**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

****

**BÀI TẬP LỚN**

**MÔN: LẬP TRÌNH WEB ASP.NET**

**ĐỀ TÀI: XÂY DỰNG WEBSITE BÁN QUẦN ÁO PHỤ KIỆN NAM NỮ TÍCH HỢP MACHINE LEARNING**

**Giáo viên giảng dạy: Cô Nguyễn Kim Sao**

**Sinh viên thực hiên:**

* **Trần Đức Cừ (MSV: 151201506)**
* **Nguyễn Minh Đạt (MSV: 151202716)**
* **Trần Văn Hoàn (MSV: 151202513)**

**Lớp: Công nghệ thông tin 1-K56**

**LỜI NÓI ĐẦU**

Với sự phát triển không ngừng của ngành công nghệ thông tin, truyền thông nói chung và internet nói riêng, ngày nay website đóng một vai trò thiết yếu trông hoạt động kinh doanh của mỗi doanh nghiêm và cả các cơ quan báo trí, nó mang lại lợi ích to lớn cho các doanh nghiệp và cơ quan báo trí mà không thể phủ nhận.

Website ra đời đóng vai trò quan trọng trong các hoạt động của doanh nghiệp nằm quản bá sản phẩm, giới thiệu, dịch vụ đến với tất cả mọi người mà chi phí ở mức thấp nhất. Qua đó các doanh nghiệp có thể mở rộng thi trường một cách nhanh chóng, các cơ quan báo trí có thể thu hút thêm đọc giả, cũng như cập nhật tin tức một cách nhanh chóng.

Việc xây dựng một website bán hàng và muốn cạnh tranh tốt thì việc gợi ý cho người dùng về sản phẩm là điều cần thiết do vậy cần một hệ thống để có thể tính toán mối tương quan giữa sản phẩn và người dùng để thực hiện điều đó. Do vậy nhóm chúng em đã phát triển một hệ thống recommender và thông qua bộ dữ liệu của chi tiết bán hàng và tính toán tương quan giữa sản phẩn và user mỗi khi user đăng nhập vào website và hiển thị ở trang index.

**Chương I. ASP.NET MVC**

***1.1 MVC là gì?***

MVC là viết tắt của Model – View – Controller. Là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nói cho dễ hiểu, nó là mô hình phân bố source code thành 3 phần, mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

Các thành phần trong MVC:

* Controller: Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng… Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.
* Model: Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…
* View: Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images… Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.

***1.2 ASP.NET là gì?***

ASP.NET là một nền tảng ứng dụng web (web application framework) được phát triển và cung cấp bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft), cho phép những người lập trình tạo ra những trang web động, những [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) và những [dịch vụ web](https://vi.wikipedia.org/wiki/D%E1%BB%8Bch_v%E1%BB%A5_web). Lần đầu tiên được đưa ra thị trường vào tháng 2 năm 2002 cùng với phiên bản 1.0 của.NET framework, là công nghệ nối tiếp của Microsoft's Active Server Pages(ASP). ASP.NET được biên dịch dưới dạng [Common Language Runtime](https://vi.wikipedia.org/w/index.php?title=Common_Language_Runtime&action=edit&redlink=1) (CLR), cho phép những người lập trình viết mã ASP.NET với bất kỳ ngôn ngữ nào được hỗ trợ bởi.NET language.

***1.3 ASP.NET MVC là gì?***

ASP.NET MVC là một nền tảng ứng dụng web được phát triển bởi Microsoft, hỗ trợ mô hình MVC (Model-View-Controller). Đó là một phần mềm mã nguồn mỡ và là một thành phần của ASP.NET.

***1.4 Lịch sử phát triển.***

ASP.NET MVC 1.0 - Công nghệ ASP.NET MVC là mới, tuy nhiên nó có lịch sử lâu đời. Nền tảng MVC được phát minh bởi Trygve Renskaug, ông đã viết trang giấy đầu tiên nói về MVC vào năm 1978. Ban đầu nó được gọi là Trang 5 Thing Model View Editor pattern, nhưng sau đó nó được đặt tên lại là Model View Controller pattern.

ASP.NET MVC lần đầu tiên xuất hiện đó là trong dự án mã nguồn mở MonoRail;

Và nguồn gốc thật sự để cho ra đời công nghệ Microsoft ASP.NET MVC là do Scott Guthrie (một trong những nhà sáng tạo ra ASP.NET) trên chuyến bay đến Austin, Texas để nói về hội thảo Alt.NET vào tháng 10-2007.

Cuối cùng thì đầu năm 2009 phiên bản ASP.NET MVC 1.0 (released) được ra đời.

Điều kiện cần để xây dựng ứng dụng ASP.NET MVC:

* Microsoft.Net Framework 3.5 SP 1
* Microsoft ASP.NET MVC.
* Visual Studio 2008 SP 1.

Phiên bản mới nhất là ASP.NET MVC5, được hỗ trợ trên Visual Studio 2015 trở lên.

**Chương II. Phân tích và thiết kế hệ thống**

**I. Phân thích hệ thống**

**1.1 Giới thiệu bài toán**.

Việc kinh doanh - mua bán là nhu cầu không thể thiếu đối với mỗi chúng ta. Trong thời đại cạnh tranh hiện nay việc giới thiệu sản phẩm kinh doanh đến từng cá nhân với chi phí thấp, hiệu quả cao mua sắm với những sản phẩm đa chủng loại, đạt chất lượng và hợp túi tiền của người tiêu dùng vì vậy thương mại điện thử đã được ra đời và dần phát triển trên toàn thế giới

Việc phổ biến các sản phẩm của cửa hàng kinh doanh đến khách hàng thông qua các bảng báo giá tuy nhiên chi phí quá cao vì số lượng sản phẩm ngày một đa dạng và giá cả thay đổi liên tục tính phổ biến không cao chưa đáp ứng được nhu cầu người dùng. Mặt khác cửa hàng còn gặp nhiều khó khăn như chưa qua quản lý sản phẩm…. Nắm bắt được tình hình trên em tiến hành thiết kế trang web mua bán hàng online để mong sao giúp cho các cửa hàng phát triển nhanh hơn.

