LÊ HOÀNG PHÚC

ĐẶNG THỊ KIM THẢO

Điện thoại: Email:

[6351071056@st.utc2.edu.vn](mailto:6351071056@st.utc2.edu.vn) [6351071057@st.utc2.edu.vn](mailto:6351071057@st.utc2.edu.vn) [6351071066@st.utc2.edu.vn](mailto:6351071066@st.utc2.edu.vn)

# LỜI CẢM ƠN

Qua thời gian học tập và rèn luyện tại Trường Đại học Giao thông Vận tải phân hiệu tại TP. Hồ Chí Minh, đến nay chúng em đã được trang bị những kĩ năng, kiến thức cơ bản để có thể hoàn thành được bài tập cuối kì do giảng viên giao.

Chúng em cảm ơn tập thể các thầy cô giáo Bộ môn Công Nghệ Thông Tin và các thầy cô thỉnh giảng đã giảng dạy, quan tâm và không ngần ngại dành thời gian để chỉ bài và giải đáp những thắc mắc của chúng em trong những tiết học và cả những lúc ngoài giờ.

Và chúng em cảm ơn thầy Thạc sĩ Trần Phong Nhã đã luôn quan tâm nhiệt tình hướng dẫn, giúp đỡ chúng em trong quá trình triển khai và thực hiện bài tập cuối kì. Thầy cũng luôn nhắc nhỡ, giúp đỡ mỗi khi chúng em gặp khó khăn, nhờ vậy mà chúng em đã hoàn thành bài tập cuối kì của nhóm mình đúng thời hạn được giao. Nếu không có sự hướng dẫn của thầy thì có lẽ chúng em đã khó có thể thực hiện được bài tập đúng theo mong muốn của mình.

Chúng em đã bỏ ra nhiều thời gian để tìm hiểu và trang bị thêm kiến thức nhằm phục vụ cho việc thực hiện ý tưởng, nhưng chắc chắn rằng chúng em sẽ không thể tránh khỏi những sai sót không đáng có vì kiến thức còn hạn chế. Chúng em hi vọng rằng sẽ nhận được những lời góp ý quý báo của thầy để có thể hoàn thiện ý tưởng của nhóm một cách tốt nhất có thể.

***TP. Hồ Chí Minh, ngày 26 tháng 10 năm 2025***

**Sinh viên thực hiện**

**Võ Thành Hoàng Phúc Lê Hoàng Phúc**

**Đặng Thị Kim Thảo**

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

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

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

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

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

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

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

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

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

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

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

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

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

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

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

***Tp. Hồ Chí Minh, ngày 26 tháng 10 năm 2025***

**Giáo viên hướng dẫn**

**ThS. Trần Phong Nhã**

# MỤC LỤC

[NHIỆM VỤ THIẾT KẾ BÁO CÁO GIỮA KỲ i](#_bookmark0)

[LỜI CẢM ƠN iv](#_bookmark1)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN v](#_bookmark2)

[DANH MỤC BẢNG BIỂU vii](#_bookmark3)

[DANH MỤC HÌNH ẢNH viii](#_bookmark4)

[CHƯƠNG 1. MỞ ĐẦU 1](#_bookmark5)

* 1. [Giới thiệu chung về đề tài 1](#_bookmark6)
     1. [Bối cảnh và sự phát triển của thương mại điện tử và nhu cầu cá nhân hóa 1](#_bookmark7)
     2. [Sự cần thiết của việc ứng dụng Trí tuệ Nhân tạo (AI) và Thuật toán tiên tiến 1](#_bookmark8)
  2. [Mục tiêu nghiên cứu 2](#_bookmark9)
     1. [Mục tiêu tổng quát 2](#_bookmark10)
     2. [Mục tiêu cụ thể 2](#_bookmark11)
  3. [Phạm vi nghiên cứu 4](#_bookmark12)
     1. [Đối tượng áp dụng 4](#_bookmark13)
     2. [Phạm vi chức năng 4](#_bookmark14)
     3. [Phạm vi công nghệ 5](#_bookmark15)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 7](#_bookmark16)

* 1. [Tổng quan về ngôn ngữ lập trình Python 7](#_bookmark17)
  2. [Google Colab 12](#_bookmark23)

[CHƯƠNG 3. KẾT QUẢ VÀ KIẾN NGHỊ 16](#_bookmark26)

* 1. [Kết quả đạt được 16](#_bookmark27)
  2. [Kiến nghị 17](#_bookmark29)

[PHỤ LỤC 19](#_bookmark30)

[Phụ lục 1: Hướng dẫn cài đặt 19](#_bookmark31)

[Phụ lục 2: Hướng dẫn sử dụng 19](#_bookmark32)

[TÀI LIỆU THAM KHẢO 20](#_bookmark33)

# DANH MỤC BẢNG BIỂU

[Bảng 3.1 So sánh mô hình 17](#_bookmark28)

# DANH MỤC HÌNH ẢNH

[Hình 2.1 Minh họa python 7](#_bookmark18)

[Hình 2,2 Code minh họa python 8](#_bookmark19)

[Hình 2.3 Hình ảnh mẫu python 9](#_bookmark20)

[Hình 2.4 Minh họa ứng dụng python 10](#_bookmark21)

[Hình 2.5 Minh họa selenium 11](#_bookmark22)

[Hình 2.6 Minh họa google colab 13](#_bookmark24)

[Hình 2.7 Mẫu google colab 14](#_bookmark25)

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| Tên thành viên | Phân công công việc | Đánh giá | Ký tên |
| Võ Thành Hoàng Phúc |  | 35% |  |
| Lê Hoàng Phúc |  | 30% |  |
| Đặng Thị Kim Thảo |  | 35% |  |

# CHƯƠNG 1. MỞ ĐẦU

### Giới thiệu chung về đề tài

### Bối cảnh và sự phát triển của thương mại điện tử và nhu cầu cá nhân hóa

Trong bối cảnh chuyển đổi số diễn ra mạnh mẽ trên toàn thế giới, thương mại điện tử trở thành một trong những lĩnh vực trọng điểm có tốc độ tăng trưởng nhanh nhất. Người tiêu dùng hiện đại ngày càng có xu hướng ưu tiên các nền tảng trực tuyến thay vì mua sắm truyền thống. Với sự phát triển của hạ tầng Internet, thiết bị di động và các hệ thống thanh toán điện tử, hành vi mua hàng của người dùng đã thay đổi sâu sắc, tạo nên một thị trường thương mại điện tử rộng lớn và cạnh tranh khốc liệt.

Riêng tại Việt Nam, giai đoạn 2020–2024 đã chứng kiến sự bùng nổ của các nền tảng TMĐT như Shopee, Lazada, Tiki,… kéo theo việc hàng nghìn cửa hàng thời trang, phụ kiện chuyển sang bán hàng trực tuyến để tiếp cận khách hàng hiệu quả hơn. Các cửa hàng thời trang vừa và nhỏ cũng bắt đầu nhận thấy rằng nếu chỉ kinh doanh theo kiểu truyền thống thì khó cạnh tranh và khó duy trì lượng khách ổn định. Việc xây dựng một hệ thống bán hàng trực tuyến chuyên nghiệp không chỉ giúp mở rộng thị trường mà còn giúp giảm chi phí vận hành, tối ưu hóa quy trình xử lý đơn hàng và cung cấp trải nghiệm mua sắm tốt hơn.

Tuy nhiên, cùng với sự phát triển mạnh mẽ đó, hành vi người dùng cũng trở nên phức tạp và khó dự đoán hơn. Khách hàng mong muốn được tương tác trong một môi trường đem đến cảm giác “riêng tư” và “hiểu họ”. Khái niệm **cá nhân hóa trải nghiệm** ngày càng quan trọng: người mua không muốn xem hàng loạt sản phẩm không liên quan mà muốn nhìn thấy những mặt hàng phù hợp đúng nhu cầu, đúng phong cách, đúng kích cỡ và sở thích của mình.

Đây chính là lý do mà các hệ thống thương mại điện tử hiện đại không chỉ dừng lại ở việc hiển thị sản phẩm, mà còn phải biết học hỏi từ hành vi khách hàng để đưa ra các gợi ý thông minh. Những nền tảng lớn coi AI và phân tích dữ liệu là chìa khóa để giữ chân khách hàng và tăng tỉ lệ mua hàng.

Trong khi đó, đa số cửa hàng thời trang vừa và nhỏ lại chưa có điều kiện triển khai các hệ thống hiện đại như vậy. Việc quản lý đơn hàng, sản phẩm, kho hàng, theo dõi doanh thu hay phân tích hành vi khách hàng hầu hết vẫn thực hiện thủ công hoặc sử dụng các phần mềm rời rạc. Điều này tạo ra sự chậm trễ, sai sót và không có khả năng hỗ trợ ra quyết định dựa trên dữ liệu.

Từ thực tế đó, nhóm quyết định xây dựng hệ thống **“Ứng dụng Stylo – Quản lý và bán hàng trực tuyến cho cửa hàng thời trang”**, vừa đáp ứng được các chức năng cơ bản của một hệ thống thương mại điện tử, vừa kết hợp thêm các yếu tố thông minh giúp nâng cao trải nghiệm người dùng và tối ưu hoạt động kinh doanh cho cửa hàng.

### Sự cần thiết của việc ứng dụng Trí tuệ Nhân tạo (AI) và Thuật toán tiên tiến

Cùng với sự phát triển của công nghệ dữ liệu và học máy, Trí tuệ nhân tạo đã trở thành công nghệ cốt lõi trong thương mại điện tử toàn cầu. Các hệ thống AI hiện nay có khả năng phân tích lượng dữ liệu khổng lồ, học từ các mẫu hành vi của người dùng và đưa ra quyết định hoặc dự đoán với độ chính xác cao. Việc ứng dụng AI trong TMĐT không còn là xu hướng mà đã trở thành tiêu chuẩn để cạnh tranh. Một số lý do khiến việc tích hợp AI vào hệ thống bán hàng thời trang trở nên vô cùng cần thiết.

Cá nhân hóa trải nghiệm mua sắm, trí tuệ nhân tạo giúp hệ thống hiểu rõ từng khách hàng hơn thông qua: Sản phẩm họ xem và tương tác nhiều nhất, thói quen mua sắm theo thời điểm, màu sắc, kích cỡ, phong cách thời trang họ ưu tiên, các đánh giá và phản hồi của họ. Từ đó, hệ thống đưa ra danh sách sản phẩm gợi ý phù hợp, giúp: Tăng khả năng khách hàng đưa sản phẩm vào giỏ, rút ngắn thời gian tìm kiếm, tăng mức độ hài lòng, gia tăng doanh số.

Tự động hóa quy trình vận hành: Thay vì quản trị viên phải phân tích thủ công doanh thu, sản phẩm nổi bật hay tỉ lệ chuyển đổi, AI có thể: Tự động phân tích dữ liệu bán hàng, phát hiện xu hướng sản phẩm sắp “hot”, đề xuất nhập hàng, phát hiện bất thường trong hành vi mua hàng. Điều này giúp quản trị viên tiết kiệm thời gian và ra quyết định chính xác hơn.

Tìm kiếm thông minh bằng hình ảnh: Nhiều khách hàng không biết phải mô tả sản phẩm bằng từ khóa như thế nào. Vì vậy, tìm kiếm bằng ảnh giúp: Người dùng chỉ cần chụp lại sản phẩm mình thích, hệ thống sử dụng CNN để tìm sản phẩm giống nhất trong cửa hàng, tăng hiệu quả tiếp cận sản phẩm ngay cả khi khách không biết cách gọi tên.

### Mục tiêu nghiên cứu

### 1.2.1 Mục tiêu tổng quát

Mục tiêu chính của đề tài là xây dựng một hệ thống bán hàng thời trang trực tuyến hoạt động trên nền tảng di động, thân thiện với người dùng, tích hợp đầy đủ chức năng từ tìm kiếm sản phẩm, xem chi tiết, giỏ hàng đến đặt hàng và đánh giá sản phẩm. Hệ thống đồng thời phải hỗ trợ quản trị viên quản lý sản phẩm, kho hàng, đơn hàng, người dùng và thống kê doanh thu một cách trực quan và dễ sử dụng.

