ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA CÔNG NGHỆ PHẦN MỀM

HOÀNG HẢI

NGUYỄN HOÀNG HIỆP

KHÓA LUẬN TỐT NGHIỆP

ỨNG DỤNG SO SÁNH GIÁ SẢN PHẨM GIỮA CÁC TRANG THƯƠNG MẠI ĐIỆN TỬ

PRICE COMPARISON BETWEEN E-COMMERCE WEBSITES SYSTEM

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

TP. HỒ CHÍ MINH, 2017

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA CÔNG NGHỆ PHẦN MỀM

HOÀNG HẢI – 13520230

NGUYỄN HOÀNG HIỆP – 13520265

KHÓA LUẬN TỐT NGHIỆP

ỨNG DỤNG SO SÁNH GIÁ SẢN PHẨM GIỮA CÁC TRANG THƯƠNG MẠI ĐIỆN TỬ

PRICE COMPARISON BETWEEN E-COMMERCE WEBSITES SYSTEM

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

GIẢNG VIÊN HƯỚNG DẪN

THS. TRẦN ANH DŨNG

TP. HỒ CHÍ MINH, 2017

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …… ngày …….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. ……. – Chủ tịch.
2. ……. – Thư ký.
3. ……. – Ủy viên.
4. ……. – Ủy viên.

ĐHQG TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc Lập - Tự Do - Hạnh Phúc

TP. HCM, ngày……tháng……năm……

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(CỦA CÁN BỘ HƯỚNG DẪN)

Tên đề tài:

ỨNG DỤNG SO SÁNH GIÁ SẢN PHẨM GIỮA CÁC TRANG THƯƠNG MẠI ĐIỆN TỬ

Nhóm SV thực hiện:		**Cán bộ hướng dẫn:**
Hoàng Hải	13520230	ThS. Trần Anh Dũng
Nguyễn Hoàng Hiệp	13520265	

Đánh giá Khóa luận

1. Về cuốn báo cáo

   Số trang : 60 Số chương : 05

   Số bảng số liệu : 36 Số hình vẽ : 19

   Số tài liệu tham khảo : 04 Số sản phẩm : 01

   Một số nhận xét về hình thức cuốn báo cáo:

   .....

   .....

   .....

2. Về nội dung nghiên cứu:

   Đã nghiên cứu và áp dụng thành công các công nghệ mới hiện nay như Spring framework sử dụng Spring data JPA cũng như một số công cụ nổi bật khác như PhantomJs, Selenium webdriver... để hiện thực hóa những yêu cầu đã đề ra từ ban đầu. Ngoài ra trong quá trình thiết kế phát triển ứng dụng, nhóm còn vận dụng sử

   Đã hoàn thành da số các mục tiêu đặt ra ban đầu, như :

   - **Crawler data** : đã xây dựng được crawler data có khả ăng lấy dữ liệu từ nhiều loại website khác nhau, bao gồm các trang web truyền thống và các trang web load dữ liệu bằng javascript. Crawler có khả năng lấy dữ liệu mong muốn từ một cơ sở tri thức mẫu, và hình thành các luạt rút trích từ cơ sở tri thức này.
   - **Hệ thống so sánh giá cả** : hê thống bước đầu nhận diện được các sản phẩm lấy về là sản phẩm đã có rồi hay sản phẩm mới, để đưa ra thông tin so sánh chính xác.

   .....

   .....

   .....

   .....

3. Về thái độ làm việc của sinh viên:

....

....

....

....

Đánh giá chung:

....

....

....

....

....

Điểm từng sinh viên:

Hoàng Hải**:** **……/10**

Nguyễn Hoàng Hiệp**:** **……/10**

Người nhận xét

(Ký và ghi rõ họ tên)

ĐHQG TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc Lập - Tự Do - Hạnh Phúc

TP. HCM, ngày……tháng……năm……

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(CỦA CÁN BỘ PHẢN BIỆN)

Tên đề tài:

ỨNG DỤNG SO SÁNH GIÁ SẢN PHẨM GIỮA CÁC TRANG THƯƠNG MẠI ĐIỆN TỬ

Nhóm SV thực hiện:		**Cán bộ hướng dẫn:**
Hoàng Hải	13520230	ThS. Trần Anh Dũng
Nguyễn Hoàng Hiệp	13520265	

Đánh giá Khóa luận

1. Về cuốn báo cáo

   Số trang : 60 Số chương : 05

   Số bảng số liệu : 36 Số hình vẽ : 19

   Số tài liệu tham khảo : 04 Số sản phẩm : 01

   Một số nhận xét về hình thức cuốn báo cáo:

   .....

   .....

   .....

   .....

2. Về nội dung nghiên cứu:

   .....

   .....

   .....

   .....

3. Về thái độ làm việc của sinh viên:

   .....

   .....

   .....

   .....

Đánh giá chung:

.....

.....

.....

.....

.....

Điểm từng sinh viên:

Hoàng Hải**:** **……/10**

Nguyễn Hoàng Hiệp**:** **……/10**

Người nhận xét

(Ký và ghi rõ họ tên)

ĐHQG TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc Lập - Tự Do - Hạnh Phúc

ĐỀ CƯƠNG CHI TIẾT

Tên đề tài:

ỨNG DỤNG SO SÁNH GIÁ CẢ SẢN PHẨM GIỮA CÁC WEBSITE THƯƠNG MẠI ĐIỆN TỬ

PRICE COMPARISON BETWEEN E-COMMERCE WEBSITES SYSTEM

Cán bộ hướng dẫn: ThS. Trần Anh Dũng

Thời gian thực hiện: Từ tháng 08/2017 đến tháng 01/2018

Sinh viên thực hiện:

- Hoàng Hải – 13520230
- Nguyễn Hoàng Hiệp – 13520265

Nội dung đề tài:

1. Mục tiêu

Đời sống ngày được nâng cao, nhu cầu mua sắm vì thế không ngừng tăng lên. Nhưng để mua được một mặt hàng với giá cả hợp lí và chất lượng là điều khá khó khăn với muôn vàn thông tin từ nhiều nhà bán lẻ khác nhau. Với mong muốn tạo ra một công cụ so sánh giá nhằm mang lại cho người tiêu dùng những mặt hàng với giá cả hợp lí nhất, với thông tin cung cấp

chính xác, nhanh chóng và tin cậy. Chính vì vậy, khóa luận này nhằm mục tiêu nghiên cứu ứng dụng các kỹ thuật rút trích theo phương pháp máy học để xây dựng một hệ thống so sánh thông tin giá cả trực tuyến có tính khả biến cao. Tính khả biến cao mà ứng dụng hướng tới là thu thập nguồn thông tin không hạn chế về giá của các mặt hàng của nhiều website bán lẻ khác nhau trên cùng một bộ thu thập dữ liệu.

2. Phạm vi

Các trang web bán hàng ở Việt Nam. Mặt hàng : điện thoại di động.

3. Đối tượng

Không phân biệt lứa tuổi, có nhu cầu mua sắm và so sánh giá khi mua hàng.

4. Kết quả mong đợi của đề tài

Xây dựng được một Web Application có chức năng so sánh giá có tính khả chuyển, nhanh chóng, chính xác và độ tin cậy cao.

Các module sẽ xây dựng:

❖ Crawler data từ các website bán hang online:

- Khảo sát các trang web thương mai điện tử của Việt Nam mà cụ thể là mục tìm kiếm sản phẩm của từng trang để phân tích, xây dựng cách rút trích dữ liệu từ các website bán hang thông qua mục tìm kiếm của các trang này.
- Xây dậy thuật toán thu thập dữ liệu chung cho các website thương mai điện tử : mục tiêu đặt ra là chỉ cần cung cấp URL website bán hàng thì crawler có khả năng tự tìm ra form tìm kiếm và thực hiện tìm kiếm sản phẩm sau đó phân tích kết quả tìm kiếm mà website trả về. Đồng thời crawler có khả năng định nghĩa luật rút trích thông tin cho website mà nó rút trích thành công.

- Nghiên cứu cách lưu trữ ghi nhớ kết quả truy vấn của người dung để tang tốc độ truy vấn kết quả so sánh ở các lần sau

❖ Application: là 1 website gồm 2 phần, phần cho user cho phép nhập sản phẩm và query ra kết quả so sánh và phần cho admin cho phép định nghĩa các site bán hang online, danh mục các từ khóa cho crawler

Các công nghệ áp dụng:

❖ Backend:

- Java
- PostgreSQL
- Spring MVC

❖ Frontend:

- JavaScript
- Jquery
- Bootstrap

Kế hoạch thực hiện:

❖ Giai đoạn 1 (từ 15/8/2017 – 25/08/2017)

- Nghiên cứu tài liệu đã thu thập để xây dựng hướng giải quyết cho 2 module quan trọng của ứng dụng: crawler data và xác định form tìm kiếm của các website.
- Sau khi nghiên cứu thì phần module của thành viên nào đảm nhận, sẽ được người đó dịch và ghi lại kết quả tìm hiểu, cách giải quyết vấn đề, lý do chọn cách giải quyết như vậy, để làm cơ sở thảo luận và dùng cho việc viết báo cáo sau này
- Hoàng Hải: đảm nhận module crawler web
- Nguyễn Hoàng Hiệp: đảm nhận module tìm kiếm form tìm kiếm của từng website

❖ Giai đoạn 2 (từ 26/08/2017 – 08/092017)

- Xây dựng thuật toán cho 2 module trên dựa vào kết quả nghiên

cứu và hướng giải quyết đã đưa ra.

- Ghi lại chi tiết hoạt động cũa thuật toán kèm mã giã

❖ Giai đoạn 3 (từ 09/09/2017 – 22/09/2017)

- Cài đặt 2 module dựa trên thuật toán đã xây dựng bằng console app. (do đây là 2 module quyết định hệ thống có vận hành được hay không nên phải xây dựng và test trước, khi đạt yêu cầu thì mới đủ cơ sở tiến hành các giai đoạn sau)
- Sẽ thảo luận về coding convention và cách ghi document các hàm trước khi tiến hành code.

❖ Giai đoạn 4 (từ 23/09/2017 – 06/10/2017)

- Thiết kế các chức năng chính của ứng dụng, dự kiến sẽ gồm các chức năng cơ bản cần
- đạt được: đưa ra so sánh cho sản phẩm do người dùng nhập vào, chức năng quản trị hệ thống. Hỗ trợ tìm kiếm từ khoảng 5-10 trang thương mại điện tử của Việt Nam
- Chức năng nâng cao (nghiên cứu tích hợp sau): tìm kiếm nâng cao
- Chú ý: đây là các chức năng hiện tại được đề xuất, trong quá trình thảo luận, có thể sẽ bổ sung thêm.

❖ Giai đoạn 5 (từ 07/10/2017 – 20/10/2017)

- Thiết kế database
- Thiết kế prototype màn hình cho các chức năng

❖ Giai đoạn 6 (từ 21/10/2017 – 03/11/2017)

- Tổng hợp lại các tài liệu đã được ghi lại trong các giai đoạn trước thành tập tài liệu phục vụ cho giai đoạn cài đặt ứng dụng.

❖ Giai đoạn 7 (từ 04/11/2017 – 15/12/2017)

- Phân công và cài đặt các chức năng của hệ thống.
- Hoàn thành báo cáo khóa luận.

Xác nhận của CBHD (Ký tên và ghi rõ họ tên	**TP. HCM, ngày…. tháng …. năm ….** (Ký tên và ghi rõ họ tên)

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

LỜI CẢM ƠN

Chương 1. MỞ ĐẦU

1.1. Đặt vấn đề

Trong vòng hơn 10 năm trở lại đây, nền kinh tế Việt Nam đã có nhiều bước tiến rõ rệt, kéo theo đời sống của người dân được nâng cao và nhu cầu mua sắm, tiêu dùng của họ cũng được nâng lên đáng kể.

Trước tình hình đó, lĩnh vực thương mại điện tử cũng ngày càng phát triển với rất nhiều trang web bán hàng mọc lên. Mặt tích cực của vấn đề này là làm tăng tính cạnh tranh, giúp người tiêu dùng mua được hàng với mực giá tốt, nhưng nó cũng nảy sinh một vấn đề là do có quá nhiều nơi bán hàng, nên người dùng sẽ gặp khó khăn trong việc lựa chọn nơi mua hàng vừa ý.

Nắm bắt nhu cầu có một nơi để tham khảo giá cả của các của hàng trực tuyến trước khi mua, một số trang web so sánh giá cả đã xuất hiện trên mạng internet. Mặc dù đây là một lĩnh vực có tiềm năng, nhưng số lượng các đề tài nghiên cứu, thảo luận trên các diễn đàn lại quá ít và khá cũ.

Với mong muốn nghiên cứu về các phương pháp rút trích dữ liệu tự động, nên nhóm chúng em đã quyết định lựa chọn đề này.

1.2. Mục tiêu và đối tượng

Khóa luận hướng tới các mục tiêu như sau:

- Nghiên cứu và xây dựng thuật toán rút trích thông tin từ website.
- Xây dựng một website bằng ngôn ngữ Java cung cấp chức năng so sánh thông tin giá cả từ nhiều trang thương mại điện tử.
- Thử nghiệm hệ thống với các trang thương mại điện tử ở Việt Nam để đánh giá hiệu quả của hệ thống, qua đó tiến hành các cải tiến về giải thuật và cài đặt

Đối tượng thực hiện : do thời gian có hạn nên nhóm chúng em tập trung vào nhóm hàng nổi bật nhất của thì trường thương mại điện tử ở Việt Nam đó là điện thoại di động với đối tưởng khảo sát là 1 số trang chuyên về lĩnh vực này.

Chương 2. TỔNG QUAN

2.1. Các đề hệ thống đã có sẵn

Trong quá trình thực hiện khóa luận, qua việc tìm hiểu các nghiên cứu liên quan và khảo sát các trang web chuyên về so sánh giá cả ở Việt Nam, chúng em nhận thấy một số vấn đề như sau :

2.1.1. Về các nghiên cứu đã tìm được

Mặc dù đề tài này liên quan đến một nhu cầu thiết thực của người tiêu dùng hiện nay, nhưng các đề tài nghiên cứu, hoặc các thảo luận trên các diễn dàn CNTT khá ít.

Các đề tài tham khảo có thời gian khá lâu trong giai đoạn 2001 đến 2006. Điểm chung của các đề tài này là đều tiếp cần theo hướng xây dựng thuật toán rút trích dữ liệu để lấy thông tin về sản phẩm.

2.1.2. Các vấn đề còn tồn tại và nảy sinh

Mặc dù các ý tưởng về rút trích dữ liệu từ website của các nghiên cứu trên rất hữu ích, tuy nhiên do thời gian đã quá lâu, nên chúng đã mắc phải một số vấn đề như sau:

- Với sự phát triển của công nghệ web, đặc biệt là javascript ngày càng được ưa chuộng cho việc xây dựng website, việc thu thập được 1 trang html hoàn chỉnh và đầy đủ để bắt đầu giai đoạn rút trích thông tin trở thành một thử thách không hề nhỏ, do một website bây giờ có thể chưa hoàn toàn có đầy đủ dữ liệu từ lần gửi request đầu tiên mà có thể được đổ dữ liệu dần dần bằng javascript hoặc thậm chí render hoàn toàn bằng javascript với sự hỗ trợ của các javascript framework.
- Html cũng dần phát triển theo thời gian với rất nhiều thẻ và thuộc tính mới, khiến cho việc rút trích dữ liệu gặp nhiều khó khăn hơn do tài liệu html ngày càng phức tạp.

