



Website bán quần áo TwentyFive Fashion Kế hoạch test

Mã dự án: SD18316_SOF3031

Mã tài liệu: TP_V1.0

Nhóm:	5
Tên thành viên:	Vũ Việt Anh - PH37175
	Nguyễn Hồng Ánh - PH40387
	Hoàng Thu Trang - PH40450
	Nguyễn Hoài Thu - PH40458
	Ngô Mạnh Cường - PH40460
	Đào Việt Anh - PH40468

<HÀ NỘI, 15/01/2024>

<GROUP5_TEST>

Kế hoạch test trang web bán quần áo Twenty Five

TRANG KÝ

Người lập: Hoàng Thu Trang

<Ngày> 15/01/2024

<Trưởng nhóm>

Người xem xét: Nguyễn Bích Ngọc

<Ngày> _____

<Giảng viên hướng dẫn>

MỤC LỤC

1	GIỚI THIỆU	4
1.1	Mục đích	4
1.2	Thông tin chung	5
1.3	Tài liệu liên quan	6
1.4	Phạm vi test	7
1.5	Ràng buộc	10
1.6	Liệt kê các mạo hiểm	10
2	CÁC YÊU CẦU CHO TEST	11
3	CHIẾN LƯỢC TEST	12
3.1	Các kiểu test	12
3.1.1	Test chức năng (Functional Testing)	13
3.1.2	Test hiệu suất (Performance testing)	16
3.1.3	Test Bảo mật và Kiểm soát truy cập (Security and Access Control Testing)	20
3.1.4	Test hồi quy (Regression Testing)	21
3.2	Giai đoạn test	21
3.3	Các công cụ test	22
4	TÀI NGUYÊN	22
4.1	Nhân lực	22
4.2	Hệ thống	25
4.2.1	Hệ thống phần cứng cần thiết	25
4.2.2	Hệ thống phần mềm cần thiết	26
4.2.3	Công cụ hỗ trợ cho quá trình Test	26
4.2.4	Cấu hình môi trường Test	27
5	CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES)	27
6	CÁC SẢN PHẨM	28
7	NHIỆM VỤ KIỂM TRA PHẦN MỀM BÁN HÀNG THE HABITS SHOP	28
7.1	Đăng ký	28
7.2	Giỏ hàng	29
7.3	Sản phẩm	30

<GROUP5_TEST>*Kế hoạch test trang web bán quần áo Twenty Five*

7.4 Hóa đơn	30
7.5 Lọc sản phẩm	30
7.4 Chỉnh sửa thông tin cá nhân	30

1. GIỚI THIỆU

1.1 Mục đích

- Để xây dựng một trang web bán quần áo hỗ trợ cửa hàng, trang web TwentyFive Fashion tập trung phát triển các mục đích chính như:

1. Quản lý sản phẩm :

- + Tối ưu hóa quá trình nhập, xuất và theo dõi tồn kho quần áo.
- + Hỗ trợ quản lý thông tin sản phẩm, từ lô hàng đến kích thước và màu sắc.

2. Quản lý đơn hàng và giao hàng:

- + Tích hợp hệ thống đơn hàng linh hoạt với quản lý giao hàng hiệu quả.
- + Ghi chép thông tin vận chuyển, theo dõi đơn hàng và thông báo tình trạng vận chuyển.

3. Phân quyền và an toàn dữ liệu:

- + Cung cấp quản lý phân quyền để kiểm soát truy cập thông tin quan trọng.
- + Bảo mật dữ liệu khách hàng và thông tin cửa hàng.

4. Tích hợp hệ thống thanh toán:

- + Tương tác dễ dàng với các hệ thống thanh toán và ghi nhận thông tin giao dịch.
- + Hỗ trợ quy trình thanh toán linh hoạt cho khách hàng và nhân viên.

5. Báo cáo và thống kê nhanh chóng:

- + Tạo báo cáo về doanh số bán hàng, tồn kho và hiệu suất vận chuyển.
- + Hỗ trợ phân tích dữ liệu để đưa ra quyết định kinh doanh hiệu quả.

6. Đơn giản và hiệu quả:

- + Thiết kế giao diện đơn giản và dễ sử dụng cho nhân viên cửa hàng.
- + Tối ưu hóa quy trình làm việc để tiết kiệm thời gian và năng lực.

- Kế hoạch kiểm thử:

1. Giới thiệu kế hoạch kiểm thử

- + Mục tiêu kiểm tra: Đảm bảo rằng phần mềm bán hàng hoạt động mượt mà, đáp ứng đầy đủ chức năng và tuân thủ các yêu cầu đã đặt ra.
- + Kế hoạch thực hiện: Kiểm tra sẽ diễn ra theo giai đoạn phát triển để đảm bảo tính liên tục và hiệu quả.

2. Các yêu cầu cho test:

- + Tình huống test: Mô phỏng các tình huống sử dụng thực tế từ việc tìm kiếm sản phẩm đến quy trình thanh toán.
- + Yêu cầu chức năng và phi chức năng: Xác định các chức năng như quản lý sản phẩm, đơn hàng, khách hàng, cũng như các yêu cầu bảo mật.

3. Chiến lược test:

- + Những vấn đề chính trong chiến lược test là các kỹ thuật được áp dụng và điều kiện để biết khi nào việc test được hoàn thành.
- + Mô tả các kiểu test dùng trong dự án.
- + Liệt kê với mỗi kiểu test tương ứng test cho chức năng nào.
- + Việc test có thể dừng khi nào.

4. Các kiểu test:

- + Kiểm thử hệ thống: Đảm bảo sự tương thích và tương tác giữa các thành phần.
- + Kiểm thử hiệu năng: Đánh giá khả năng xử lý và độ ổn định dưới điều kiện tải cao.
- + Kiểm thử bảo mật: Đảm bảo an toàn và bảo mật thông tin người dùng.

5. Tài nguyên:

- Nhân sự: gồm chuyên viên kiểm thử, lập trình viên, và quản lý dự án.
- Tài nguyên hệ thống: máy chủ, cơ sở dữ liệu, và các công cụ kiểm thử tự động.

6. Các mốc kiểm soát của giai đoạn test:

- + Có thể độc lập với các mốc kiểm soát của dự án, cho biết chính xác thông tin về tình trạng hoàn thành của dự án.