Bên cạnh đó, đề tài hướng tới việc tích hợp các mô hình trí tuệ nhân tạo nhằm mang đến trải nghiệm cá nhân hóa, tăng hiệu quả kinh doanh và giảm tải cho quản trị viên.

### Mục tiêu cụ thể

1. **Mục tiêu về mặt kỹ thuật và kiến trúc hệ thống**

**-** Xây dựng backend theo kiến trúc ASP.NET Core Web API đảm bảo bảo mật, hiệu suất và dễ mở rộng.

**-** Thiết kế cơ sở dữ liệu SQL Server có cấu trúc rõ ràng, hỗ trợ truy vấn tối ưu.

**-** Phát triển frontend bằng Flutter với giao diện hiện đại, dễ dùng, đồng bộ trên Android.

**-** Tích hợp các mô hình AI: Gợi ý sản phẩm (CBF), Tìm kiếm bằng ảnh (CNN)

### Mục tiêu về mặt chức năng và nghiệp vụ

**-** Cung cấp đầy đủ quy trình mua sắm: xem → tìm → giỏ → thanh toán → theo dõi.

**-** Mang lại cho khách hàng trải nghiệm cá nhân hóa thông minh hơn.

**-** Tạo môi trường quản lý hiệu quả cho quản trị viên.

**-** Hỗ trợ thống kê chính xác để chủ cửa hàng ra quyết định.

**-** Đảm bảo hệ thống hoạt động mượt, an toàn, ổn định.

### Phạm vi nghiên cứu

- Hệ thống được xây dựng hướng tới: Khách hàng mua sắm thời trang trên thiết bị di động, quản trị viên cửa hàng, bộ phận xử lý dữ liệu và AI của hệ thống. Hệ thống tập trung vào quy mô cửa hàng vừa và nhỏ nhưng có khả năng mở rộng về lâu dài.

### Đối tượng áp dụng

- Hệ thống được xây dựng hướng tới: Khách hàng mua sắm thời trang trên thiết bị di động, quản trị viên cửa hàng, bộ phận xử lý dữ liệu và AI của hệ thống. Hệ thống tập trung vào quy mô cửa hàng vừa và nhỏ nhưng có khả năng mở rộng về lâu dài.

### Phạm vi chức năng

1. **Chức năng cơ bản:**

* Quản lý Người dùng và Bảo mật: Đăng ký, Đăng nhập, Khôi phục mật khẩu, Cập nhật thông tin cá nhân.
* Quản lý Sản phẩm: Hiển thị, Tìm kiếm, Lọc, Xem chi tiết sản phẩm, Giỏ hàng, Yêu thích.
* Quản lý Giao dịch: Quy trình Thanh toán (giả định), Theo dõi trạng thái đơn hàng, Gửi đánh giá/nhận xét.

### Chức năng quản trị viên:

* Quản lý Danh mục và Sản phẩm: CRUD sản phẩm và danh mục, Quản lý kho hàng.
* Quản lý Đơn hàng: Theo dõi và cập nhật trạng thái đơn hàng.
* Thống kê và Báo cáo: Dashboard trực quan hóa dữ liệu về Doanh thu, Đơn hàng, Hành vi khách hàng.

### Chức năng trí tuệ nhân tạo:

**-** Gợi ý sản phẩm thông minh.

**-** Tìm kiếm bằng hình ảnh.

### - Tìm kiếm bằng hình ảnh.Phạm vi công nghệ

1. **Công nghệ giao diện**

* Ngôn ngữ lập trình: Dart.
* Framework/Thư viện: Flutter.

### Công nghệ backend

* Ngôn ngữ lập trình: C#.
* Framework/Thư viện: ASP.NET Core Web API (RESTful API).
* Công cụ phát triển: Visual Studio.

### Công nghệ trí tuệ nhân tạo

* Ngôn ngữ lập trình: Python.
* Framework/Thư viện:
  + CBF: Là công nghệ cốt lõi để triển khai mô hình gợi ý sản phẩm cơ bản cho hệ thống
  + TensorFlow/PyTorch: Để xây dựng và huấn luyện các mô hình học sâu khác như CNN để tìm kiếm sản phẩm bằng hình ảnh.
* Môi trường: Google Colab.

### Công nghệ dữ liệu

* Hệ quản trị cơ sở dữ liệu: Microsoft SQL Server.
* Công cụ quản lý: SQL Server Management Studio (SSMS).

### Phương pháp phát triển

* Mô hình phát triển: Agile/Scrum.

# CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### Tổng quan về ngôn ngữ lập trình Python và FastAPI

2.1.1 Giới thiệu về Python

Python là một ngôn ngữ lập trình thông dịch, dễ đọc và dễ hiểu. Nền tảng nổi tiếng với cú pháp đơn giản và được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau. Điển hình như phát triển website, phân tích dữ liệu, trí tuệ nhân tạo và nhiều ứng dụng khác. Python có cú pháp linh hoạt và cấu trúc dữ liệu mạnh mẽ, công nghệ được hỗ trợ bởi một cộng đồng lớn. Điều này đã mang đến các thư viện và framework phong phú mà người dùng có thể sử dụng để xây dựng các ứng dụng phức tạp. Python cũng là một trong những ngôn ngữ phổ biến cho người mới học lập trình nhờ vào tính linh hoạt của nó.



Hình 2.1 Ảnh minh họa về Python

Những lợi ích của Python:

* Cú pháp đơn giản và dễ đọc, phù hợp cho người mới học lập trình và cũng dễ dàng cho những người có kinh nghiệm.
* Python được sử dụng trong nhiều lĩnh vực như phát triển website, phân tích dữ liệu, trí tuệ nhân tạo, khoa học dữ liệu và nhiều ứng dụng khác. Điều này làm cho Python trở thành một ngôn ngữ lập trình linh hoạt và tiện lợi.
* Cộng đồng Python khá lớn và luôn hoạt động tích cực nhằm cung cấp nhiều thư viện, framework hữu ích. Người dùng sẽ được nhận sự hỗ trợ từ cộng đồng thông qua tài liệu, diễn đàn và các nguồn thông tin trực tuyến.
* Python cung cấp nhiều thư viện mạnh mẽ cho phân tích dữ liệu như Pandas, NumPy, và Matplotlib. Bộ ngôn ngữ giúp người dùng xử lý và thể hiện dữ liệu một cách hiệu quả.
* Python có thể chạy trên nhiều hệ điều hành khác nhau và trở thành một trong những ngôn ngữ lập trình chéo phổ biến.
* Ngôn ngữ Python là mã nguồn mở và miễn phí, cho phép người dùng tự do sử dụng, phân phối và thay đổi phiên bản của nó.

Phương thức hoạt động chính của Python:

* Python có nhiều framework phát triển web phổ biến, ví dụ như Django, Flask, Pyramid và Fast API. Đây là các công cụ tối ưu giúp lập trình viên phát triển ứng dụng web từ phía máy chủ một cách dễ dàng và hiệu quả.
* Các framework này cung cấp nhiều tính năng như xử lý URL, tương tác cơ sở dữ liệu, quản lý phiên và tạo giao diện người dùng. Django được xem là một trong những framework phát triển web phổ biến nhất trong cộng đồng Python. Nền tảng cung cấp sẵn các tính năng linh hoạt và nhiều công cụ hữu ích giúp cho việc xây dựng ứng dụng web phức tạp trở nên thuận lợi hơn.
* Flask là một framework nhỏ gọn và linh hoạt hơn nên nó sẽ phù hợp cho việc xây dựng ứng dụng web từ nhỏ đến trung bình. Công nghệ cho phép người dùng tùy chỉnh nhiều hơn trên một hệ thống. Pyramid cũng là tiện ích cung cấp chế độ hoạt động mạnh mẽ, đặc biệt là việc xây dựng ứng dụng lớn và phức tạp.
* Fast API là một framework mới nhưng có tốc độ làm việc khá nhanh. Nền tảng được sử dụng để xây dựng các ứng dụng web hiệu suất cao và API. Tùy thuộc vào yêu cầu cụ thể của dự án mà người phát triển nên lựa chọn framework phù hợp với ngôn ngữ Python.

Ứng dụng trong khoa học dữ liệu: Ngôn ngữ Python được sử dụng phổ biến trong lĩnh vực khoa học dữ liệu và máy học. Cách sử dụng các thư viện mạnh mẽ trong Python giúp cho việc phân tích dữ liệu hoặc triển khai mô hình lập trình trở nên dễ dàng hơn.

Dưới đây là một số thư viện quan trọng mà người làm việc trong lĩnh vực này thường sử dụng:

* NumPy: Thư viện mạnh mẽ cho các phép toán trên mảng đa chiều và ma trận, cung cấp các hàm để thao tác dữ liệu số học một cách hiệu quả
* Pandas: Pandas cung cấp cấu trúc dữ liệu và công cụ phân tích dữ liệu mạnh mẽ, đặc biệt là trong việc làm việc với dữ liệu có cấu trúc như bảng và chuỗi thời gian
* Matplotlib và Seaborn: Đây là các thư viện hỗ trợ vẽ đồ thị và trực quan hóa dữ liệu một cách dễ dàng và mạnh mẽ
* Scikit-learn: Scikit-learn là một trong những thư viện học máy phổ biến nhất trong Python, cung cấp nhiều thuật toán lập trình và công cụ cho tiền xử lý dữ liệu, đánh giá mô hình và tinh chỉnh tham số
* TensorFlow và PyTorch: Đây là hai framework phổ biến cho việc triển khai mô hình học sâu (deep learning). Cả hai đều cung cấp API mạnh mẽ để xây dựng và huấn luyện mạng nơ-ron

Những thư viện này cung cấp hệ sinh thái để phân tích dữ liệu, xây dựng mô hình và triển khai ứng dụng trong lĩnh vực khoa học dữ liệu. Python được xem là ngôn ngữ lập trình hàng đầu trong lĩnh vực này.

2.1.2 FastAPI Framework

Để tự động hóa kiểm thử phần mềm bằng Python cần sử dụng nhiều thư viện và framework mạnh mẽ. Dưới đây giới thiệu một số công cụ và thư viện phổ biến được tích hợp trong quá trình tự động hóa kiểm thử phần mềm bằng Python:

2.2 Ảnh minh họa về selenium

Selenium: Selenium là một công cụ tự động hóa trình duyệt web, cho phép làm tự động các tác vụ như điều hướng, điền thông tin và kiểm tra các yếu tố trên trang web. Selenium có thể được sử dụng để viết kịch bản kiểm thử tự động cho ứng dụng web.

PyTest: PyTest là một framework kiểm thử mạnh mẽ và dễ sử dụng cho Python. Công nghệ cung cấp các cách linh hoạt để viết các bài kiểm tra và hỗ trợ chạy tự động các bài kiểm tra, báo cáo kết quả.

Robot Framework: Robot Framework là một framework kiểm thử tự động mã nguồn mở cho việc tự động hóa kiểm thử phần mềm. Nền tảng cung cấp các thư viện mở rộng cho nhiều mục đích, bao gồm kiểm thử giao diện người dùng, dịch vụ web và ứng dụng di động. Unittest: Unittest là một framework kiểm thử chuẩn có sẵn trong thư viện chuẩn của Python. Tiện ích cung cấp giải pháp tổ chức các bài kiểm tra và thực hiện kiểm thử tự động.

### Google Colab

Google Colab là một dịch vụ miễn phí từ Google cho phép bạn viết và chia sẻ mã Python. Người dùng có thể lưu trữ và chạy Notebook trên đám mây thông qua trình duyệt mà không cần kích hoạt cấu hình bộ đệm phức tạp cho máy tính cá nhân.