2.2. Các vấn đề mà khóa luận tập trung giải quyết

Từ các vấn đề còn tồn tại ở trên, khóa luận của chúng em sẽ tập trung giải quyết những vấn đề sau :

- Nghiên cứu, xây dựng một thuật toán có khả năng lấy được một tài liệu html hoàn chỉnh từ các trang web sử dụng nhiều javascript.
- Xây dựng thuật toán rút trích dữ liệu có thể xử lý các tài liệu html phức tạp để lấy được thông tin sản phẩm

Chương 3. NGHIÊN CỨU VÀ THỰC NGHIỆM

3.1. Khảo sát các trang thương mại điện tử về điện thoại di động

Mục đích của nghiên cứu là thu thập thông tin giá cả từ các trang thương mại điện tử (được giới hạn trong mặt hàng điện thoại di động trong khóa luận này), nên trước hết cần phải khảo sát các trang này. Một số website tiêu biểu có thể kể đến như :

STT	Địa chỉ website
1	https://www.thegioididong.com/
2	https://vienthonga.vn/
3	https://fptshop.com.vn/
4	https://cellphones.com.vn
5	https://hoanghamobile.com/

Bảng 3.1 Một số trang bán đtdđ ở Việt Nam

Một số kết luận rút ra được từ việc khảo sát :

- Phần lớn các trang web này sử javascript để hiển thị một phần trang web, do đó ngày từ request đầu tiên chưa thể lấy hết thông tin của một trang mong muốn
- Để lấy thông tin sản phẩm một cách hiệu quả từ các trang này thì việc dựa vào trang kết quả tìm kiếm của từng trang là thích hợp hơn cả.

3.1.1. Nhận diện trang tìm kiếm

Ở trang chủ của các trang bán hàng đều có một khung search nhanh, chúng ta có thể tận dụng khung search này để gửi request và được dẫn đến trang tìm kiếm sản phẩm.

Vấn đề đặt ra là làm sao để tìm ra được được khung search này. Vì đa phần các trang đặt khung search trong thẻ form, nên ta có thể lợi dụng điểm này, lấy ra tất cả các form trong trang, sau đó điền dữ liệu vào và submit từng form rồi phân tích trang kết quả trả về để xem nó có phải là trang kết quả tìm kiếm hay không. Cách làm này hiệu

quả khá cao, nhưng có thể gặp trường hợp input không được đặt trong form hoặc có nhiều form trong trang, gây tốn nhiều thời gian hơn chi việc kiểm tra.

Sau khi xem xét, nhóm chúng em nhận ra rằng tất cả các input để tìm kiếm đều có phần placeholder, và nội dung có nhiều nhất trong các placeholder này là chuỗi "tìm", "search". Nên nhóm bọn em quyết định chọn cách tìm kiếm khung nhập dựa vào placeholder. Phương án này không phải gửi nhiều request để kiểm tra, mà chỉ cần tìm được thẻ input nào có placegolder khớp với placeholder lưu trong database là được.

Sau khi đã xác định được thẻ input dẫn tới trang kết quả tìm kiếm, ta sẽ tiến hành gửi thử request thông qua thẻ input đó để xem đó có đúng là trang kết quả tìm kiếm hay không. Có 2 cách để kiểm tra, cách trực tiếp là nhập vào input một đoạn truy vấn chắc chắn sẽ có sản phẩm và kiểm tra nội dung trang trả về, tuy nhiên độ chính xác của cách này không cao. Một cách khác cho độ chính xác rất cao nhưng lại vô cùng đơn giản đó là gửi 1 chuỗi truy vấn chắc chắn không có kết quả và chuỗi đó cũng không có khả ăng xuất hiện trước trong website, ta chỉ việc kiểm tra xem chuỗi đó có hay không trong trang html nhận được là có thể xác định được xem trang đó có phải trang kết quả tìm kiếm hay không.

3.1.2. Duyệt các trang sử dụng javascript để hiển thị.

Trong quá trình nghiên cứu các tài liệu có liên quan về bài toán rút trích dữ liệu trên website của đề tài xây dựng hệ thống, do thời gian thực hiện các nghiên cứu này đã khá lâu, trong thời kỳ javascript chưa phát triển mạnh như hiện nay, nên việc duyệt các trang web để lấy nội dung phục vụ cho việc rút trích khá đơn giản, vì thường phần lớn nội dung của các trang sẽ được trả về thông qua một request duy nhất.

Ở bối cảnh giai đoạn hiện nay, các thư viện , framework javascript như jQuery, AngularJS, ReactJS … được sử dụng rất phổ biến để hiển thị cho website được sinh động và để load dữ liệu thông qua ajax. Điều này dẫn đến một vấn đề là nội dung của một trang web sẽ không hiển thị đầy ngay lần đầu tiên nó được request, mà sau khi server trả về nội dung, javascript sẽ được thực thi để hoàn tất công việc hiển thị. Do

đó, các công cụ và kỹ thuật duyệt web tự động ở các nghiên cứu này hoàn toàn không còn sử dụng được đối với các website mới.

Do dó, cần phải có hướng tiếp cận mới trong việc tìm cách lấy được toàn bộ html của website, vì đây là đầu vào của giai đoạn rút trích dữ lệu. Cuối cùng, nhóm đã tìm ra một giải pháp hữu hiệu, đó là sử dụng các công cụ test web tự động để thực hiện giai đoạn này. Ưu điểm của các công cụ này là vì chúng phục vụ cho việc test, nên chúng có khả năng thao tác như một trình duyệt web thực thụ, bao gồm cả việc thực thi javascript hoặc đợi cho website được javascirpt load hoàn chỉnh. Phần sau sẽ trình bày các công cụ mà nhóm đã tìm hiểu và sử dụng.

3.1.2.1. Selenium Webdriver

Selenium Webdriver (Se driver) là một tool open source giúp việc thực thi các hành động lên trang web một cách tự động. Se driver hỗ trợ viết script trên nhiều ngôn ngữ khác nhau: Java, C#, python, PHP….

Điểm nội trội nhất của Se chính là khả năng tái hiện 100% hành vi tương tự như khi ta duyệt web bằng các trình duyệt web thông dụng. Để thực hiện được chức năng này, Se cần một webdriver kèm theo, có thể là firefox webdriver (để giả lập trình duyệt firefox), chrome webdriver (giả lập trình duyệt chrome) …..

Ở đây nhóm chọn sử dụng PhantomJs để làm webdriver cho Se.

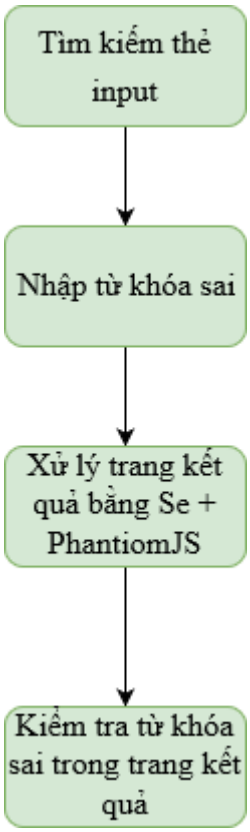
3.1.2.2. PhantomJs

Việc truy vấn các website bằng Se hoàn toàn có thể thực hiện qua firefox webdriver hoặc chorme webdriver, tuy nhiên các driver này có một hạn chế khi thực hiện mục tiếu khác với mục tiêu test web đó là chúng tiêu tốn tài nguyên như một trình duyệt thực sự, điều này là không cần thiết và cần phải tránh để tối ưu hiệu năng cho hệ thống. Do vậy, PhantomJs đã được chọn để giải quyết vấn đề này.

PhantomJs là một open source headless browser, nó có thể thực hiện đầy đủ chức năng của một trình duyệt, nhưng hoàn toàn không có giao diện, nên mức độ tiêu tốn

tài nguyên rất thấp. Se cũng có một package gọi là PhantomDriver để thực thi PhantomJs nên sử dụng PhantomJs là hợp lý nhất.

Quá trình nhận diện trang tìm kiếm của một website được mô tả qua sơ đồ sau :



Hình 3.1 Trình tự nhận diện trang tìm kiếm của một website

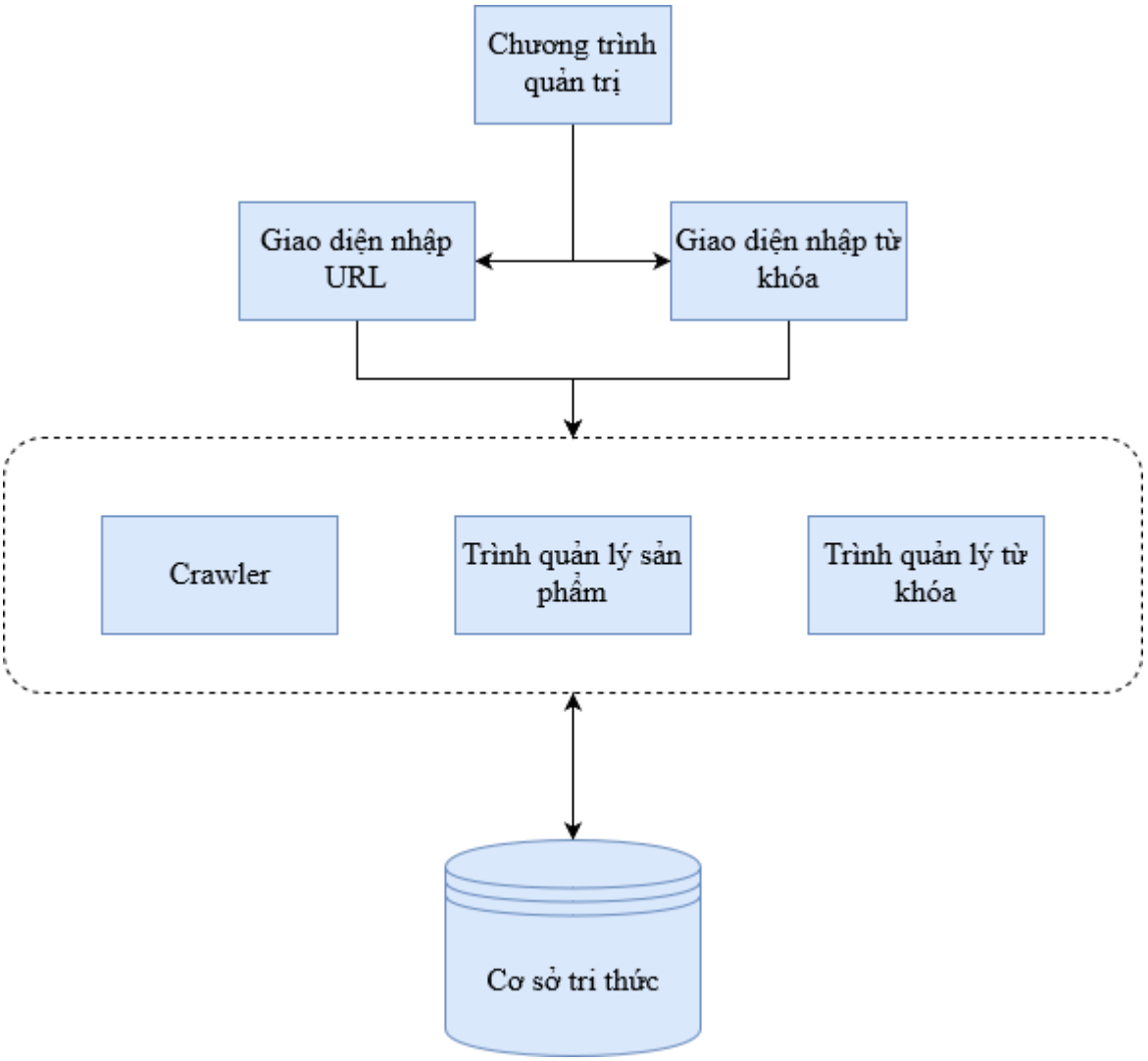
3.2. Xây dựng hệ thống rút trích dữ liệu

Hình 3.2 mô tả mô hình tổng quát của hệ thống trích rút dữ liệu. Hệ thống gồm có hai giao diện chính giúp cho người quản trị có thể tương tác với chương trình. Bây giờ chúng ta sẽ tìm hiểu chi tiết từng thành phần của hệ thống.

Giao diện nhập URL cho phép người dùng chương trình nhập vào địa chỉ URL của website bán hàng trực tuyến.

Giao diện nhập thuật ngữ cho phép người dùng chương trình nhập vào các thuật ngữ (từ khoá) mới để nhận biết ra các thông tin về thuộc tính sản phẩm, các từ cần loại bỏ hoặc các thẻ html cần bỏ qua. Crawler có khả năng truy nhập đến các Website bán

hàng trực tuyến tự động học cấu trúc của Website, định dạng của sản phẩm và tìm cách trích rút thông tin của các sản phẩm trong Website đó.



Hình 3.2 Kiến trúc của hệ thống rút trích sản phẩm

Trình quản lý từ khóa có chức năng quản lý những từ khoá và cho phép nhập thêm từ khoá mới thông qua giao diện nhập từ khóa.

Cơ sở tri thức là những tri thức mà hệ thống thu được bằng việc kết hợp kết quả của crawler và trình quản lý thuật ngữ.

Chương trình cập nhật thông tin vào CSDL dùng để đọc giá trị của thuộc tính miêu tả sản phẩm và cập nhật một cách chính xác.

CSDL chứa các thông tin về sản phẩm và nhà sản xuất. Các thông tin về sản phẩm mà chương trình cập nhật thông tin vào CSDL sẽ được lưu lại một cách chính xác theo từng thuộc tính.