1.2 Thông tin chung

1. Test Chức Năng:

- + Test chức năng (Function Testing): Kiểm tra từng chức năng cụ thể của cửa hàng, bao gồm quản lý sản phẩm, đơn hàng, tài khoản khách hàng.
- + Test giao diện người sử dụng (User Interface Testing): Đảm bảo giao diện người dùng thân thiện, dễ sử dụng và thẩm mỹ.
- + Test dữ liệu và tích hợp dữ liệu (Data and Database Integrity Testing): Kiểm tra độ chính xác và tính toàn vẹn của dữ liệu trong cơ sở dữ liệu.
- + Test chu trình nghiệp vụ (Business Cycle Testing): Kiểm tra toàn bộ chu trình kinh doanh từ lúc khách hàng chọn sản phẩm đến khi thanh toán.

2. Test chức năng (Non-Function Testing):

- + Test hiệu suất (Performance Testing): Đánh giá khả năng xử lý của hệ thống dưới áp lực tải lớn và đồng thời.
- + Performance Profiling: Đo lường và phân tích hiệu suất của các thành phần để tối ưu hóa.
- + Load Testing: Kiểm tra khả năng xử lý dữ liệu đồng thời và đồng loạt.
- + Stress Testing: Kiểm tra sự ổn định và độ tin cậy dưới áp lực tải và điều kiện khắc nghiệt.
- + Volume Testing: Kiểm tra khả năng xử lý dữ liệu lớn và thời gian đáp ứng.

3. Test hồi quy (Regression Testing):

- + Kiểm tra hồi quy: Đảm bảo rằng mọi thay đổi không gây ảnh hưởng tiêu cực đến các chức năng khác.
- Mục tiêu chung:
- + Phát hiện lỗi và vấn đề: Mục tiêu chính là phát hiện càng nhiều lỗi và vấn đề càng tốt để cải thiện chất lượng và hiệu suất của phần mềm.
- + Nâng cao trải nghiệm người dùng: Những lỗi và cải tiến được áp dụng để tối ưu hóa trải nghiệm người dùng và tính năng của cửa hàng.

1.3 Tài liệu liên quan

ST T	Tên tài liệu	Nguồn	Ghi chú
1	Group 5-Test V1.0	Group 5	
2	Slides Software Testing	SoftwareTesting Class	
3	CSC Testplan Sample FPT Testplan Sample	CSC FPT	Thu thập từ Internet

1.4 Phạm vi test

Các giai đoạn kiểm tra được thực hiện : *(Khái quát định nghĩa từng mức độ trong các giai đoạn , thành viên cần phải nắm rõ để biết được quy trình kiểm tra website Twentyfive sẽ được diễn ra như thế nào)*

- **Unit Test – kiểm thử mức đơn vị**

Mục đích của Unit Test là bảo đảm thông tin được xử lý và xuất (khỏi Unit) là chính xác, trong mối tương quan với dữ liệu nhập và **chức năng của từng đơn vị thành phần nhỏ nhất của phần mềm**

Kiểm tra từng đơn vị thành phần nhỏ nhất của *website Twentyfive* gồm : **các hàm (Function), thủ tục (Procedure), lớp (Class), hoặc các phương thức (Method)**

Một kinh nghiệm đúc kết từ thực tiễn: thời gian tốn cho Unit Test sẽ được đền bù bằng việc tiết kiệm rất nhiều thời gian và chi phí cho việc kiểm thử và sửa lỗi ở các mức kiểm thử sau đó do đó chúng ta sẽ cố gắng thực hiện Unit Test thật tốt

Vì Unit Test thường thường do lập trình viên thực hiện trong giai đoạn viết code và xuyên suốt chu kỳ phát triển phần mềm. Do đó, Unit Test đòi hỏi kiểm thử viên có kiến thức về thiết kế và code của chương trình

Unit Test cũng đòi hỏi phải chuẩn bị trước các tình huống (test case) hoặc kịch bản (script), trong đó chỉ định rõ dữ liệu vào, các bước thực hiện và dữ liệu mong chờ sẽ xuất ra. Các test case và script này nên được giữ lại để tái sử dụng

- **Integration Test – kiểm thử tích hợp**

Integration test **kết hợp các thành phần của một ứng dụng và kiểm thử như một ứng dụng đã hoàn thành.**

Integration Test có 2 mục tiêu chính:

- Phát hiện **lỗi giao tiếp** xảy ra giữa các Unit
- **Tích hợp các Unit đơn lẻ** thành các hệ thống nhỏ (subsystem) và cuối cùng là nguyên hệ thống hoàn chỉnh (system) chuẩn bị cho kiểm thử ở mức hệ thống (System Test)

Integration Test chỉ nên thực hiện trên những Unit đã được kiểm tra cẩn thận trước đó bằng Unit Test, và tất cả các lỗi mức Unit đã được sửa chữa

Có 4 loại kiểm thử trong Integration Test:

- Kiểm thử cấu trúc (Structure Test): Tương tự White Box Test
- Kiểm thử chức năng (Functional Test): Tương tự Black Box Test
- Kiểm thử hiệu năng (Performance Test): kiểm thử việc vận hành của hệ thống
- Kiểm thử khả năng chịu tải (Stress Test): kiểm thử các giới hạn của hệ thống

- **System Test - kiểm thử mức hệ thống**

Mục đích System Test là kiểm thử **thiết kế và toàn bộ hệ thống (sau khi tích hợp) có thỏa mãn yêu cầu đặt ra hay không**

System Test .bắt đầu ngay sau Integration Test,trọng tâm là đánh giá về hoạt động, thao tác, sự tin cậy và các yêu cầu khác liên quan đến chất lượng của toàn hệ thống

Điểm khác nhau then chốt giữa Integration Test và System Test là System Test chú trọng các hành vi và lỗi trên toàn hệ thống, còn Integration Test chú trọng sự giao tiếp giữa các đơn thể hoặc đối tượng khi chúng làm việc cùng nhau

Bản thân System Test lại gồm nhiều loại kiểm thử khác nhau ,phổ biến nhất gồm:

- Kiểm thử chức năng (Functional Test)
- Kiểm thử khả năng vận hành (Performance Test)
- Kiểm thử khả năng chịu tải (Stress Test hay Load Test)
- Kiểm thử cấu hình (Configuration Test)
- Kiểm thử khả năng bảo mật (Security Test)
- Kiểm thử khả năng phục hồi (Recovery Test)