*Hình 2.6 Minh họa google colab*

Nền tảng cũng cung cấp GPU và TPU để huấn luyện mô hình học và thực hiện các yêu cầu tính toán lớn. Điều này có tác dụng mở rộng sức mạnh xử lý trong điện toán và giảm thời gian chờ đợi khi thực hiện các nhiệm vụ tính toán nặng nề.

Google Colab được phát triển bởi nhóm Google Research nhằm hỗ trợ quá trình nghiên cứu và phát triển trong lĩnh vực học máy tính và khoa học dữ liệu. Công nghệ xuất phát từ nền tảng Jupyter Notebook phổ biến. Sau đó, Colab được kết hợp thêm khả năng tính toán trên tiện ích đám mây vô cùng mạnh mẽ của Google.

Từ đó, quá trình phát triển Google Colab tập trung vào mục tiêu cung cấp một môi trường lập trình linh hoạt và mạnh mẽ. Nền tảng hỗ trợ đa dạng công nghệ tính toán như Python, CUDA, Tensorflow. Kèm theo đó có nhiều thư viện khoa học máy tính và khoa học dữ liệu khác. Đồng thời, dịch cụ còn cung cấp GPU và TPU để tăng cường sức mạnh tính toán.

Các tính năng chính của google colab

* Sử dụng Jupyter Notebooks trực tuyến: Google Colab cho phép tạo và chạy các Jupyter Notebooks trực tuyến mà không cần cài đặt môi trường phát triển phức tạp trên máy tính cá nhân. Giao diện sử dụng tương tự như Jupyter Notebook truyền thống với các cell cho phép thực thi mã Python hoặc viết markdown để tạo nội dung hướng dẫn.
* Khả năng chia sẻ và cộng tác: Người dùng có thể chia sẻ notebook với những người khác để cùng làm việc trên cùng một notebook, tạo điều kiện thuận lợi cho việc học tập và làm việc nhóm. Các tính năng như bình luận và chế độ chỉnh sửa đồng thời giúp tăng tính tương tác và hiệu quả của quá trình cộng tác.
* Dùng GPU và TPU miễn phí: Google Colab cung cấp truy cập miễn phí đến GPU và TPU, đặc biệt hữu ích cho các tác vụ tính toán nặng về mặt số học, đặc biệt là trong lĩnh vực học máy và deep learning. Việc sử dụng các card GPU hoặc TPU có sẵn giúp tăng tốc độ xử lý và huấn luyện mô hình so với việc sử dụng CPU thông thường.
* Lưu trữ dữ liệu trên Google Drive và tích hợp Google Cloud: Người dùng có thể truy cập và lưu trữ dữ liệu trực tiếp từ Google Drive, tạo điều kiện thuận lợi cho việc làm việc với tập tin dữ liệu lớn. Tích hợp với Google Cloud Platform (GCP) cũng cho phép sử dụng các dịch vụ như BigQuery, Cloud Storage, và các API khác từ GCP trong quá trình làm việc

Ứng dụng của google colab

* Học máy và khoa học dữ liệu: Google Colab là một công cụ mạnh mẽ cho việc học máy và nghiên cứu khoa học dữ liệu. Với khả năng sử dụng miễn phí GPU và TPU, người dùng có thể xây dựng, huấn luyện và kiểm định các mô hình máy học một cách hiệu quả. Google Colab cung cấp môi trường Jupyter Notebook trực tuyến, cho phép viết và chạy code Python, đồng thời cung cấp các thư viện phổ biến như TensorFlow, Keras, và scikit-learn.
* Phát triển ứng dụng AI và ML: Với sức mạnh của Colab's GPU và TPU, các nhà phát triển có thể tận dụng để phát triển ứng dụng trí tuệ nhân tạo (AI) và máy học (ML). Điều này bao gồm việc xây dựng và đánh giá các mô hình học máy, tạo ứng dụng dự đoán, xử lý ngôn ngữ tự nhiên, nhận diện hình ảnh và nhiều ứng dụng AI khác.
* Nghiên cứu và phân tích dữ liệu: Các nhà nghiên cứu và chuyên gia phân tích dữ liệu thường sử dụng Google Colab để thực hiện phân tích số liệu, xử lý dữ liệu lớn, và thực hiện các thí nghiệm khoa học. Colab cung cấp khả năng tích hợp dữ liệu từ Google Drive hoặc các nguồn khác, cho phép họ thực hiện các phân tích phức tạp mà không cần tải về dữ liệu về máy cục bộ.
* Giáo dục và đào tạo: Google Colab cũng được sử dụng rộng rãi trong lĩnh vực giáo dục và đào tạo. Nó cung cấp môi trường lập trình Python trực tuyến mà không yêu cầu cài đặt, giúp sinh viên và giáo viên dễ dàng tiếp cận và chia sẻ các notebook học tập. Điều này làm tăng tính tương tác và khả năng học tập cộng đồng trong việc chia sẻ kiến thức và dự án.

## 2.3 Flutter

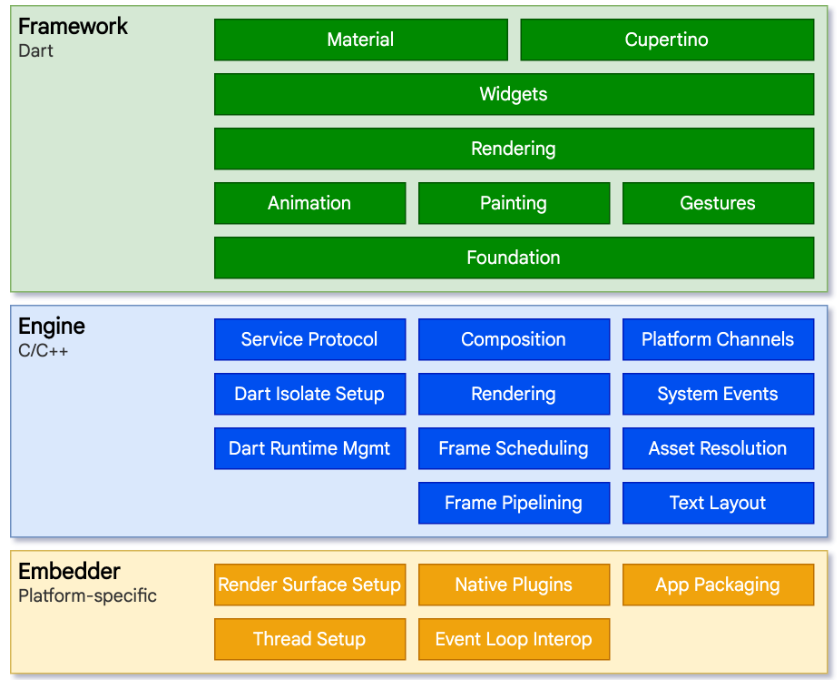
Flutter là một bộ công cụ giao diện người dùng đa nền tảng được thiết kế để cho phép tái sử dụng mã trên các hệ điều hành như iOS, Android, web và máy tính để bàn, đồng thời cho phép các ứng dụng giao tiếp trực tiếp với các dịch vụ nền tảng bên dưới. Mục tiêu là giúp các nhà phát triển tạo ra các ứng dụng hiệu suất cao, hoạt động mượt mà trên các nền tảng khác nhau, chấp nhận sự khác biệt khi cần thiết trong khi vẫn chia sẻ càng nhiều mã càng tốt.

Trong quá trình phát triển, ứng dụng Flutter chạy trên máy ảo (VM) cung cấp tính năng tải lại nóng (hot reload) có trạng thái mà không cần biên dịch lại toàn bộ. Khi phát hành, ứng dụng Flutter được biên dịch trực tiếp thành mã máy, có thể là lệnh Intel x64 hoặc ARM, hoặc thành JavaScript nếu nhắm mục tiêu đến web. Framework này là mã nguồn mở, với giấy phép BSD cho phép sử dụng rộng rãi, và có một hệ sinh thái phát triển mạnh mẽ gồm các gói bên thứ ba bổ sung chức năng cho thư viện cốt lõi.

Tổng quan này được chia thành nhiều phần:

* Mô hình lớp : Các thành phần cấu tạo nên Flutter.
* Giao diện người dùng tương tác : Một khái niệm cốt lõi trong phát triển giao diện người dùng Flutter.
* Giới thiệu về widget : Các khối xây dựng cơ bản của giao diện người dùng.
* Quá trình hiển thị : Flutter chuyển đổi mã giao diện người dùng thành các điểm ảnh như thế nào.
* Tổng quan về các trình nhúng nền tảng : Mã cho phép hệ điều hành di động và máy tính để bàn chạy các ứng dụng Flutter.
* Tích hợp Flutter với các mã nguồn khác : Thông tin về các kỹ thuật khác nhau có sẵn cho ứng dụng Flutter.
* Hỗ trợ cho web : Kết luận về các đặc điểm của Flutter trong môi trường web.

Flutter được thiết kế như một hệ thống phân lớp, có khả năng mở rộng. Nó tồn tại dưới dạng một chuỗi các thư viện độc lập, mỗi thư viện phụ thuộc vào lớp nền tảng bên dưới. Không có lớp nào có quyền truy cập đặc quyền vào lớp bên dưới, và mọi phần của khung phần mềm đều được thiết kế để tùy chọn và có thể thay thế.



Hình 2.8 Các lớp kiến trúc

Đối với hệ điều hành nền tảng, các ứng dụng Flutter được đóng gói theo cùng một cách như bất kỳ ứng dụng gốc nào khác. Một trình nhúng dành riêng cho nền tảng cung cấp điểm truy cập; phối hợp với hệ điều hành nền tảng để truy cập các dịch vụ như bề mặt hiển thị, khả năng truy cập và nhập liệu; và quản lý vòng lặp sự kiện thông báo. Trình nhúng được viết bằng ngôn ngữ phù hợp với nền tảng: hiện tại là Java và C++ cho Android, Swift và Objective-C/Objective-C++ cho iOS và macOS, và C++ cho Windows và Linux. Sử dụng trình nhúng, mã Flutter có thể được tích hợp vào một ứng dụng hiện có dưới dạng một mô-đun, hoặc mã đó có thể là toàn bộ nội dung của ứng dụng. Flutter bao gồm một số trình nhúng cho các nền tảng mục tiêu phổ biến, nhưng cũng tồn tại các trình nhúng khác.

Cốt lõi của Flutter là **công cụ Flutter (Flutter engine )**, chủ yếu được viết bằng C++ và hỗ trợ các thành phần cơ bản cần thiết để vận hành tất cả các ứng dụng Flutter. Công cụ này chịu trách nhiệm tạo ảnh bitmap cho các cảnh ghép nối mỗi khi cần vẽ một khung hình mới. Nó cung cấp triển khai cấp thấp của API cốt lõi của Flutter, bao gồm bố cục văn bản đồ họa, nhập/xuất tệp và mạng, môi trường chạy Dart và chuỗi công cụ biên dịch.

Công cụ này được tích hợp vào khung Flutter thông qua thư viện dart:ui, thư viện này bao bọc mã C++ cơ bản trong các lớp Dart. Thư viện này cung cấp các thành phần cơ bản nhất, chẳng hạn như các lớp điều khiển hệ thống nhập liệu, đồ họa và hiển thị văn bản.

Thông thường, các nhà phát triển tương tác với Flutter thông qua framework Flutter, một framework hiện đại, phản ứng nhanh được viết bằng ngôn ngữ Dart. Nó bao gồm một bộ thư viện nền tảng, bố cục và cơ bản phong phú, được cấu thành từ nhiều lớp.

[Các lớp nền tảng](https://api.flutter.dev/flutter/foundation/foundation-library.html) cơ bản và các dịch vụ khối xây dựng như [hoạt hình](https://api.flutter.dev/flutter/animation/animation-library.html)**,**[vẽ](https://api.flutter.dev/flutter/painting/painting-library.html) và [cử chỉ](https://api.flutter.dev/flutter/gestures/gestures-library.html) cung cấp các khái niệm trừu tượng thường được sử dụng trên nền tảng cơ bản đó.

