

**Lớp**: DHKTPM16FTT

**Nhóm 6 - Thành viên tham gia :**

1. Nguyễn Minh Nhật - 20040831
2. Nguyễn Quốc Huy - 20040451
3. Lâm Bá Phúc - 20044381

**BÁO CÁO BÀI TẬP LỚN**

**KIẾN TRÚC VÀ THIẾT KẾ PHẦN MỀM**

**Quy Trình Xử Lý Tái Chế Các**

**Thiết Bị Điện Tử Cũ**

**Phân tích kiến trúc phần mềm của dự án hệ thống tái chế điện tử**

**Kiểu kiến trúc lựa chọn cho dự án:** Kiến trúc hướng mô hình (Model-View-Controller - MVC)

**Lý do lựa chọn:**

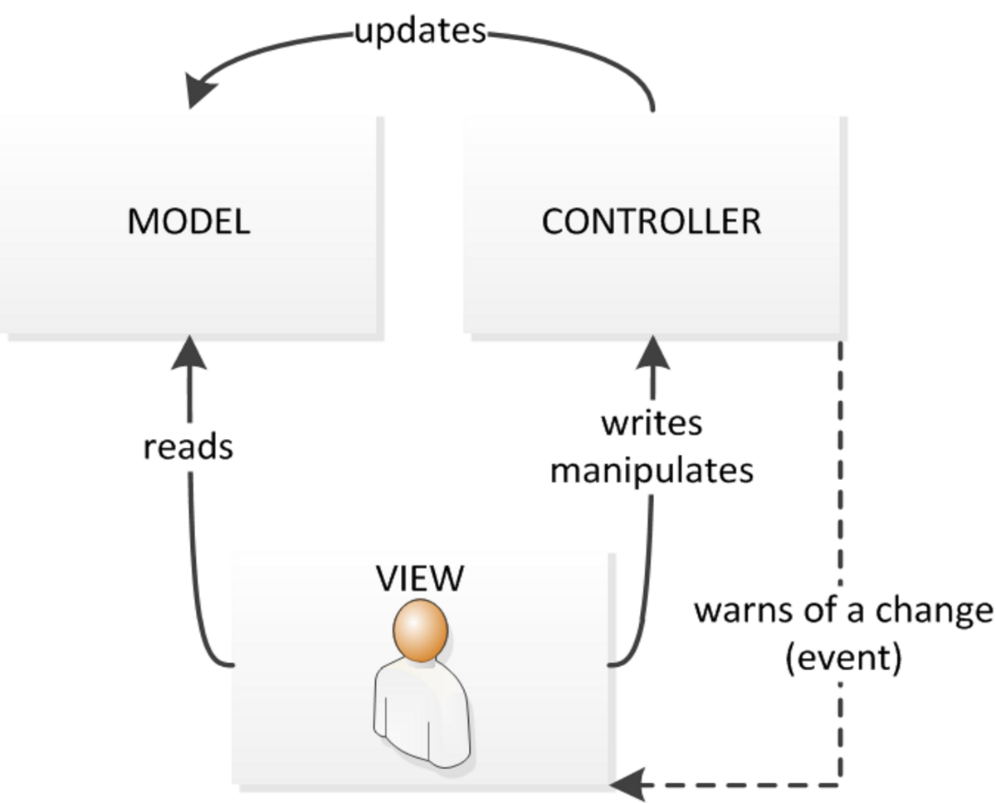
* **Tính phù hợp**
  + Hệ thống tái chế điện tử thường bao gồm các chức năng thu thập, phân loại, xử lý và quản lý dữ liệu về rác thải điện tử. Mô hình MVC phù hợp với các ứng dụng quản lý dữ liệu, giúp chia nhỏ hệ thống thành các thành phần logic rõ ràng, dễ dàng phát triển, bảo trì và mở rộng.
  + MVC tách biệt các khía cạnh giao diện, xử lý logic và dữ liệu, giúp các lập trình viên có thể tập trung vào từng phần việc riêng biệt, nâng cao hiệu quả phát triển.
* **Khả năng tái sử dụng**
  + Các thành phần Model, View và Controller trong MVC có thể được tái sử dụng cho các chức năng khác nhau trong hệ thống, tiết kiệm thời gian và công sức phát triển.
* **Dễ dàng bảo trì**
  + Việc chia nhỏ hệ thống theo mô hình MVC giúp dễ dàng xác định và sửa lỗi, cũng như nâng cấp các thành phần riêng lẻ mà không ảnh hưởng đến toàn bộ hệ thống.
* **Tính linh hoạt**
  + MVC cho phép sử dụng các công nghệ khác nhau cho từng thành phần, giúp hệ thống dễ dàng thích ứng với những thay đổi về yêu cầu.

**Mô tả kiến trúc:**

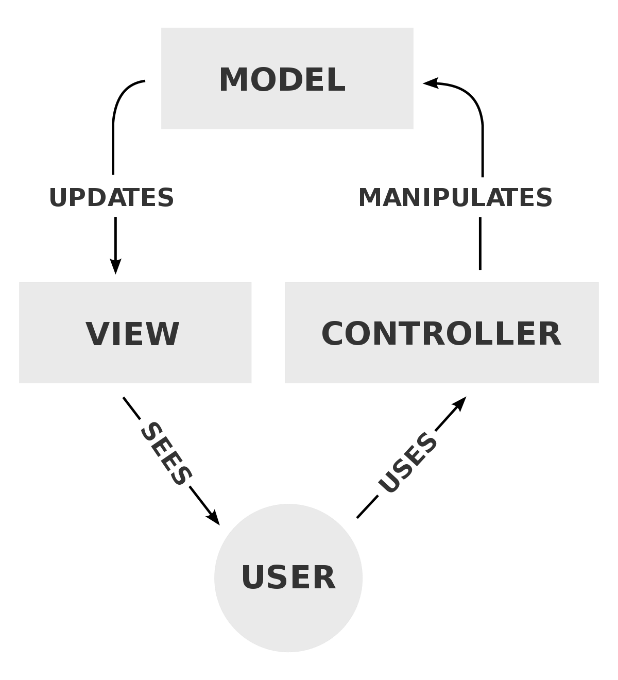
Hệ thống bao gồm các dịch vụ sau:

1. **Model (Mô hình):**

* Chứa các lớp đại diện cho các thực thể trong hệ thống, ví dụ như: loại rác thải điện tử, thông tin về điểm thu gom, dữ liệu về quá trình xử lý,...
* Quản lý logic nghiệp vụ liên quan đến dữ liệu, ví dụ như: xác định loại rác thải, tính toán chi phí xử lý, lưu trữ hồ sơ thu gom và xử lý,...
* Cung cấp API để View và Controller truy cập và thao tác với dữ liệu.



1. **View (Giao diện):**

* Hiển thị dữ liệu cho người dùng thông qua các giao diện web, ứng dụng di động hoặc các giao diện người dùng khác.
* Bao gồm các thành phần giao diện như: bảng điều khiển quản trị, biểu mẫu thu thập thông tin, báo cáo thống kê,...
* Sử dụng dữ liệu từ Model để hiển thị thông tin chính xác và cập nhật cho người dùng.
* Gửi yêu cầu và phản hồi từ người dùng đến Controller để xử lý.

1. **Controller (Bộ điều khiển):**

* Nhận yêu cầu từ người dùng thông qua View.
* Giao tiếp với Model để lấy dữ liệu hoặc cập nhật dữ liệu.
* Xử lý logic nghiệp vụ cần thiết dựa trên yêu cầu của người dùng.
* Cập nhật View với dữ liệu mới hoặc thông báo kết quả xử lý.
* A diagram of a model

  Description automatically generatedĐảm bảo sự tương tác chính xác giữa View và Model.

**Ưu điểm của kiến trúc:**

* **Dễ dàng phát triển và bảo trì:** Việc chia nhỏ hệ thống thành các thành phần logic rõ ràng giúp các lập trình viên dễ dàng phát triển và bảo trì từng phần, giảm thiểu lỗi và tiết kiệm thời gian.
* **Khả năng mở rộng:** Hệ thống có thể dễ dàng mở rộng bằng cách thêm các chức năng mới hoặc tăng quy mô các thành phần hiện có mà không ảnh hưởng đến toàn bộ hệ thống.
* **Tính linh hoạt:** MVC cho phép sử dụng các công nghệ khác nhau cho từng thành phần, giúp hệ thống dễ dàng thích ứng với những thay đổi về yêu cầu.
* **Tái sử dụng code:** Các thành phần Model, View và Controller có thể được tái sử dụng cho các chức năng khác nhau trong hệ thống, tiết kiệm thời gian và công sức phát triển.
* **Trải nghiệm người dùng tốt:** MVC giúp tạo ra các giao diện người dùng trực quan, dễ sử dụng và cung cấp thông tin chính xác cho người dùng.

A diagram of a model

Description automatically generated

**Nhược điểm của kiến trúc:**

* **Có thể phức tạp hơn cho các ứng dụng đơn giản:** Việc chia nhỏ hệ thống thành nhiều thành phần có thể khiến cho việc phát triển trở nên phức tạp hơn cho các ứng dụng đơn giản.
* **Yêu cầu sự hiểu biết rõ ràng về kiến trúc MVC:** Để sử dụng hiệu quả mô hình MVC, các lập trình viên cần có sự hiểu biết rõ ràng về kiến trúc và cách thức hoạt động của nó.
* **Khó khăn trong việc quản lý các ứng dụng lớn và phức tạp:** Việc quản lý một hệ thống MVC với nhiều thành phần có thể phức tạp, đòi hỏi các công cụ và quy trình quản lý phù hợp.

**Kết luận:**

MVC là lựa chọn phù hợp cho dự án hệ thống tái chế điện tử nhờ tính linh hoạt, dễ dàng phát triển, bảo trì và mở rộng. Tuy nhiên, cần lưu ý đến độ phức tạp tiềm ẩn và yêu cầu về kiến thức chuyên môn khi sử dụng MVC cho các dự án lớn.

Ngoài ra, cần cân nhắc các yếu tố khác như quy mô dự án, trình độ kỹ thuật của đội ngũ phát triển, ngân sách và thời gian để lựa chọn kiến trúc phần mềm phù hợp nhất.