Nhìn từ quan điểm người dùng, các cấp độ kiểm thử trên rất quan trọng: bảo đảm hệ thống đủ khả năng làm việc trong môi trường thực

Lưu ý không nhất thiết phải thực hiện tất cả các loại kiểm thử nêu trên. Tùy yêu cầu và đặc trưng của từng hệ thống, tùy khả năng và thời gian cho phép của dự án, khi lập kế hoạch, người Quản lý dự án sẽ quyết định áp dụng những loại kiểm thử nào. Chính vì thế, ***đối với TwentyFive Fashion chúng ta sẽ kiểm thử những chức năng thiết yếu nhất đối với 1 Website : chức năng, chịu tải, vận hành và bảo mật***

- **Acceptance Test - kiểm thử chấp nhận sản phẩm**

Thông thường, sau giai đoạn System Test là Acceptance Test, được khách hàng thực hiện (hoặc ủy quyền cho một nhóm thứ ba thực hiện). Mục đích của Acceptance Test là để **chứng minh PM thỏa mãn tất cả yêu cầu của khách hàng và khách hàng chấp nhận sản phẩm** (và trả tiền thanh toán hợp đồng)

Acceptance Test có ý nghĩa hết sức quan trọng, mặc dù trong hầu hết mọi trường hợp, các phép kiểm thử của System Test và Acceptance Test gần như tương tự, nhưng bản chất và cách thức thực hiện lại rất khác biệt

Việc kiểm tra *TwentyFive Fashion* được thực hiện lần đầu tiên từ lúc hiện thực đến khi hoàn thành, chính vì thế **Group 5 sẽ phải test tất cả chức năng hiện có của trang web này**, bao gồm :

Bảng test chức năng:

Độ	Mã	Nội dung	Ghi chú
01	G5-TP-01	Đăng nhập	Test GUI
02	G5-TP-02	Đăng ký	Test GUI
03	G5-TP-03	Giỏ hàng	Test GUI
04	G5-TP-04	Sản phẩm	Test GUI
05	G5-TP-05	Danh mục sản phẩm	Test GUI
06	G5-TP-06	Hóa đơn	Test GUI
07	G5-TP-07	Lọc sản phẩm	Test GUI
08	G5-TP-08	Khuyến mãi	Test GUI
09	G5-TP-09	Quên mật khẩu	Test GUI
10	G5-TP-10	Cập nhật thông tin tài khoản	Test GUI
11	G5-TP-11	Đăng xuất	Test GUI

1.5 Ràng buộc

- Đảm bảo website chạy được trên Win 2003, XP phiên bản sau cùng sử dụng browser IE6, IE7, Google chrome, Microsoft Edge và FireFox
- Mọi thành viên trong nhóm đều phải đảm bảo hoàn thành lịch trình trong Testplan V1.0
- Mọi vấn đề phát sinh trong quá trình Test cần phải liên hệ với nhóm trưởng để tìm giải pháp và phải báo cáo thường xuyên những vấn đề này
- Xác định phạm vi kiểm thử bao gồm những chức năng cụ thể, kịch bản thử nghiệm, hoặc các yếu tố mà kiểm thử tập chung vào
- Đặt ra các hạn chế hoặc điều kiện đặc biệt mà kiểm thử cần tuân theo (không được tùy ý theo đổi mã nguồn khi test)

1.6 Liệt kê các mạo hiểm

STT	Mạo hiểm	Phương án khắc phục & phòng ngừa	Mức độ ảnh hưởng (MD)
1	Môi trường kiểm thử không phản ánh môi trường thực tế	Tạo một môi trường kiểm thử giống với môi trường thực tế nhất có thể và thực hiện kiểm thử tích hợp để đảm bảo sự tương thích	Trung bình
2	Thiếu nguồn tài nguyên cả phần cứng lẫn phần mềm phục vụ cho quá trình test	Kiểm tra hệ thống máy tính hiện có, lấy đề xuất từ thành viên về phần cứng và phần mềm máy tính đủ khả năng phục vụ cho quá trình test. Từ đó kiểm tra chi phí, lên kế hoạch và mua những thiết bị phần cứng và phần mềm tốt nhất	Cao
3	Trễ thời gian so với lịch trình	Dùng phần mềm DotProject để quản lý và phân công cho thành viên Thường xuyên giám sát, nhắc nhở và hỗ trợ các thành viên Nếu thấy cần thiết, có thể xin ban lãnh đạo cho tăng ca, thêm thời gian	Cao
4	Mâu thuẫn xảy ra giữa các thành viên	Nhóm trưởng phụ trách chính trong việc giải quyết Tạo các buổi gặp mặt riêng để tìm ra phương án tốt nhất giải quyết vấn đề Ghi chú lại tất cả trường hợp để tạo thành tài liệu quy cho công ty	Cao

5	Không đủ kiểm thử tương tự người dùng cuối	Sử dụng kiểm thử người dùng cuối(User Acceptance Testing -UAT) để đảm bảo sản phẩm đáp ứng được mong đợi của người dùng cuối	Trung bình
---	--	--	------------

2. CÁC YÊU CẦU CHO TEST

Độ ưu tiên	Mã	Nội dung	Mức độ công việc	Ghi chú
1	G5-TP-01	Đăng nhập	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
2	G5-TP-02	Đăng ký	Design TC 1 man - days, test 1 man- days	Test GUI
3	G5-TP-03	Giỏ hàng	Design TC 1 man - days, test 2 man- days	Test GUI
4	G5-TP-04	Sản phẩm	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
5	G5-TP-05	Danh mục sản phẩm	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
6	G5-TP-06	Hóa đơn	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
7	G5-TP-07	Lọc sản phẩm	Design TC 0,5 man - days, test 1 man- days	Test GUI
8	G5-TP-08	Khuyến mãi	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
9	G5-TP-09	Quên mật khẩu	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
10	G5-TP-10	Cập nhật thông tin tài khoản	Design TC 0,5 man - days, test 0,5 man- days	Test GUI
11	G5-TP-11	Đăng xuất	Design TC 0,5 man - days, test 0,5 man- days	Test GUI

3. CHIẾN LƯỢC TEST

- Chiến lược test giới thiệu **phương án tiếp cận để test các mục tiêu test**
- Những vấn đề chính trong chiến lược test là các kỹ thuật được áp dụng và điều kiện **để biết khi nào việc test được hoàn thành**
- Mô tả **các kiểu test** dùng trong dự án
- Có thể liệt kê với mỗi kiểu test tương ứng test cho chức năng nào
- Việc test có thể dừng khi nào.