Lớp [hiển thị](https://api.flutter.dev/flutter/rendering/rendering-library.html) cung cấp một lớp trừu tượng để xử lý bố cục. Với lớp này, bạn có thể xây dựng một cây các đối tượng có thể hiển thị. Bạn có thể thao tác các đối tượng này một cách linh hoạt, với cây tự động cập nhật bố cục để phản ánh những thay đổi của bạn.

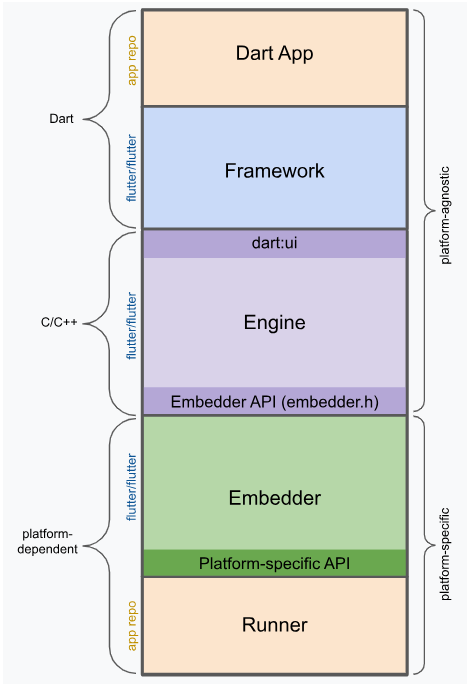
Lớp [widget](https://api.flutter.dev/flutter/widgets/widgets-library.html) là một sự trừu tượng hóa về thành phần. Mỗi đối tượng hiển thị trong lớp hiển thị đều có một lớp tương ứng trong lớp widget. Ngoài ra, lớp widget cho phép bạn định nghĩa các tổ hợp lớp mà bạn có thể tái sử dụng. Đây là lớp mà mô hình lập trình phản ứng được giới thiệu.

Các thư viện [Material](https://api.flutter.dev/flutter/material/material-library.html) và [Cupertino](https://api.flutter.dev/flutter/cupertino/cupertino-library.html) cung cấp bộ điều khiển toàn diện sử dụng các thành phần cơ bản của lớp widget để triển khai ngôn ngữ thiết kế Material hoặc iOS.

Khung Flutter tương đối nhỏ; nhiều tính năng cấp cao mà các nhà phát triển có thể sử dụng được triển khai dưới dạng các gói, bao gồm các plugin nền tảng như camera và webview, cũng như các tính năng độc lập với nền tảng như ký tự , HTTP và hoạt ảnh được xây dựng dựa trên các thư viện Dart và Flutter cốt lõi. Một số gói này đến từ hệ sinh thái rộng lớn hơn, bao gồm các dịch vụ như thanh toán trong ứng dụng , xác thực Apple.

Phần còn lại của tổng quan này sẽ đi sâu vào từng lớp, bắt đầu với mô hình phản ứng trong phát triển giao diện người dùng. Sau đó, chúng ta sẽ mô tả cách các widget được kết hợp với nhau và chuyển đổi thành các đối tượng có thể được hiển thị như một phần của ứng dụng. Chúng ta sẽ mô tả cách Flutter tương tác với các mã khác ở cấp độ nền tảng, trước khi đưa ra tóm tắt ngắn gọn về sự khác biệt giữa hỗ trợ web của Flutter so với các nền tảng khác.

Sơ đồ sau đây cung cấp cái nhìn tổng quan về các thành phần tạo nên một ứng dụng Flutter thông thường được tạo ra bởi flutter create. Nó cho thấy vị trí của Flutter Engine trong cấu trúc này, làm nổi bật các ranh giới API và xác định các kho lưu trữ nơi các thành phần riêng lẻ được đặt. Chú thích bên dưới làm rõ một số thuật ngữ thường được sử dụng để mô tả các thành phần của một ứng dụng Flutter.



Hình 2.9 Cấu trúc của 1 ứng dụng

**Ứng dụng Dart:** Sắp xếp các widget thành giao diện người dùng mong muốn, thực thi logic nghiệp vụ, thuộc sở hữu của nhà phát triển ứng dụng.

**Khung phần mềm:** Cung cấp API cấp cao hơn để xây dựng các ứng dụng chất lượng cao (ví dụ: widget, kiểm thử va chạm, phát hiện cử chỉ, khả năng truy cập, nhập văn bản). Tổng hợp cây widget của ứng dụng thành một khung cảnh.

Công cụ: Chịu trách nhiệm chuyển đổi các cảnh ghép thành ảnh bitmap. Cung cấp triển khai cấp thấp các API cốt lõi của Flutter (ví dụ: đồ họa, bố cục văn bản, môi trường chạy Dart). Nó cung cấp chức năng của mình cho framework bằng cách sử dụng API dart:ui. Tích hợp với một nền tảng cụ thể bằng cách sử dụng API Embedder của Engine.

Trình nhúng: Phối hợp với hệ điều hành cơ bản để truy cập các dịch vụ như hiển thị bề mặt, khả năng truy cập và nhập liệu, quản lý vòng lặp sự kiện, cung cấp API dành riêng cho từng nền tảng để tích hợp Embedder vào các ứng dụng.

Người chạy bộ: Công cụ này kết hợp các thành phần được cung cấp bởi API dành riêng cho nền tảng của Embedder thành một gói ứng dụng có thể chạy trên nền tảng mục tiêu, Một phần của mẫu ứng dụng được tạo bởi flutter create, thuộc sở hữu của nhà phát triển ứng dụng.

Trong trường hợp có thể, số lượng khái niệm thiết kế được giữ ở mức tối thiểu trong khi vẫn cho phép tổng vốn từ vựng lớn. Ví dụ, trong lớp widget, Flutter sử dụng cùng một khái niệm cốt lõi (a Widget) để biểu diễn việc vẽ lên màn hình, bố cục (định vị và kích thước), tương tác người dùng, quản lý trạng thái, chủ đề, hoạt ảnh và điều hướng. Trong lớp hoạt ảnh, một cặp khái niệm, Animations và Tweens, bao phủ hầu hết không gian thiết kế. Trong lớp hiển thị, RenderObjects được sử dụng để mô tả bố cục, vẽ, kiểm tra va chạm và khả năng truy cập. Trong mỗi trường hợp này, vốn từ vựng tương ứng cuối cùng đều rất lớn: có hàng trăm widget và đối tượng hiển thị, và hàng chục loại hoạt ảnh và chuyển động trung gian.

Cấu trúc phân cấp lớp được thiết kế đơn giản và rộng rãi nhằm tối đa hóa số lượng kết hợp có thể, tập trung vào các widget nhỏ, có thể kết hợp với nhau, mỗi widget thực hiện tốt một chức năng duy nhất. Các tính năng cốt lõi được trừu tượng hóa, ngay cả các tính năng cơ bản như khoảng cách lề và căn chỉnh cũng được triển khai dưới dạng các thành phần riêng biệt thay vì được tích hợp vào lõi. (Điều này cũng trái ngược với các API truyền thống hơn, nơi các tính năng như khoảng cách lề được tích hợp vào lõi chung của mọi thành phần bố cục.) Vì vậy, ví dụ, để căn giữa một widget, thay vì điều chỉnh một Alignthuộc tính giả định, bạn sẽ bọc nó trong một Center widget khác.

Có các widget dành cho khoảng cách lề, căn chỉnh, hàng, cột và lưới. Các widget bố cục này không có hình ảnh trực quan riêng. Thay vào đó, mục đích duy nhất của chúng là kiểm soát một khía cạnh nào đó trong bố cục của một widget khác. Flutter cũng bao gồm các widget tiện ích tận dụng cách tiếp cận theo thành phần này.

**CHƯƠNG 3: THUẬT TOÁN VÀ MÔ HÌNH**

**3.1 Tổng quan**

Trong chương này, nhóm sẽ trình bày tổng quan về hai kỹ thuật cốt lõi được áp dụng trong hệ thống: **thuật toán gợi ý sản phẩm dựa trên nội dung Content-Based Filtering (CBF)** và **mạng nơ-ron tích chập Convolutional Neural Network (CNN)** dùng cho chức năng tìm kiếm sản phẩm bằng hình ảnh. Đây là hai thành phần quan trọng góp phần tạo nên tính thông minh, khả năng cá nhân hóa và sự tiện lợi của ứng dụng Stylo.

Việc ứng dụng các thuật toán này không chỉ nâng cao trải nghiệm người dùng mà còn giúp cửa hàng thời trang phân tích dữ liệu một cách hiệu quả, từ đó tối ưu chiến lược kinh doanh. Dưới đây là trình bày chi tiết về từng thuật toán, nguyên lý hoạt động và cách nhóm áp dụng vào hệ thống.

**3.2 Thuật toán Content-Based Filtering (CBF)**

Khái niệm: CBF là thuật toán gợi ý sản phẩm dựa trên **đặc điểm của sản phẩm** và **lịch sử hành vi của người dùng.**

CBF phân tích: Thuộc tính sản phẩm: loại sản phẩm, kiểu, màu sắc, chất liệu, thương hiệu…, mô tả sản phẩm, từ khóa liên quan, lịch sử xem hoặc mua của người dùng. Sau đó tính độ tương đồng giữa: Sản phẩm mà người dùng tương tác và các sản phẩm khác trong cửa hàng. Kết quả là hệ thống đưa ra danh sách gợi ý có mức độ phù hợp cao.

Thuật toán gợi ý dựa trên nội dung CBF là một trong những phương pháp nền tảng của hệ thống gợi ý và được áp dụng rộng rãi trong thương mại điện tử, đặc biệt khi hệ thống cần hiểu rõ đặc điểm của từng sản phẩm để phục vụ mục tiêu cá nhân hóa trải nghiệm người dùng. Thuật toán này hoạt động dựa trên nguyên tắc phân tích nội dung sản phẩm, tức là xem xét những đặc trưng mô tả sản phẩm như màu sắc, chất liệu, kiểu dáng, phong cách, danh mục, mô tả chi tiết hoặc các thuộc tính liên quan khác, sau đó chuyển các đặc trưng này thành dạng vector số hóa bằng các kỹ thuật như Bag of Words, TF-IDF hoặc Word Embedding. Mỗi sản phẩm vì thế được biểu diễn như một điểm dữ liệu trong không gian nhiều chiều, mang theo toàn bộ “dấu hiệu nhận dạng” về nội dung của chính sản phẩm đó.

Trong khi đó, để đưa ra các gợi ý phù hợp cho người dùng, thuật toán cần xác định hồ sơ sở thích của họ. Hồ sơ này được hình thành từ lịch sử tương tác của người dùng, bao gồm các sản phẩm mà họ đã xem, đã nhấn yêu thích hoặc đã mua trong quá khứ. Những sản phẩm này cũng được chuyển thành vector đặc trưng, và hồ sơ người dùng chính là trung bình trọng số của các vector đó, phản ánh xu hướng phong cách hoặc sở thích đặc trưng của từng cá nhân. Sau khi có hồ sơ người dùng và đặc trưng của toàn bộ sản phẩm trong hệ thống, CBF sử dụng độ đo tương đồng – phổ biến nhất là Cosine Similarity – để tính toán mức độ giống nhau giữa hồ sơ người dùng với từng sản phẩm còn lại trong kho dữ liệu. Các sản phẩm có điểm tương đồng cao nhất sẽ được đề xuất cho người dùng.