## Mô hình con từ mô hình hệ thống tái chế điện tử

Dựa trên mô hình hệ thống tái chế điện tử được cung cấp, có thể vẽ các mô hình con sau:

**1. Mô hình quản lý thu gom thiết bị điện tử:**

**Model:**

* Thông tin điểm thu gom (địa chỉ, tên liên hệ, thời gian thu gom,...)
* Loại thiết bị điện tử cần thu gom (điện thoại, máy tính, tivi,...)
* Số lượng thiết bị điện tử cần thu gom
* Trạng thái thu gom (đã đặt lịch, đang thu gom, đã thu gom)

**View:**

* Giao diện người dùng cho phép người dùng đăng ký điểm thu gom, nhập thông tin về loại thiết bị điện tử và số lượng.
* Hiển thị danh sách các điểm thu gom đã được ghi nhận.
* Hiển thị trạng thái thu gom của từng điểm thu gom.
* Cung cấp chức năng theo dõi vị trí của nhân viên thu gom.

**Controller:**

* Xử lý các yêu cầu thu gom thiết bị điện tử từ người dùng.
* Lưu trữ thông tin thu gom vào cơ sở dữ liệu.
* Cập nhật trạng thái thu gom của từng điểm thu gom.
* Gửi thông báo cho bộ phận thu gom khi có yêu cầu mới.
* Quản lý lịch trình thu gom và phân công nhân viên thu gom.

A diagram with text and words

Description automatically generated

**Hình 1. Mô hình quản lý thu gom thiết bị điện tử**

**2. Mô hình phân loại thiết bị điện tử:**

**Model:**

* Loại thiết bị điện tử (điện thoại, máy tính, tivi,...)
* Số lượng thiết bị điện tử đã phân loại
* Hiệu quả phân loại (tỷ lệ phân loại chính xác)

**View:**

* Giao diện điều khiển cho phép người dùng điều khiển các thiết bị phân loại tự động.
* Hiển thị thông tin về loại thiết bị điện tử và số lượng đã phân loại.
* Hiển thị hiệu quả phân loại của hệ thống.
* Cung cấp chức năng theo dõi quy trình phân loại.

**Controller:**

* Gửi tín hiệu điều khiển đến các thiết bị phân loại tự động.
* Thu thập dữ liệu phân loại từ các cảm biến.
* Lưu trữ dữ liệu phân loại vào cơ sở dữ liệu.
* Tính toán hiệu quả phân loại của hệ thống.
* Ghi nhận các lỗi phân loại và điều chỉnh hệ thống.

A diagram of a model

Description automatically generated

**Hình 2. Mô hình phân loại thiết bị điện tử**

**3. Mô hình quy trình xử lý thiết bị điện tử:**

**Model:**

* Loại thiết bị điện tử cần xử lý
* Số lượng thiết bị điện tử cần xử lý
* Phương pháp xử lý (tái chế, tiêu hủy,...)
* Lượng khí thải và chất thải độc hại sinh ra

**View:**

* Giao diện điều khiển cho phép người dùng điều khiển các thiết bị xử lý.
* Hiển thị thông tin về loại thiết bị điện tử và số lượng đang được xử lý.
* Hiển thị tình trạng hoạt động của các thiết bị xử lý.
* Cung cấp chức năng theo dõi quy trình xử lý.

**Controller:**

* Gửi tín hiệu điều khiển đến các thiết bị xử lý.
* Thu thập dữ liệu về hoạt động xử lý từ các cảm biến.
* Lưu trữ dữ liệu xử lý vào cơ sở dữ liệu.
* Theo dõi và điều chỉnh quy trình xử lý để đảm bảo an toàn và hiệu quả.
* Ghi nhận các sự cố trong quá trình xử lý và xử lý sự cố.

A diagram of a model

Description automatically generated

**Hình 3. Mô hình quy trình xử lý thiết bị điện tử**

**4. Mô hình báo cáo:**

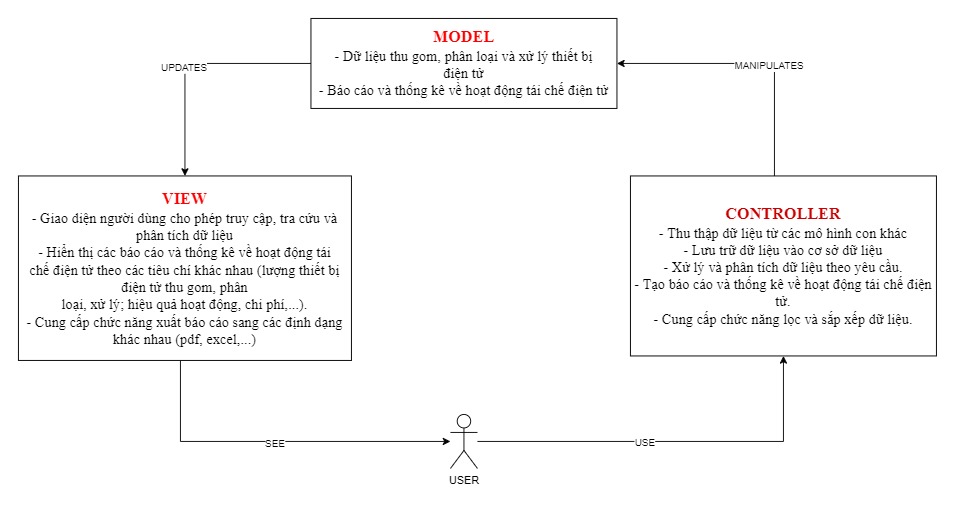
**Model:**

* Dữ liệu thu gom, phân loại và xử lý thiết bị điện tử
* Báo cáo và thống kê về hoạt động tái chế điện tử

**View:**

* Giao diện người dùng cho phép truy cập, tra cứu và phân tích dữ liệu.
* Hiển thị các báo cáo và thống kê về hoạt động tái chế điện tử theo các tiêu chí khác nhau (lượng thiết bị điện tử thu gom, phân loại, xử lý; hiệu quả hoạt động, chi phí,...).
* Cung cấp chức năng xuất báo cáo sang các định dạng khác nhau (pdf, excel,...)

**Controller:**

* Thu thập dữ liệu từ các mô hình con khác.
* Lưu trữ dữ liệu vào cơ sở dữ liệu.
* Xử lý và phân tích dữ liệu theo yêu cầu.
* Tạo báo cáo và thống kê về hoạt động tái chế điện tử.
* Cung cấp chức năng lọc và sắp xếp dữ liệu.

**Hình 4. Mô hình báo cáo**

**5. Mô hình thanh toán**

**Model:**

* Thông tin khách hàng (tên, địa chỉ, số điện thoại, email)
* Loại dịch vụ sử dụng (thu gom, phân loại, xử lý thiết bị điện tử)
* Số lượng thiết bị điện tử cần thanh toán
* Phương thức thanh toán (tiền mặt, chuyển khoản, thẻ ngân hàng)
* Lịch sử thanh toán

**View:**

* Giao diện thanh toán cho phép khách hàng nhập thông tin thanh toán.
* Hiển thị thông tin chi tiết về hóa đơn thanh toán (loại dịch vụ, số lượng thiết bị, số tiền).
* Hỗ trợ các phương thức thanh toán khác nhau (tiền mặt, chuyển khoản, thẻ ngân hàng).
* Xác nhận thanh toán thành công và cung cấp hóa đơn điện tử.

**Controller:**

* Xử lý các yêu cầu thanh toán từ khách hàng.
* Tính toán số tiền thanh toán dựa trên loại dịch vụ và số lượng thiết bị điện tử.
* Kết nối với các cổng thanh toán để thực hiện thanh toán trực tuyến.
* Lưu trữ thông tin thanh toán vào cơ sở dữ liệu.
* Gửi email xác nhận thanh toán và cung cấp hóa đơn điện tử cho khách hàng.
* Quản lý lịch sử thanh toán của khách hàng.