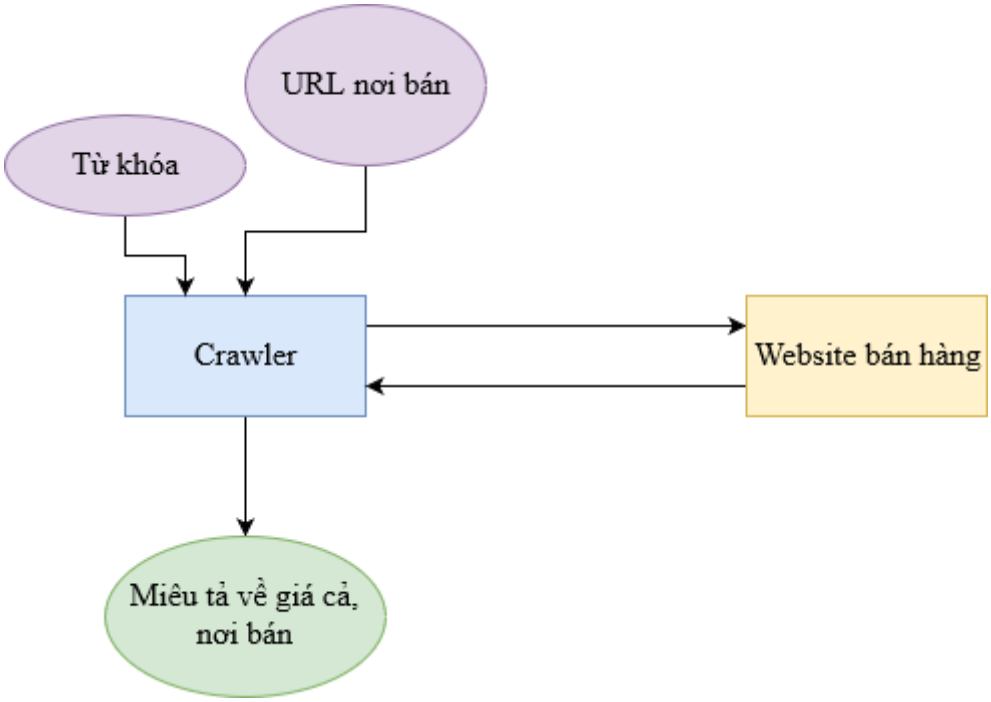
Nhiệm vụ chính của hệ thống là làm sao có thể trích rút được một cách chính xác thông tin của sản phẩm và cập nhật vào CSDL. Để làm được việc này thì hệ thống phải nhận dạng được mẫu biểu tìm kiếm trên Website bán hàng trực tuyến. Sau khi xác định được mẫu biểu tìm kiếm nó phải gửi truy vấn tên sản phẩm cần tìm để nhận được trang kết quả tìm kiếm là một danh sách các sản phẩm. Tiếp theo hệ thống phải có khả năng trích rút được những thông tin liên quan đến sản phẩm (phải loại bỏ các thông tin không liên quan). Cuối cùng hệ thống phải hiểu được các thuộc tính miêu tả sản phẩm như hiểu được đâu là giá, tên, hay nhà sản xuất ra sản phẩm đó và cập nhật những thông tin đó vào CSDL. Công việc này được thực hiện đối với mỗi Website và lưu thành các mẫu (template) để sau này chương trình chỉ việc lấy ra những thông tin về sản phẩm, không quan tâm đến cấu trúc của Website cũng như các trường khác.

3.2.1. Crawler

Crawler sẽ đi thu thập các thông tin về nơi bán hàng cũng như định dạng miêu tả sản phẩm và cấu trúc của thông tin sản phẩm được hiển thị. Crawler chỉ thực hiện một lần đối với mỗi Website bán hàng trực tuyến. Mô hình của giai đoạn học như hình 3.3.

Để xác định được các thông tin về nơi bán thì Crawler cần:

- Xác định mẫu biểu tìm kiếm (địa chỉ URL của form tìm kiếm).
- Lấy ra đơn vị miêu tả sản phẩm.
- Xác định giá trị của thuộc tính sản phẩm.



Hình 3.3 Minh họa quá trình crawler lấy dữ liệu

3.2.2. Rút trích ra đơn vị sản phẩm

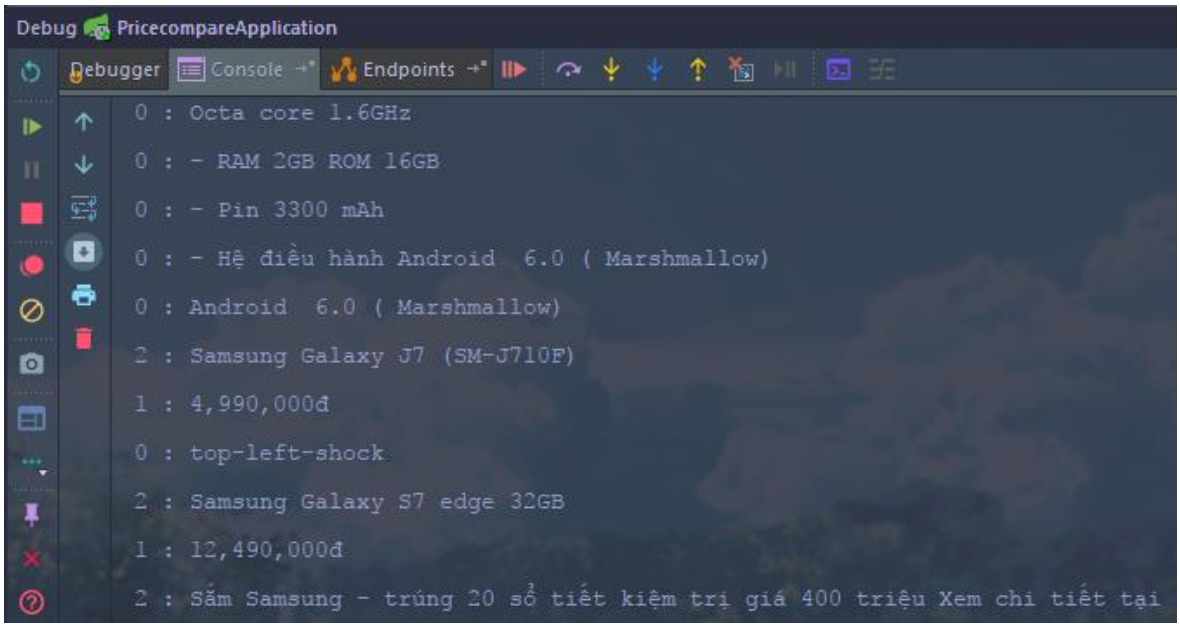
Kết quả đầu ra sau khi xử lý bằng Selenium sẽ cho ta một đoạn html hoàn chỉnh của trang kết quả tìm kiếm bao gồm danh sách các sản phẩm. Nhiệm vụ của crawler là sẽ đi rút trích thông tin sản phẩm từ đoạn html này, ở giai đoạn hiện tại crawler có khả năng lấy được tên sản phẩm và giá, đây là hai thông tin tối quan trọng cho hệ thống so sánh giá.

Do mỗi website có một cách trình bày danh sách sản phẩm riêng, cho nên để crawler có khả năng lấy được thông tin từ nhiều nguồn, cần xây dựng một thuật toán có tính tổng quát hóa cao.

Ý tưởng của thuật toán là mỗi website sẽ có một cấu trúc html riêng cho từng đơn vị sản phẩm, tạm gọi là Product Description Unit (PDU) . Chúng ta sẽ phân tách file html được thành các dòng, gọi là logical line, sau đó đánh số cho từng dòng ứng với thông tin mà nó cung cấp. Hình 3.4 minh họa kết quả phân tích logial line từ trang https://vienthonga.vn/ với từ khóa là “samsung” :

STT	Id	Ý nghĩa
1	0	Thông tin không xác định
2	1	Giá của sản phẩm
2	2	Tên của sản phẩm

Bảng 3.2 Danh sách các loại logical line hiện tại của hệ thống



Hình 3.4. Kết quả phân tích logical line cho trang của vienthonga

Sau khi đã có được chuỗi logical line của toàn bộ đoạn html thu được ta sẽ đi tìm ra mẫu PDU với nguyên tắc sau : **PDU phải chứa toàn bộ các thuộc tính cần rút trích, mỗi thuộc tính xuất hiện không quá một lần, trừ nhóm thuộc tính không xác định.**

VD : nếu ta có chuỗi logical line 20012001210021 thì các PDU hợp lệ đó là : 2001, 001200, 01200, 1200, 0012, 012, 12, 21, 1002, 0021,021, 21.

Khi đã có danh sách các chuỗi có khả năng là PDU trong đoạn html trả về, ta sẽ tìm ra chuỗi nào có số lần xuất hiện cao nhất và lấy đó làm PDU cho website được khảo sát.

Để crawler có khả năng nhận diện đâu là giá, đâu là tên sản phẩm, ta cần có một cơ sở tri thức mẫu, bao gồm các mẫu có khả năng mang thông tin giá cả, tên sản phẩm v.v… Khi lần đầu tiên rút trích thông tin cho một website, crawler sẽ dựa vào các tri thức mẫu này để xác định xem đâu là giá, đâu là tên, đâu là thông tin chưa xác định. Sau đó sẽ hình thành lên danh sách logical line và tiến hành tìm ra PDU và lưu lại template cho website này, từ lần thứ hai trở đi, cralwer sẽ dựa vào template để tiến hành rút trích dữ liệu để cải thiện tốc độ xử lý.

Mấu chốt cho hiệu quả của cralwer nằm ở cơ sở tri thức mẫu, do đó dữ liệu này phải được cập nhật thường xuyên, để đáp ứng cho việc thu thập dữ liệu từ các website mới, hoặc trường hợp website đã có templete rồi nhưng bị sai do website đó họ thay đổi nội dung.

Dưới đây là minh họa cơ sở tri thức mẫu :

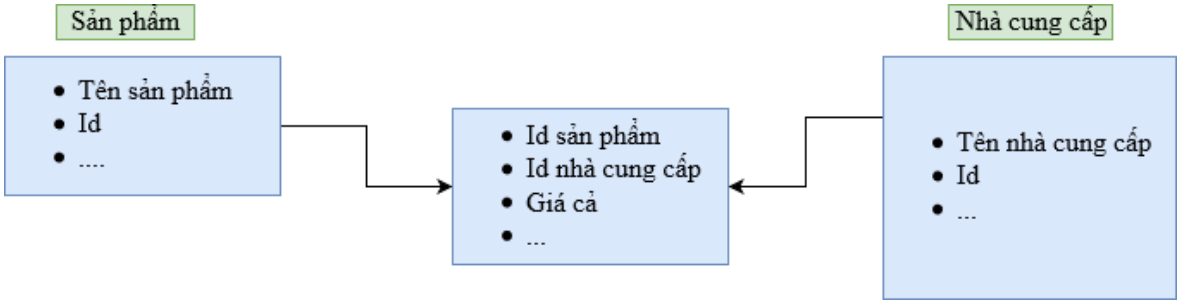
STT	Loại thông tin	Các định dạng có thể có
1	Giá cả	đ, vnđ, d, vnd, ₫
2	Tên sản phẩm	“tên sản phẩm :”, “tên mặt hàng :”, hoặc chỉ có tên mà không có các tiền tố kể trên

Bảng 3.3 Cơ sở tri thức mẫu của hệ thống

Ở lần đầu tiên khảo sát, crawler sẽ tìm xem từng logical line sẽ trùng với định dạng nào trong cơ sở tri thức mẫu, bằng cách so sánh nội dung của nó với **tất cả** các định dạng có thể có. Tuy nhiên, từ lần thứ 2, do đã có sẵn template, nên mỗi loại thông tin ứng với một website chỉ có một định dạng duy nhất, do đó, số lần kiểm tra sẽ được giảm đi đáng kể.

3.2.3. Nhận diện sản phẩm trùng lặp

Trước khi trình bày vấn đề này và giải thích tại sao phải nhận diện sản phẩm trùng lặp, chúng ta hãy xem xét cách lưu trữ sản phẩm của hệ thống ở hình 3.5 như sau:



Hình 3.4 Minh họa cách lưu sản phẩm trong hệ thống

Khi lấy dữ liệu sản phẩm hoàn tất, chúng ta phải xác định xem sản phẩm đó là san phẩm nào, đã có trong database hay chưa? Nếu có thì chúng ta chỉ cần cập nhật giá của nhà cung cấp hoặc thêm giá của nhà cung cấp đó, nếu là sản phẩm mới hoàn toàn thì chúng ta phải cập nhật cả bản "Sản phẩm" . Đây là một giai đoạn hết sức quan trọng, vì nếu không biết các sản phẩm lấy về thuộc vào sản phẩm nào thì không thể nhóm chúng lại một cách chính xác cho việc so sánh, sẽ dẫn đến kết quả sai.

Do mỗi trang web có thể thêm, bớt thông tin vào chuỗi tên sản phẩm, nên chỉ so sánh thuần túy chuỗi tên sản phẩm thì có thể sẽ dẫn đến sai sót. Cần phải có một thuật toán đảm nhiệm việc này. Thông qua việc phân tích cách đặt tên sản phẩm thì ta có thể nhận thấy, tên sản phẩm luôn có một công thức như sau :

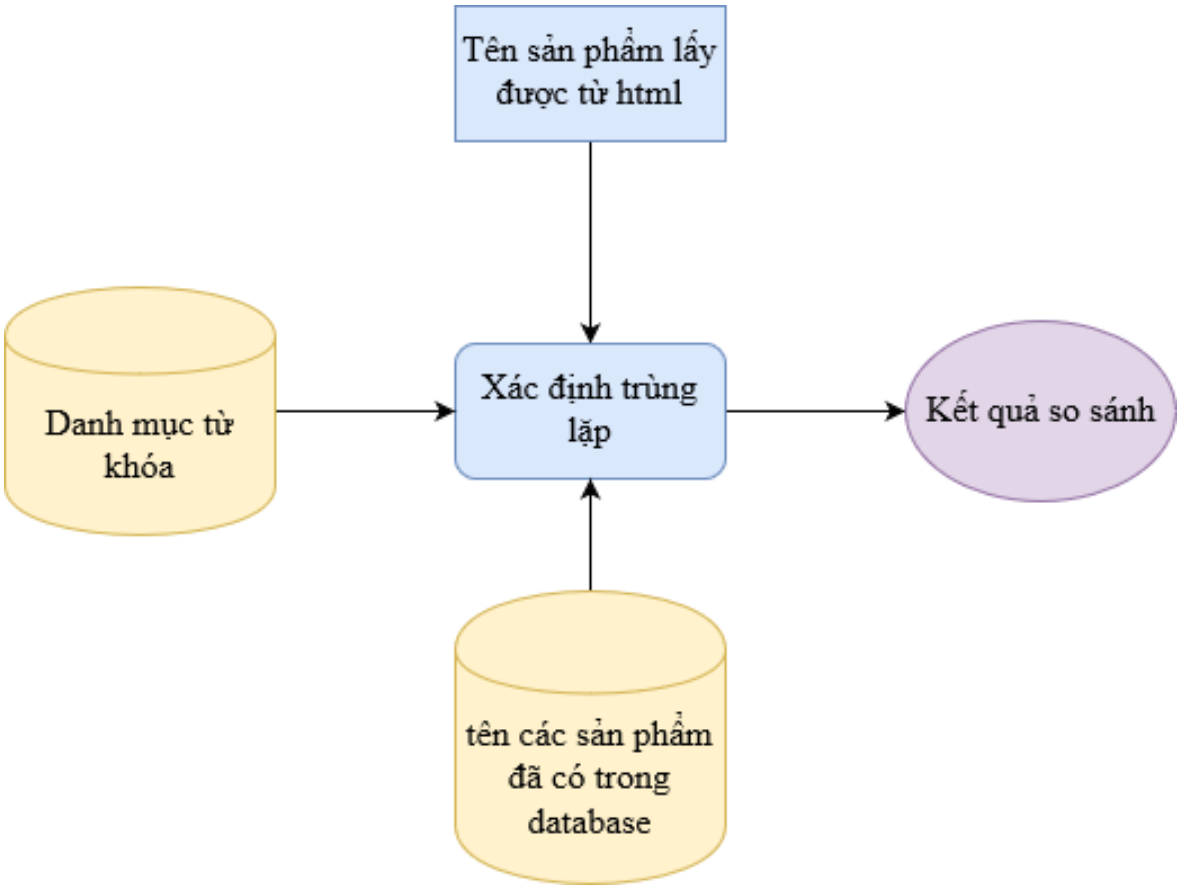
$$Tên\ mặt\ hàng = Tên\ Nhà\ sản\ xuất + Mã + [Các\ thuộc\ tính\ khác]$$

Do đặt thù trên, nên việc so sánh tên 2 mặt hàng có thể áp dụng một số cách đơn giãn nhưng vẫn hiệu quả để xác định xem 2 chuỗi ký có phải là cùng một sản phẩm hay không :

- Điều kiện cần để 2 sản phẩm là một là chúng phải có cùng Tên NSX và Mã
- Các thuộc tính khác có thể bao gồm màu sắc, độ phân giải của camera, dung lượng pin… Có thể được trang bán hàng ghi thêm vào dòng tên sản phẩm. Tùy vào từng tình huống cụ thể mà cùng một loại thông tin thêm có thể quyết định sự khác nhau của cùng một mã sản phẩm hay không . Và việc

tùy chọn thông tin nào có thể bỏ qua sẽ tùy thuộc vào người quản trị hệ thống.