Việc áp dụng CBF trong lĩnh vực thời trang mang lại nhiều hiệu quả vì sản phẩm thời trang thường có mô tả phong phú và mang tính phân loại cao. Khi người dùng xem một chiếc áo thun unisex form rộng, hệ thống có thể dễ dàng phân tích các từ khóa quan trọng như “form rộng”, “unisex”, “streetwear”, “cotton”, và tìm ra các sản phẩm tương tự trong hệ thống có đặc trưng kỹ thuật gần giống, từ đó tạo ra danh sách gợi ý mang tính cá nhân hóa cao. Điều này giúp người dùng tiếp cận nhanh chóng những sản phẩm phù hợp với phong cách của mình mà không cần mất thời gian tìm kiếm thủ công. Ngoài ra, CBF còn có ưu điểm lớn ở khả năng hoạt động hiệu quả ngay cả khi hệ thống mới triển khai và lượng người dùng chưa nhiều, bởi thuật toán không phụ thuộc vào dữ liệu của người dùng khác như các phương pháp Collaborative Filtering. Chính vì vậy, việc áp dụng CBF trong đề tài mang ý nghĩa quan trọng, giúp hệ thống xây dựng khả năng hiểu nội dung sản phẩm và tự động đưa ra những gợi ý thông minh, phù hợp với sở thích của từng khách hàng dựa trên hành vi tương tác của họ.

**3.3 Mô hình** **Convolutional Neural Network (CNN)**

Mô hình CNN là một trong những kiến trúc mạng nơ-ron sâu mang tính đột phá, được thiết kế chuyên biệt để xử lý dữ liệu hình ảnh và tự động học các đặc trưng trực quan mà không cần con người phải lập trình thủ công. CNN được ứng dụng rộng rãi trong nhiều lĩnh vực như nhận diện khuôn mặt, phân loại ảnh, phát hiện vật thể và đặc biệt phù hợp với bài toán tìm kiếm sản phẩm bằng hình ảnh trong thương mại điện tử. Điểm mạnh của CNN nằm ở khả năng trích xuất đặc trưng một cách tự động và hiệu quả. Khi một hình ảnh được đưa vào mô hình, các lớp tích chập sẽ thực hiện việc “quét” trên ảnh bằng các bộ lọc (filter) nhỏ để phát hiện ra những đặc trưng cơ bản như đường biên, cạnh, họa tiết hay sự thay đổi ánh sáng. Khi đi sâu vào mô hình, các lớp tiếp theo tiếp tục học các đặc trưng phức tạp hơn như form dáng quần áo, cấu trúc họa tiết, kiểu cổ áo, chất liệu vải hoặc phong cách tổng thể của trang phục.

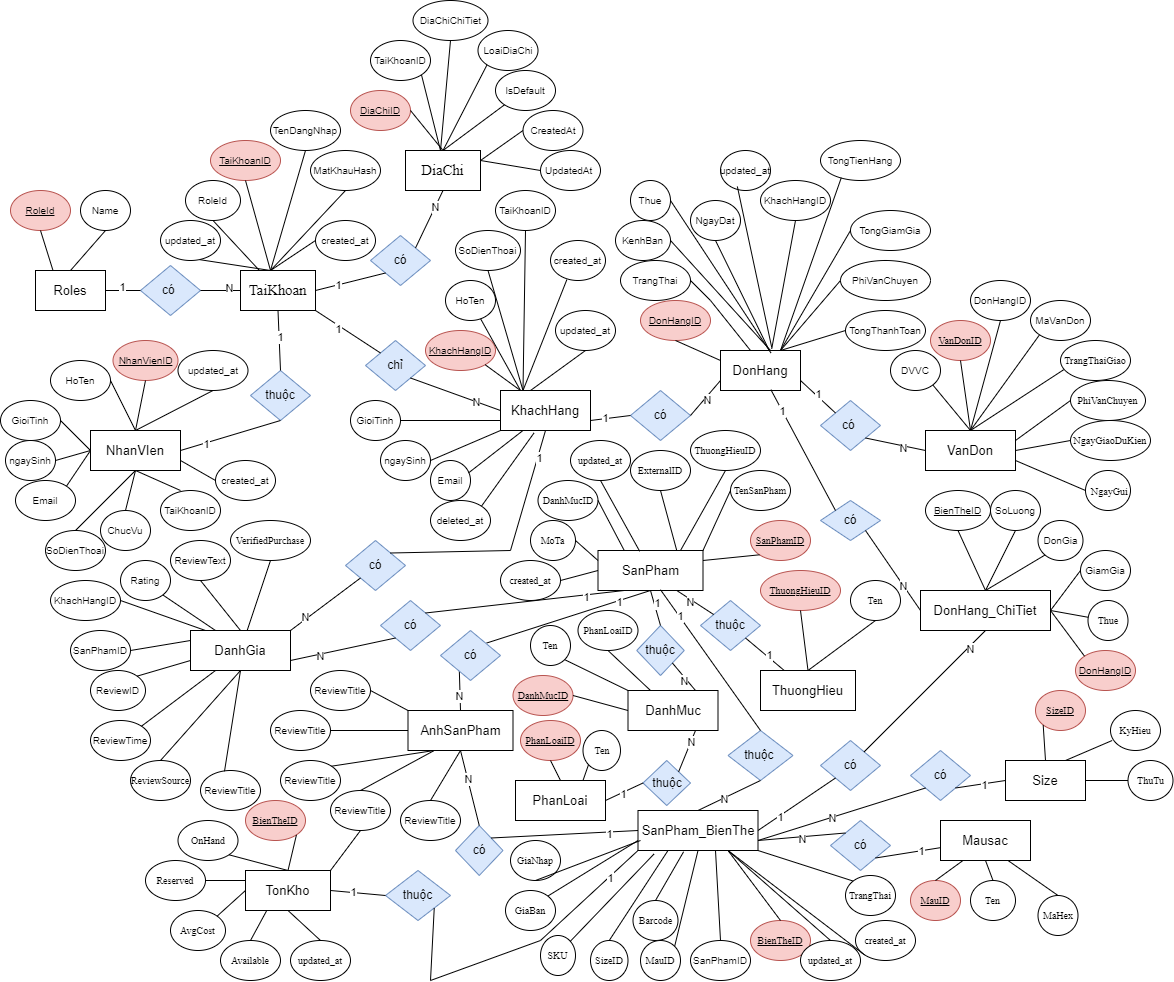
Sau quá trình trích xuất đặc trưng, CNN tạo ra một vector đặc trưng đại diện cho hình ảnh đầu vào. Vector này có thể được dùng để so sánh với các vector hình ảnh sản phẩm có sẵn trong cơ sở dữ liệu để tìm ra sản phẩm có mức độ tương đồng cao nhất. Việc so sánh mục tiêu này thường sử dụng các thước đo như Euclidean Distance hoặc Cosine Similarity. Sản phẩm có vector gần nhất với ảnh input sẽ được xem là sản phẩm giống nhất và hiển thị cho người dùng. Điều này đặc biệt hữu ích trong ngành thời trang – nơi mà hình ảnh đóng vai trò quan trọng hơn nhiều so với mô tả bằng văn bản. Người dùng thường khó mô tả bằng từ khóa các chi tiết như “kiểu sọc caro mảnh pha màu pastel” hoặc “dáng váy xếp tầng nhẹ, có bèo trước ngực”, nhưng CNN có thể nhận ra các đặc điểm này một cách tự nhiên chỉ dựa trên ảnh.

Việc áp dụng CNN vào hệ thống tìm kiếm của ứng dụng Stylo không chỉ giúp nâng cao trải nghiệm người dùng mà còn tạo ra sự tiện lợi vượt trội. Người dùng có thể bắt gặp một kiểu quần áo yêu thích khi xem video, đọc bài đăng trên mạng xã hội hay nhìn thấy ngoài đời thật và chỉ cần chụp hoặc tải lên hình ảnh đó, hệ thống sẽ chủ động tìm kiếm các sản phẩm tương tự trong cửa hàng. Điều này giúp rút ngắn quá trình tìm kiếm và làm tăng khả năng tiếp cận sản phẩm mong muốn một cách nhanh chóng. Không chỉ vậy, CNN còn mang lại tính năng hiện đại và tạo nên lợi thế cạnh tranh đáng kể cho ứng dụng so với các hệ thống bán hàng truyền thống.

Tổng quan lại, mô hình CNN và thuật toán CBF đóng vai trò quan trọng trong việc tạo nên hệ thống gợi ý và tìm kiếm thông minh của ứng dụng Stylo. CNN giúp hệ thống hiểu được hình ảnh sản phẩm, còn CBF giúp hệ thống hiểu được nội dung và mối quan hệ giữa các sản phẩm bên trong ứng dụng. Sự kết hợp này giúp tạo nên một nền tảng bán hàng thời trang trực tuyến vừa trực quan, vừa cá nhân hóa, đáp ứng tốt xu hướng công nghệ hiện đại và mang lại trải nghiệm tối ưu cho người dùng.

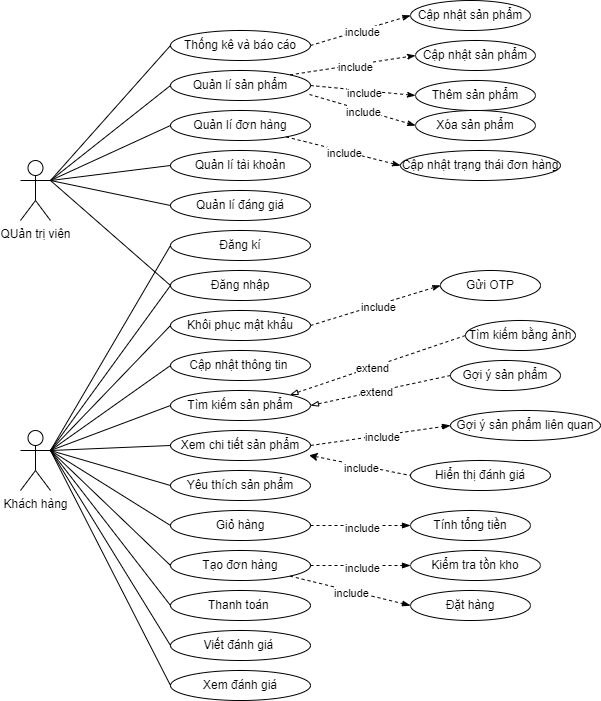
**CHƯƠNG 4: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