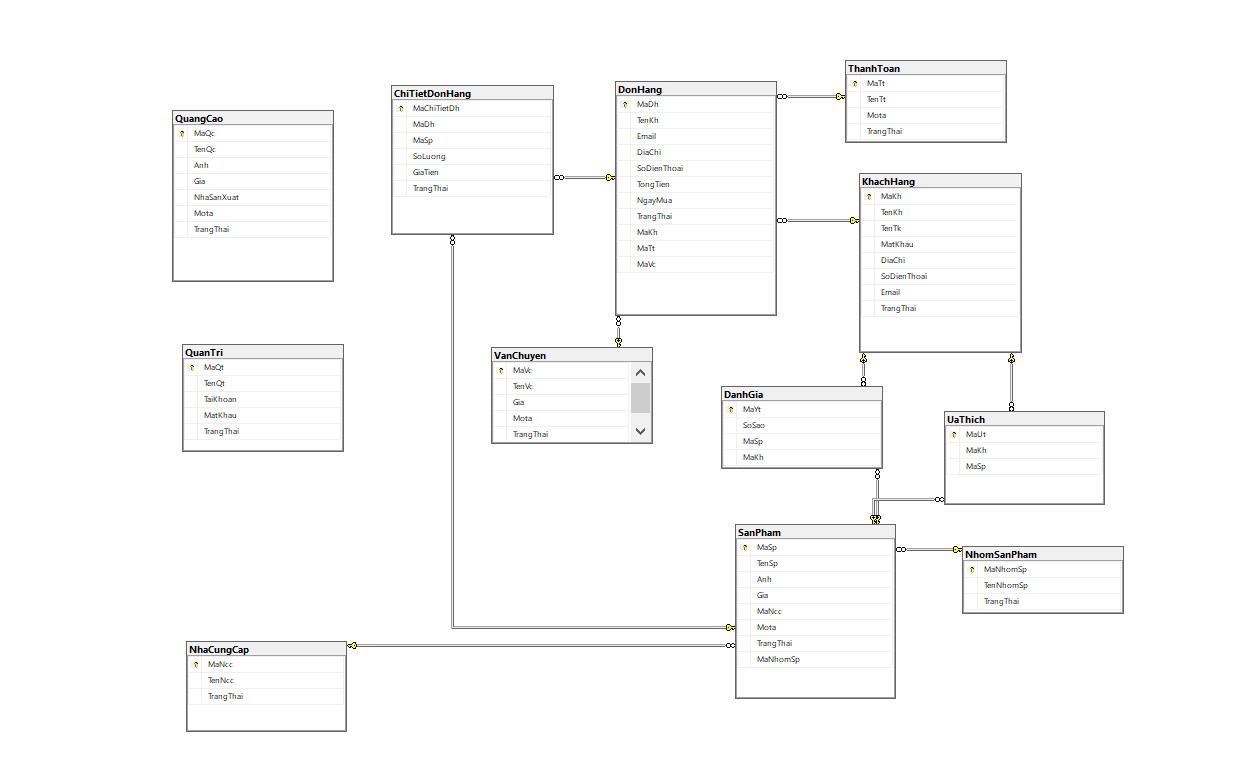
## 

**Hình 4. Mô hình thanh toán**

## Mô hình dự án hệ thống tái chế điện tử

1. **Yêu cầu báo giá (Quoting):**
   * Khách hàng truy cập trang web hoặc máy kiosk của công ty để hỏi giá tiền mà họ có thể nhận được cho thiết bị điện tử cũ của mình.
   * Sau khi nhận được báo giá, nếu khách hàng hài lòng, họ sẽ gửi thiết bị điện tử cũ cho công ty tái chế.
2. **Tiếp nhận thiết bị (Receiving):**
   * Công ty tái chế nhận thiết bị điện tử cũ từ khách hàng sau khi đã được thông báo báo giá.
   * Quá trình tiếp nhận này có thể diễn ra thông qua các kênh trực tuyến hoặc tại các điểm thu gom được chỉ định.
3. **Đánh giá thiết bị (Assessment):**
   * Sau khi tiếp nhận, công ty tái chế đánh giá tình trạng của thiết bị để xác định xem nó có hoạt động tốt hay không.
   * Nếu thiết bị hoạt động tốt, công ty sẽ tiến hành thanh toán cho khách hàng theo hợp đồng.
4. **Tài khoản thanh toán (Accounting):**
   * Công ty tái chế sẽ thực hiện thanh toán cho khách hàng dựa trên giá trị đã thỏa thuận của thiết bị.
   * Thông tin về các giao dịch thanh toán này sẽ được ghi vào hệ thống tài chính.
5. **Trạng thái của mặt hàng (Item Status):**
   * Khách hàng có thể truy cập trang web của công ty bất kỳ lúc nào để kiểm tra trạng thái của thiết bị của họ, từ khi nó được gửi cho đến khi được xử lý và thanh toán.
6. **Tái chế (Recycling):**
   * Dựa trên kết quả đánh giá, thiết bị điện tử cũ sau đó sẽ được tái chế, có thể bằng cách an toàn phá hủy hoặc bán lại.
   * Quyết định tái chế được đưa ra dựa trên tình trạng cụ thể của thiết bị.
7. **Báo cáo (Reporting):**
   * Công ty thực hiện báo cáo tài chính và hoạt động tái chế, bao gồm cả báo cáo tài chính và hoạt động thường xuyên và không định kỳ.
   * Thông tin từ các báo cáo này có thể được sử dụng để cải thiện hiệu suất và quản lý hoạt động tái chế.

## Sơ đồ mô hình UML



* **Lớp (Class):**

1. Quảng cáo
2. Chi tiết đơn hàng
3. Đơn hàng
4. Thanh toán
5. Khách hàng
6. Vận chuyển
7. Đánh giá
8. Ưa thích
9. Sản phẩm
10. Quản trị
11. Nhà cung cấp
12. Nhóm sản phẩm

* **Mối quan hệ (Relationship):**

1. ChiTietDonHang - DonHang: Mối quan hệ 1-N. Một đơn hàng (DonHang) có thể có nhiều chi tiết đơn hàng (ChiTietDonHang).
2. ChiTietDonHang - SanPham: Mối quan hệ N-1. Một chi tiết đơn hàng (ChiTietDonHang) liên kết với một sản phẩm (SanPham).
3. DonHang - KhachHang: Mối quan hệ N-1. Một đơn hàng (DonHang) thuộc về một khách hàng (KhachHang).
4. DonHang - VanChuyen: Mối quan hệ N-1. Một đơn hàng (DonHang) sử dụng một phương thức vận chuyển (VanChuyen).
5. DonHang - ThanhToan: Mối quan hệ N-1. Một đơn hàng (DonHang) có một phương thức thanh toán (ThanhToan).
6. DanhGia - SanPham: Mối quan hệ N-1. Một đánh giá (DanhGia) thuộc về một sản phẩm (SanPham).
7. DanhGia - KhachHang: Mối quan hệ N-1. Một đánh giá (DanhGia) được thực hiện bởi một khách hàng (KhachHang).
8. UaThich - KhachHang: Mối quan hệ N-1. Một sở thích (UaThich) của một khách hàng (KhachHang).
9. UaThich - SanPham: Mối quan hệ N-1. Một sở thích (UaThich) liên quan đến một sản phẩm (SanPham).
10. SanPham - NhaCungCap: Mối quan hệ N-1. Một sản phẩm (SanPham) được cung cấp bởi một nhà cung cấp (NhaCungCap).
11. SanPham - NhomSanPham: Mối quan hệ N-1. Một sản phẩm (SanPham) thuộc về một nhóm sản phẩm (NhomSanPham).

## Mô tả quy trình UML

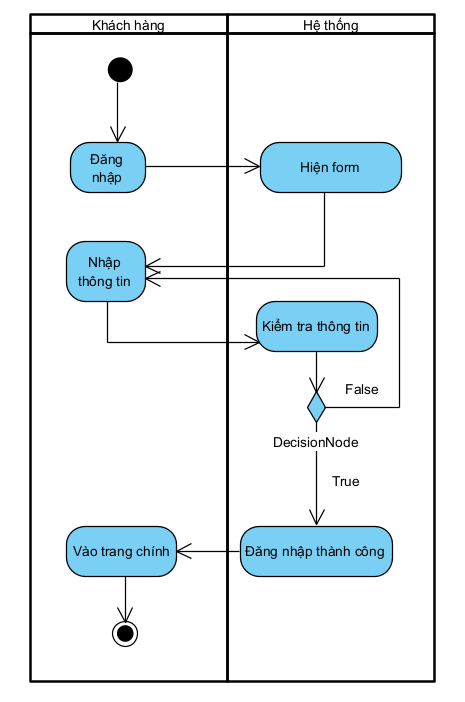
1. Khách Hàng Đăng Nhập hoặc Đăng Ký Tài Khoản

* KhachHang cung cấp thông tin đăng nhập hoặc đăng ký.
* Hệ thống kiểm tra thông tin đăng nhập hoặc tạo tài khoản mới và lưu vào bảng KhachHang.

**Use Case đăng nhập**

|  |  |
| --- | --- |
| **Tên Use Case:** | Đăng nhập |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã có tài khoản để đăng nhập vào hệ thống |
| **Hậu điều kiện:** | Khách hàng đăng nhập thành công |
| **Luồng sự kiện chính:**  1. Khách hàng chọn chức năng đăng nhập trên giao diện chính của hệ thống  2. Hệ thống hiển thị form đăng nhập  3. Khách hàng nhập thông tin  4. Hệ thống kiểm tra tính hợp lệ của thông tin  5. Hệ thống hiển thị giao diện chính tương ứng với các chức năng của tác nhân | |
| **Luồng sự kiện thay thế:**  4.1. Khách hàng nhập tài khoản hay mật khẩu không chính xác và quay lại bước 3 | |
| **Ngoại lệ:** | |

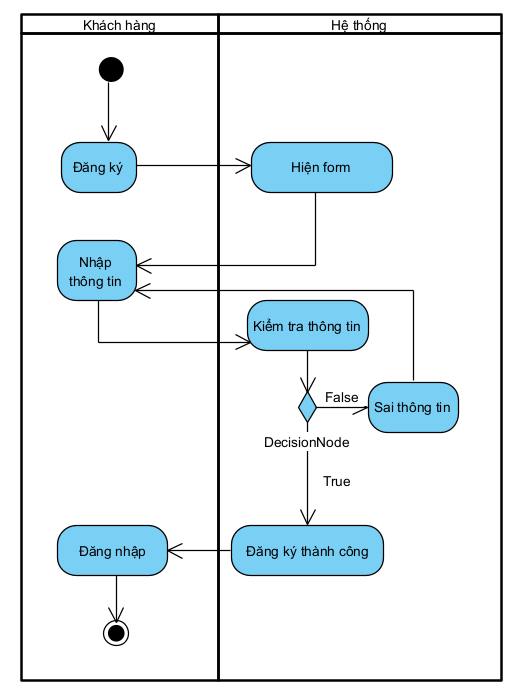
**Activity Diagram**

****

**Use Case đăng ký**

|  |  |
| --- | --- |
| **Tên Use Case:** | Đăng ký |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã truy cập vào trang |
| **Hậu điều kiện:** | Khách hàng đăng ký thành công |
| **Luồng sự kiện chính:**  1. Khách hàng truy cập vào trang  2. Khách hàng chọn chức năng đăng ký tài khoản  3. Hệ thống yêu cầu nhập thông tin đăng ký vào form  4. Khách hàng nhập thông tin vào đăng ký  5. Hệ thống kiểm tra thông tin, lưu vào cơ sở dữ liệu và thông báo đăng ký thành công | |
| **Luồng sự kiện thay thế:**  5.1. Hệ thống báo tên tài khoản bị trùng  5.1.1. Hệ thống thông báo thông tin nhập sai và quay lại bước 4 | |
| **Ngoại lệ:** | |

**Activity Diagram**

****

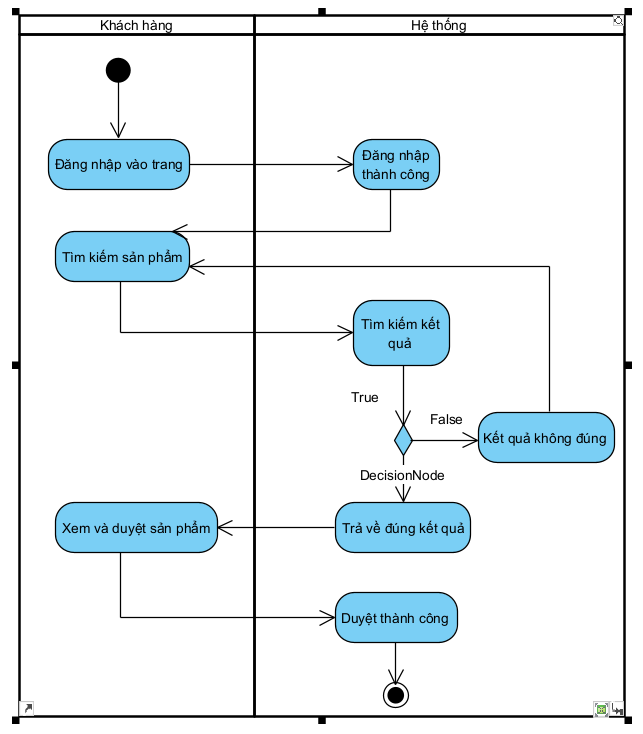
2. Khách Hàng Duyệt Sản Phẩm

* KhachHang duyệt qua các sản phẩm trong SanPham.
* SanPham lấy dữ liệu từ NhaCungCap và NhomSanPham để hiển thị thông tin chi tiết.