Hình 3.6 minh họa quá trình xác định trùng lặp sản phẩm :



Hình 3.5 Quá trình xác định sản phẩm trùng lặp

Đầu tiên, dựa vào các tùy chọn của người quản trị, mà phần [Các thuộc tính khác] trong tên sản phẩm sẽ được lược bỏ hoặc giữ lại, việc này được thực hiện với cả sản phẩm được lấy về va sản phẩm có trong database sau đó, chương trình sẽ so sánh các tên sau khi được xử lý với nhau để xác định sự trùng lặp. Cách làm này có hai lợi ích sau :

- Các thông tin bắt buộc (tên nhà sản xuất, mã) được giữ lại

- Người quản trị được quyền quyết định các thông tin tùy chọn nào có thể được dùng để phân biệt các sản phẩm có trùng tên nhà sản xuất và mã
- Danh mục các thuộc tính khác có thể được cập nhật

Hình 3.7 minh họa kết quả kiểm tra trùng lặp với từ khóa “iphone” cho trang www.fptshop.com.vn :

Product name	Price	Product in Db
Bao da iPhone 8 UAG Metropolis Blue	950.000 ₫	New product
iPhone 5s 16GB	5.999.000 ₫	iPhone 5s 16GB
iPhone 6 16GB	11.999.000 ₫	New product
iPhone 6 32GB (2017)	8.999.000 ₫	iPhone 6 32GB (2017)
iPhone 6 Plus 16GB	11.999.000 ₫	New product
iPhone 6 Plus 64GB	14.999.000 ₫	New product
iPhone 6s 16GB	12.999.000 ₫	New product
iPhone 6s 32GB	14.999.000 ₫	New product
iPhone 6s 64GB	15.999.000 ₫	iPhone 6s 64GB
iPhone 6s Plus 16GB	13.999.000 ₫	New product

Showing 1 to 10 of 32 entries Previous 1 2 3 4 Next

Hình 3.6 Kết quả kiểm tra trùng lặp với sản phẩm “iphone” ở trang fptshop

Chương 4. XÂY DỰNG HỆ THỐNG SO SÁNH GIÁ CẢ

4.1. Mô tả tổng quan hệ thống

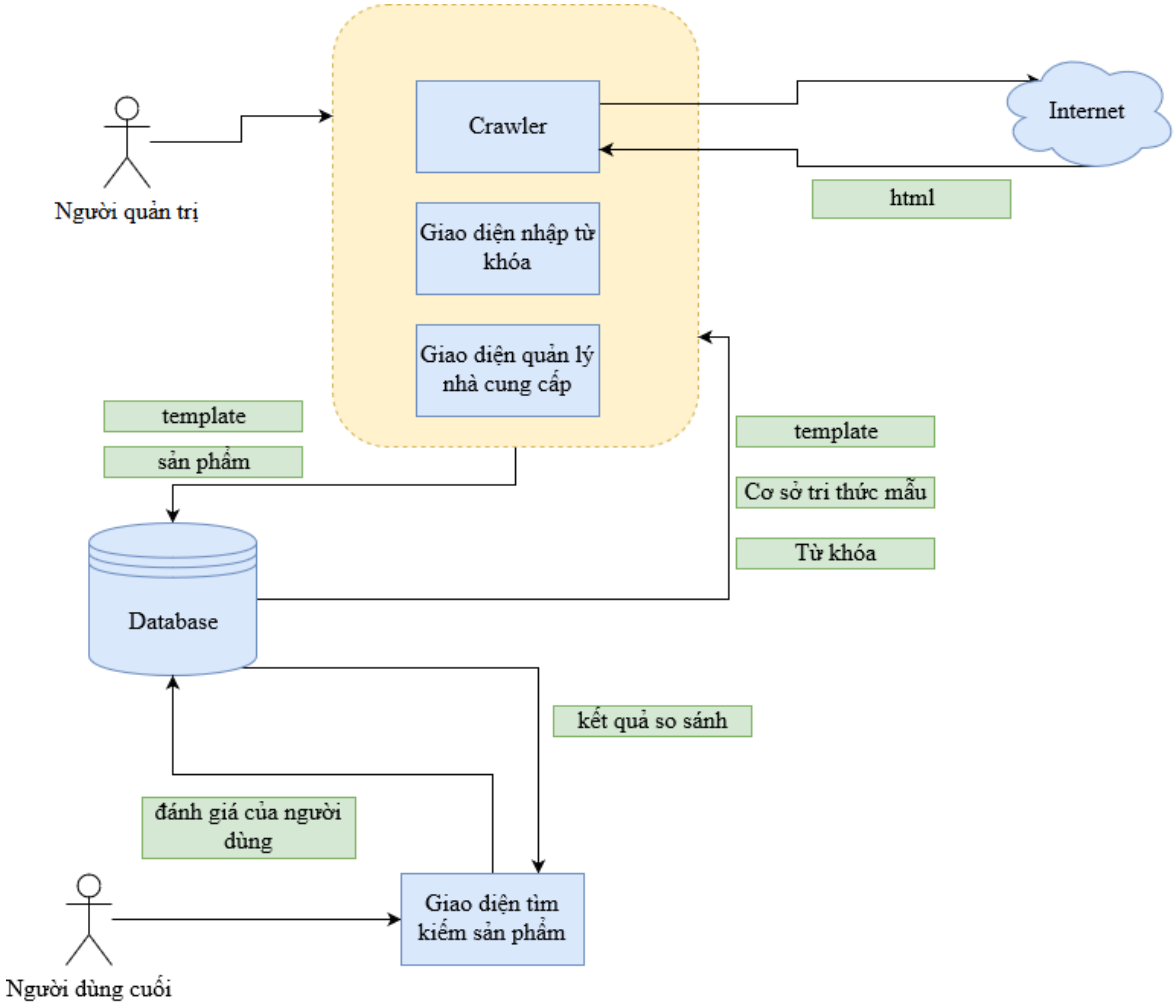
Hệ thống so sánh giá cả được xây dựng hoàn toàn trên nền web, sử dụng ngôn ngữ Java và Spring Framework

Hệ thống so sánh giá cả gồm có hai phần : phần cho người quản trị và phần cho người dùng bình thường.

Phần cho người quản trị sẽ nhận nhiệm vụ thu thập thông tin về sản phẩm, lưu trữ vào database.

Phần cho người dùng bình thường sẽ đáp ứng các yêu cầu tham khảo giá cả từ người dùng, bằng cách lấy dữ liệu đã thu thập được và hiển thị lên.

Hình 4.1 mô tả tổng quan sự tương tác giữa 2 phần :



Hình 4.1 Tổng quan hoạt động của hệ thống và luông dữ liệu

4.2. Nền tảng và công nghệ sử dụng

4.2.1. Ngôn ngữ lập trình Java

4.2.1.1. Tổng quan

Java là một ngôn ngữ lập lập trình, được phát triển bởi Sun Microsystem vào năm 1995, là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng.

Vì sao ngôn ngữ này lại được đặt tên là Java? Java là tên một hòn đảo ở Indonesia - hòn đảo nổi tiếng với loại coffee Peet và cũng là loại nước uống phổ biến của các kỹ

sư Sun. Ban đầu Ngôn ngữ này được đặt tên là "Oak" (có nghĩa là "Cây sồi" - 1991), nhưng các luật sư của Sun xác định rằng tên đó đã được đăng ký nhãn hiệu nên các nhà phát triển đã phải thay thế bằng một tên mới - và cũng vì lý do trên mà cái tên Java đã ra đời và trở thành tên gọi chính thức của Ngôn ngữ này - Ngôn ngữ Lập trình Java.

Java rất mạnh về lập trình ứng dụng web với hiều framework phổ biến như Spring, Strut 2, JSF…

4.2.1.1. Ưu điểm

- Có thể viết mã nguồn trên một IDE ở một máy tính và thực thi chương trình ở bất cứ máy tính sử dụng hệ điều hành nào
- Là ngôn ngữ hoàn toàn hướng đối tượng.
- Cộng đồng công nghệ trong JAVA rất lớn và nhiều công nghệ miễn phí.

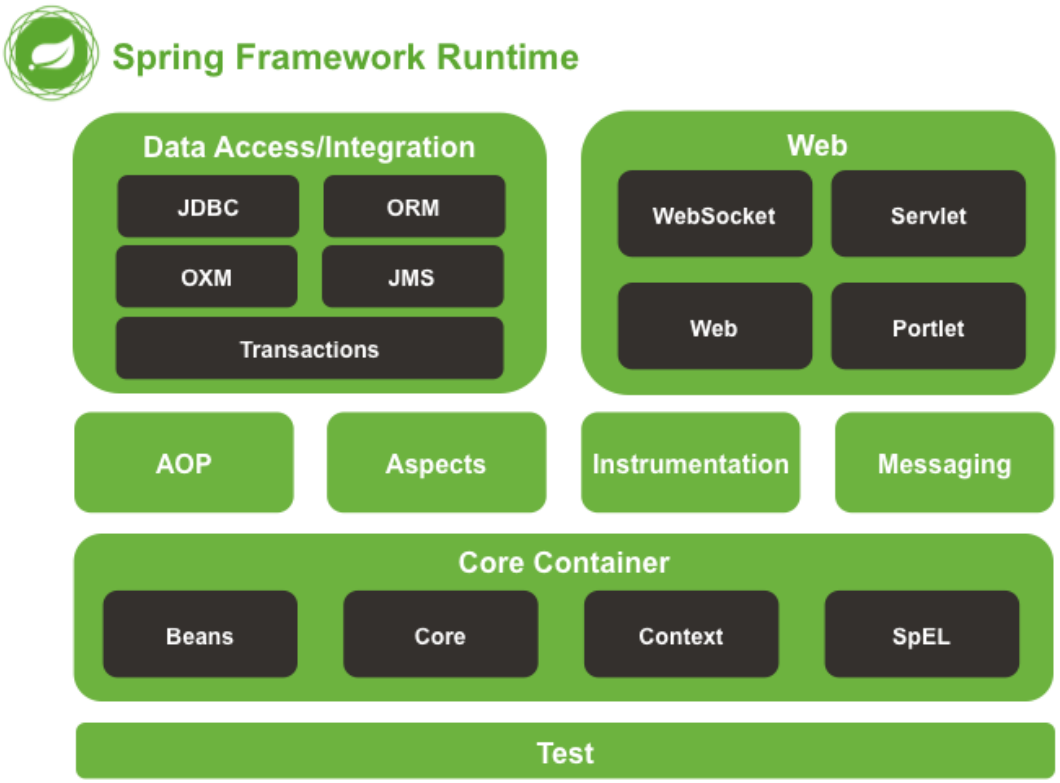
4.2.2. Spring framework

4.2.2.1. Tổng quan

Spring là framework phát triển ứng dụng phổ biến nhất dành cho Java Enterprise. Ban đầu nó được viết bởi Rod Johnson và lần đầu tiên được phát hành theo giấy phép Apache 2.0 vào tháng 6 năm 2003. Spring có kích thướng nhẹ, phiên bản cơ bản của Spring framework có kích thước khoảng 2MB.

Spring framework là một Java Platform mã nguồn mở, một giải pháp gọn nhẹ dành cho Java Enterprise. Với Spring Framework các nhà phát triển có thể tạo ra các mã có hiệu suất cao, dễ kiểm thử và có thể sử dụng lại được.

Các tính năng core của Spring Framework có thể được sử dụng trong việc phát triển bất kỳ ứng dụng Java nào. Bên cạnh đó, phần mở rộng được sử dụng để xây dựng các ứng dụng web trên nền tảng Java EE. Mục tiêu của Spring Framework là làm cho việc phát triển ứng dụng J2EE dễ dàng hơn và thúc đẩy việc lập trình tốt hơn bằng mô hình POJO-based.



Hình 4.2 Tổng quan kiến trúc của Spring framework

4.2.2.2. Ưu điểm

- Nhẹ và đơn giản trong việc phát triển ứng dụng của bạn.
- Dependency Injection hoặc Inversion of Control được sử dụng để giúp các component tách rời, độc lập với nhau. Spring container sẽ giúp gắn kết những components này lại với nhau theo đặc tả business của bạn.
- Spring IoC container quản lý vòng đời của Spring Bean và các cấu hình của project chẳng hạn như JNDI lookup.
- Spring MVC framework được sử dụng cho phát triển ứng dụng web rất dễ dàng với việc hỗ trợ rất tốt các tính năng web services, json,...

- Hỗ trợ quản lý transaction, JDBC operations, File uploading, Exception Handling,... rất dễ dàng bằng cách cấu hình được rút gọn, thay vào đó là sử dụng annotation hoặc spring bean configuration file.
- Làm giảm đi sự phụ thuộc giữa các components khác nhau của ứng dụng, Spring IoC container làm nhiệm vụ khởi tạo resources hoặc beans và "tiêm - inject" chúng theo sự phụ thuộc khác nhau.
- Thực hiện unit test case rất dễ bởi vì business logic của bạn không có sự phụ thuộc trực tiếp. Việc thực hiện chỉ là viết test configuration và inject mock bean cho các mục đích test khác nhau.
- Làm giảm đi khối lượng code rất nhiều, chẳng hạn như việc khởi tạo đối tượng, open/close các resources,...
- Spring framework chia thành nhiều module riêng biệt, do đó việc sử dụng các features trong Spring framework rất tự do... Ví dụ như ứng dụng không sử dụng tính năng transaction, thì không cần thiết phải thêm dependency này vào.
- Spring framework hỗ trợ hầu hết các tính năng của Java EE, thậm chí còn nhiều hơn nữa.