3.1 Các kiểu test

- Các thành viên trong nhóm 5 cần nắm vững phần ghi chú dưới đây để có thể hiểu rõ cách trình bày, nội dung của các kiểu test đã được liệt kê.
- Đối với mỗi kiểu test phải giải thích **kỹ thuật, điều kiện hoàn thành và các vấn đề đặc biệt liên quan**:
 - **Kỹ thuật**: Kỹ thuật phải mô tả việc test được thực hiện như thế nào, bao gồm cả những gì sẽ được test, các hoạt động chính sẽ được thực hiện trong quá trình test và các phương pháp dùng để đánh giá kết quả.
 - **Điều kiện hoàn thành**: Điều kiện hoàn thành được phát biểu nhằm hai mục đích:
 - ❖ Xác định chất lượng sản phẩm được chấp nhận.
 - ❖ Xác định thời điểm mà các nỗ lực test được thực hiện thành công.
 - Một điều kiện hoàn thành được phát biểu rõ ràng phải bao gồm:
 - ❖ Phương pháp tính toán
 - ❖ Điều kiện hoặc mức độ thích ứng với phép đo
 - **Các vấn đề đặc biệt**: Phần này phải chỉ ra các ảnh hưởng hoặc phụ thuộc có thể tác động hoặc ảnh hưởng đến nguồn lực test mô tả trong chiến lược. Các ảnh hưởng có thể bao gồm: Nhân công (ví dụ sự sẵn sàng hoặc cần thiết của các nguồn lực khác test để hỗ trợ/tham gia trong test); các ràng buộc (ví dụ hạn chế về thiết bị hoặc sự sẵn sàng hoặc cần thiết/thiếu các thiết bị đặc biệt); các yêu cầu đặc biệt (ví dụ lịch test hoặc truy cập vào hệ thống).

Test chức năng (Function Testing)

- Mục đích của test chức năng là tập trung vào các yêu cầu test có thể được lưu vết trực tiếp trong các chức năng và quy tắc nghiệp vụ.
- Mục tiêu của kiểu test này là kiểm tra tính đúng đắn của các dữ liệu, quy trình và báo cáo cũng như việc thực hiện đúng những quy tắc nghiệp vụ.
- Kiểu test này dựa vào **kỹ thuật Black Box**, tức là kiểm tra ứng dụng và các xử lý nội tại bằng cách tương tác với ứng dụng thông qua giao diện người sử dụng và phân tích các kết quả hoặc đầu ra. Bảng sau liệt kê một số gợi ý đối với mỗi ứng dụng:

Mục đích test	Đảm bảo mục tiêu test đúng đắn của chức năng, bao gồm định hướng, dữ liệu đầu vào, xử lý và dữ liệu nhận được
Cách thực hiện	Thực hiện mỗi đơn vị, chu trình đơn vị hoặc chức năng, sử dụng dữ liệu hợp lệ và không hợp lệ để kiểm tra: <ul style="list-style-type: none"> - Kết quả mong đợi với dữ liệu hợp lệ. - Lỗi thích hợp hoặc thông báo hiển thị khi dữ liệu không hợp lệ. - Mỗi quy tắc nghiệp vụ đều được áp dụng đúng.
Điều kiện hoàn thành	<ul style="list-style-type: none"> - Toàn bộ kế hoạch test đã được thực hiện. - Toàn bộ các lỗi phát hiện ra đã được ghi nhận.
Các vấn đề đặc biệt	Xác định hoặc mô tả các vấn đề (nội bộ hoặc bên ngoài) ảnh hưởng đến việc test chức năng.

Test giao diện người sử dụng (User Interface Testing)

- Test giao diện người dùng (UI) kiểm tra các tương tác của người dùng với phần mềm.
- Mục tiêu của test UI là để đảm bảo rằng giao diện người dùng cung cấp cho người sử dụng cách truy cập và sử dụng thích hợp thông qua các chức năng trong mục tiêu test. Ngoài ra, test UI còn để đảm bảo rằng các đối tượng trong phạm vi chức năng UI giống như mong đợi và phù hợp với yêu cầu khách hàng.

Mục đích test	Kiểm tra: - Việc sử dụng thông qua mục tiêu test phản ánh đúng các chức năng và yêu cầu nghiệp vụ, bao gồm màn hình đến màn hình, trường đến trường và sử dụng các phương pháp truy cập (phím tabs, di chuột, tổ hợp phím). - Các đối tượng và thuộc tính màn hình như menus, size, position, state, và tập trung vào việc tương thích với chuẩn.
Cách thực hiện	Tạo ra và chỉnh sửa test cho mỗi màn hình để kiểm tra việc sử dụng đúng cách và tình trạng các đối tượng cho mỗi màn hình và đối tượng của ứng dụng.
Điều kiện hoàn thành	Mỗi màn hình được kiểm tra thành công đúng với phiên bản kiểm tra hoặc phạm vi chấp nhận được.
Các vấn đề đặc biệt	Không phải toàn bộ các thuộc tính của các đối tượng đều truy cập được.