**Use Case duyệt sản phẩm**

|  |  |
| --- | --- |
| **Tên Use Case:** | Duyệt sản phẩm |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã đăng nhập vào hệ thống thành công |
| **Hậu điều kiện:** | Khách hàng duyệt được sản phẩm |
| **Luồng sự kiện chính:**  1. Khách hàng đăng nhập vào hệ thống thành công  2. Khách hàng chọn chức năng tìm kiếm và xem sản phầm cần duyệt  3. Hệ thống trả lại kết quả tương ứng với sản phẩm mà khách hàng cần tìm kiếm  4. Khách hàng chọn sản phẩm và xem chi tiết trước khi duyệt sản phẩm | |
| **Luồng sự kiện thay thế:**  1.1. Khách hàng tìm kiếm sản phẩm không hợp lệ  1.1.1. Hệ thống hiển thị thông báo không tìm thấy sản phẩm  1.1.2. Khách hàng tìm kiếm lại sản phẩm | |
| **Ngoại lệ:** | |

**Activity Diagram**



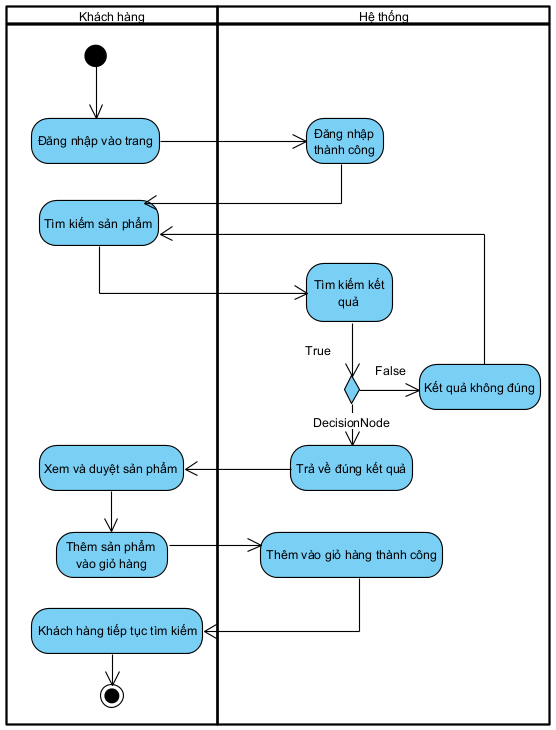
3. Khách Hàng Thêm Sản Phẩm vào Giỏ Hàng

* KhachHang chọn sản phẩm và số lượng.
* Hệ thống thêm sản phẩm và chi tiết vào ChiTietDonHang tạm thời.

**Use Case thêm sản phẩm**

|  |  |
| --- | --- |
| **Tên Use Case:** | Thêm sản phẩm |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã đăng nhập vào hệ thống thành công |
| **Hậu điều kiện:** | Khách hàng thêm sản phẩm vào giỏ hàng thành công |
| **Luồng sự kiện chính:**  1. Khách hàng đăng nhập vào hệ thống thành công  2. Khách hàng tìm kiếm sản phẩm và nhập tên sản phẩm cần thêm vào giỏ hàng  3. Hệ thống hiển thị danh sách sản phẩm tương ứng với sản phẩm khách hàng tìm kiếm  4. Khách hàng chọn sản phẩm và thêm vào giỏ hảng  5. Thêm vào giỏ hàng thành công và tiếp tục chọn | |
| **Luồng sự kiện thay thế:**  1.1. Khách hàng tìm kiếm sản phẩm không hợp lệ  1.1.1. Hệ thống hiển thị thông báo không tìm thấy sản phẩm  1.1.2. Khách hàng tìm kiếm lại sản phẩm | |
| **Ngoại lệ:** | |

**Activity Diagram**



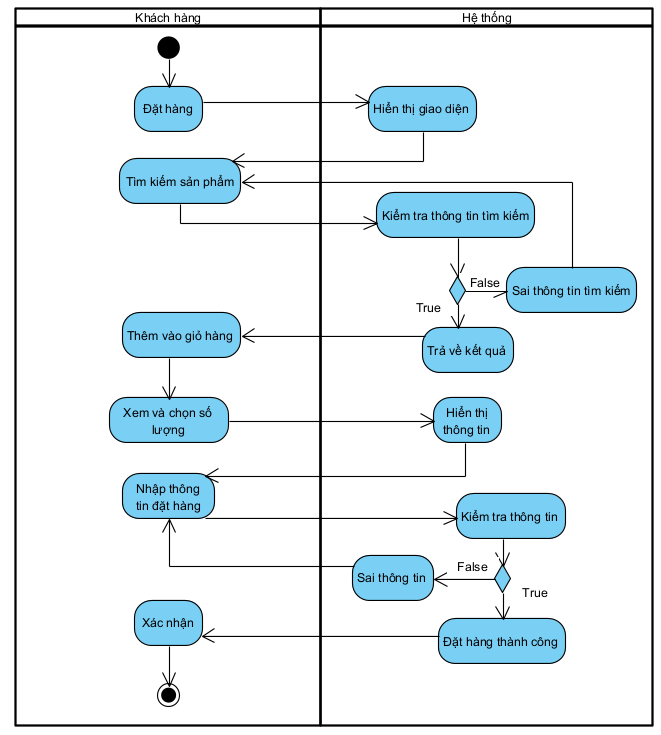
4. Khách Hàng Đặt Hàng

* KhachHang xác nhận đơn hàng, chọn phương thức thanh toán và vận chuyển.
* Hệ thống tạo một bản ghi mới trong DonHang với trạng thái chờ xử lý.
* Thông tin chi tiết đơn hàng từ ChiTietDonHang tạm thời được liên kết với DonHang.

**Use Case đặt hàng**

|  |  |
| --- | --- |
| **Tên Use Case:** | Đặt hàng |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã đăng nhập vào hệ thống thành công |
| **Hậu điều kiện:** | Khách hàng đặt hàng thành công |
| **Luồng sự kiện chính:**  1. Khách hàng chọn chức năng đặt hàng  2. Khách hàng tìm kiếm sản phẩm cần nhập thông qua việc nhập tên sản phẩm cần đặt  3. Hệ thống kiểm tra thông tin tìm kiếm và trả về kết quả  4. Khách hàng chọn sản phẩm cần đặt hàng và thêm vào giỏ hàng  5. Khách hàng chọn số lượng sản phẩm  6. Hệ thống kiểm tra thông tin  7. Khách hàng nhập đầy đủ thông tin và xác nhận đặt hàng  8. Hệ thống kiểm tra thông tin, lưu vào cơ sở dữ liệu và thông báo thành công | |
| **Luồng sự kiện thay thế:**  1.1. Khách hàng tìm kiếm sản phẩm không hợp lệ  1.1.1. Hệ thống hiển thị thông báo không tìm thấy sản phẩm  1.1.2. Khách hàng tìm kiếm lại sản phẩm  2.1. Khách hàng nhập sai thông tin  2.1.1. Hệ thống hiển thị thông báo không nhập sai  2.1.2. Khách hàng nhập lại thông tin | |
| **Ngoại lệ:** | |

**Activity Diagram**



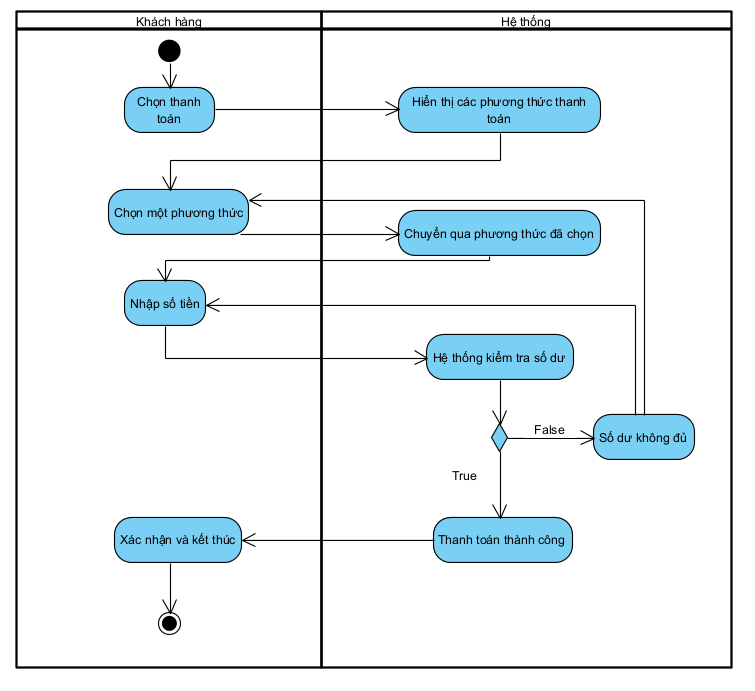
5. Thanh Toán

* KhachHang chọn phương thức thanh toán từ ThanhToan.
* Hệ thống cập nhật trạng thái thanh toán trong DonHang.