**4.1 Sơ đồ ER**



Hình 4.1 Sơ đồ ER

**4.2 Sơ đồ Use case tổng quát**



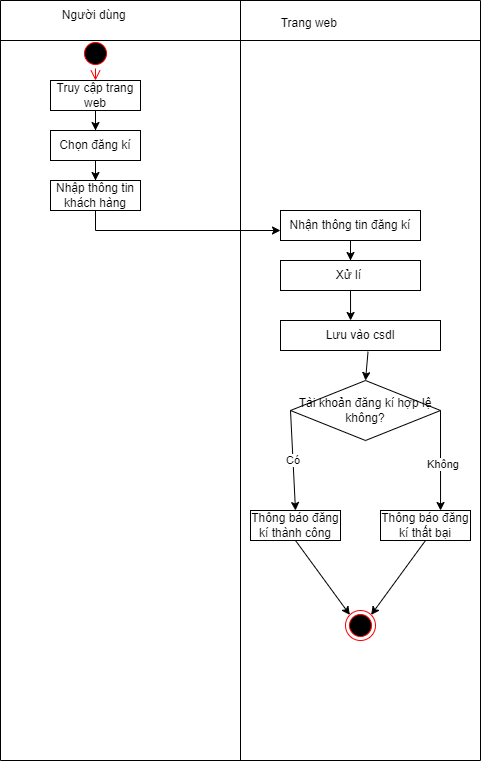
Hình 4.2 Sơ đồ Use case tônge quát

**4.3 Đặc tả một số Use case chính**

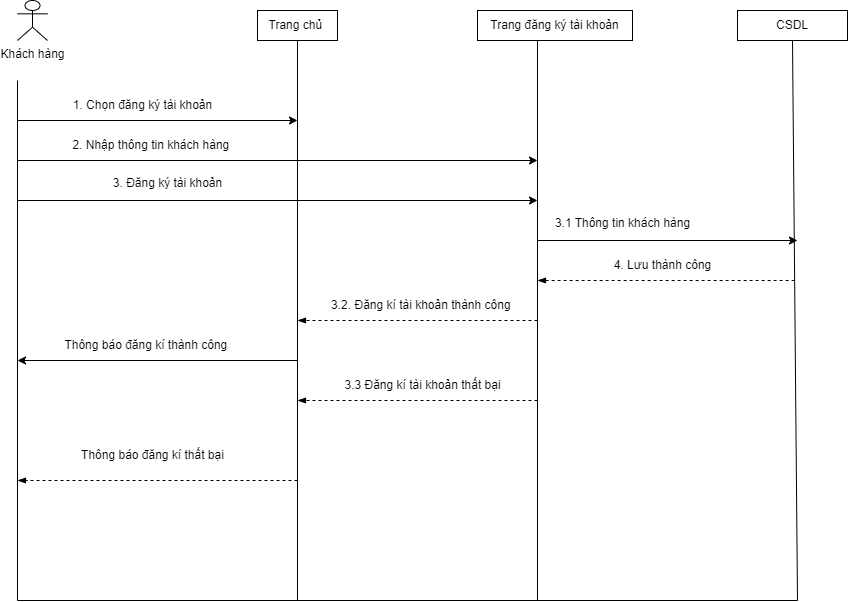
**4.3.1 Chức năng đăng kí**

|  |  |
| --- | --- |
| **Tên Use case** | Đăng kí |
| **Tên tác nhân** | Admin, Khách hàng |
| **Điều kiện kiên quyết** | Người dùng chưa có tài khoản hoặc chưa đăng nhập vào hệ thống. |
| **Đảm bảo tối thiểu** | Hệ thống giữ nguyên trạng thái hiện tại, không tạo tài khoản mới nếu thông tin không hợp lệ. |
| **Đảm bảo thành công** | Tài khoản mới được tạo thành công, thông tin được lưu vào CSDL và người dùng có thể đăng nhập. |
| **Kích hoạt** | Người dùng chọn chức năng "Đăng ký" trên giao diện. |
| **Chuỗi sự kiện chính** | - Người dùng nhập các thông tin bắt buộc (Họ tên, Email, Mật khẩu, Số điện thoại...).  - Hệ thống kiểm tra tính hợp lệ của dữ liệu (định dạng email, độ mạnh mật khẩu).  - Hệ thống kiểm tra sự tồn tại của Email trong cơ sở dữ liệu.  - Hệ thống thực hiện mã hóa mật khẩu và lưu thông tin người dùng mới vào bảng TaiKhoan.  - Hệ thống gửi thông báo đăng ký thành công.  - Ca sử dụng kết thúc. |
| **Ngoại lệ** | - Email đã tồn tại: Hệ thống thông báo Email đã được sử dụng, yêu cầu nhập Email khác.  - Dữ liệu không hợp lệ: Hệ thống báo lỗi các trường nhập sai (ví dụ: mật khẩu quá ngắn, email sai định dạng).  - Lỗi kết nối CSDL: Hệ thống thông báo thử lại sau. |

Bảng 4.1 Đặc tả chức năng đăng kí



Hình 4.3 Sơ đồ hoạt động chức năng đăng kí

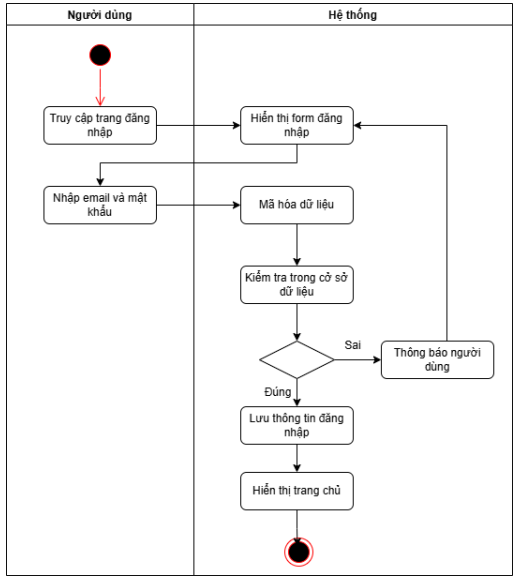


Hình 4.4 Sơ đồ tuần tự Use case đăng kí

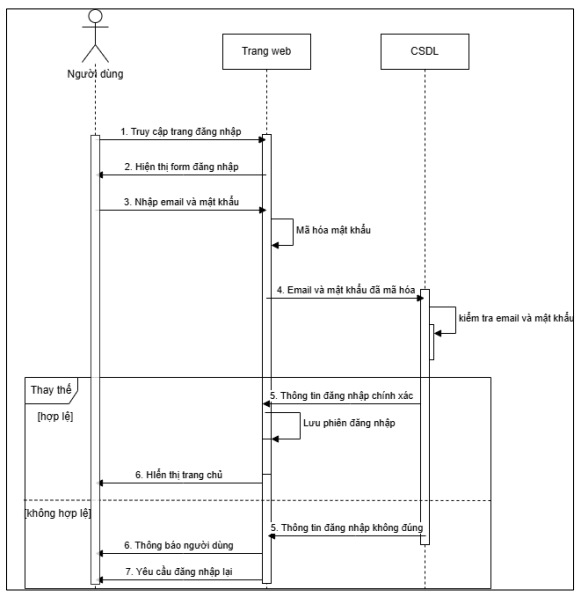
**4.3.2 Chức năng đăng nhập**

|  |  |
| --- | --- |
| **Tên Use case** | Đăng nhập |
| **Tên tác nhân** | Admin, Khách hàng |
| **Điều kiện kiên quyết** | Đã đăng xuất khỏi hệ thống |
| **Đảm bảo tối thiểu** | Bỏ thông tin đăng nhập và quay về trạng thái ban đầu |
| **Đảm bảo thành công** | Đăng nhập thành công vào hệ thống |
| **Kích hoạt** | Truy cập vào hệ thống |
| **Chuỗi sự kiện chính** | - Nhập email và mật khẩu. - Hệ thống mã hóa mật khẩu và đối chiếu với cơ sở dữ liệu. - Nếu thông tin chính xác, người dùng sẽ được chuyển đến trang chủ của hệ thống. - Ca sử dụng kết thúc |
| **Ngoại lệ** | - Đăng nhập thất bại - Hệ thống thông báo sai thông tin đăng nhập - Người dùng đăng nhập lại hệ thống. |

Bảng 4.2 Đặc tả Use case đăng nhập

****

Hình 4. 5 Sơ đồ hoạt động Use case đăng nhập

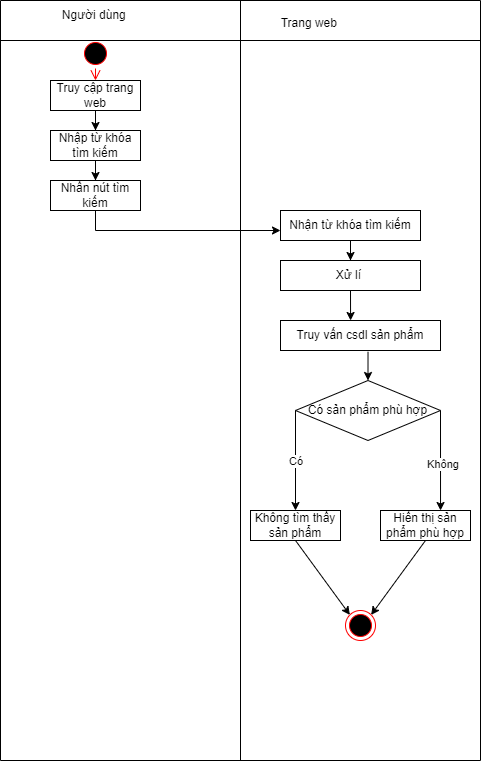
****

Hình 4.6 Sơ đồ tuần tự Use case đăng nhập

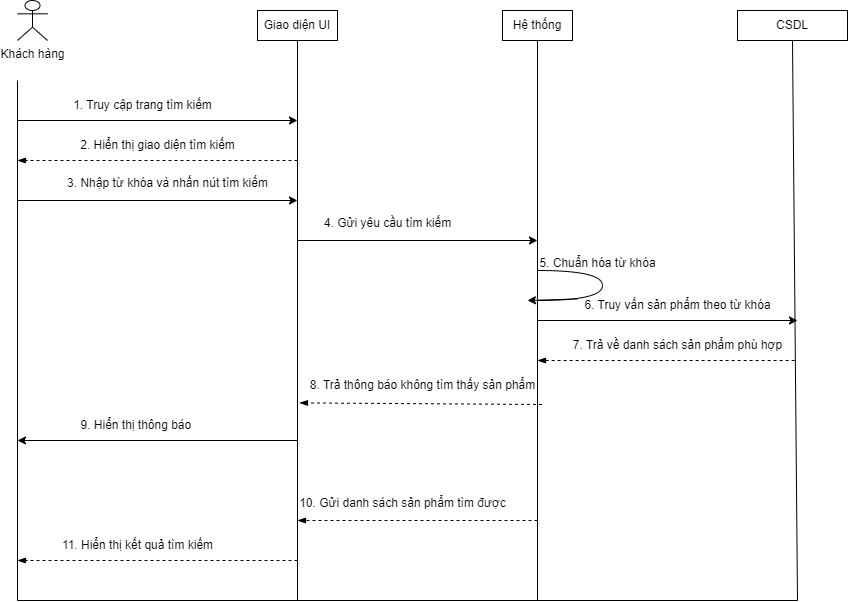
**4.3.3 Chức năng tìm kiếm**

|  |  |
| --- | --- |
| **Tên Use case** | Tìm kiếm sản phẩm |
| **Tên tác nhân** | Khách hàng |
| **Điều kiện kiên quyết** | Khách hàng đã truy cập trang web. |
| **Đảm bảo tối thiểu** | Hệ thống nhận yêu cầu tìm kiếm từ khách hàng. |
| **Đảm bảo thành công** | Hệ thống hiển thị danh sách các sản phẩm phù hợp với từ khóa tìm kiếm. |
| **Kích hoạt** | Khách hàng nhập từ khóa tìm kiếm và nhấn nút tìm kiếm. |
| **Chuỗi sự kiện chính** | - Khách hàng truy cập vào trang tìm kiếm.  - Giao diện UI hiển thị giao diện tìm kiếm.  - Khách hàng nhập từ khóa và nhấn nút tìm kiếm.  - Giao diện UI gửi yêu cầu tìm kiếm đến Hệ thống.  - Hệ thống chuẩn hóa từ khóa và xử lý từ khóa tìm kiếm.  - Hệ thống truy vấn sản phẩm theo từ khóa từ CSDL.  - CSDL trả về danh sách sản phẩm phù hợp cho Hệ thống.  - Hệ thống gửi danh sách sản phẩm tìm được đến Giao diện UI.  - Giao diện UI (Trang web) hiển thị kết quả tìm kiếm (các sản phẩm phù hợp) cho Khách hàng. |
| **Ngoại lệ** | Không tìm thấy sản phẩm: CSDL không trả về sản phẩm nào phù hợp. Hệ thống trả thông báo không tìm thấy sản phẩm. Giao diện UI hiển thị thông báo "Không tìm thấy sản phẩm" cho Khách hàng. |