4.2.2.3. Ứng dụng vào đề tài

Ứng dụng được xây dựng hoàn toàn bằng Spring framework theo mô hình MVC

4.2.3. PostgresSQL

4.2.3.1. Tổng quan

PostgreSQL là hệ quản trị cơ sở dữ liệu được viết theo hướng mã nguồn mở và rất mạnh mẽ. Hệ quản trị cơ sở dữ liệu này đã có hơn 15 năm phát triển, đồng thời kiến trúc đã được kiểm chứng và tạo được lòng tin với người sử dụng về độ tin cậy, tính toàn vẹn dữ liệu, và tính đúng đắn. PostgreSQL có thể chạy trên tất cả các hệ điều hành, bao gồm cả Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), và Windows. Do nó hoàn toàn tuân thủ ACID, có hỗ trợ đầy đủ các foreign

keys, joins, views, triggers, và stored procedures (trên nhiều ngôn ngữ). Hệ quản trị này còn bao gồm các kiểu dữ liệu SQL: 2008 như INTEGER, NUMBER, BOOLEAN, CHAR, VARCHAR, DATE INTERVAL, và TIMESTAMPs. PostgreSQL cũng hỗ trợ lưu trữ các đối tượng có kiểu dữ liệu nhị phân lớn, bao gồm cả hình ảnh, âm thanh, hoặc video. Hệ quản trị cơ sở dữ liệu này được sử dụng thông qua giao diện của các ngôn ngữ C / C + +, Java,. Net, Perl, Python, Ruby, Tcl, ODBC…

Là một hệ quản trị cơ sở dữ liệu mạnh, PostgreSQL tự hào có các tính năng phức tạp như kiểm soát truy cập đồng thời nhiều phiên bản (MVCC), khôi phục dữ liệu tại từng thời điểm (Recovery), quản lý dung lượng bảng (tablespaces), sao chép không đồng bộ, giao dịch lồng nhau (savepoints), sao lưu trực tuyến hoặc nội bộ, truy vấn phức tạp và tối ưu hóa, và viết trước các khai báo để quản lý và gỡ lỗi. PostgreSQL hỗ trợ bộ ký tự quốc tế, hỗ trợ bảng mã nhiều byte, Unicode, và cho phép định dạng, sắp xếp và phân loại ký tự văn bản (chữ hoa, thường). PostgreSQL còn được biết đến với khả năng mở rộng để nâng cao cả về số lượng dữ liệu quản lý và số lượng người dùng truy cập đồng thời. Đã từng có những hệ thống PostgreSQL hoạt động trong môi trường thực tế thực hiện quản lý vượt quá 4 terabyte dữ liệu.

4.2.3.2. Ưu điểm

- Đầy đủ các chức năng theo chuẩn:

  PostgreSQL tự hào tuân thủ theo các các tiêu chuẩn về hệ quản trị cơ sở dữ liệu. Thực hiện ngôn ngữ truy vấn của nó mạnh mẽ và phù hợp với tiêu chuẩn ANSI-SQL: 2008. Hệ quản trị cơ sở dữ liệu này còn hỗ trợ đầy đủ cho các truy vấn con (bao gồm cả subselects trong mệnh đề FROM), xác nhận đọc và mức độ biệt lập giao dịch riêng. Đồng thời, PostgreSQL có đầy đủ danh mục các loại quan hệ và hỗ trợ nhiều lược đồ (Diagram) cho mỗi cơ sở dữ liệu. Các cơ sở dữ liệu của hệ quản trị này cũng thể truy cập thông qua các scheme như tiêu chuẩn của ngôn ngữ truy vấn SQL.

  Các tính năng toàn vẹn dữ liệu bao gồm khóa chính, khóa ngoại, tầng cập nhật / xóa, kiểm tra hạn chế, ràng buộc duy nhất, và những hạn chế không null.

  PostgreSQL cũng có một loạt các phần mở rộng và các tính năng tiên tiến. Trong số các tiện ích đó như cột tự động tăng theo trình tự, và LIMIT /

OFFSET cho phép trả về kết quả từng phần. PostgreSQL hỗ trợ compound, unique, partial, và functional indexes mà ta có thể sử dụng các phương thức như B-tree, R-tree, hash hoặc GiST.

Gist lập chỉ mục (Generalized Search Tree) là một hệ thống tiên tiến trong đó tập hợp một mảng rộng các thuật toán khác nhau nhằm sắp xếp và tìm kiếm trên các cây bao gồm B-tree, B+-tree, R-tree, partial sum trees, ranked B+-trees những loại cây khác. Nó cũng cung cấp một giao diện cho phép tạo ra các kiểu dữ liệu tùy chỉnh cũng như các phương pháp truy vấn mở rộng để tìm kiếm chúng. Như vậy, nhìn chung Gist cung cấp sự linh hoạt để xác định những gì bạn lưu trữ, cách bạn lưu trữ, và xác định cách thức mới để tìm kiếm --- cách vượt xa những gì được cung cấp bởi các tiêu chuẩn B-tree, R-tree và những thuật toán thông thường khác.

Gist là nền tảng cho nhiều dự án nào sử dụng PostgreSQL như OpenFTS và PostGIS. OpenFTS ((Open Source Full Text Search engine) cung cấp lập chỉ mục dữ liệu trực tuyến và lập thứ tự để tìm kiếm cơ sở dữ liệu. PostGIS là một dự án hỗ trợ thêm cho các đối tượng địa lý trong PostgreSQL. PostgreSQL cho phép nó được sử dụng như một cơ sở dữ liệu chung cho các hệ thống thông tin địa lý (GIS), giống như SDE của ESRI hoặc không gian mở rộng của Oracle.

Các tính năng tiên tiến khác bao gồm thừa kế bảng, một hệ thống quy tắc, và các sự kiện với cơ sở dữ liệu. Bảng thừa kế đặt sử dụng cách hướng đối tượng để tạo ra bảng. Tính năng này cho phép thiết kế cơ sở dữ liệu mới lấy từ các bảng khác, xử lý chúng như các lớp cơ sở. Thậm chí tốt hơn, PostgreSQL hỗ trợ cả đơn và đa thừa kế.

Hệ thống quy định, còn được gọi là hệ thống viết lại truy vấn, cho phép các nhà thiết kế cơ sở dữ liệu tạo ra các quy tắc xác định các hoạt động cụ thể cho một bảng hoặc view, và tự động chuyển đổi chúng thành hoạt động thay thế khi chúng được xử lý.

- Tính tùy biến cao :

  PostgreSQL chạy thủ tục lưu trữ trong hơn mườingôn ngữ lập trình, bao gồm cả Java, Perl, Python, Ruby, Tcl, C/C++ và trong chính PL / pgSQL, hệ thốngtương tự như Oracle PL / SQL. Thư viện với hàng trăm chức năng được xây dựng từ những chức năng cơ bản như chuỗi số đến các thuật toán phức tạp như mã hóa và đặc biệt tương thích với Oracle. Trigger và các thủ tục lưu trữ có thể được viết bằng C thêm vào cơ sở dữ liệu như là một thư viện, cho phép linh hoạt mở rộng khả năng của mình. Tương tự như vậy, PostgreSQL bao gồm một frameworkcho phép các nhà phát triển để xác định và tạo ra các kiểu dữ liệu của riêng mình cùng với việc hỗ trợ xậy dựng các chức năng với những

toán tử mới nhằm sử dụng hiệu quả. Như vậy, người dùng có thể tạo ra một loạt các loại dữ liệu mới, tiên tiến từ các dữ liệu gốc ban đầu như địa chỉ địa lý, địa chỉ mạng…
Vì có nhiều ngôn ngữ hỗ trợ thủ tục PostgreSQL, nên ta có thể khai thác nhiều thư viện giao tiếp tốt. Sau đây là một số giao tiếp với Java (JDBC), ODBC, Perl, Python, Ruby, C, C++, PHP, Lisp, Scheme, và Qt.

Trên hết, mã nguồn của PostgreSQL được phát hành theo mã nguồn mở tự do. Với quyền này sẽ mang lại cho bạn sự tự do để sử dụng, sửa đổi và cài đặt PostgreSQL trong bất kỳ hình thức nào mà bạn thích. Bất kỳ sửa đổi, cải tiến, hoặc những thay đổi bạn thực hiện là phục vụ nhu cầu của bản thân các bạn. Với PostgreSQL, đây không chỉ một hệ thống cơ sở dữ liệu mạnh mẽ chuyên nghiệp, PostgreSQL còn là một nền tảng phát triển ứng dụng đơn, web, hoặc các sản phẩm phần mềm thương mại yêu cầu một RDBMS.

4.2.3.1. Ứng dụng vào đề tài

PostgresSQL được chọn làm CSDL cho ứng dụng mà khóa luận thực hiện

4.3. Phân tích thiết kế hệ thống

4.3.1. Đặc tả yêu cầu

4.3.1.1. Mục đích, phạm vi

Việc xây dựng hệ thống giá cả trực tuyến giải quyết các nội dung sau:

- **Crawler** : Có nhiệm vụ thu thập thông tin sản phẩm từ các trang thương mại điện tử. Trong phạm vi khóa luận này, sẽ tập trung phát triển cralwer thu thập thông tin sản phẩm điện thoại di động.
- **So sánh giá cả :** có nhiệm vụ xử lý thông tin của cralwer, hiển thị và đưa ra kết quả so sánh giá cả sản phẩm.

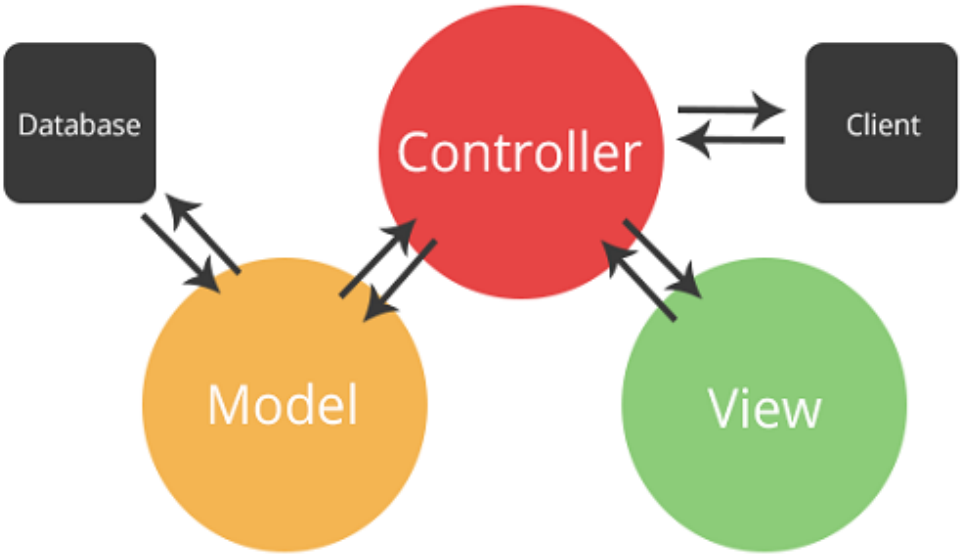
Phạm vi của ứng dụng : Ứng dụng được xây dựng trên nền tảng Spring framework sử dụng PostgresSQL. Hệ thống được xây dựng là kết quả của việc khảo sát yêu cầu, tìm hiểu, nghiên cứu các tài liệu liên quan trong chương 2 và 3.

4.3.1.2. Yêu cầu hệ thống

- **Crawler** : có khả năng thu thập dữ liệu từ nhiều nguồn website khác nhau, xử lý được nội dung của các website dùng nhiều javascript để load trang.
- **So sánh giá cả** : có hệ thống hình ảnh, biểu đồ sinh động, giúp người đọc dễ dàng nắm bắt được sự khác biệt của từng nơi bán.

4.3.2. Kiến trúc hệ thống

Hệ thống được phát triển dựa trên Spring framework và sử dụng mô hình MVC nên tuân thủ đầy đủ các nguyên tắc của mô hình MVC.



Hình 4.3 Mô hình kiến trúc của hệ thống

4.3.3. Thiết kế dữ liệu

STT	Tên bảng	Ý nghĩa
1	products	Lưu các sản phẩm mà crawler thu thập được
2	agents	Lưu các trang web bán hàng mà hệ thống hỗ trợ
3	products_agents	Lưu thông tin của sản phẩm ứng với từng trang bán hàng
4	agent_rules	Lưu luật rút trích dữ liệu cho từng agents
5	require_terms	Lưu các label có thể có cũa nội dung cần rút trích cho một sản phẩm
6	agent_loadmore_methods	Lưu cách thức crawler xử lý một trang web để lấy được nội dung html đầy đủ
7	require_formats	Lưu các tổ hợp format có thể có của các require_term
8	crawling_requires	Lưu các nội dung cần rút trích của một sản phẩm
9	inputs_styles	Lưu các tổ hợp có thể có của thẻ input trong các website
10	attributes	Lưu các attribute có thể có của thẻ input (Vd : class, id,..)
11	values	Lưu các value có thể có của các attribute của thẻ input
12	placeholder	Lưu các nội dung thường được dùng làm place holder cho thẻ input

13	ignored_words	Lưu các từ crawler sẽ bỏ qua khi xử lý nội dung html
14	ignored_tags	Lưu các tag html cralwer sẽ bỏ qua khi xử lý nội dung html
15	format_tags	
16	removed_tag	Lưu các tag thường dùng để định dạng giá tiền cũ (gạch ngang, làm mờ…)
17	product_specifics	Lưu các thuộc tính phụ có thể có của sản phẩm
18	Specific_details	Lưu nội dung chi tiết của các thuộc tính

Bảng 4.1 Danh sách các bản dữ liệu của hệ thống

- **Bảng products:**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	name	varchar	Tối đa 100 ký tự	
3	image	varchar	Tối đa 100 ký tự	Đường dẫn đến hình ảnh sản phẩm
4	visit_count	integer		Số lượt truy cập sản phẩm
5	rating	numeric		
6	agent_count	integer		Tổng số nơi bán sản phẩm

Bảng 4.2 Bảng products

- **Bảng agents**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	code	varchar	Tối đa 10 ký tự	
3	name	varchar	Tối đa 50 ký tự	Name cũng là địa chỉ web của trang bán hàng
4	search_url	varchar	Tối đa 200 ký tự	Đường liên kết đến trang tìm kiếm sản phẩm của website
5	Is_deleted	boolean		Xác định xem một webiste có còn nằm trong danh sách truy vấn hay không

Bảng 4.3 Bảng agents

- **Bảng products_agents**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	product_id	integer	Khóa ngoại	
3	agent_id	integer	Khóa ngoại	
4	price	numeric		Giá của sản phẩm
5	url	varchar	Tối đa 200 ký tự	Đường liên kết tới trang sản phẩm

Bảng 4.4 Bảng prodcuts_agents

- **Bảng agent_rules**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	agent_id	integer	Khóa ngoại	
2	require_id	integer	Khóa ngoại	
3	format	varchar		
4	Rule_index	integer		Cho biết index của nội dung trong PDU

Bảng 4.5 Bảng agent_rules

- **Bảng require_terms :**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	require_id	integer	Khóa ngoại	
3	term	varchar	Tối đa 50 ký tự	Nội dung của term

Bảng 4.6 Bảng require_terms

- **Bảng require_formats**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	require_id	integer	Khóa ngoại	