**Use Case thanh toán**

|  |  |
| --- | --- |
| **Tên Use Case:** | Thanh toán |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã đăng nhập vào hệ thống thành công |
| **Hậu điều kiện:** | Khách hàng thanh toán thành công |
| **Luồng sự kiện chính:**  1. Khách hàng chọn phương thức thanh toán  2. Hệ thống hiển thị các phương thức thanh toán  3. Khách hàng chọn  4. Hệ thống chuyển trang và nhập số tiền  5. Khách hàng nhập số tiền cần thanh toán  6. Hệ thống thông báo thanh toán thành công | |
| **Luồng sự kiện thay thế:**  1.1. Hệ thống thông báo thanh toán không thành công  1.1.1. Khách hàng nhập lại số tiền hoặc chuyển qua phương thức thanh toán khác | |
| **Ngoại lệ:** | |

**Activity Diagram**



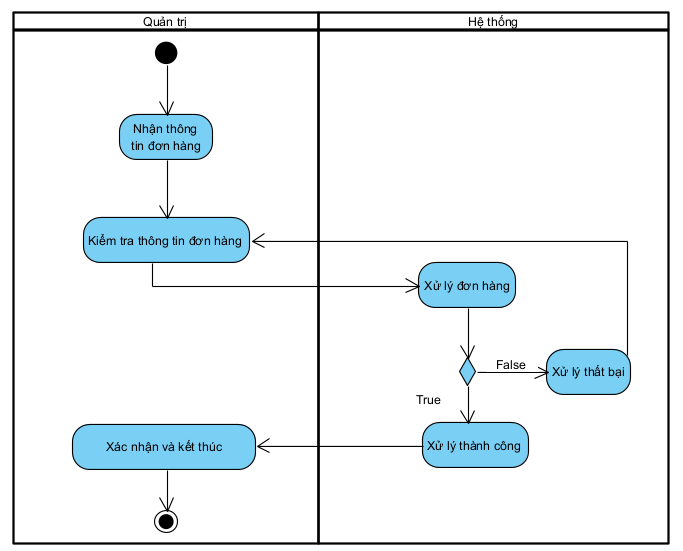
6. Xử Lý và Giao Hàng

* Đơn hàng được chuyển đến bộ phận xử lý.
* DonHang được cập nhật với thông tin vận chuyển từ VanChuyen.
* DonHang cập nhật trạng thái khi hàng được gửi đi và khi khách hàng nhận hàng.

**Use Case xử lý đơn hàng**

|  |  |
| --- | --- |
| **Tên Use Case:** | Xử lý đơn hàng |
| **Actor:** | Quản trị |
| **Tiền điều kiện:** | Đơn hàng đã được đặt thành công và lưu trong cơ sở dữ liệu |
| **Hậu điều kiện:** | Đơn hàng đã được xử lý và thông tin vận chuyển được cập nhật |
| **Luồng sự kiện chính:**  1. Quản trị xử lý đơn hàng từ hệ thống  2. Quản trị kiểm tra thông tin đơn hàng để đảm bảo tính hợp lệ và sẵn sàng để xử lý  3. Quản trị xử lý đơn hàng và chọn phương thức vận chuyển  4. Hệ thống cập nhật thông tin vận chuyển | |
| **Luồng sự kiện thay thế:**  1.1. Nếu thông tin đơn hàng không hợp lệ  1.1.1. Hệ thống sẽ thông báo lỗi và yêu cầu nhân viên xử lý kiểm tra lại | |
| **Ngoại lệ:** | |

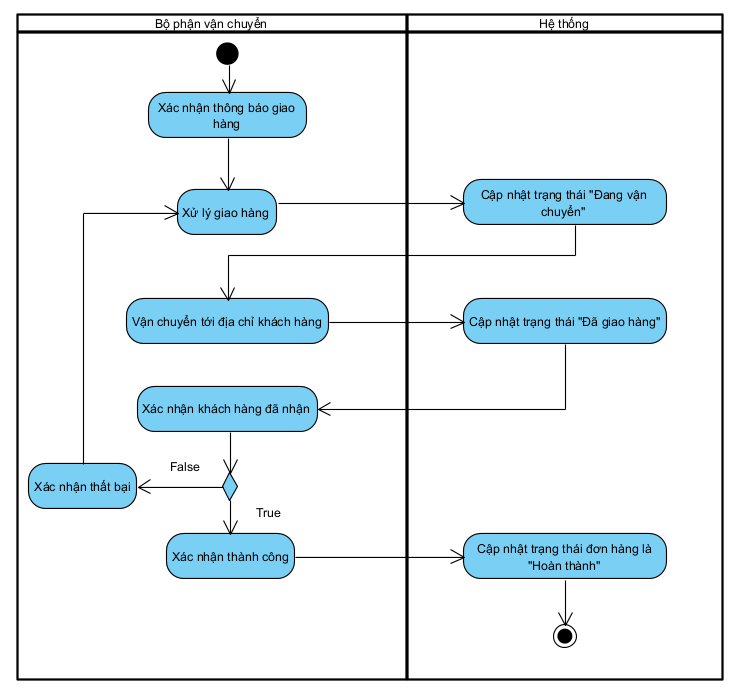
**Activity Diagram**



**Use Case giao hàng**

|  |  |
| --- | --- |
| **Tên Use Case:** | Giao hàng |
| **Actor:** | Bộ phận vận chuyển |
| **Tiền điều kiện:** | Đơn hàng đã được xử lý và thông tin vận chuyển đã được cập nhật |
| **Hậu điều kiện:** | Đơn hàng được giao thành công và trạng thái đơn hàng được cập nhật. |
| **Luồng sự kiện chính:**  1. Bộ phận vận chuyển nhận hàng và xác nhận đã gửi hàng đi.  2. Hệ thống cập nhật trạng thái đơn hàng là "Đang vận chuyển".  3. Bộ phận vận chuyển giao hàng đến địa chỉ khách hàng.  4. Khách hàng xác nhận đã nhận hàng.  5. Hệ thống cập nhật trạng thái đơn hàng là "Hoàn thành". | |
| **Luồng sự kiện thay thế:**  1.1. Nếu khách hàng thông báo không nhận được hàng,  1.1.1. Hệ thống sẽ cập nhật trạng thái đơn hàng là "Giao hàng thất bại" và thông báo cho bộ phận xử lý để điều tra và xử lý lại đơn hàng. | |
| **Ngoại lệ:** | |

**Activity Diagram**

****

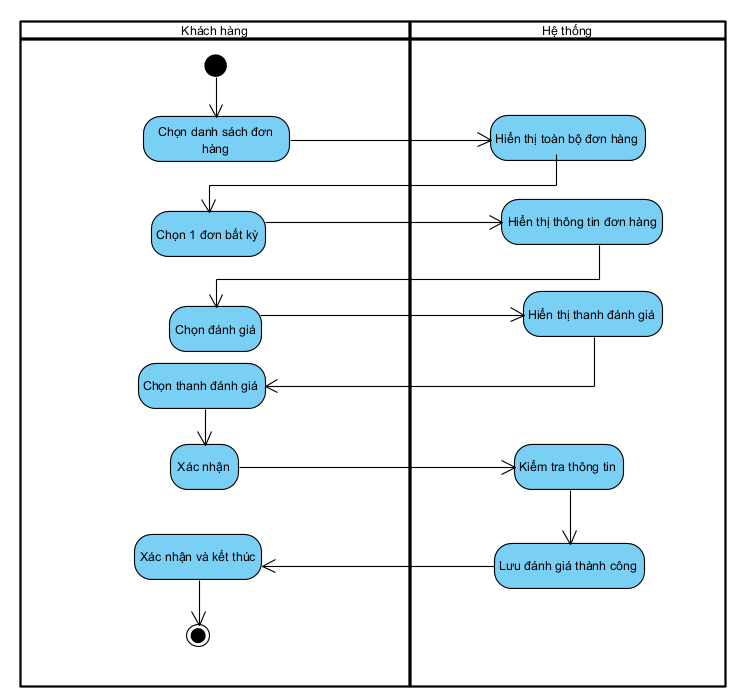
7. Khách Hàng Đánh Giá

* Sau khi nhận hàng, KhachHang có thể đánh giá sản phẩm.
* Đánh giá được lưu vào DanhGia và liên kết với SanPham và KhachHang.

**Use Case dánh giá**

|  |  |
| --- | --- |
| **Tên Use Case:** | Đánh giá |
| **Actor:** | Khách hàng |
| **Tiền điều kiện:** | Khách hàng đã đăng nhập vào hệ thống thành công và thanh toán thành công |
| **Hậu điều kiện:** | Đánh giá được lưu thành công |
| **Luồng sự kiện chính:**  1. Khách hàng chọn danh sách đơn hàng đã thanh toán thành công  2. Hệ thống hiển thị danh sách các đơn hàng đã thanh toán thành công  3. Khách hàng chọn 1 đơn hàng từ danh sách  4. Khách hàng chọn hiển thị thông tin đơn hàng đó  5. Khách hàng chọn đánh giá  6. Hệ thống hiển thị mức đánh giá  7. Khách hàng chọn điểm và xác nhận lưu đánh giá của mình  8. Hệ thống kiểm tra và lưu vào cơ sở dữ liệu, trả về thông báo đánh giá thành công | |
| **Luồng sự kiện thay thế:** | |
| **Ngoại lệ:** | |