Test dữ liệu và tích hợp dữ liệu (Data and Database Integrity Testing)

- Cơ sở dữ liệu và xử lý cơ sở dữ liệu phải được test như một hệ thống con trong dự án (hệ thống con này phải được test không cần thông qua giao diện người dùng để giao tiếp với dữ liệu)
- Nghiên cứu thêm về DBMS (SQL Server) để xác định các công cụ và kỹ thuật có thể có giúp hỗ trợ cho việc test

Mục đích test	Đảm bảo rằng các phương pháp truy cập và chức năng xử lý là đúng và không có sai lệch dữ liệu.
Cách thực hiện	<ul style="list-style-type: none"> - Thực hiện từng phương pháp truy cập và xử lý, thử từng trường hợp với dữ liệu hợp lệ và không hợp lệ hoặc các yêu cầu dữ liệu. - Kiểm tra cơ sở dữ liệu để đảm bảo rằng dữ liệu được lưu trữ như mong đợi, toàn bộ các sự kiện với cơ sở dữ liệu xảy ra đều đúng, hoặc xem xét các dữ liệu trả về để đảm bảo rằng đã nhận được dữ liệu đúng cho các lý do đúng.
Điều kiện hoàn thành	Tất cả các phương pháp truy cập và chức năng xử lý đều giống như thiết kế và không có sai lệch dữ liệu.
Các vấn đề đặc biệt	<ul style="list-style-type: none"> - Việc test có thể đòi hỏi phải môi trường phát triển SQL Server hoặc drivers để truy cập hoặc sửa dữ liệu trực tiếp trong cơ sở dữ liệu. - Các xử lý phải được thực hiện bằng tay. - Cơ sở dữ liệu có kích thước nhỏ hoặc tối thiểu (giới hạn số bản ghi) phải được dùng để làm rõ thêm các sự kiện không được phép chấp nhận.

Test chu trình nghiệp vụ (Business Cycle Testing)

- Test chu trình nghiệp vụ phải thực hiện các hoạt động trong dự án qua thời gian (phải xác định một chu kỳ, ví dụ một năm, và các giao dịch và hoạt động có thể xảy ra trong chu kỳ của năm đó phải được thực hiện).
- Việc này bao gồm cả các chu kỳ hàng ngày, hàng tuần hoặc hàng tháng và các sự kiện là ảnh hưởng bởi ngày tháng, ví dụ như ứng dụng ngân hàng.

Mục đích test	Đảm bảo mục đích của test là đúng đắn và các tiến trình chạy ngầm thực hiện đúng yêu cầu về mô hình nghiệp vụ và lịch trình
Cách thực hiện	<p>Việc test sẽ giả lập vài chu trình nghiệp vụ bằng cách thực hiện các công việc sau:</p> <ul style="list-style-type: none"> - Các test dùng cho việc test chức năng sẽ được sửa lại hoặc nâng cấp để tăng số lần mỗi chức năng được thực hiện để giả lập một số người dùng khác nhau trong chu kỳ đã định. - Toàn bộ các chức năng theo ngày tháng sẽ được thực hiện với dữ liệu hợp lệ và không hợp lệ hoặc chu kỳ thời gian. - Toàn bộ các chức năng xảy ra trong lịch trình chu kỳ sẽ được thực hiện vào thời gian thích hợp. - Việc test sẽ bao gồm cả dữ liệu hợp lệ và không hợp lệ để kiểm tra: <ul style="list-style-type: none"> + Kết quả xảy ra khi dữ liệu hợp lệ. + Lỗi tương tự hoặc cảnh báo hiển thị khi dữ liệu không hợp lệ. - Mỗi quy tắc nghiệp vụ đều được áp dụng.
Điều kiện hoàn thành	<ul style="list-style-type: none"> - Toàn bộ kế hoạch test đã được thực hiện. - Toàn bộ các lỗi phát hiện ra đều được ghi nhận
Các vấn đề đặc biệt	<ul style="list-style-type: none"> - Ngày và các sự kiện của hệ thống có thể đòi hỏi các hoạt động hỗ trợ đặc biệt. - Mô hình nghiệp vụ đòi hỏi xác định các yêu cầu và thủ tục test thích hợp.

Test hiệu suất (Performance testing)

Performance Profiling

- Performance profiling là một dạng test hiệu suất trong đó thời gian phản hồi, tỷ lệ giao dịch và các yêu cầu phụ thuộc thời gian khác được đo đạc và đánh giá.
- Mục đích của Performance Profiling là kiểm tra các yêu cầu về hiệu suất có đạt được hay không.
- Performance profiling là tiến hành và thực hiện để mô tả sơ lược và điều chỉnh các hành vi hiệu suất của mục tiêu test như một hàm của các điều kiện ví dụ workload hoặc cấu hình phần cứng.

Mục đích test	<p>Kiểm tra các biểu hiện về hiệu suất cho các giao dịch hoặc chức năng nghiệp vụ đã thiết kế theo những điều kiện sau:</p> <ul style="list-style-type: none"> - Workload bình thường đã biết trước (normal anticipated workload). - Workload xấu đã biết trước (anticipated worst case workload).
Cách thực hiện	<ul style="list-style-type: none"> - Sử dụng các thủ tục test cho test chức năng và chu trình nghiệp vụ. - Chỉnh sửa file dữ liệu để tăng số lượng các giao dịch hoặc scripts để tăng số tương tác xảy ra trong mỗi giao dịch. - Scripts phải được chạy trên một máy (trường hợp tốt nhất để đánh giá người dùng đơn lẻ, giao dịch đơn lẻ) và phải lặp lại trên nhiều máy trạm (ảo hoặc thực, xem các vấn đề đặc biệt dưới đây).
Điều kiện hoàn thành	<ul style="list-style-type: none"> - Giao dịch đơn lẻ hoặc người dùng đơn lẻ: Thực hiện thành công test script không có lỗi và trong phạm vi mong đợi hoặc thời gian phản hồi cho mỗi giao dịch. - Nhiều giao dịch hoặc nhiều người dùng: Thực hiện thành công test script không có lỗi và trong thời gian chấp nhận được.
Các vấn đề đặc biệt	<p>Việc test hiệu suất toàn diện bao gồm phải có một workload nền trên máy chủ.</p> <p>Có một số phương pháp để thực hiện, bao gồm:</p> <ul style="list-style-type: none"> - “Drive transactions” trực tiếp đến máy chủ, thường trong các form gọi SQL. - Sử dụng nhiều người dùng, mỗi người chạy một test script để load lên hệ thống. <p>Test hiệu suất phải được thực hiện trên máy chuyên dụng hoặc thời gian chuyên dùng. Điều đó cho phép việc tính toán được đầy đủ và chính xác.</p> <p>Cơ sở dữ liệu sử dụng để test hiệu suất phải có kích thước thực tế hoặc đo bằng nhau.</p>

Load Testing

- Load testing là một kiểu test hiệu suất mà mục tiêu là kiểm tra workload để tính toán và đánh giá hiệu suất và khả năng của mục đích test để tiếp tục thực hiện các chức năng thích hợp với các workload khác.
- Mục đích của load testing là xác định và đảm bảo các chức năng hệ thống thích hợp với nhiều nhất các workload.
- Ngoài ra load testing còn đánh giá các tính năng hiệu suất như thời gian phản hồi, tỉ lệ giao dịch và các vấn đề liên quan đến thời gian khác.