3	format	varchar	Tối đa 100 ký tự	Nội dung của format

Bảng 4.7 Bảng require_formats

- **Bảng crawling_requires**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	code	varchar	Tối đa 10 ký tự	
3	text	varchar	Tối đa 20 ký tự	Nội dung của require

Bảng 4.8 bảng crawling_requires

- **Bảng input_styles**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	attribute_id	integer	Khóa ngoại	
2	value_id	integer	Khóa ngoại	

Bảng 4.9 bảng input_stlyes

- **Bảng attributes**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	

2	attribute	varchar	Tối đa 50 ký tự	

Bảng 4.10 Bảng attributes

- **Bảng values**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	value	varchar	Tối đa 50 ký tự	

Bảng 4.11 Bảng values

- **Bảng agent_loadmore_methods**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	agent_id	integer	Khóa ngoại	
3	method	varchar	Tối đa 50 ký tự	Hiện tại có 2 method là ajax (load bằng javascript) hoặc url (load trang kế bằng url)
4	value	varchar	Tối đa 50 ký tự	Nộ dung của method
5	xpath	varchar	Tối đa 200 ký tự	Xpath của element html thực hiện việc load trang kế tiếp

Bảng 4.12 agent_loadmore_methods

- **Bảng placeholder**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	value	varchar	Tối đa 50 ký tự	

Bảng 4.13 Bảng placeholder

- **Bảng product_specifics**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	code	varchar	Tối đa 10 ký tự	
3	text	varchar	Tối đa 20 ký tự	Nội dung của specific

Bảng 4.14 Bảng product_specifics

- **Bảng specific_details**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	

2	specific_id	integer		
3	possible_text	varchar	Tối đa 50 ký tự	Các text có thể có của một thuộc tính

Bảng 4.15 Bảng spceific_details

- **Bảng ignored_words**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	word	varchar	Tối đa 50 ký tự	

Bảng 4.16 Bảng ignored_words

- **Bảng ignored_tag**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	tag	varchar	Tối đa 50 ký tự	

Bảng 4.17 Bảng ignored_tags

- **Bảng format_tags**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	tag	varchar	Tối đa 50 ký tự	

Bảng 4.18 bảng format_tags

- **Bảng remove_tags**

STT	Tên	Kiểu	Ràng buộc	Ý nghĩa
1	id	serial	Khóa chính, tăng tự động	
2	tag	varchar	Tối đa 50 ký tự	

Bảng 4.19 Bảng remove_tags

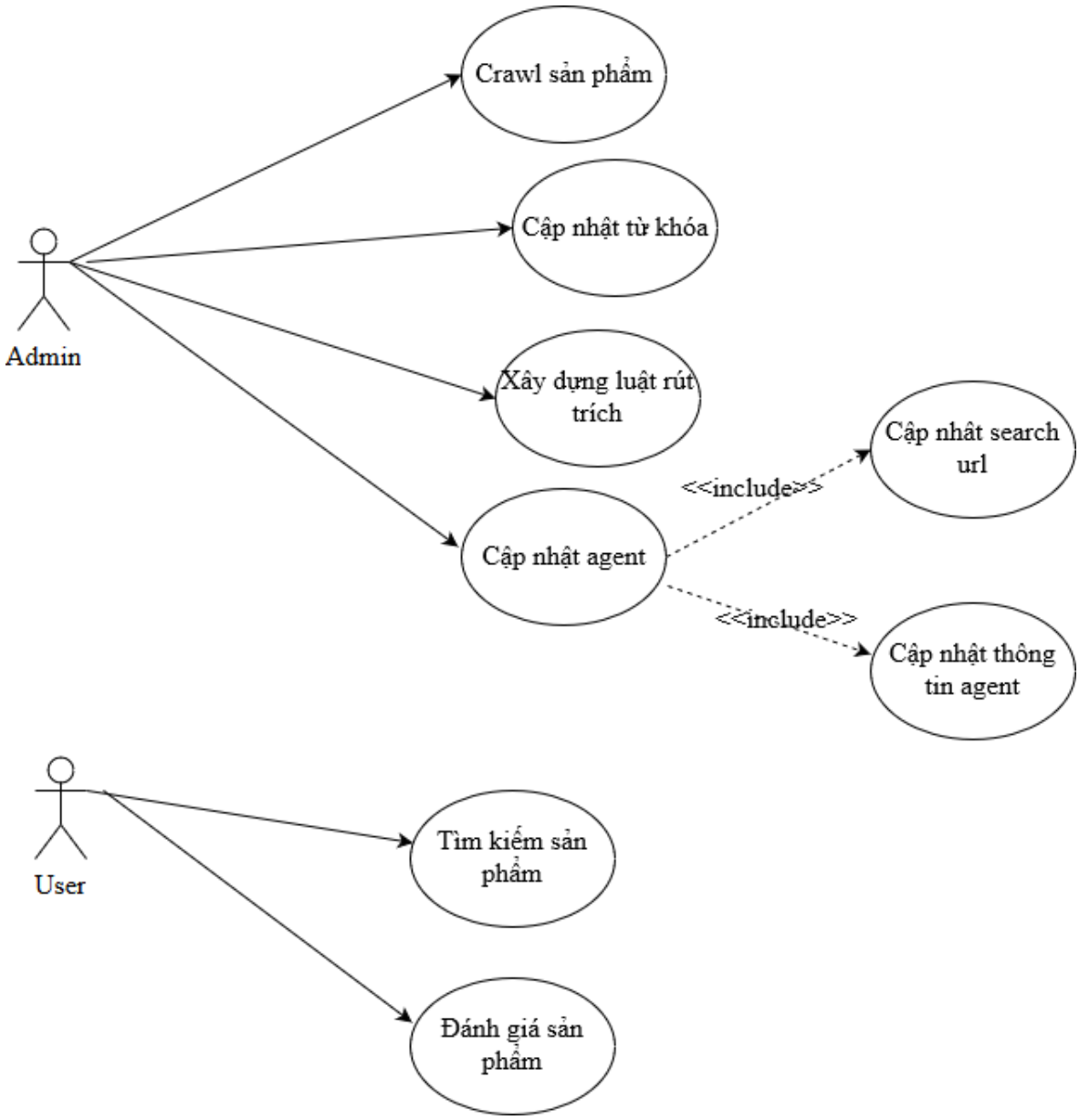
4.3.4. Sơ đồ usecase

4.3.4.1. Danh sách actor

STT	Actor	Ý nghĩa
1	Admin	Là người thực hiện các thao tác chạy crawler, cập nhật cơ sở tri thức, cập nhật các rule rút trích dữ liệu
2	User	Là người có nhu cầu tham khảo giá cả giữa các sản phẩm, truy cập vào website để tìm kiếm thông tin so sánh

Bảng 4.20 Danh sách Actor

4.3.4.2. Sơ đồ usecase tổng quát



Hình 4.4 Sơ đồ use case tổng quát

STT	Tên use case	Ý nghĩa
1	Crawl sản phẩm	Cho crawler thu thập thông tin sản phẩm từ các web thương mại điện tử
2	Cập nhật từ khóa	Cập nhật các loại từ khóa, làm cơ sở tri thức cho crawler
3	Xây dựng luật rút trích	Cho crawler test các trang thương mại điện tử để xây dựng luật rút trích
4	Cập nhật agent	Cập nhật các thông tin của webiste bán hàng, bao gồm cả để crawler tìm kiếm search url
5	Tìm kiếm sản phẩm	Người dùng tìm kiếm sản phẩm muốn so sánh giá
6	Đánh giá sản phẩm	Người dùng đưa ra đánh giá về sản phẩm

Bảng 4.21 Danh sách các use case của hệ thống

4.3.4.3. Đặc tả usecase

- **Usecase Crawl sản phẩm**

Tên Usecase	Crawl sản phẩn
Tóm tắt	Cho crawler thu thập thông tin sản phẩm từ các website bán hành
Actor	Admin
Dòng sử kiện chính	Admin chọn website cần lấy sản phẩm. Amin nhập tên hãng sản xuất muốn tìm và nhất nút Go

	Hệ thống sẽ kiểm tra xem website này đã có rule rút trích chưa. Nếu đã có, sẽ tiến hành crawl data. Hệ thống hiển thị danh sách sản phẩm thu thập được
Dòng sự kiện khác	Nếu website chưa có rule trích xuất, hệ thống sẽ hiện thông báo lỗi : • *Website thiếu rule trích xuất*
Các yêu cầu đặc biệt	Nội dung nhập nên là các nhãn hiệu điện thoại
Trạng thái hệ thống trước khi thực hiện Use case	Actor: Admin Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển thị danh sách các sản phẩm thu thập được
Ngoại lệ	Không có

Bảng 4.22 Usecase crawl data

- **Usecase cập nhật từ khóa**

Tên Usecase	Cập nhật từ khóa
Tóm tắt	Cập nhật từ khóa làm cơ sở tri thức cho crawler
Actor	Admin
Dòng sử kiện chính	Admin chọn website cần lấy sản phẩm. Amin nhập tên hãng sản xuất làm dữ liệu mậu và nhất nút Go.

	Hệ thống sẽ khảo sát webiste dựa trên cơ sở tri thức hiện có, hiển thị ra màn hình danh sách sản phẩm thu thập được và luật rút trích cho website.
Dòng sự kiện khác	Hệ thống không thể tính toán được
Các yêu cầu đặc biệt	Không có
Trạng thái hệ thống trước khi thực hiện Use case	Actor: Admin Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển thị danh sách các từ khóa sau khi được cập nhật
Ngoại lệ	Không có

Bảng 4.23 Usecase cập nhật từ khóa

- **Usecase xây dựng luật rút trích**

Tên Usecase	xây dựng luật rút trích
Tóm tắt	Xây dựng luật rút trích cho từng website
Actor	Admin
Dòng sử kiện chính	Admin chọn loại từ khóa cần cập nhật. Amin nhập từ kháo và ấn save nếu muốn lưu hoặc chọn từ khóa và ấn xóa nếu muốn xóa Hệ thống sẽ cập nhật lại thông tin từ khóa và hiển thị danh sách từ khóa mới.

Dòng sự kiện khác	Hệ thống không thể tính toán được luật rút trích cho trang, yêu cầu admin cập nhật cơ sở tri thức và thử lại
Các yêu cầu đặc biệt	Nội dung nhập nên là các nhãn hiệu điện thoại
Trạng thái hệ thống trước khi thực hiện Use case	Actor: Admin Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển thị luật rút trích cho webiste
Ngoại lệ	Không có

Bảng 4.24 Usecase xây dựng luật rút trích

- **Usecase cập nhật agent**

Tên Usecase	Cập nhật agent
Tóm tắt	Cập nhật các thông tin của webiste bán hàng, bao gồm cả để crawler tìm kiếm search url
Actor	Admin
Dòng sử kiện chính	Admin chọn agent cần cập nhật. Amin nhập thông tin mới cho agent và nhấn lưu hoặc chọn nút “check url” để hệ thống cập nhật lại toàn bộ search url cho các agent. Hệ thống sẽ cập nhật lại thông tin agent và hiển thị danh sách agent được cập nhật

Dòng sự kiện khác	Hệ thống không thể tính toán được search url cho các agent, hiển thị thông báo lỗi
Các yêu cầu đặc biệt	
Trạng thái hệ thống trước khi thực hiện Use case	Actor: Admin Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển danh sách agent sau khi cập nhật
Ngoại lệ	Không có

Bảng 4.25 Usecase cập nhật agent

- **Usecase tìm kiếm sản phẩm**

Tên Usecase	tìm kiếm sản phẩm
Tóm tắt	Tìm kiếm sản phẩm và so sánh giá cả
Actor	User
Dòng sử kiện chính	User nhập tên sản phẩm muốn tìm và nhấn nút tìm kiếm Hệ thống sẽ kiểm tra dữ liệu và trả về danh sách sản phẩm theo từ khóa tìm kiếm
Dòng sự kiện khác	Hệ thống không tìm thấy sản phẩm
Các yêu cầu đặc biệt	

Trạng thái hệ thống trước khi thực hiện Use case	Actor: User Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển danh sách sản phẩm dựa vào khóa tìm kiếm
Ngoại lệ	Không có

Bảng 4.26 Usecase tìm kiếm sản phẩm

- **Usecase đánh giá sản phẩm**

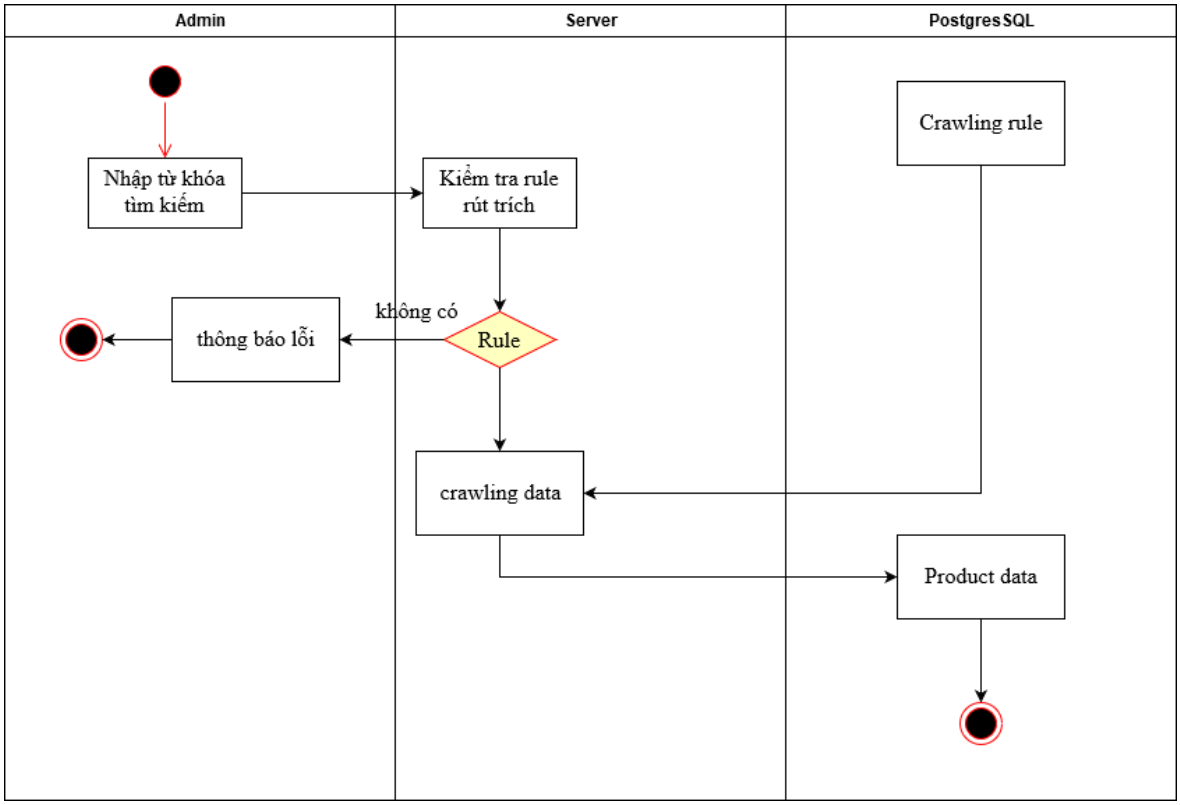
Tên Usecase	Đánh giá sản phẩm
Tóm tắt	Người dùng đưa ra đánh giá về sản phẩm
Actor	User
Dòng sử kiện chính	Từ màn hình sản phẩm User chọn mức điểm đánh giá. Hệ thống hiển thị form nhập thông tin cá nhân. User nhập email, họ tên và nhấn nút “rate” Hê thống cập nhật điểm đánh giá cho sản phẩm
Dòng sự kiện khác	Trường hợp email đã được dùng để đánh giá cho sản phẩm rồi thì hệ thống sẽ hiển thị thông báo lỗi
Các yêu cầu đặc biệt	