**Activity Diagram**



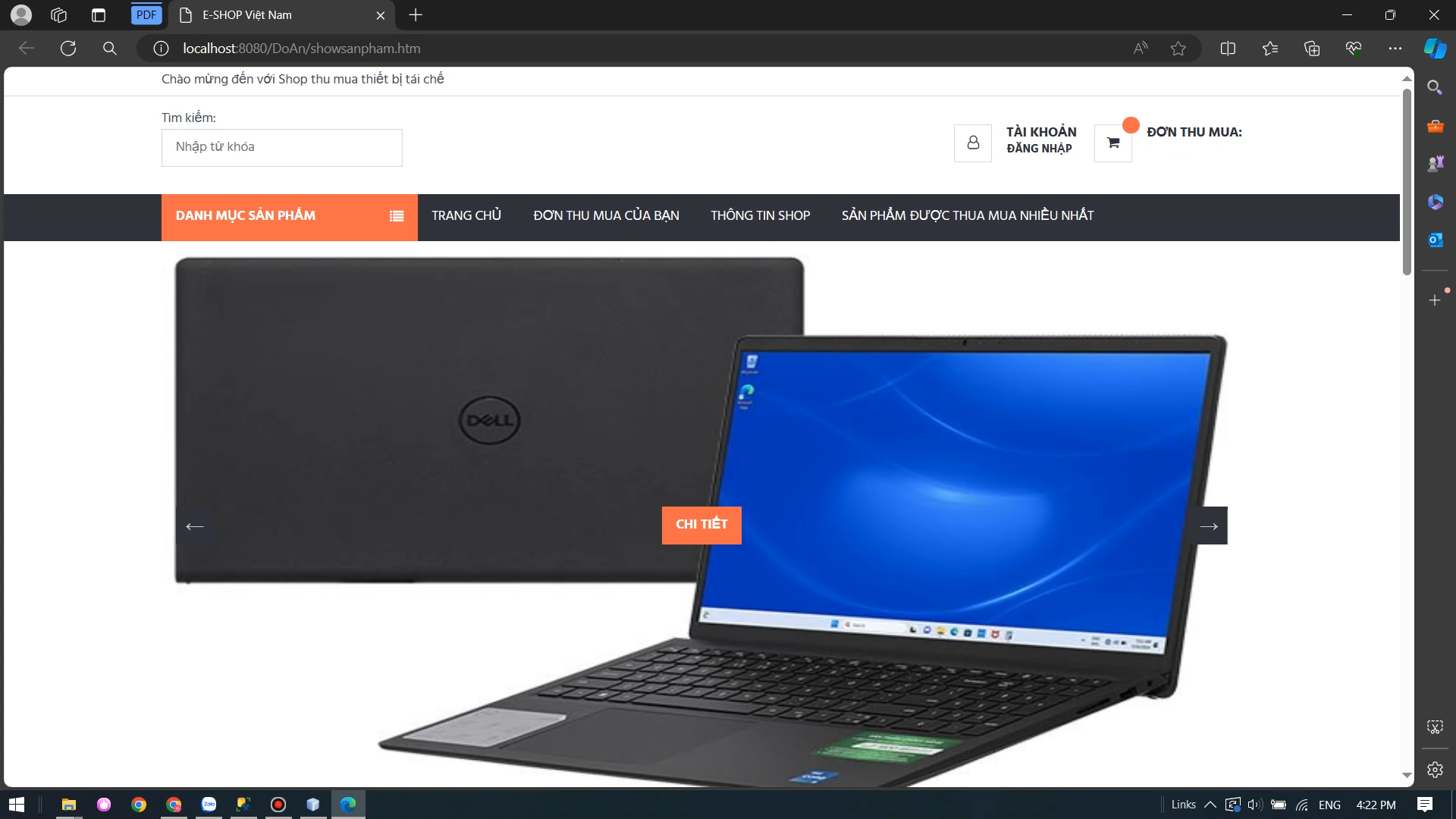
8. Cập Nhật Trạng Thái Đơn Hàng

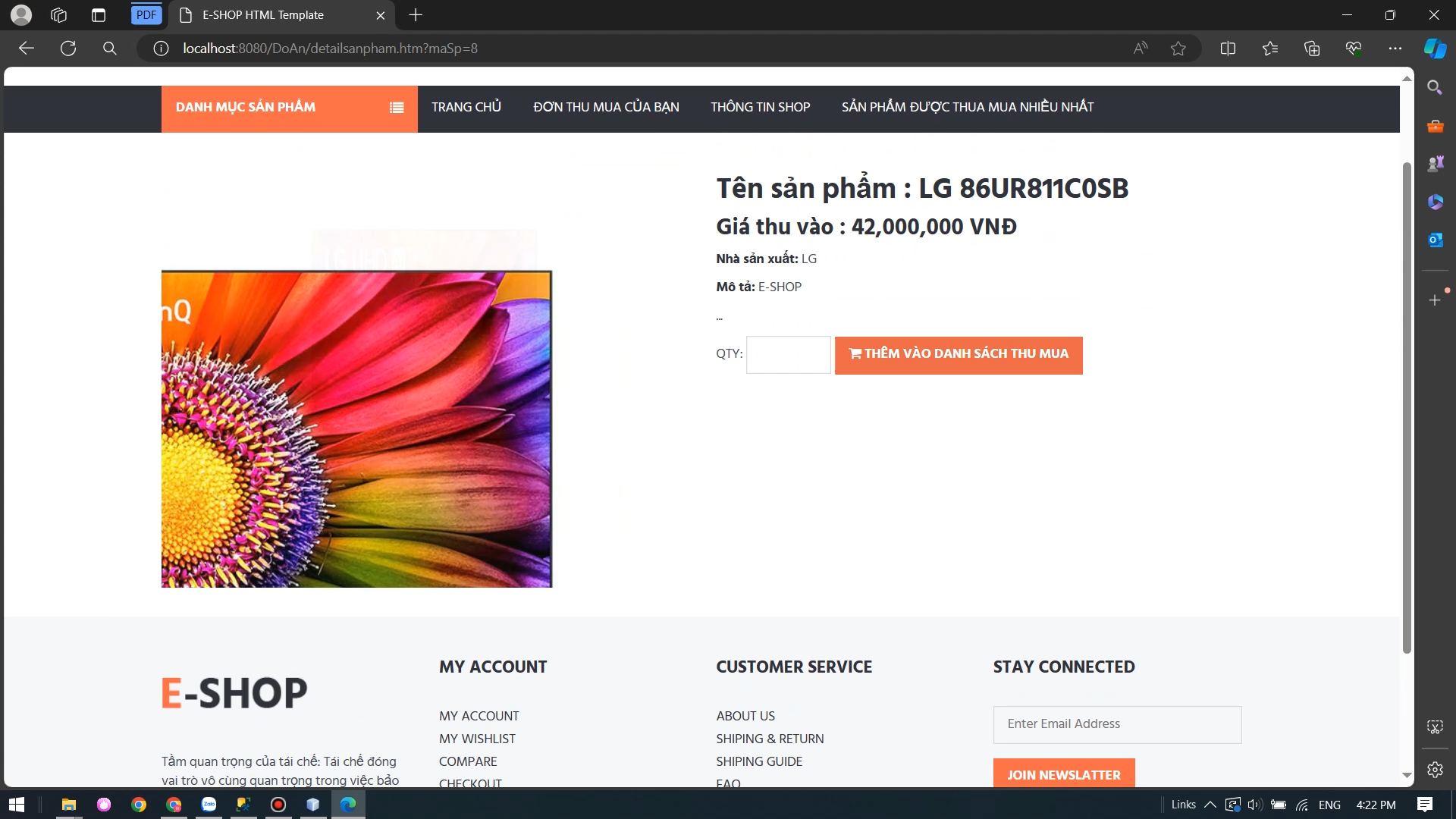
* Hệ thống cập nhật trạng thái cuối cùng của đơn hàng trong DonHang.