Việc gợi ý sản phẩm để nhằm cạnh tranh và đưa lại sự tiện lợi nhất cho người dùng. Người dùng có thể nhận được những gợi ý gần như chính xác tới 70 – 80 % về những sản phẩm mà họ thích và có như cầu mua và sẽ tạo được sự tin tưởng và đánh giá ca từ người dùng. Nếu phát triển thành công hệ thống này cũng sẽ đem lại nguồn lợi nhuận không hề nhỏ. Vì vậy, với nền công nghệ phát triển như hiện nay thì mỗi một trang website bán hàng cần có một hệ thống gợi ý như vậy

**1.2 Sơ đồ chức năng.**

Sơ đồ phân giã chức năng

Quản lý website bán hàng

Gợi ý sản phẩm cho user

Xử lý tìm kiếm

Gửi thư phản hồi

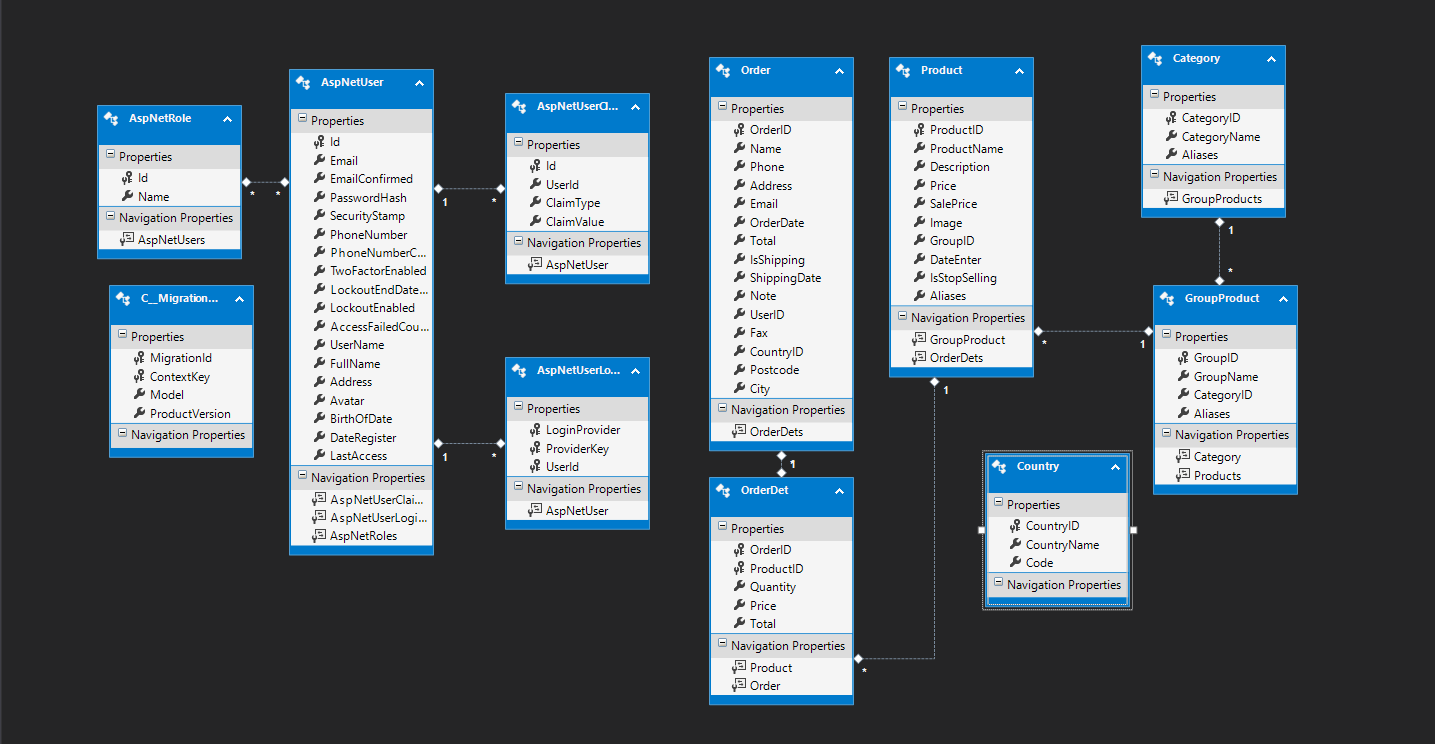
Xử lý giỏ hàng

Quản lý sản phẩm

Quản lý qua trang Admin

**1.3 Cơ sở dữ liệu**

Mô hình dữ liệu quạn hệ

****

**II. Thiết kế hệ thống.**

**1. Cài đặt hệ thống Recommender**

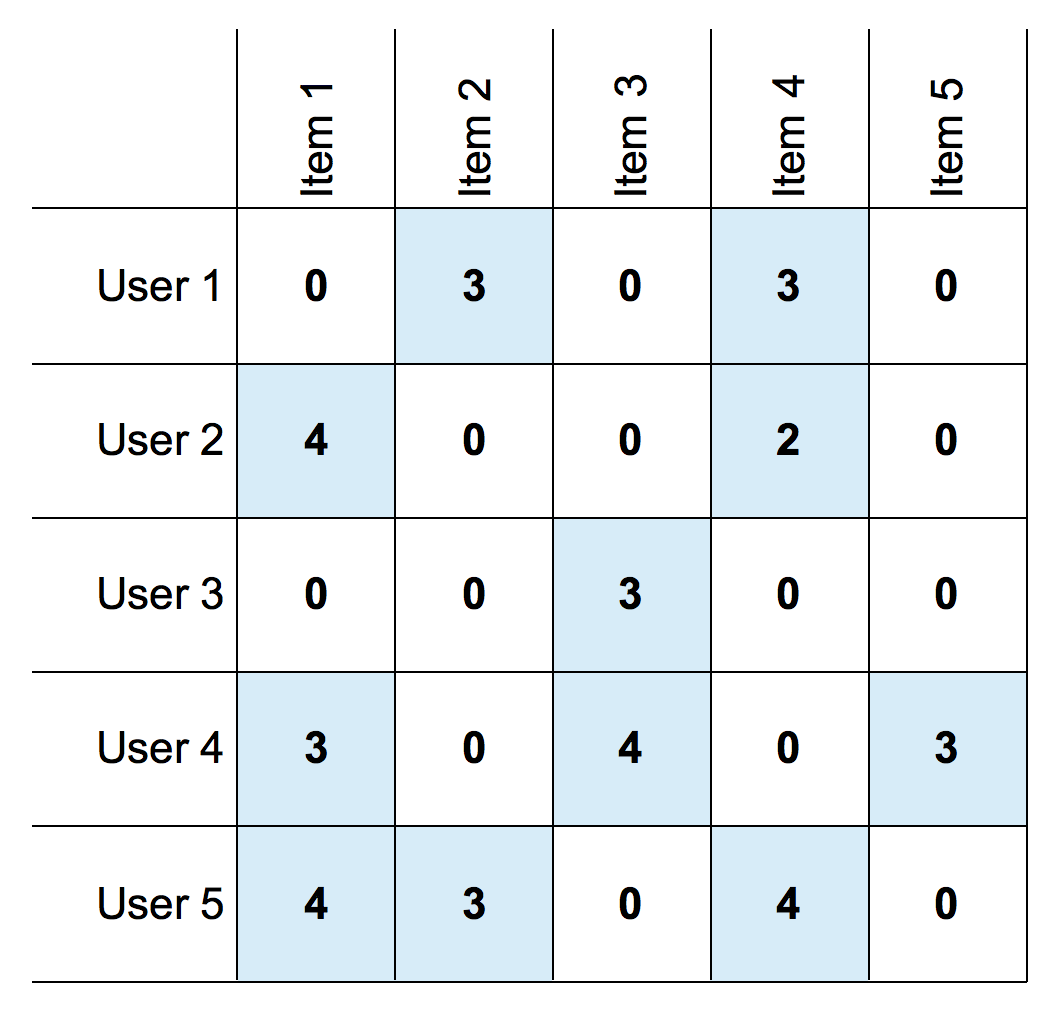
**a. Giới thiệu về hệ thống**

Một trong những công nghệ giúp cho đời sống con người ngày càng dễ dàng hơn đó là recommender system. Recommender system giúp kết nối người dùng với sản phẩm mà họ tìm kiếm được thuận tiện và nhanh chóng hơn từ đó mang lại lợi thế cạnh tranh của sản phẩm so với các đối thủ khác. Bạn có thể bắt gặp hệ thống này ở các trang xem phim, nghe nhạc, mua bán, mạng xã hội, … Về cơ bản, bạn có thể áp dụng ngay kỹ thuật này vào hệ thống của bạn thông qua các hướng tiếp cận như: Popularity – liệt kê top các sản phẩm được nhiều người quan tâm nhất, Classification – dựa vào các chủ đề mà bạn cung cấp để lọc ra danh sách sản phẩm tương ứng. Tuy nhiên, các hướng tiếp cận này đều mang tính đại chúng, không nhắm vào một cá nhân cụ thể nào. Hơn nữa, không ai dễ dàng cung cấp thông tin cá nhân cho ứng dụng của bạn để có thể lọc thông tin phù hợp.

Do đó, chúng em sẽ đi theo hướng tiếp cận Collaborative Filtering với hai phương pháp gồm Memory-Based Collaborative Filtering và Model-Based Collaborative filtering giúp trả lời hai câu hỏi “user mua sản phẩm này thì sẽ có xu hướng mua những sản phẩm nào…” và “user có gu ăn mặc như bạn thì sẽ có xu hướng nghe mua sản phẩm nào …”.

**b. Cài đặt hệ thống**

* Cài đặt Memory-Based Collaborative: để có thể áp dụng phương pháp user – item filtering và item-item filtering, ta cần xây dựng ma trận user-item



Load dữ liệu từ OrderDet tạo matrix tương quan giữa user và item

1. **void** LoadData()