Bảng 4.3 Đặc tả Use case tìm kiếm



Hình 4. 7 Sơ đồ hoạt động Use case tìm kiếm

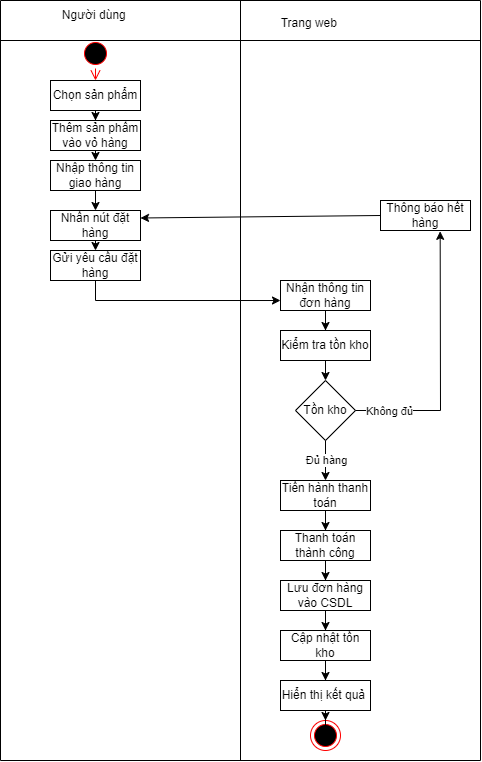


Hình 4.8 Sơ đồ tuần tự Use case tìm kiếm

**4.3.4 Chức năng đặt hàng**

|  |  |
| --- | --- |
| **Tên Use case** | Mua hàng |
| **Tên tác nhân** | Khách hàng |
| **Điều kiện kiên quyết** | Khách hàng đã chọn sản phẩm và thêm vào giỏ hàng. |
| **Đảm bảo tối thiểu** | Hệ thống ghi nhận yêu cầu và giữ nguyên trạng thái giỏ hàng nếu giao dịch bị hủy hoặc lỗi. |
| **Đảm bảo thành công** | - Thanh toán thành công và đơn hàng được khởi tạo trên hệ thống.  - Thông tin giao dịch được lưu trữ trong CSDL.  - Tồn kho được cập nhật. |
| **Kích hoạt** | - Khách hàng truy cập trang giỏ hàng.  - Khách hàng nhấn nút đặt hàng. |
| **Chuỗi sự kiện chính** | - Khách hàng truy cập trang giỏ hàng.  - Khách hàng nhập thông tin giao hàng, chọn phương thức thanh toán và gửi yêu cầu đặt hàng.  - Hệ thống tiếp nhận thông tin đơn hàng và thực hiện kiểm tra tồn kho.  - Nếu đủ hàng, hệ thống tiến hành điều hướng sang cổng thanh toán.  - Sau khi thanh toán thành công, hệ thống thực hiện lưu thông tin đơn hàng vào CSDL.  - Hệ thống gửi yêu cầu tạo đơn hàng sang Hệ thống đơn hàng để xác nhận.  - Hệ thống thực hiện cập nhật lại số lượng tồn kho của sản phẩm.  - Trang web hiển thị thông báo kết quả đặt hàng thành công và kết thúc ca sử dụng. |
| **Ngoại lệ** | - Hết hàng: (Theo Sơ đồ Hoạt động) Trang web kiểm tra tồn kho, nếu không đủ sẽ thông báo hết hàng cho Người dùng.  - Lỗi thanh toán: Cổng thanh toán thông báo thất bại. Trang web thông báo lỗi thanh toán và yêu cầu Khách hàng thử lại.  - Lỗi tạo đơn: Hệ thống đơn hàng xác nhận tạo đơn hàng thất bại. |

Bảng 4.4 Đặc tả Use case mua hàng



Hình 4.9 Sơ đồ hoạt động Use case đặt hàng



Hình 4.10 Sơ đồ tuần tự Use case đặt hàng

**4.4 Thiết kế cơ sở dữ liệu**

**4.4.1 Xác định thực thể**

Thực thể Roles (Vai trò)

* RoleId: Mã vai trò (Khóa chính).
* Name: Tên vai trò (ví dụ: Admin, Customer).

Thực thể TaiKhoan (Tài khoản)

* TaiKhoanID: Mã tài khoản (Khóa chính).
* TenDangNhap: Tên đăng nhập (duy nhất).
* MatKhauHash: Chuỗi băm mật khẩu.
* RoleId: Mã vai trò (Khóa ngoại).
* created\_at: Thời gian tạo tài khoản.
* updated\_at: Thời gian cập nhật tài khoản.

Thực thể KhachHang (Khách hàng)

* KhachHangID: Mã khách hàng (Khóa chính).
* HoTen: Họ và tên.
* GioiTinh: Giới tính.
* ngaySinh: Ngày sinh.
* Email: Địa chỉ Email (duy nhất).
* SoDienThoai: Số điện thoại.
* TaiKhoanID: Mã tài khoản liên kết (Khóa ngoại, duy nhất).
* created\_at: Thời gian tạo.
* updated\_at: Thời gian cập nhật.
* deleted\_at: Thời gian xóa mềm.

Thực thể NhanVien (Nhân viên)

* NhanVienID: Mã nhân viên (Khóa chính).
* HoTen: Họ và tên.
* GioiTinh: Giới tính.
* ngaySinh: Ngày sinh.
* Email: Địa chỉ Email (duy nhất).
* SoDienThoai: Số điện thoại.
* TaiKhoanID: Mã tài khoản liên kết (Khóa ngoại, duy nhất).
* created\_at: Thời gian tạo.
* updated\_at: Thời gian cập nhật.
* ChucVu: Chức vụ.

Thực thể VanDon (Vận đơn)

* VanDonID: Mã vận đơn (Khóa chính).
* DonHangID: Mã đơn hàng (Khóa ngoại).
* DVVC: Đơn vị vận chuyển.
* MaVanDon: Mã vận đơn.
* TrangThaiGiao: Trạng thái giao hàng.
* PhiVanChuyen: Phí vận chuyển
* NgayGui: Ngày gửi
* NgayGiaoDuKien: Ngày giao dữ kiện

Thực thể PhanLoai (Phân loại)

* PhanLoaiID: Mã phân loại (Khóa chính).
* Ten: Tên phân loại.

Thực thể DanhMuc (Danh mục sản phẩm)

* DanhMucID: Mã danh mục (Khóa chính).
* Ten: Tên danh mục .
* PhanLoaiID: Mã phân loại cấp cao (Khóa ngoại).

Thực thể ThuongHieu (Thương hiệu)

* ThuongHieuID: Mã thương hiệu (Khóa chính).
* Ten: Tên thương hiệu (duy nhất).

Thực thể SanPham (Sản phẩm)

* SanPhamID: Mã sản phẩm (Khóa chính).
* TenSanPham: Tên sản phẩm.
* MoTa: Mô tả chi tiết.
* DanhMucID: Mã danh mục (Khóa ngoại).
* ThuongHieuID: Mã thương hiệu (Khóa ngoại).
* ExternalID: Mã ngoài.
* created\_at: Thời gian tạo.
* updated\_at: Thời gian cập nhật.

Thực thể MauSac (Màu sắc)

* MauID: Mã màu (Khóa chính).
* Ten: Tên màu (duy nhất).
* MaHex: Mã màu Hex.

Thực thể Size (Kích cỡ)

* SizeID: Mã kích cỡ (Khóa chính).
* KyHieu: Ký hiệu kích cỡ (duy nhất).
* ThuTu: Thứ tự sắp xếp.

Thực thể SanPham\_BienThe (Biến thể/SKU)

* BienTheID: Mã biến thể (Khóa chính).
* SanPhamID: Mã sản phẩm (Khóa ngoại).
* MauID: Mã màu (Khóa ngoại).
* SizeID: Mã kích cỡ (Khóa ngoại).
* SKU: Mã SKU (duy nhất).
* Barcode: Mã vạch.
* GiaBan: Giá bán.
* GiaNhap: Giá nhập.
* TrangThai: Trạng thái.
* created\_at: Thời gian tạo.
* updated\_at: Thời gian cập nhật.

Thực thể AnhSanPham (Ảnh sản phẩm)

* AnhID: Mã ảnh (Khóa chính).
* SanPhamID: Mã sản phẩm (Khóa ngoại).
* BienTheID: Mã biến thể (Khóa ngoại).
* URL: Đường dẫn ảnh.
* IsPrimary: Cờ ảnh chính.

Thực thể TonKho (Tồn kho)

* BienTheID: Mã biến thể (Khóa chính, Khóa ngoại).
* OnHand: Số lượng có sẵn trong kho.
* Reserved: Số lượng đang được đặt giữ.
* AvgCost: Giá vốn trung bình.
* Available: Số lượng khả dụng (tính toán).
* updated\_at: Thời gian cập nhật.

Thực thể DanhGia (Đánh giá)

* ReviewID: Mã đánh giá.
* SanPhamID: Mã sản phẩm (Khóa ngoại).
* KhachHangID: Mã khách hàng (Khóa ngoại).
* Rating: Điểm xếp hạng.
* ReviewText: Nội dung đánh giá.
* ReviewTime: Thời gian đánh giá.
* VerifiedPurchase: Đã xác nhận mua hàng.
* ReviewSource: Nguồn đánh giá
* ReviewTitle: Tiêu đề đánh giá

Thực thể DonHang (Đơn hàng)

* DonHangID: Mã đơn hàng (Khóa chính).
* KhachHangID: Mã khách hàng (Khóa ngoại).
* TrangThai: Trạng thái đơn hàng.
* KenhBan: Kênh bán hàng.
* TongTienHang: Tổng tiền hàng.
* TongGiamGia: Tổng giảm giá.
* Thue: Thuế.
* PhiVanChuyen: Phí vận chuyển.
* TongThanhToan: Tổng số tiền phải thanh toán.
* NgayDat: Ngày đặt hàng.
* updated\_at: Thời gian cập nhật.

Thực thể DonHang\_ChiTiet (Chi tiết đơn hàng)

* DonHangID: Mã đơn hàng (Khóa chính, Khóa ngoại).
* BienTheID: Mã biến thể (Khóa chính, Khóa ngoại).
* SoLuong: Số lượng đặt mua.
* DonGia: Đơn giá tại thời điểm đặt. **-**
* GiamGia: Số tiền giảm giá cho mục này.
* Thue: Thuế cho mục này.

Thực thể DiaChi (Địa chỉ)

* DiaChiID: Mã địa chỉ( Khóa chính)
* TaiKhoanID: Mã tài khoản ( Khóa ngoại)
* DiaChiChiTiet: Nội dung địa chỉ chi tiết
* LoaiDiaChi: Loại địa chỉ
* IsDefault: Trạng thái địa chỉ
* CreatedAt: Thời gian tạo bản ghi
* UpdatedAt: Thời gian cập nhật bản ghi

**4.4.2 Biểu đồ Diagram**

Hình 4.11 Biểu đồ Diagram

**4.4.3 Đặc tả các bảng dữ liệu**

**\* Vai trò**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | RoleId | Int | Chính | Mã vai trò |
| **2** | Name | Nvarchar |  | Tên vai trò |

Bảng 4.5 Đặc tả bảng Role

**\* Tài khoản**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | TaiKhoanID | Int | Chính | Mã tài khoản |
| **2** | TenDangNhap | Nvarchar |  | Tên đăng nhập |
| **3** | MatKhauHash | Nvarchar |  | Chuỗi băm mật khẩu |
| **4** | RoleId | Nvarchar | Ngoại | Mã vai trò |
| **5** | created\_at | Datetime |  | Thời gian tạo tài khoản |
| **6** | updated\_at | Datetime |  | Thời gian cập nhật tài khoản |

4.6 Đặc tả bảng TaiKhoan

**\* Khách hàng**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | KhachHangID | Int | Chính | Mã khách hàng |
| **2** | HoTen | Nvarchar |  | Họ tên |
| **3** | GioiTinh | Nvarchar |  | Giới tính |
| **4** | ngaySinh | Datetime |  | Ngày sinh |
| **5** | Email | Nvarchar |  | Email |
| **6** | SoDienThoai | Nvarchar |  | Số điện thoại |
| **7** | TaiKhoanID | Int | Ngoại | Mã tài khoản |
| **8** | created\_at | Datetime |  | Thời gian tạo |
| **9** | updated\_at | Datetime |  | Thời gian cập nhật |
| **10** | deleted\_at | Datetime |  | Thời gian xóa mềm |

4.7 Đặc tả bảng KhachHang

**\* Nhân viên**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | NhanVienID | Int | Chính | Mã nhân viên |
| **2** | HoTen | Nvarchar |  | Họ tên |
| **3** | GioiTinh | Nvarchar |  | Giới tính |
| **4** | ngaySinh | Datetime |  | Ngày sinh |
| **5** | Email | Nvarchar |  | Email |
| **6** | SoDienThoai | Nvarchar |  | Số điện thoại |
| **7** | TaiKhoanID | Int | Ngoại | Mã tài khoản |
| **8** | created\_at | Datetime |  | Thời gian tạo |
| **9** | updated\_at | Datetime |  | Thời gian cập nhật |
| **10** | ChucVu | Nvarchar |  | Chức vụ |