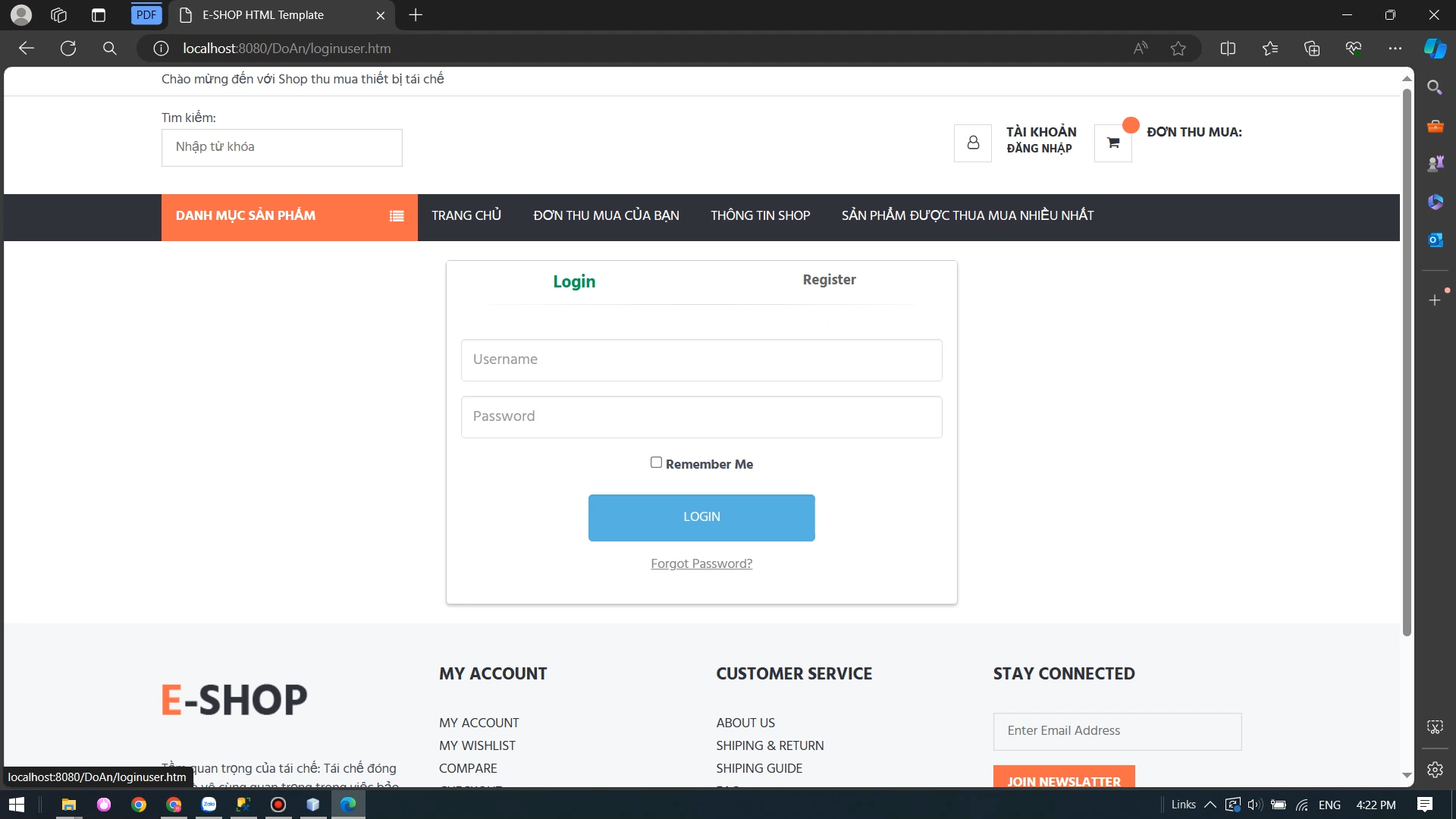
## Công cụ tạo mô hình UML

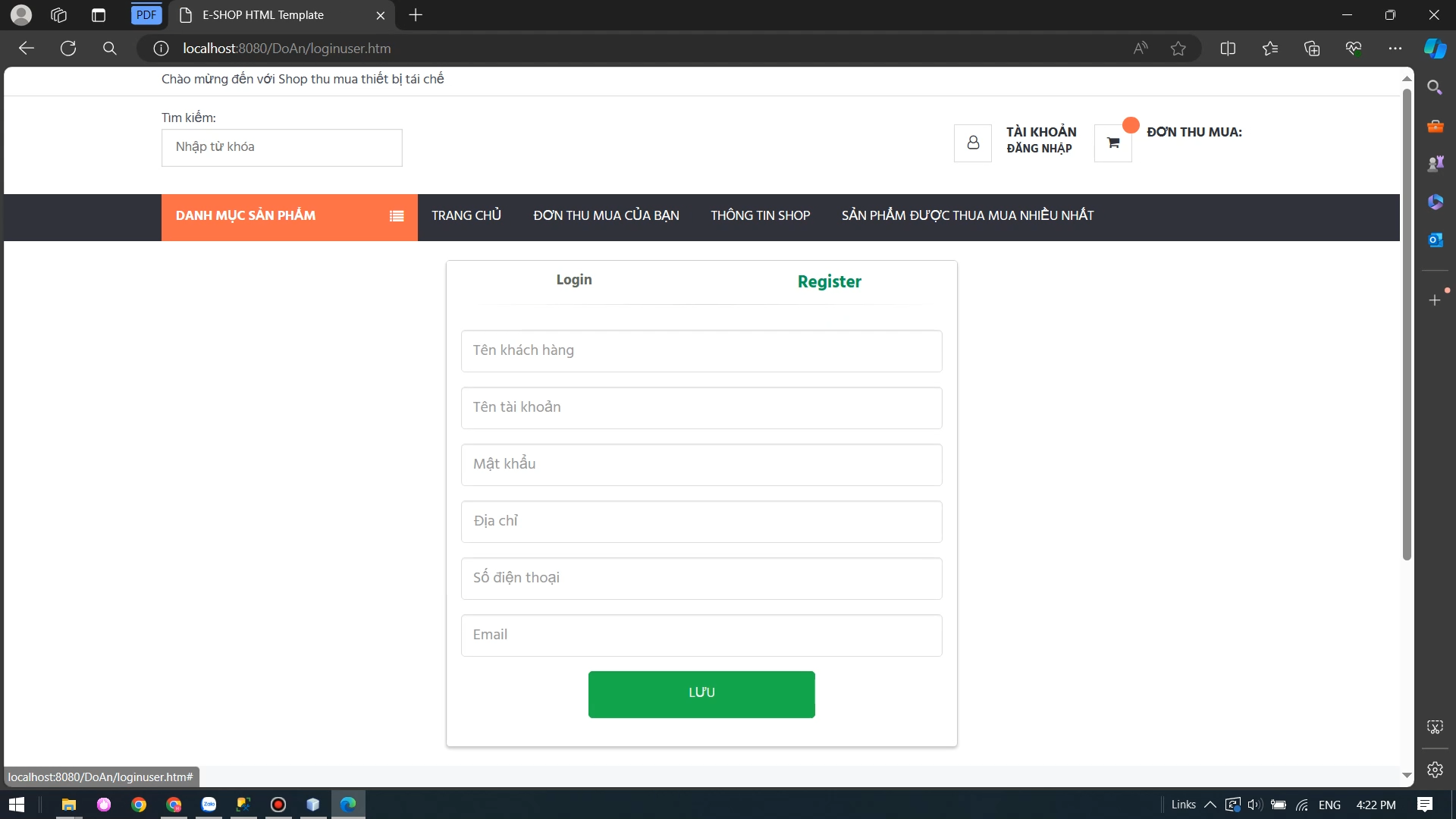
* StarUML
* UMLet
* Visual Paradigm

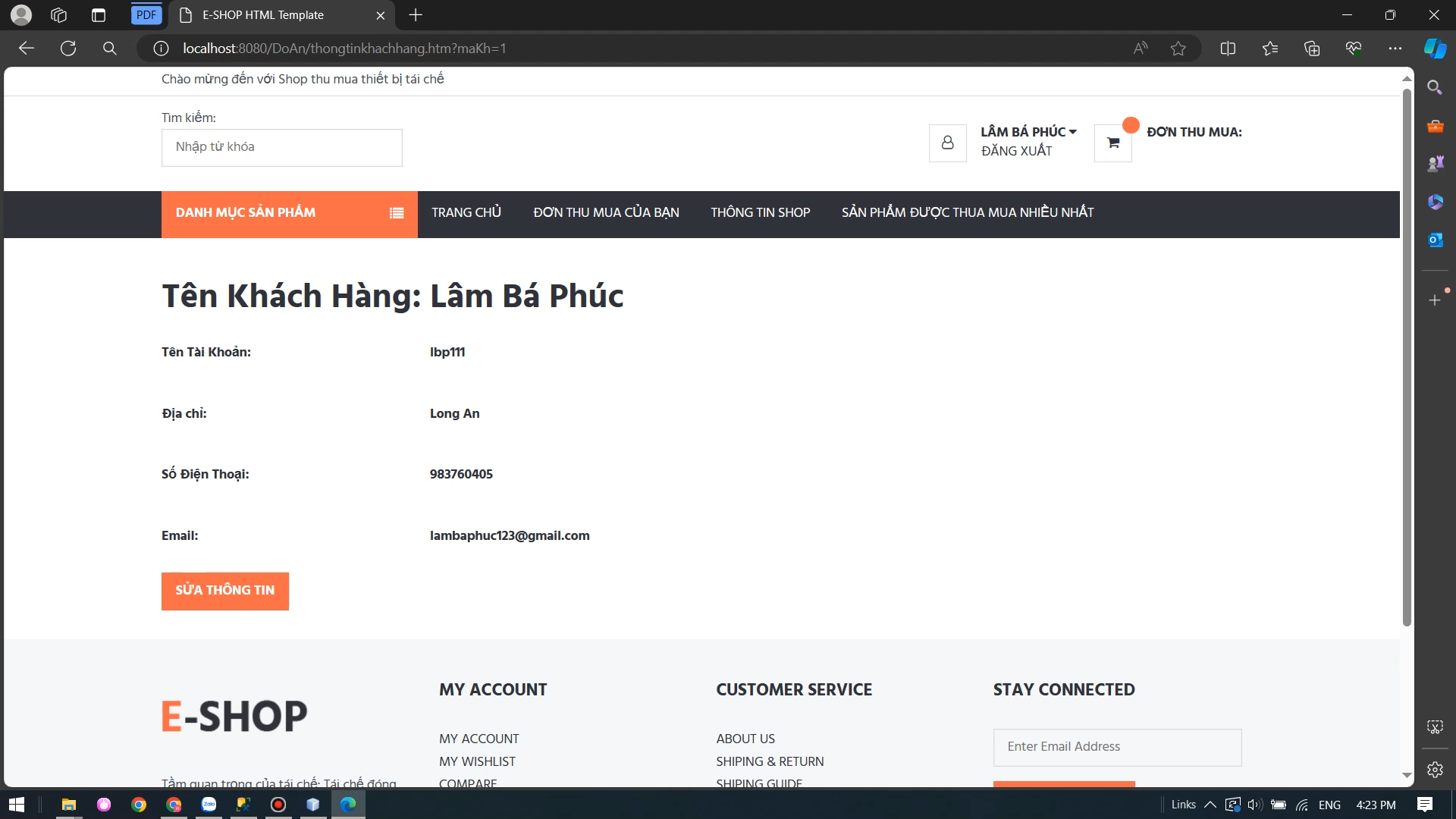
## Lập trình

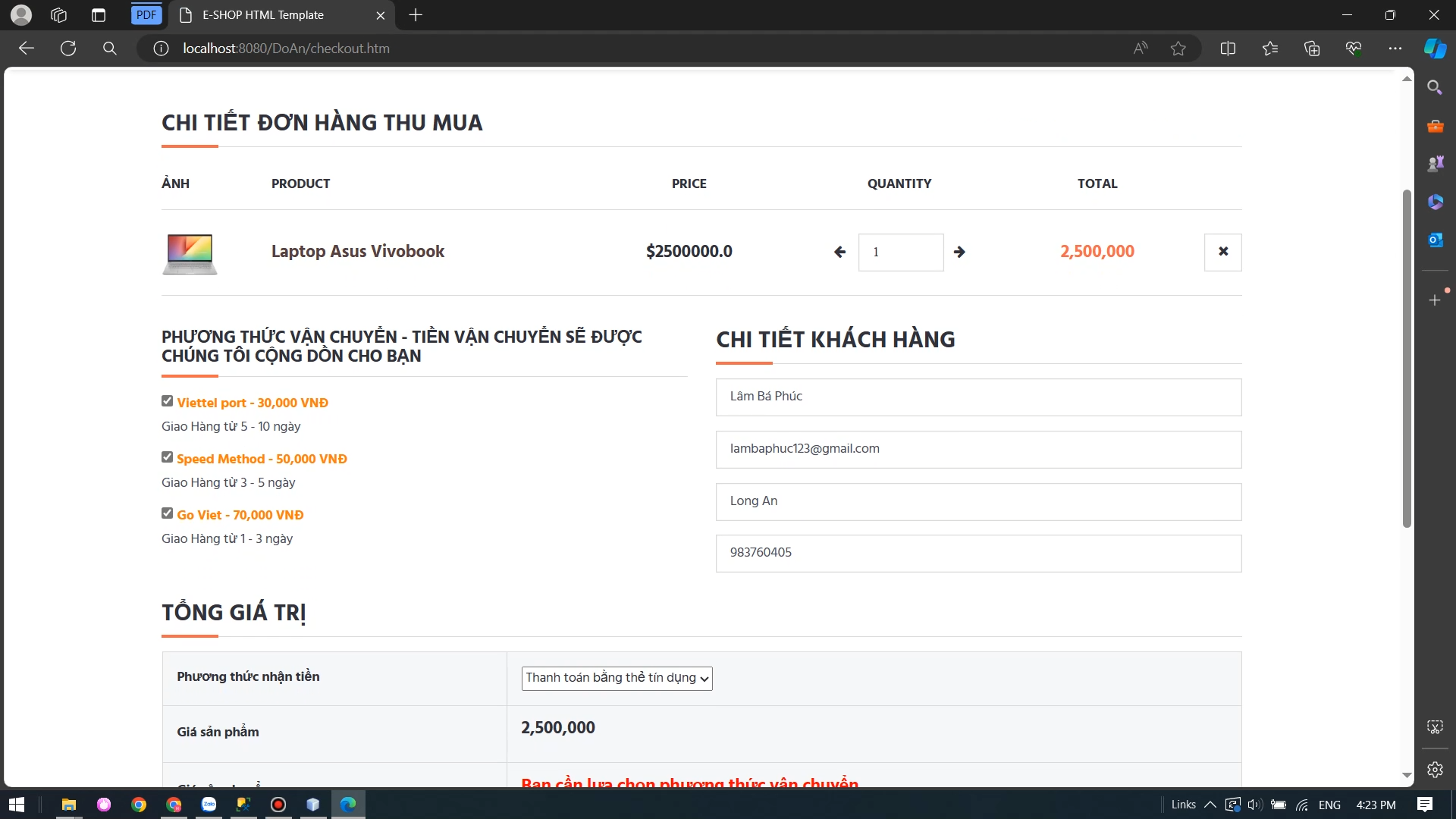