Mục tiêu test	Kiểm tra hiệu suất về thời gian cho các giao dịch hoặc tình huống nghiệp vụ đã thiết kế với nhiều điều kiện workload.
Cách thực hiện	<ul style="list-style-type: none"> - Sử dụng các test đã xây dựng cho test chức năng và chu trình nghiệp vụ. - Sửa lại file dữ liệu để tăng số lượng giao dịch hoặc test nhằm tăng thêm số lần thực hiện mỗi giao dịch.
Điều kiện hoàn thành	Nhiều giao dịch hoặc nhiều người dùng: Thực hiện thành công việc test không có lỗi và trong thời gian chấp nhận được.
Các vấn đề đặc biệt	<ul style="list-style-type: none"> - Load testing phải được thực hiện trên máy chuyên dụng hoặc vào những giờ chuyên biệt. Nó cho phép đo đạc đầy đủ và chính xác. - Cơ sở dữ liệu dùng cho load testing phải có kích thước thực tế hoặc đo bằng nhau.

Stress Testing

- Stress testing là một kiểu test hiệu suất được thực hiện để tìm ra các lỗi trong trường hợp thiếu tài nguyên hoặc cạnh tranh về tài nguyên (bộ nhớ hoặc dung lượng đĩa ít có thể làm xuất hiện lỗi trong mục đích test mà nó không xuất hiện dưới điều kiện bình thường)
- Các lỗi khác có thể là kết quả của việc cạnh tranh hoặc chia sẻ tài nguyên như khóa cơ sở dữ liệu hoặc băng thông mạng.
- Stress testing cũng được dùng để xác định workload tối đa mà mục đích test có thể điều khiển được

Mục đích test	Kiểm tra các chức năng của mục đích test là đúng đắn và không có lỗi với những điều kiện sau: <ul style="list-style-type: none"> - Có ít hoặc không có bộ nhớ phù hợp trên máy chủ (RAM và DASD).
----------------------	--

	<ul style="list-style-type: none"> - Số lượng máy trạm tối đa trong thực tế hoặc giả lập kết nối vào máy chủ - Nhiều người dùng thực hiện cùng một giao dịch với cùng dữ liệu hoặc account. - Độ lớn các giao dịch xấu hoặc hỗn hợp (xem phần Performance Testing ở trên). <p>Chú ý: Mục đích của Stress Testing có thể được phát biểu rõ và ghi ra các điều kiện mà hệ thống có thể lỗi, không thể tiếp tục thực hiện các chức năng một cách thích hợp.</p>
Cách thực hiện	<ul style="list-style-type: none"> - Sử dụng các test đã xây dựng để thực hiện Performance Profiling hoặc Load Testing. - Để test việc hạn chế tài nguyên, test phải chạy trên máy đơn lẻ và RAM và DASD trên máy chủ phải giảm đi hoặc hạn chế. - Để thực hiện các stress tests khác phải sử dụng nhiều người dùng cùng chạy một TC hoặc bổ sung các test để thực hiện độ lớn giao dịch số hoặc hỗn hợp.
Điều kiện hoàn thành	Toàn bộ kế hoạch test được thực hiện và các hạn chế của hệ thống được xác định thỏa mãn các điều kiện tối thiểu đã đặt ra hoặc chỉ sai trong trong hợp các điều kiện không nằm trong điều kiện đã xác định.
Các vấn đề đặc biệt	<ul style="list-style-type: none"> - Việc test Stressing mạng có thể đòi hỏi những công cụ để load mạng với nhiều thông báo hoặc gói dữ liệu. - DASD dùng cho hệ thống phải tạm thời giảm xuống để hạn chế khả năng chỗi trống cho tăng trưởng cơ sở dữ liệu. - Đồng bộ hóa các máy trạm đồng thời truy cập vào cùng một bản ghi hoặc các account dữ liệu

Volume Testing

- Mục tiêu của Volume Testing là để kiểm tra giới hạn của độ lớn của dữ liệu có thể làm phần mềm bị sai.
- Volume Testing cũng xác định load lớn nhất liên tục hoặc độ lớn mà mục đích test có thể điều khiển được trong chu kỳ đã cho (Ví dụ, nếu mục đích test là xử lý một tập các bản ghi để tạo báo cáo, Volume Test có thể dùng một cơ sở dữ liệu test lớn và kiểm tra xem phần mềm có chạy bình thường và cho ra báo cáo đúng không).

Mục đích test	<p>Kiểm tra xem mục tiêu test có thực hiện thành công các chức năng theo những điều kiện sau không:</p> <ul style="list-style-type: none"> - Số máy trạm lớn nhất kết nối (thực tế hoặc vật lý – có thể), hoặc giả lập, tất cả đều thực hiện cùng một chức năng nghiệp vụ trong một chu kỳ mở rộng. - Kích thước cơ sở dữ liệu lớn nhất có thể (thực tế hoặc đo được) và nhiều query hoặc giao dịch báo cáo được thực hiện đồng thời.
Cách thực hiện	<ul style="list-style-type: none"> - Sử dụng các test đã xây dựng cho Performance Profiling hoặc Load Testing. - Có thể dùng nhiều người dùng, chạy cùng một test hoặc bổ sung các test để thực hiện trường hợp giao dịch volume hoặc hỗn hợp xấu nhất (xem Stress Testing ở trên) trong một chu kỳ mở rộng. - Tạo ra cơ sở dữ liệu lớn nhất (thực tế, quy đổi, hoặc lọc các dữ liệu đại diện) và nhiều người dùng chạy các query và giao dịch báo cáo đồng thời trong một chu kỳ mở rộng.
Điều kiện hoàn thành	Toàn bộ kế hoạch test được thực hiện và các giới hạn của hệ thống được xác định là đạt tới hoặc xử lý mà không có lỗi.
Các vấn đề đặc biệt	Chu kỳ thời gian như thế nào là chấp nhận được cho điều kiện cơ sở dữ liệu lớn, như đã nói ở trên.