2. {
3. products = **new** Dictionary<**int**, **int**>();
4. users = **new** Dictionary<**string**, **int**>();
6. StoreOlineEntities db = **new** StoreOlineEntities();
7. var listProduct = db.Products.ToList();
8. **int** ii = 0;
9. **foreach**(var item **in** listProduct) products.Add(item.ProductID, ii++);
11. var temp = db.OrderDets.Select(t => **new** UserToProduct() {UserID = t.Order.UserID,ProductID = t.ProductID,Quantity = t.Quantity }).ToList();
12. var buffer = temp.GroupBy(t => t.UserID).ToList();
13. **for** (**int** i = 0; i < buffer.Count; i++) users.Add(buffer[i].Key, i);
15. N = users.Count;
16. M = products.Count;
17. data = **new** **int**[N, M];
19. **for** (**int** i = 0; i < N; i++)
20. {
21. var item = buffer[i].GroupBy(t => t.ProductID).Select(t => **new** { ID = t.Key, Total = t.Sum(o => o.Quantity) }).ToList();
22. **int** index = 0;
23. **int** j = 0;
24. **foreach** (var item1 **in** products)
25. {
26. **if** (item[index].ID == item1.Value)
27. {
28. **if** (j >= M) **break**;
29. data[i, j] = item[index].Total;
30. index++;
31. j++;
32. }
33. }
34. }
36. }

* Sau khi xây dựng được ma trận user-item, ta sẽ xây dựng ma trận khoảng cách để tính độ tương tự (similarity distance) giữa các item và user lẫn nhau. Ta có nhiều độ đo khoảng cách để áp dụng tính toán. Thông thường, ta sử dụng độ đo khoảng cách cosine để tính. Cosine similiarity giữa user k và a được tính dựa vào công thức như bên dưới. Trong đó, x_{k,m} là số lượng sản phẩm m của user k, x_{a,m} là số lượng mua sản phẩm m của user a.

****

1. **double** user\_similarity(**int** u1, **int** u2)
2. {
3. **double** res = 0;
4. **int** sum1 = 0;
5. **int** sum2 = 0;
6. **int** sum3 = 0;
7. **for** (**int** i = 0; i < M; i++)
8. {
9. sum1 += (data[u1, i] \* data[u2, i]);
10. sum2 += (data[u1, i] \* data[u1, i]);
11. sum3 += (data[u2, i] \* data[u2, i]);
12. }
14. **double** tmp = (Math.Sqrt(sum2) \* Math.Sqrt(sum3));
15. res = (**double**)sum1 / tmp;
16. **return** res;
17. }

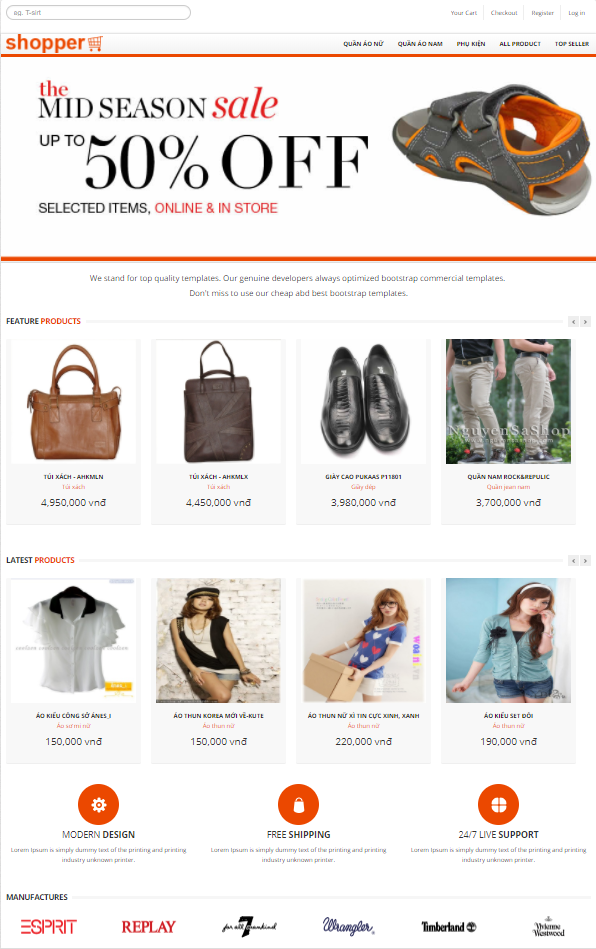
* Để có thể dự đoán số lượng mua sản phẩm m của user k. Số lượng mua càng cao tương đương với user quan tâm đến sản phẩm này càng nhiều. Ta có thể sắp xếp lại kết quả dự đoán để cho ra các sản phẩm gợi ý cho user này. Công thức dự đoán theo user-based CF được tính như bên dưới:

****

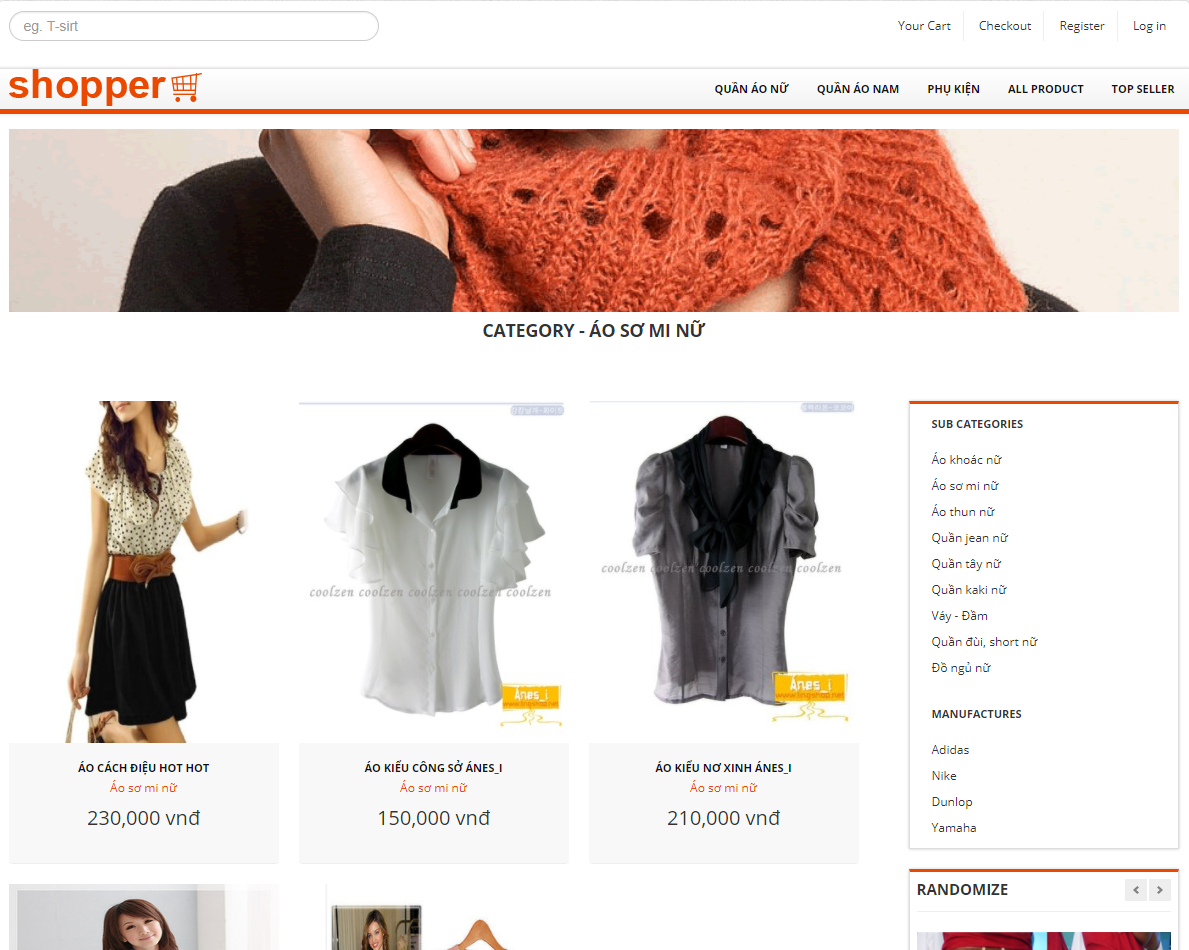
1. **double** predict(**int** k, **int** m)
2. {
3. **double** res = AVG(k);
4. **double** sum1 = 0;
5. **double** sum2 = 0;
6. **for** (**int** i = 0; i < N; i++)
7. {
8. sum1 += user\_similarity(k, i) \* (data[i, m] - AVG(i));
9. sum2 += Math.Abs(user\_similarity(k, i));
10. }
11. **if** (sum2 != 0) res += sum1 / sum2;
12. **return** res;
13. }

**2. Cài đặt website**

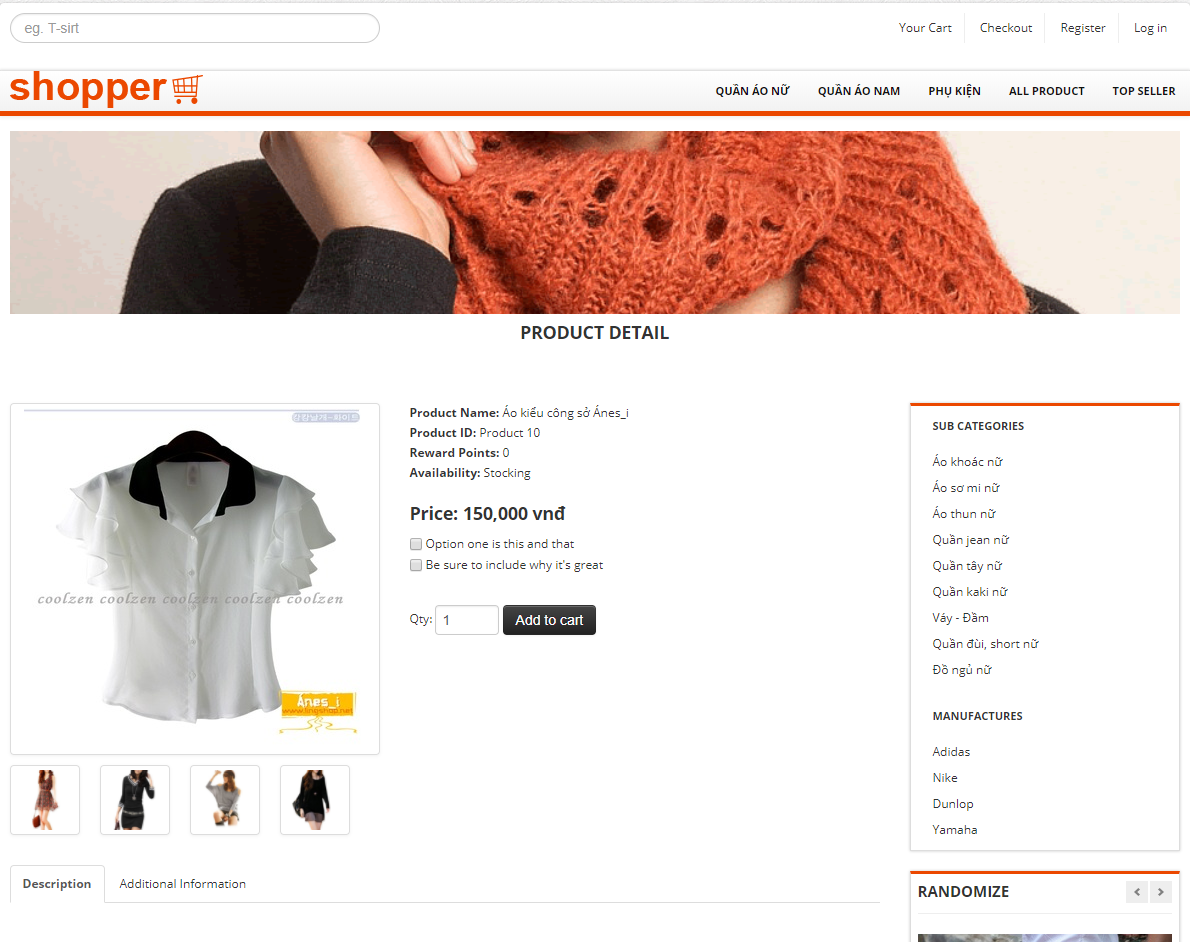
* Trang chủ gồm các chức năng:
  + menu tùy chọn
  + Hiển thị các sản phẩm nổi bật và gợi ý nếu có user đăng nhập
  + Hiển thị các sản phẩm mới nhất.
  + Một số thông tin khác và các mục để điều hướng.