Trạng thái hệ thống trước khi thực hiện Use case	Actor: User Yêu cầu : không có
Trạng thái hệ thống sau khi thực hiện Use case	Hiển màn hình chi tiết sản phẩm
Ngoại lệ	Không có

Bảng 4.27 Usecase đánh giá sản phẩm

4.3.5. Sơ đồ một số hoạt động chính

- **Sơ đồ hoạt động Crawl data**



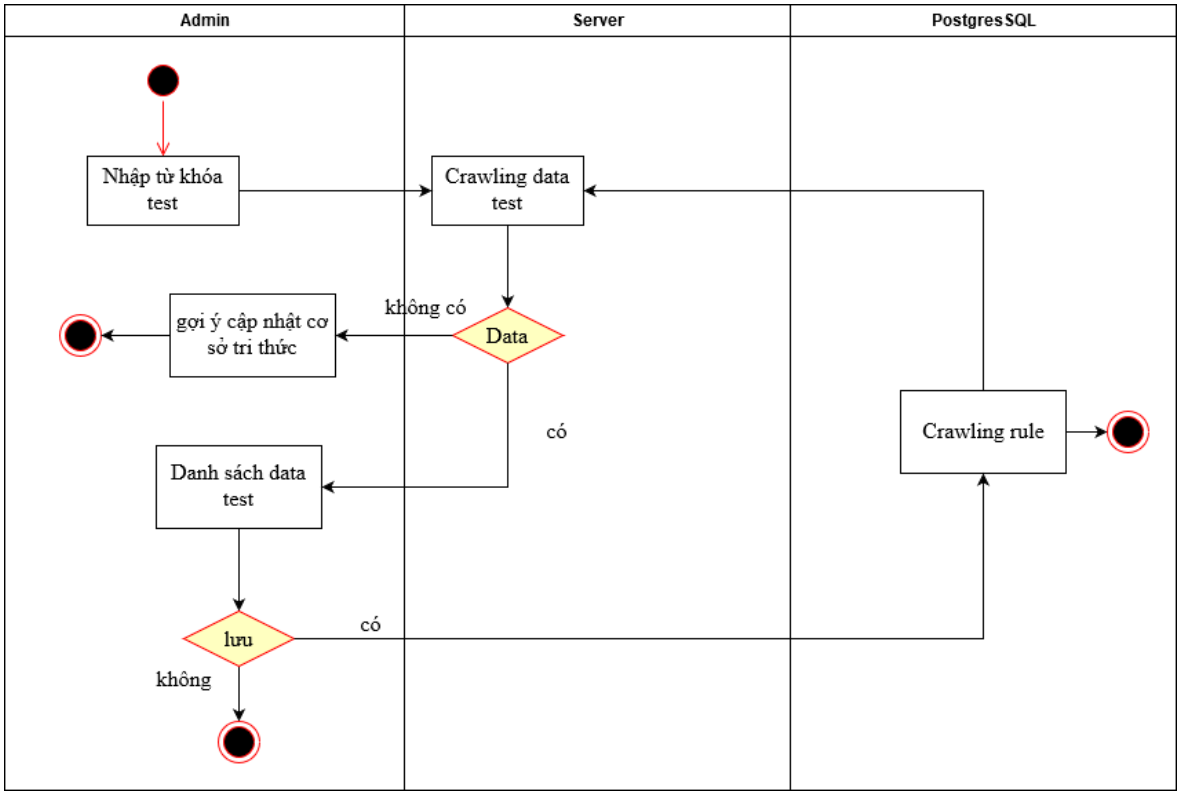
Hình 4.5 Sơ đồ hoạt động "Crawl data"

Mục đích : thu thập dữ liệu sản phẩm cho hệ thống.

Mô tả :

1. Chọn agent muốn thu thập dữ liệu
2. Nhập từ khóa
3. Nhấn nút “Go”

- **Sơ đồ hoạt động xây dựng luật rút trích**



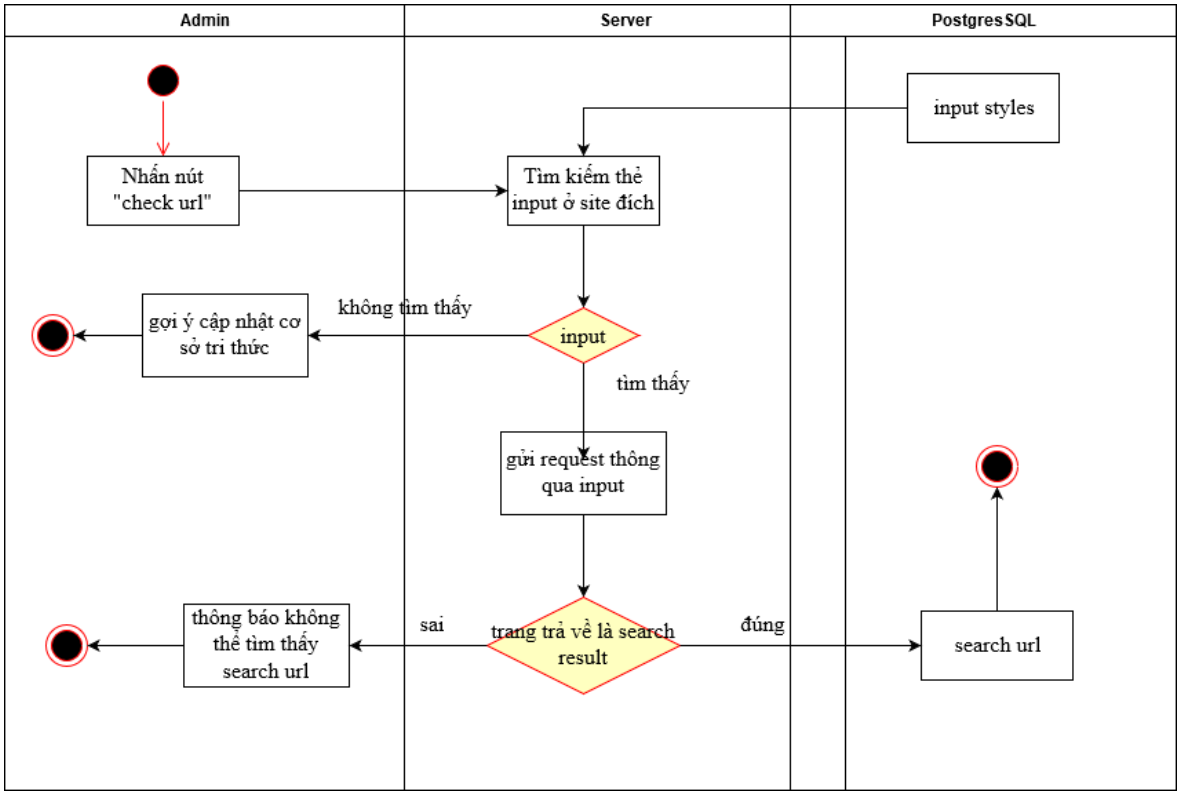
Hình 4.6 sơ đồ hoạt động “xây dựng luật rút trích”

Mục đích : xây dựng các bọ luật craw dữ liệu cho từng website, giúp rút ngắn thời gian lấy data khi phải thực hiện craw data nhiều lần trên cùng một website.

Mô tả :

1. Chọn agent muốn xây dựng luật rút trích
2. Nhập keyword test
3. Nhấn nút “Go”

- **Sơ đồ hoạt động tạo search url**



Hình 4.7 Sơ đồ hoạt động “tạo search url”

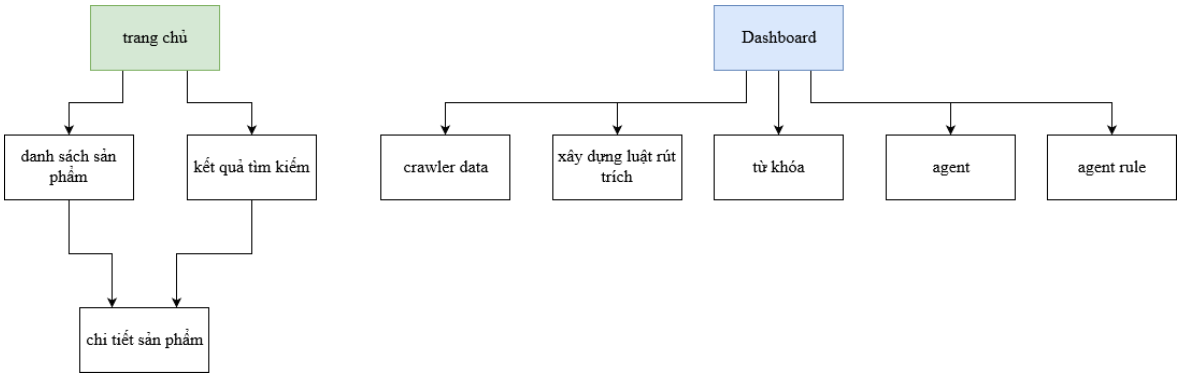
Mục đích: mỗi trang bán hàng đều có một đường dẫn đến trang tìm kiếm sản phẩm, việc tìm thấy đường dẫn này và lưu lại để sử dụng sẽ giúp tăng tốc tìm kiếm vì ta không phải tìm submit form mỗi lần crawling data.

Mô tả :

1. Chọn agent management.
2. Nhấn nút “check url”.

4.3.6. Thiết kế giao diện

4.3.6.1. Sơ đồ liên kết màn hình



Hình 4.8 Sơ đồ liên kết màn hình

4.3.6.2. Danh sách màn hình

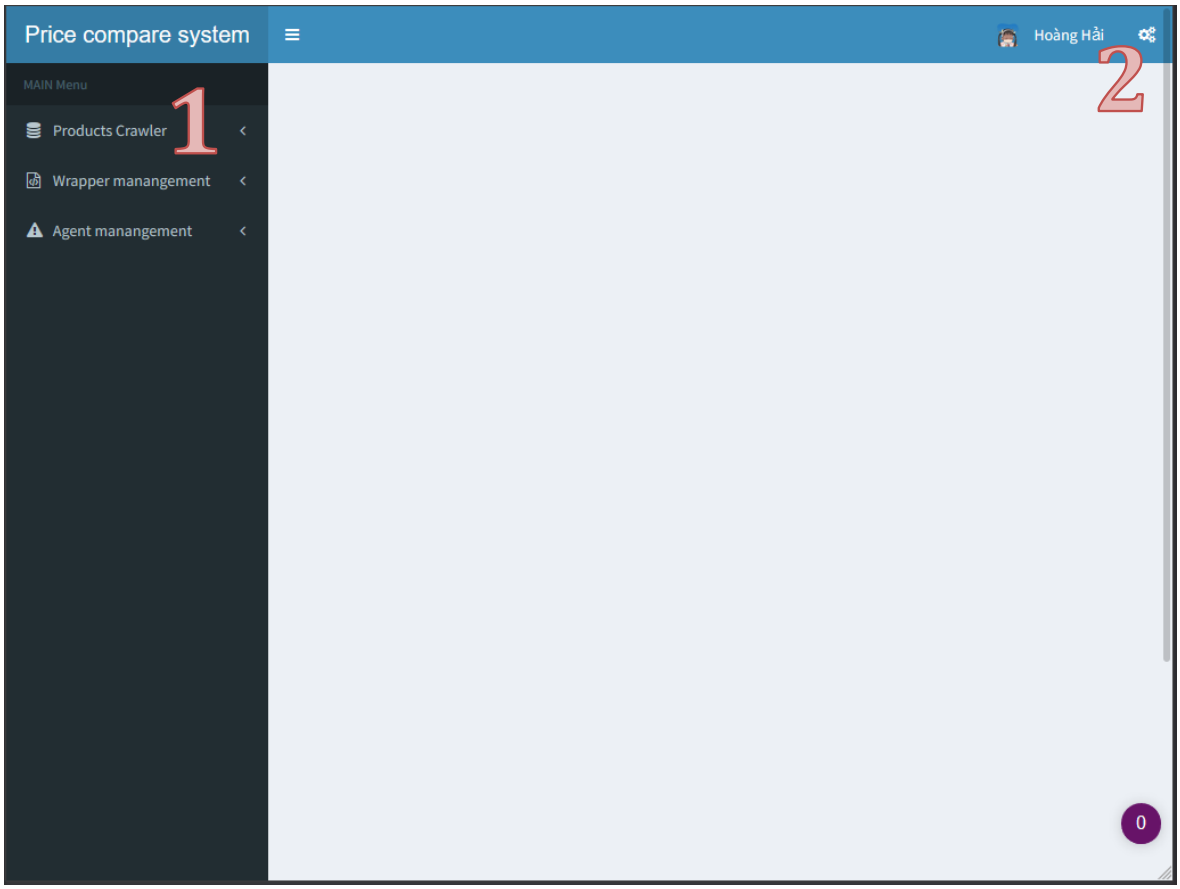
STT	Khối	Tên	Ý nghĩa
1	Admin	Dashboard	Màn hình trang chủ của khối admin
2		Cawler Data	Màn hình thực hiện việc lấy dữ liệu từ website
3		Xây dựng luật rút trích	Màn hình thực hiện việc tạo các luật rút trích cho website
4		Từ khóa	Màn hình quản lý các từ khóa của hệ thống
5		Agent	Màn hình quản lý các agent và kiểm tra search url của agent

6		Agent rule	Màn hình nhập các rule cho một số agent đặc biệt
7	User	Trang chủ	Trang chủ của khối user
8		Danh sách sản phẩm	Hiển thị danh sách sản phẩm theo từng mục
9		Kết quả tìm kiếm	Kết quả tìm kiếm sản phẩm
10		Chi tiết sản phẩm	Chi tiết của sản phẩm, bao gồm so sánh giá, biểu đồ và rating.