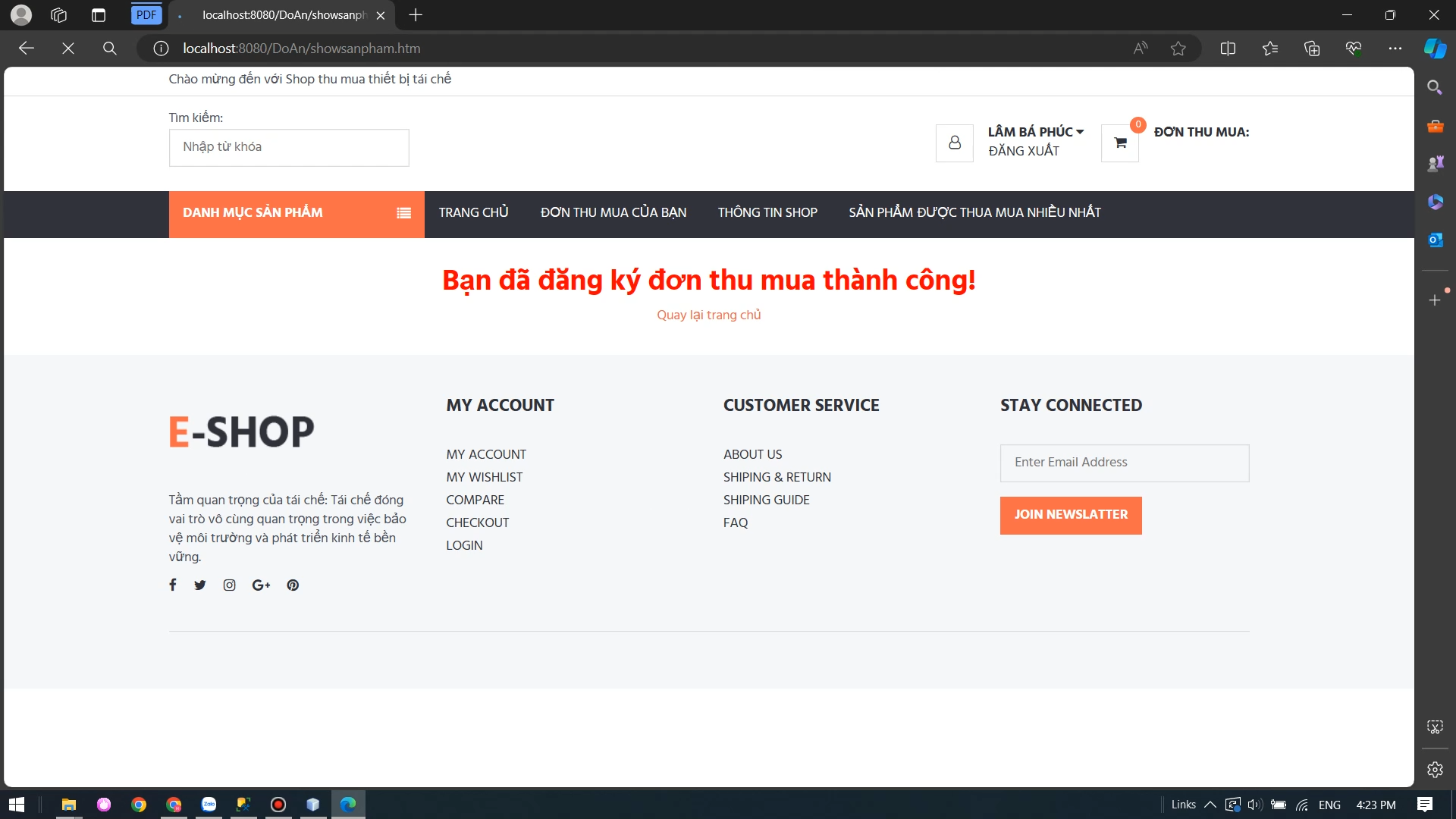


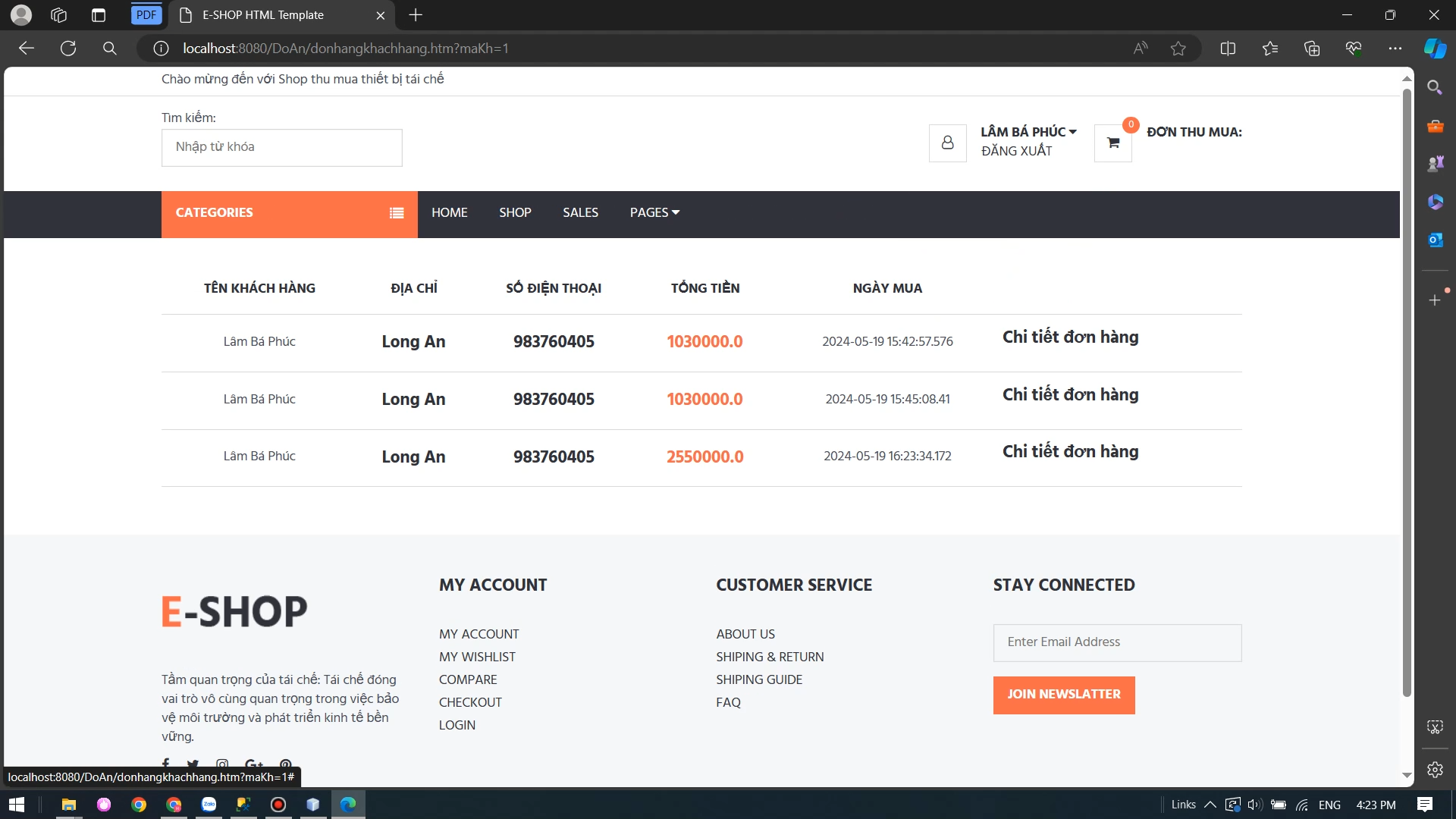












## Video DEMO

Link youtube: <https://youtu.be/r9vqEQv8zI0>

Link drive: https://drive.google.com/file/d/1gWyrHGKIufmryyJiEUVVjuan9Fj2Kcd7/view?usp=sharing