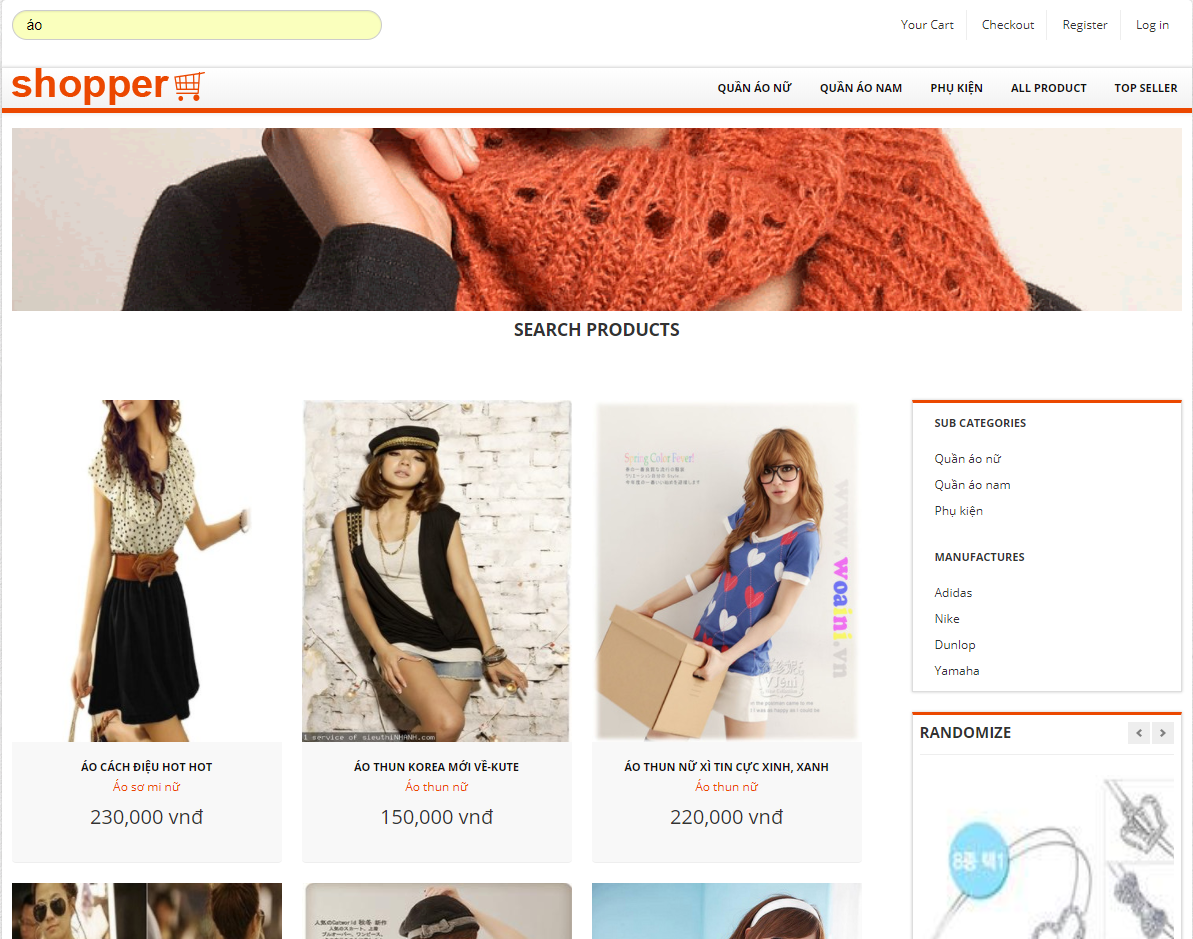
* Loại sản phẩm gồm các chức năng:
  + Hiển thị các sản phẩm theo loại sản phẩm đã chọn.
  + Hiển thị một menu gồm các loại sản phẩm liên quan.
  + Một số sản phẩm random và được bán nhiều nhất



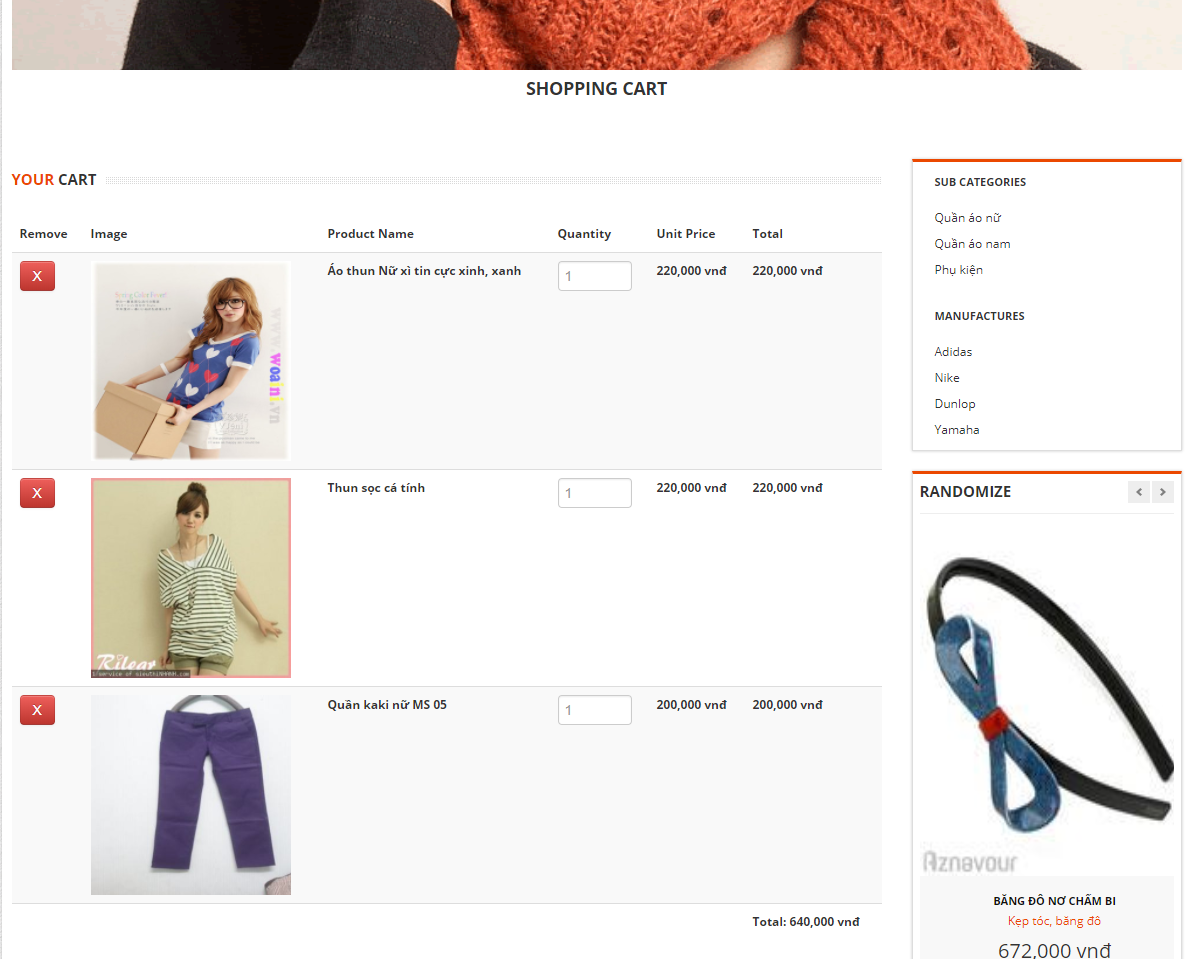
* Chi tiết sản phẩm gồm các chức năng:
  + Hiển thị chi tiết của một mặt hàng
  + Có nút thêm sản phẩm vào giỏ đồ
  + Hiển thị một số sản phẩm có liên quan đến sản phẩm đó