4.8 Đặc tả bảng NhanVien

**\*Vận đơn**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | VanDonID | Int | Chính | Mã vận đơn |
| **2** | DonHangID | Int | Ngoại | Mã đơn hàng |
| **3** | DVVC | Nvarchar |  | Đơn vị vận chuyển |
| **4** | MaVanDon | Int |  | Mã vận đơn |
| **5** | TrangThaiGiao | Nvarchar |  | Trạng thái giao hàng |
| **6** | PhiVanChuyen | Decimal |  | Phí vận chuyển |
| **7** | NgayGui | Datetime |  | Ngày gửi |
| **8** | NgayGiaoDuKien | Datetime |  | Ngày giao dự kiến |

4.9 Đặc tả bảng VanDon

**\* Địa chỉ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | DiaChiID | Int | Chính | Mã địa chỉ |
| **2** | TaiKhoanID | Int | Ngoại | Mã tài khoản |
| **3** | DiaChiChiTiet | Nvarchar |  | Địa chỉ chi tiết |
| **4** | LoaiDiaChi | Nvarchar |  | Loại địa chỉ |
| **5** | IsDefault | Bit |  | Trạng thái địa chỉ |
| **6** | CreatedAt | Datetime |  | Ngày tạo |
| **7** | UpdatedAt | Datetime |  | Ngày cập nhật |

4.10 Đặc tả bảng DiaChi

**\* Phân loại**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | PhanLoaiID | Int | Chính | Mã phân loại |
| **2** | Ten | Nvarchar |  | Tên phân loại |

4.11 Đặc tả bảng PhanLoai

**\* Danh mục**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | DanhMucID | Int | Chính | Mã danh mục |
| **2** | PhanLoaiID | Int | Ngoại | Mã phân loại |
| **3** | Ten | Nvarchar |  | Tên danh mục |

4.12 Đặc tả bảng DanhMuc

**\* Thương hiệu**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | ThuongHieuID | Int | Chính | Mã thương hiệu |
| **2** | Ten | Nvarchar |  | Tên thương hiệu |

4.13 Đặc tả bảng ThuongHieu

**\* Sản phẩm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | SanPhamID | Int | Chính | Mã sản phẩm |
| **2** | TenSanPham | Nvarchar |  | Tên sản phẩm |
| **3** | MoTa | Nvarchar |  | Mô tả |
| **4** | DanhMucID | Int | Ngoại | Mã danh mục |
| **5** | ThuongHieuID | Int | Ngoại | Mã thương hiệu |
| **6** | ExternalID | Int |  | Mã ngoài |
| **7** | created\_at | Datetime |  | Ngày tạo |
| **8** | updated\_at | Datetime |  | Ngày cập nhật |

4.14 Đăch tả bảng SanPham

**\* Màu sắc**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | MauID | Int | Chính | Mã màu |
| **2** | Ten | Nvarchar |  | Tên màu |
| **3** | MaHex | Nchar |  | Mã màu hex |

4.15 Đặc tả bảng MauSac

**\* Kích cỡ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | SizeID | Int | Chính | Mã kích thước |
| **2** | KyHieu | Nvarchar |  | Kí hiệu |
| **3** | ThuTu | Int |  | Thứ tự |

4.16 Đặc tả bảng Size

**\* Sản phẩm biến thể**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | BienTheID | Int | Chính | Mã biến thể |
| **2** | SanPhamID | Int | Ngoại | Mã sản phẩm |
| **3** | MauID | Int | Ngoại | Mã màu |
| **4** | SizeID | Int | Ngoại | Mã kích thước |
| **5** | SKU | Nvarchar |  | Mã sku |
| **6** | Barcode | Nvarchar |  | Mã vạch |
| **7** | GiaBan | Decimal |  | Giá bán |
| **8** | GiaNhap | Decimal |  | Giá nhập |
| **9** | TrangThai | Nvarchar |  | Trạng thái |
| **10** | created\_at | Datetime |  | Ngày tạo |
| **11** | updated\_at | Datetime |  | Ngày cập nhật |

4.17 Đặc tả bảng SanPham\_BienThe

**\* Ảnh sản phẩm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | AnhID | Int | Chính | Mã ảnh |
| **2** | SanPhamID | Int | Ngoại | Mã sản phẩm |
| **3** | BienTheID | Int | Ngoại | Mã biến thể |
| **4** | URL | Nvarchar |  | Đường dẫn ảnh |
| **5** | IsPrimary | Bit |  | Cờ ảnh chính |

4.18 Đặc tả bảng AnhSanPham

**\* Tồn kho**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | BienTheID | Int | Chính, ngoại | Mã biến thể |
| **2** | OnHand | Int |  | Số lượng tồn kho |
| **3** | Reserved | Int |  | Số lượng đang được đặt |
| **4** | AvgCost | Decimal |  | Giá vốn trung bình |
| **5** | Available | Persisted |  | Số lượng khả dụng |
| **6** | updated\_at | Datetime |  | Thời gian cập nhật |

4.19 Đặc tả bảng TonKho

**\* Đánh giá**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | ReviewID | BigInt | Chính | Mã đánh giá |
| **2** | SanPhamID | Int | Ngoại | Mã sản phẩm |
| **3** | KhachHangID | Int | Ngoại | Mã khách hàng |
| **4** | Rating | Decimal |  | Điểm đánh giá |
| **5** | ReviewText | Nvarchar |  | Nội dung đánh giá |
| **6** | ReviewTime | Datetime |  | Thời gian đánh giá |
| **7** | VerifiedPurchase | Bit |  | Đã xác nhận mua hàng |
| **8** | ReviewSource | Nvarchar |  | Nguồn đánh giá |
| **9** | ReviewTitle | Nvarchar |  | Tiêu đề đánh giá |

4.20 Đặc tả bảng DanhGia

**\* Đơn hàng**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | DonHangID | Int | Chính | Mã đơn hàng |
| **2** | KhachHangID | Int | Ngoại | Mã khách hàng |
| **3** | TrangThai | Nvarchar |  | Trạng thái |
| **4** | KenhBan | Nvarchar |  | Kênh bán |
| **5** | TongTienHang | Decimal |  | Tổng tiền hàng |
| **6** | TongGiamGia | Decimal |  | Tổng giảm giá |
| **7** | Thue | Decimal |  | Thuế |
| **8** | PhiVanChuyen | Decimal |  | Phí vân chuyển |
| **9** | TongThanhToan | Decimal |  | Tổng thanh toán |
| **10** | NgayDat | Datetime |  | Ngày đặt |
| **11** | updated\_at | Datetime |  | Ngày cập nhật |

4.21 Đặc tả bảng DonHang

**\* Đơn hàng chi tiết**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Khóa** | **Diễn giải** |
| **1** | DonHangID | Int | Chính | Mã đơn hàng |
| **2** | BienTheID | Int | Ngoại | Mã biến thể |
| **3** | SoLuong | Int |  | Số lượng |
| **4** | DonGia | Decimal |  | Đơn giá |
| **5** | GiamGia | Decimal |  | Giảm giá |
| **6** | Thue | Decimal |  | Thuế |

4.22 Đặc tả bảng ChiTiet\_DonHang

# CHƯƠNG 5 XÂY DỰNG CHƯƠNG TRÌNH

### 5.1 Giao diện chính

### Hình 5.1

### Hình 5.2

### Hình 5.3

### Hình 5.4

### Hình 5.5

### Hình 5.6

### Hình 5.7

### Hình 5.8

### 5.2 Giao diện xử lí

### Hình 5.9

### Hình 5.10

### Hình 5.11

### Hình 5.12

### Hình 5.13

### Hình 5.14

### Hình 5.15

### Hình 5.16

### Hình 5.17

### Hình 5.18

### Hình 5.19

### Hình 5.20

**KẾT LUẬN**

**1. Kết quả đạt được**

Hoàn thiện ứng dụng đa nền tảng (Android/iOS) sử dụng framework Flutter. Xây dựng thành công hệ thống Backend bằng ASP.NET Core Web API và cơ sở dữ liệu Microsoft SQL Server để quản lý thông tin ổn định.

Đã phát triển đầy đủ các tính năng cần thiết cho một hệ thống thương mại điện tử từ đăng ký, đăng nhập, tìm kiếm sản phẩm đến giỏ hàng, đặt hàng và quản lý đơn hàng cho Admin.

Tích hợp thành công các mô hình học máy như gợi ý sản phẩm tương tự (CBF) và tìm kiếm bằng ảnh (CNN).

Cung cấp Dashboard thống kê doanh thu, đơn hàng và hành vi khách hàng trực quan, giúp người quản trị ra quyết định kinh doanh chính xác hơn.

Các mô hình gợi ý sản phẩm và kích cỡ đạt được độ chính xác ≥ 80% so với lựa chọn thực tế của khách hàng.

**2. Hạn chế**

Dữ liệu sử dụng để huấn luyện mô hình Machine Learning chủ yếu từ hệ thống thử nghiệm nên có thể chưa bao quát được toàn bộ sự đa dạng của thị trường thực tế.

Việc tìm kiếm sản phẩm bằng hình ảnh có độ chính xác cao nhưng tốc độ xử lý vẫn phụ thuộc vào chất lượng đường truyền Internet và tài nguyên máy chủ.

Hiện tại ứng dụng mới chỉ tập trung tối ưu cho các cửa hàng thời trang quy mô vừa và nhỏ.

**3. Hướng phát triển**

Phát triển cấu trúc hệ thống để có thể áp dụng cho các ngành hàng thương mại điện tử khác như mỹ phẩm, phụ kiện, giày dép.

Bổ sung các cổng thanh toán trực tuyến phổ biến (Momo, VNPay, thẻ ngân hàng) để tăng sự thuận tiện cho người dùng.

Nghiên cứu thêm về công nghệ AR (Thực tế ảo tăng cường) để hỗ trợ tính năng "thử đồ ảo", giúp khách hàng chọn kích cỡ chính xác hơn.

# PHỤ LỤC

### Phụ lục 1: Hướng dẫn cài đặt

Git clone <https://github.com/KimThao26042004/Demo_Train_CART_TabPFNv2_Model>

### Phụ lục 2: Hướng dẫn sử dụng

B1: Tạo 1 folder mới và paste folder vừa clone về vào folder mới tạo B2: Mở folder bằng Visual Studio Code

B3 Cài các thư viện cần thiết và chạy chương trình

# TÀI LIỆU THAM KHẢO

[1]. Đại học Bách Khoa TP.HCM (2021), *Giáo trình Học máy (Machine Learning),* Bộ môn Khoa học Máy tính..

[2]. Grinberg, M. (2018), *Flask Web Development: Developing Web Applications with Python,* O’Reilly Media.

[3]. W3Schools, (2024), *Python, C#, SQL and Flutter Tutorials,* Retrieved from [https://www.w3schools.com](https://www.w3schools.com/)

[4]. El Kharoua, R. (2024), *Predict Customer Purchase Behavior Dataset* [Data set], Kaggse,[https://www.kaggle.com/datasets/rabieelkharoua/predict-customer-](https://www.kaggle.com/datasets/rabieelkharoua/predict-customer-purchase-behavior-dataset)

[purchase-behavior-dataset](https://www.kaggle.com/datasets/rabieelkharoua/predict-customer-purchase-behavior-dataset)

[5]. KimThao26042004, (2025), *Demo\_Train\_CART\_TabPFNv2\_Model* [Repository], GitHub,[https://github.com/KimThao26042004/Demo\_Train\_CART\_TabPFNv2\_M](https://github.com/KimThao26042004/Demo_Train_CART_TabPFNv2_Model) [odel](https://github.com/KimThao26042004/Demo_Train_CART_TabPFNv2_Model)