Bảng 4.28 Danh sách các màn hình

4.3.6.3. Chi tiết các màn hình

- Màn hình Dashboard

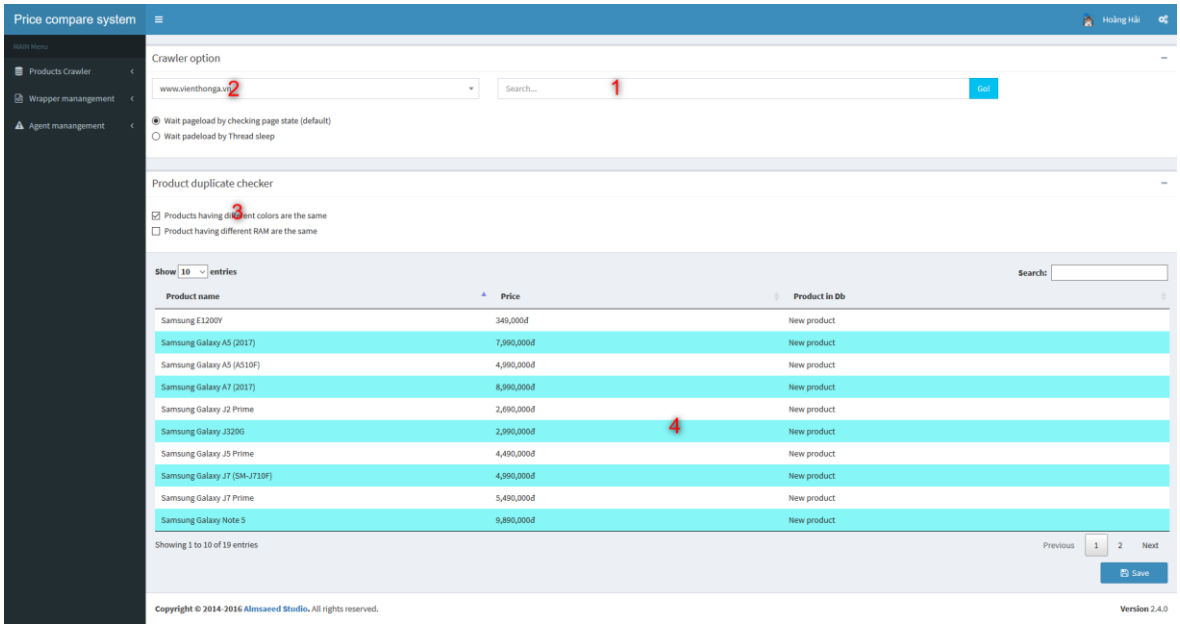


Hình 4.9 Màn hình dashboard

STT	Tên	Chức năng
1	Nhóm menu chính	Các chức năng chính của hệ thống : crawl data, quản lý từ khóa, ….
2	Nhóm menu giao diện	Các tùy chọn theme cho website

Bảng 4.29 Mô tả các thành phần của màn hình Dashboard

- **Màn hình crawler data:**

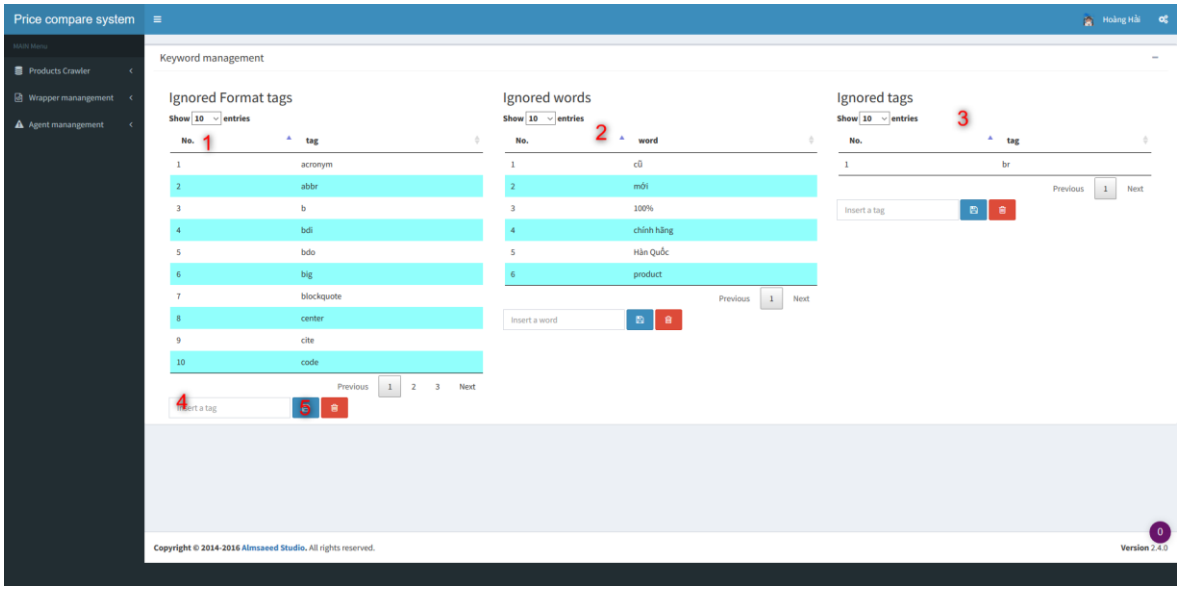


Hình 4.10 Màn hình crawler data

STT	Tên	Chức năng
1	Thanh search	Nhập dữ liệu để crawler gửi request tìm kiếm
2	Danh mục các agent	Danh mục các agent hiện được hệ thống hỗ trợ
3	Các quy tùy chọn so sánh sản phẩm	Hiển thị các tùy chọn dùng để so sánh tên của 2 sản phẩm với nhau
4	Kết quả crawling data	Hiển thị dữ liệu crawl được từ agent

Bảng 4.30 Mô tả các thành phần màn hình crawler data

- **Màn hình từ khóa**

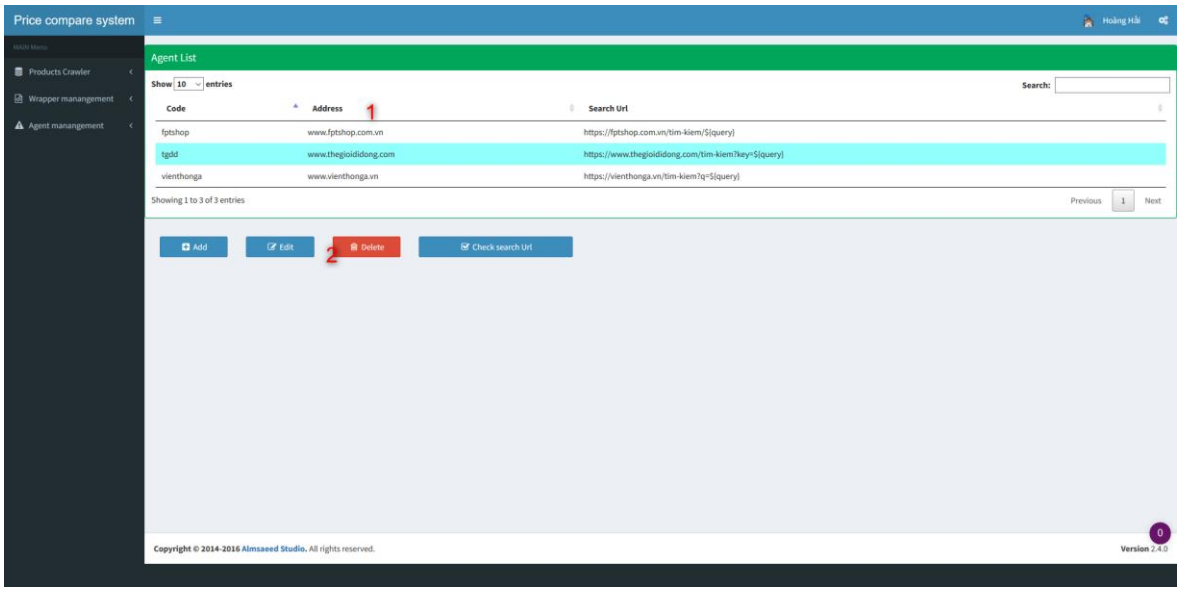


Hình 4.11 màn hình từ khóa

STT	Tên	Chức năng
1	Bảng fotmat tags	Hiển thị các format tag
2	Bảng ignored words	Hiển thị các ignored word
3	Bảng ignored tags	Hiển thị các ignored tag
4	Các thanh nhập liệu	Thanh input để nhập mới từ khóa
5	Nhóm phím chức năng	Lưu hoặc xóa từ khóa

Bảng 4.31 mô tả chi tiết màn hình từ khóa

- **Màn hình agent**

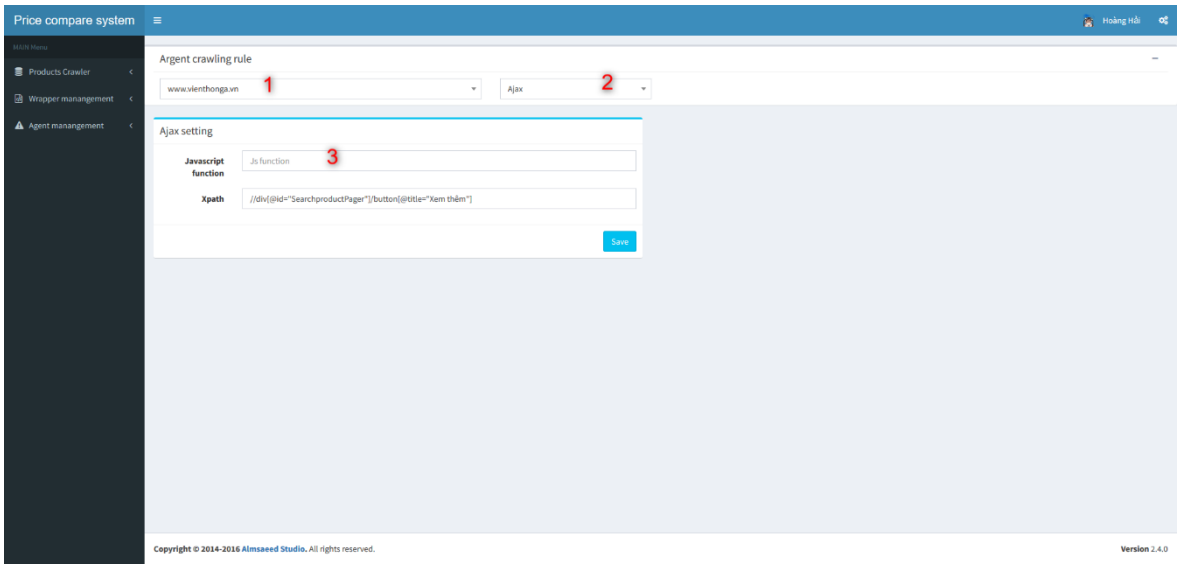


Hình 4.12 Màn hình agent

STT		Tên	Chức năng
1		Bảng agent	Hiển thị danh sách các agent hiện có
2	2.1	Nút add	Mở form nhập agent mới
	2.2	Nút edit	Edit agent được chọn
	2.3	Nút delete	Xóa một agent được chọn
	2.1	Nút Check url	Thực hiện tạo search url cho các agent

Bảng 4.32 Mô tả chi tiết màn hình agent

- **Màn hình agent rule**



Hình 4.13 Màn hình agent rule

STT	Tên	Chức năng
1	Danh sách agent	Hiển thị các agent của hệ thống
2	Danh sách method	Hiển thị các phương thức load tiếp trang
3	Cấu hình method	Hiển thị các cài đặt cho từng method cụ thể

Bảng 4.33 Mô tả màn hình agent rule

Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Báo cáo này đề cập đến việc xây dựng một hệ thống thu thập thông tin sản phẩm và đưa ra so sánh về giá cả giữa các sản phẩm cùng loại.

Trong phần này, nhóm tác giả xin tổng kết đề tài, những đóng góp và đưa ra hướng phát triển tiếp theo của đề tài này.

5.1. Môi trường phát triển và triển khai

5.1.1. Môi trường phát triển

Phần cứng: không yêu cầu.

Phần mềm:

- Hệ quản trị Cơ sở dữ liệu: PostgresSQL 9.6
- Công cụ phân tích, thiết kế: draw.io (online tool)
- Công cụ xây dựng ứng dụng: IntelliJ IDEA 2017, DBVisualizer
- Các thư viện sử dụng: Spring framework, Lombok, thymeleaf, boostrap, Selenium webdriver..,
- Các phần mềm external : PhantomJs.

5.1.2. Môi trường riển khai

Phần cứng: không yêu cầu.
Phần mềm:

- Các trình duyệt web cả trên desktop lẫn smartphone như Chrome, Firefox
- Hiển thị tốt nhất trên FireFox.

5.2. Kết quả đạt được

5.2.1. Về mặt công nghệ

Đã nghiên cứu và áp dụng thành công các công nghệ mới hiện nay như Spring framework sử dụng Spring data JPA cũng như một số công cụ nổi bật khác như PhantomJs, Selenium webdriver... để hiện thực hóa những yêu cầu đã đề ra từ ban đầu. Ngoài ra trong quá trình thiết kế phát triển ứng dụng, nhóm còn vận dụng sử

dụng các mô hình thiết kế kiến trúc như MVC, sử dụng inverse of control thông qua dependency injection để đảm bảo khả năng bảo trì và phát triển của ứng dụng trong tương lai.

5.2.2. Về nội dung nghiên cứu

Đã hoàn thành da số các mục tiêu đặt ra ban đầu, như :

- **Crawler data** : đã xây dựng được crawler data có khả ăng lấy dữ liệu từ nhiều loại website khác nhau, bao gồm các trang web truyền thống và các trang web load dữ liệu bằng javascript. Crawler có khả năng lấy dữ liệu mong muốn từ một cơ sở tri thức mẫu, và hình thành các luạt rút trích từ cơ sở tri thức này.
- **Hệ thống so sánh giá cả** : hê thống bước đầu nhận diện được các sản phẩm lấy về là sản phẩm đã có rồi hay sản phẩm mới, để đưa ra thông tin so sánh chính xác.

5.3. Kết luận

Trong quá trình thực hiện Khóa luận tốt nghiệp, nhóm đã có thêm cơ hội củng cố, tích lũy thêm kiến thức chuyên môn về lập trình web, nghiên cứu các đề tài khoa học cũng như kinh nghiệm làm việc nhóm, lên kế hoạch cho đề tài, dự án, viết báo cáo.... Nắm rõ hơn về quy trình phát triển ứng dụng cũng như xây dựng được 1 framework hoàn chỉnh. Biết cách vận dụng những kiến thức đã học về mẫu thiết kế về hệ thống cũng như xây dựng cơ sở dữ liệu.

5.4. Hướng phát triển

Tuy đã hoàn thành ứng dụng với những tính năng chính nhưng với thời gian có hạn, ứng dụng vẫn chưa thể đáp ứng hết những yêu cầu của người sử dụng. vẫn còn một số vấn đề cần lưu tâm và cải tiến :

- Nâng cao khả năng tự động hóa của crawler thông qua việc động tìm và kiểm tra các phần tử của website.

- Tăng số lượng nội dung mà cralwer có thể lấy được (hiện tại chỉ lấy được tên và giá của sản phẩm)

DANH MỤC TÀI LIỆU THAM KHẢO

- Intelligent Agents and Multi-Agent Systems 6th Pacific Rim International Workshop on Multi-Agents, PRIMA 2003, Seoul, Korea, November 7-8, 2003.
- Web Intelligence (Editors: Zhong, Ning, Liu, Jiming, Yao, Yiyu)
- Selenium webdriver documents : http://www.seleniumhq.org/docs/
- Spring framework documents : https://docs.spring.io/spring/docs/current/spring-framework-reference/