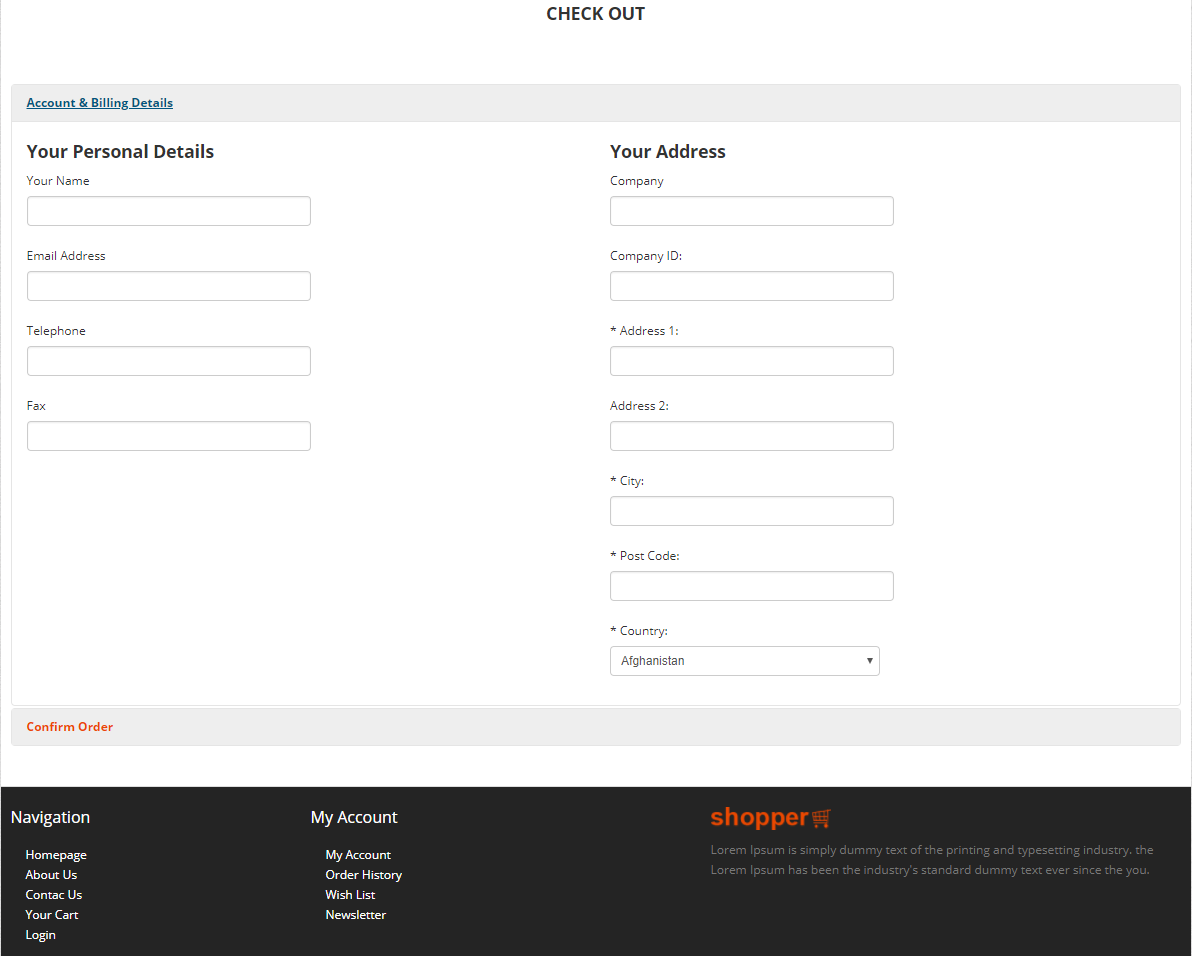
* Chức năng tìm kiếm sản phẩm theo tên và bí danh: Người dùng có thể nhập vào chuỗi ký tự liên quan đến tên sản phẩm hoặc bí danh của sản phẩm đó để tìm kiếm. Sau khi tìm kiếm sẽ trả về một danh sách các sản phẩm với tên và bí danh chứa ký tự vừa nhập