Test hồi quy (Regression Testing)

Test hồi quy là một hoạt động cần thiết để chỉ ra rằng việc thay đổi code không gây ra những ảnh hưởng bất lợi.

Mục đích test	Test hồi quy dùng để kiểm tra các phần được sửa chữa trong phần mềm, để đảm bảo rằng những sự thay đổi đó không gây ra lỗi trong những phần khác.
Cách thực hiện	<ul style="list-style-type: none"> - Tái sử dụng các TC từ những phần test trước để test các module đã được sửa chữa. - 80% các TC được chọn ngẫu nhiên. - Xây dựng một chương trình phân tích cơ sở hạ tầng. Chúng ta dựng một cơ sở hạ tầng có thể mở rộng được để thực hiện và đánh giá chương trình phân tích. Dựa vào

	kết quả phân tích chúng ta xác định phạm vi cần test hồi quy.
Điều kiện hoàn thành	<ul style="list-style-type: none"> - Toàn bộ các TC được thực hiện và đạt yêu cầu. - Toàn bộ các TC được chọn được thực hiện và đạt yêu cầu.

3.2 Giai đoạn test

Kiểu test	Giai đoạn test			
	Unit	Integration	System	Acceptance
Functional Tests (Function, User Interface)	X	X	X	X
Performance Tests (Performance profiles of individual components)	X	X		
Performance Tests (Load, Stress, Contention)			X	X
Reliability (Integrity, Structure)	X	X		

3.3 Các công cụ test

Mục đích	Công cụ	Tự xây dựng	Phiên bản
Phần mềm bán quần áo TwentyFive Fashion			

4. TÀI NGUYÊN

4.1 Nhân lực

Nguồn nhân lực		
Vai trò	Họ và tên	Trách nhiệm cụ thể
Test Manager	Hoàng Thu Trang	<ul style="list-style-type: none">- Quản lý toàn bộ dự án- Xác định phương hướng dự án- Có được tài nguyên phù hợp
Test Analyst	Vũ Việt Anh	<ul style="list-style-type: none">- Identifies and defines the specific tests to be conducted.- Responsibilities include:<ul style="list-style-type: none">● identify test ideas● define test details● determine test results● document change requests● evaluate product quality
Test Designer	Nguyễn Hoài Thu	<ul style="list-style-type: none">- Defines the technical approach to the implementation of the test effort.- Responsibilities include:<ul style="list-style-type: none">● define test approach● define test automation architecture● verify test techniques● define testability elements● structure test implementation

Nguồn nhân lực		
Vai trò	Họ và tên	Trách nhiệm cụ thể
Tester	Vũ Việt Anh Nguyễn Hồng Ánh Hoàng Thu Trang Nguyễn Hoài Thu Ngô Mạnh Cường Đào Việt Anh	<ul style="list-style-type: none"> • Xác định và mô tả các techniques/tools/automation architecture. • Xác minh và đánh giá Phương pháp tiếp cận (Test Approach). • Thực hiện các bài kiểm thử, Log results, Report defects. • Tester có thể là thành viên in-sourced hoặc outsourced, dựa trên ngân sách dự án.
Test System Administrator	Đào Việt Anh	<ul style="list-style-type: none"> - Ensures test environment and assets are managed and maintained. - Responsibilities include: <ul style="list-style-type: none"> • administer test management system • install and support access to, and recovery of, test environment configurations and test labs
Database Administrator, Database Manager	Ngô Mạnh Cường	<ul style="list-style-type: none"> - Ensures test data (database) environment and assets are managed and maintained. - Responsibilities include: <ul style="list-style-type: none"> • support the administration of test data and test beds (database).

Nguồn nhân lực		
Vai trò	Họ và tên	Trách nhiệm cụ thể
Designer	Nguyễn Hồng Ánh	<ul style="list-style-type: none"> - Identifies and defines the operations, attributes, and associations of the test classes. - Responsibilities include: <ul style="list-style-type: none"> • defines the test classes required to support testability requirements as defined by the test team
Implementer	Nguyễn Hoài Thu	<ul style="list-style-type: none"> - Implements and unit tests the test classes and test packages. - Responsibilities include: <ul style="list-style-type: none"> • creates the test components required to support testability requirements as defined by the designer

4.2 Hệ thống

Hệ thống phần cứng cần thiết

Tài nguyên hệ thống		
Resource	Quantity	Name and Type
Database Server	1	Database Server – My SQL
CPU	1	Intel Dual Core 3.0 GHz
RAM	2	2 G
HardWare	1	250GB
—Network or Subnet	1	Ip Public

Tài nguyên hệ thống		
Resource	Quantity	Name and Type
—Server Name		TBD
—Database Name	1	Ngô Mạnh Cường Data-MySQL
Client Test PCs	4	
CPU	1	Intel Dual Core 3.0 GHz
RAM	2	2 G
HardWare	1	250GB
—Include special configuration requirements		TBD
Test Repository	1	Test_Data – Bugzilla
—Network or Subnet	1	
—Server Name	1	
Test Development PCs	4	
CPU	1	Intel Dual Core 3.0 GHz
RAM	2	2 G
HardWare	1	250GB

Hệ thống phần mềm cần thiết

Tên phần mềm	Version	Type and Other Notes
Window XP	SP3	Hệ điều hành
Windows 2003	SP2	Hệ điều hành
Internet Explorer	7.0 , 8.0	Trình duyệt Web
Firefox	3.5,3.6	Trình duyệt Web

Tên phần mềm	Version	Type and Other Notes
Kaspersky Internet Security	9.0	Bảo vệ hệ thống và chống Virus

Công cụ hỗ trợ cho quá trình Test

Danh sách	Tên công cụ	Nhà cung cấp	Version
Quản lý Test	SilkCentral Test Manager	Borland	
Tìm lỗi	BugZilla	Open Sources	
Quản lý dự án	Microsoft Project	Microsofts	11.0
DBMS tools	Microsoft SQL Server	Microsofts	2008

Cấu hình môi trường Test

Tên	Description	Ghi chú
Người dùng trung bình	6 người : 1 nhóm trưởng và 5 thành viên kiểm tra	

CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES)