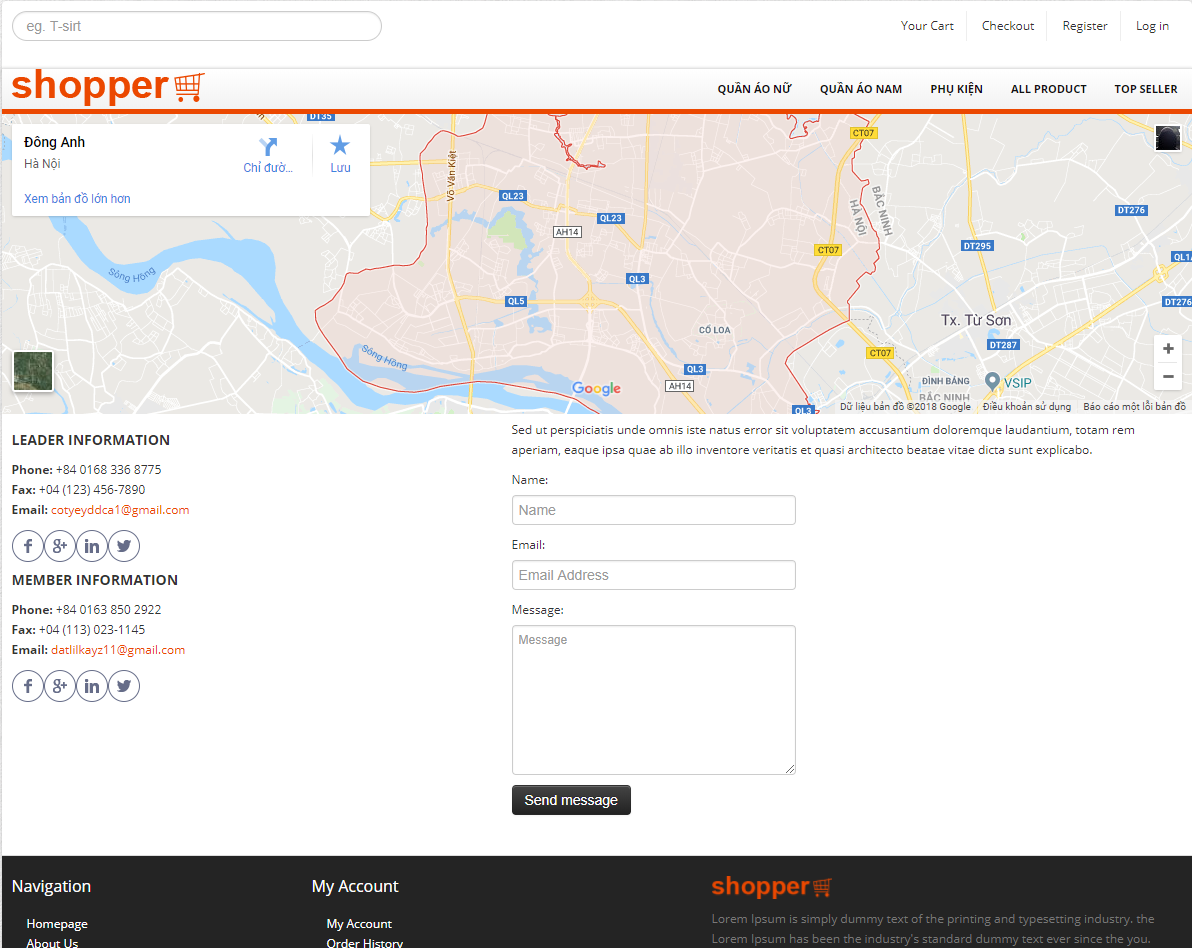
* Xử lý giỏ hàng gồm các chức năng:
  + Hiển thị chi tiết một danh sách các sản phẩm mà người dùng đã chọn
  + Hiển thị giá của toàn bộ các sản phẩm đó



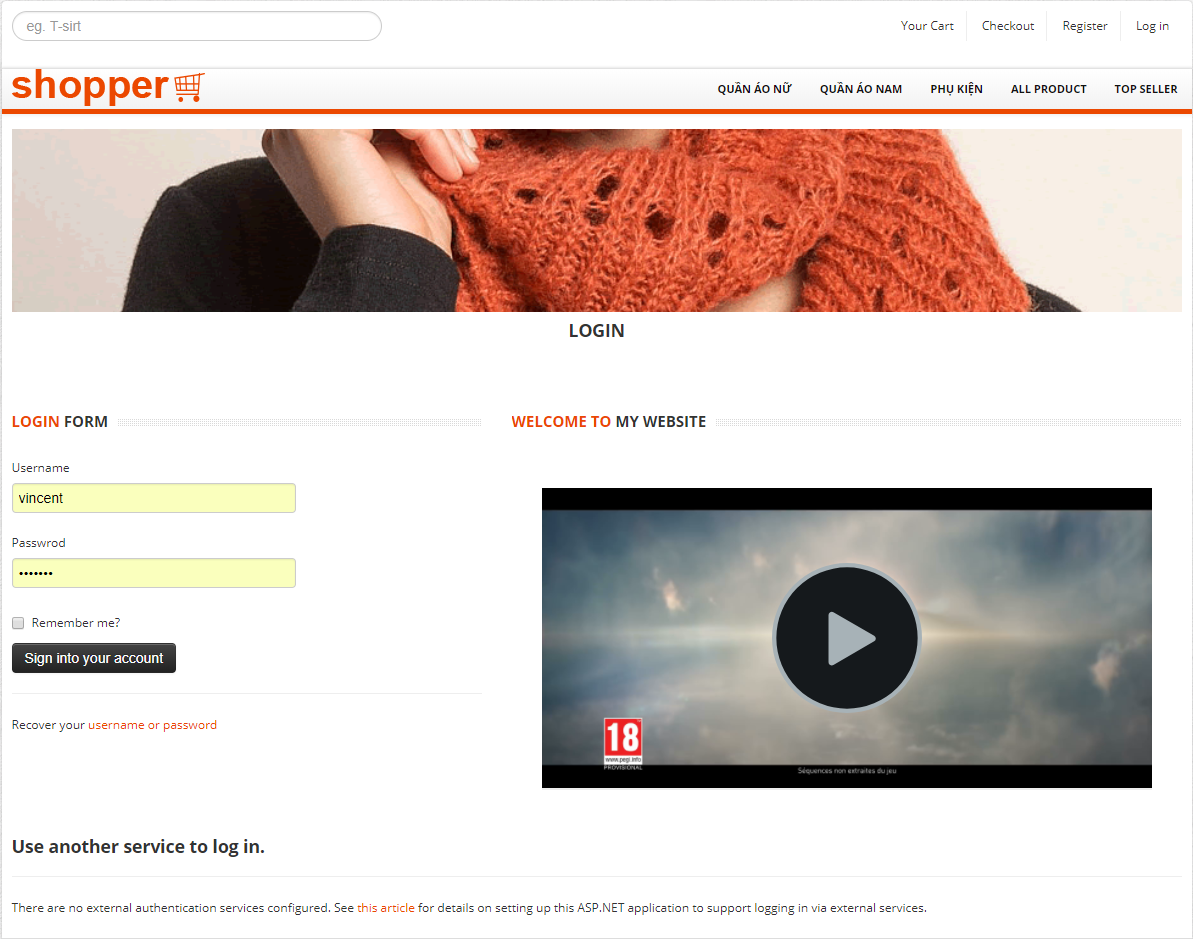
* Thanh toán gồm các chức năng:
  + Khi người dùng điền đầy đủ thông tin thì sẽ được lưu vào cơ sở dữ liệu của shop
  + Gửi một email hiển thị chi tiết các thông tin về sản phẩm cũng như một số thông tin người dùng về email của người dùng đã nhập.



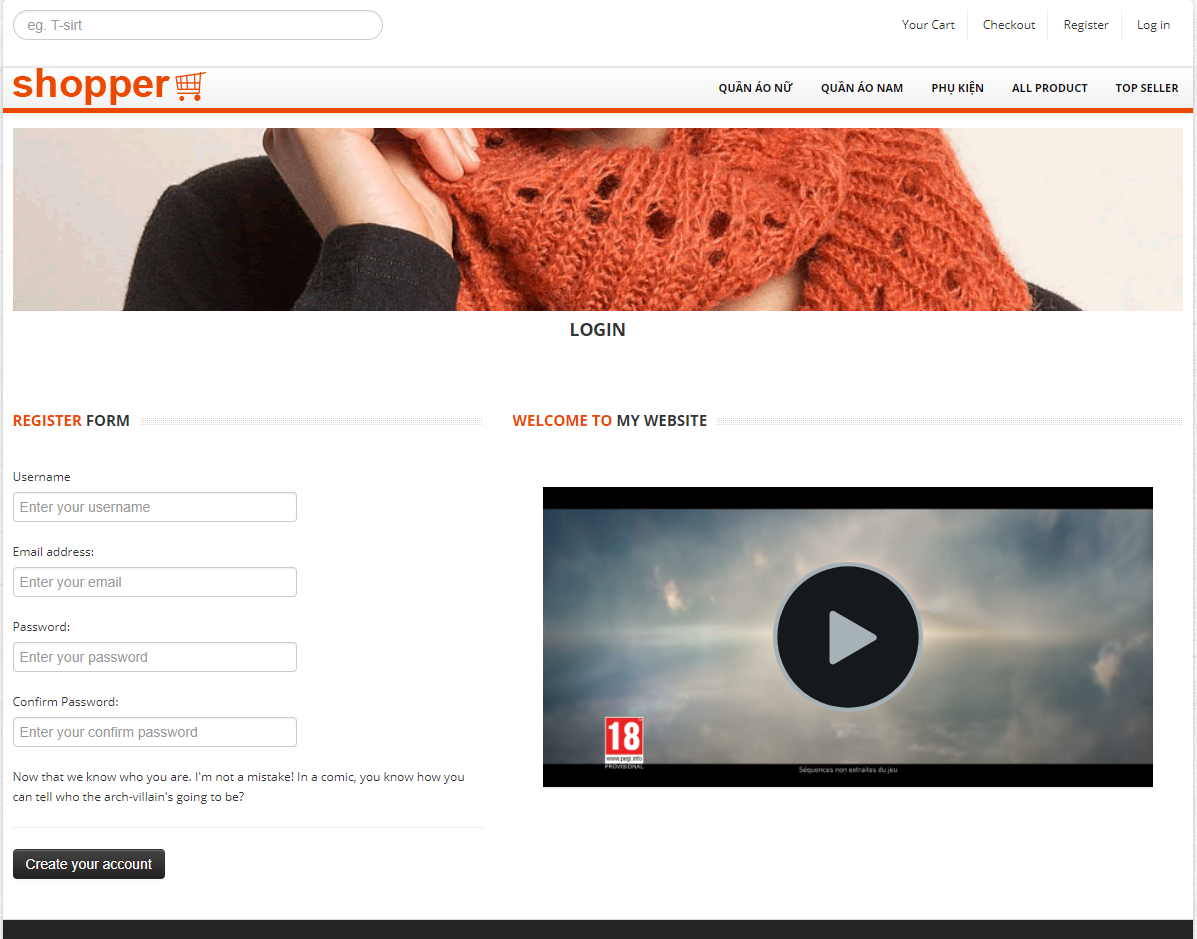
* Chức năng gửi thư phản hồi : Người dùng có thể tìm kiếm shop thông qua map được hiển thị và có thể gửi các phản hồi về shop.



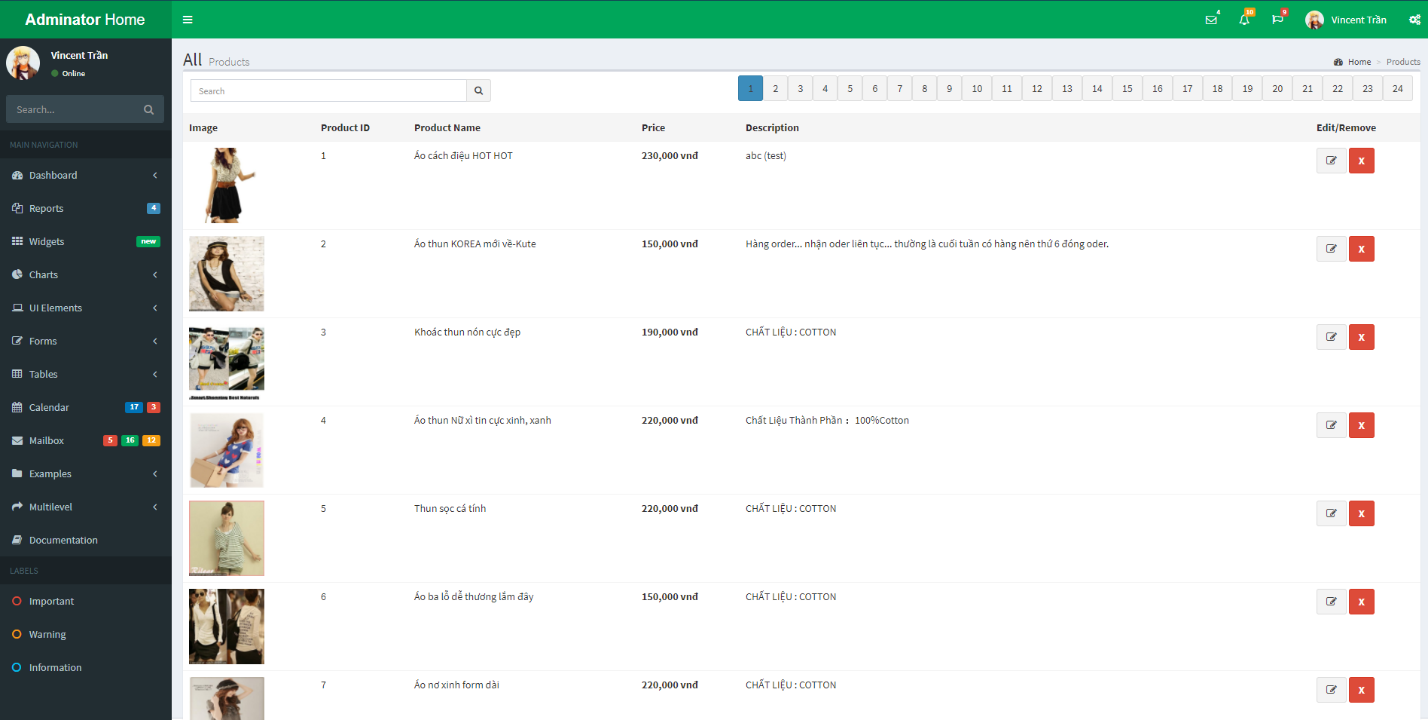
* Chức năng đăng nhập, đăng kí
  + Người dùng có thể đăng nhập vào website để thực hiện mua hàng và nhận được nhưng gợi ý về sản phẩm.
  + Nếu chưa có tài khoản thì người dùng có thể đăng ký một tài khoản mới để truy cập
  + Trong khi ở view đăng nhập hoặc đăng ký người dùng sẽ được xem một đoạn giới thiệu về shop.
  + Đăng nhập



* + Đăng ký



* Quản lý sản phẩm: Hiển thị chi tiết các sản phẩm thông qua một bảng và có phân trang. Trên bảng này người quản trị có thể thực hiện thêm/sửa/xóa sản phẩm.



* Quản lý báo cáo : Hiển thị các thông báo theo các tháng trong năm và cũng có thể theo ngày trong mục day report