Nhiệm vụ	Người	Ngày bắt đầu	Ngày kết thúc
Lập kế hoạch test	Hoàng Thu Trang	15/01/2024	31/01/2024
Test: Đăng nhập	Vũ Việt Anh	15/01/2024	31/01/2024
Test: Giỏ hàng	Nguyễn Hồng Ánh	15/01/2024	31/01/2024
Test: Hóa đơn	Hoàng Thu Trang	15/01/2024	31/01/2024
Test: Cập nhật thông tin tài khoản	Nguyễn Hoài Thu	15/01/2024	31/01/2024
Test: Khuyến mãi	Ngô Mạnh Cường	15/01/2024	31/01/2024
Test: Quên mật khẩu	Đào Việt Anh	15/01/2024	31/01/2024

CÁC SẢN PHẨM

ST	Sản phẩm	Ngày bàn giao	Người bàn giao	Người nhận bàn giao
1	<Test cases>	31/01/2024	Hoàng Thu Trang Nguyễn Hoài Thu Nguyễn Hồng Ánh Ngô Mạnh Cường Vũ Việt Anh Đào Việt Anh	Hoàng Thu Trang
2	<Defect reports>	31/01/2024	Nguyễn Hồng Ánh Ngô Mạnh Cường	Hoàng Thu Trang
3	<Test Plan>	31/01/2024	Vũ Việt Anh Đào Việt Anh	Hoàng Thu Trang

NHIỆM VỤ KIỂM TRA PHẦN MỀM BÁN HÀNG TWENTYFIVE FASHION

Nhóm 5 được phân công thực hiện “test case” và test cho 6 chức năng chính cần kiểm tra trong 2 môi trường tự chọn. Nhóm 5 đã chọn 6 mục trong phần

1.2 Phạm vi test

Độ	Mã	Nội dung	Ghi chú
02	G5-TP-02	Đăng ký	Test GUI
03	G5-TP-03	Giỏ hàng	Test GUI
04	G5-TP-04	Sản phẩm	Test GUI
06	G5-TP-06	Hóa đơn	Test GUI
07	G5-TP-07	Lọc sản phẩm	Test GUI
10	G5-TP-10	Cập nhật thông tin tài khoản	Test GUI

Tất cả tester cần phải nắm rõ từng chức năng đã được liệt kê dưới đây để có thể hoàn thành tốt yêu cầu đã đề ra.

7.1 Đăng ký

- Muốn tham gia và thực hiện thao tác giao dịch trên *TwentyFive Fashion* cần phải đăng ký tài khoản, thông tin đăng ký là các thông tin cá nhân cơ bản của khách hàng (tài khoản, mật khẩu, họ tên, số điện thoại, địa chỉ nhà, địa chỉ email, v.v) sdt và email sẽ được gửi mã xác nhận nhằm đảm bảo thông tin chính xác. Sau khi đăng ký thì tài khoản và mật khẩu sẽ được lưu về cơ sở dữ liệu, muốn sử dụng các chức năng hay thực hiện giao dịch thì sẽ cần phải đăng nhập, khi đăng nhập hệ thống sẽ kiểm tra xem tài khoản đó liệu có hợp lệ, nếu có thì thông qua và ngược lại
- Khi một tài khoản có các hành vi không chuẩn mực (boom hàng nhiều lần, lời lẽ thô tục, v.v) thì quản lý có thông qua số điện thoại, mail để thực hiện trừng phạt tài khoản đó

Mục đích test	Chức năng đăng ký tài khoản mà đa số trang web nào cũng cần có
Cách thực hiện	Ấn vào nút ‘đăng ký’ dùng cả test case positive và negative -Đề ra kết quả mong đợi VD: Click vào nút mua hàng sẽ hiển thị thông báo ‘chưa đăng nhập’ -áp dụng nguyên tắc nghiệp vụ khi test chức năng này
Điều kiện hoàn thành	-Toàn bộ kế hoạch test đã được thực hiện -Toàn bộ các lỗi phát hiện ra đã được ghi nhận
Các vấn đề đặc biệt	Các vấn đề (trong hoặc ngoài phần mềm) ảnh hưởng tới việc test chức năng VD:Quỹ thời gian và nguồn lực gặp khó khăn khi đang thực hiện test plan này

7.2 Giỏ hàng

- Khách hàng có thể thêm sản phẩm vào giỏ hàng bằng cách nhấn vào nút "Thêm vào giỏ" trên trang sản phẩm. Sau khi thêm sản phẩm vào giỏ hàng, thông tin sản phẩm sẽ được hiển thị trong giỏ hàng, bao gồm: Mã sản phẩm, Tên sản phẩm, Hình ảnh sản phẩm, Giá sản phẩm, Số lượng sản phẩm
- Chỉnh sửa số lượng sản phẩm trong giỏ hàng: Khách hàng có thể chỉnh sửa số lượng sản phẩm trong giỏ hàng bằng cách nhấn vào nút "+" hoặc "-" ở bên cạnh sản phẩm. Khách hàng có thể xóa sản phẩm khỏi giỏ hàng bằng cách nhấn vào nút "Xóa" ở bên cạnh sản phẩm. Tổng giá trị giỏ hàng sẽ được hiển thị ở phía trên cùng của giỏ hàng. Tổng giá trị giỏ hàng bao gồm giá sản phẩm, phí vận chuyển và các ưu đãi (nếu có).

7.3 Sản phẩm

- Sản phẩm là chức năng cơ bản của hầu hết các trang web bán hàng. Nó sẽ chứa thông tin cơ bản về một sản phẩm đặc biệt, các thông tin có thể hiển thị bao gồm: ảnh mẫu, giá thành, mẫu mã, màu sắc, kích thước, số lượng đã bán, số lượng còn lại và đánh giá của người mua.. Ảnh mẫu và giá

thành sẽ luôn được hiển thị trong trang chủ, và có thể tìm kiếm sản phẩm dựa trên các thông tin trên

- Sản phẩm hết hàng sẽ được đánh dấu là ‘Hết hàng’ mà không ẩn đi để khách hàng biết là sản phẩm này có tại cửa hàng, nhưng mà đang hết, một sản phẩm sẽ chỉ bị xóa trong các trường hợp đặc thù ví dụ như ngừng sản xuất và đã bán hết sạch

7.4 Hóa đơn

- Hóa đơn sẽ hiển thị số tiền (Tiền sản phẩm+bảo hiểm thời trang+vận chuyển) mà khách hàng sẽ phải thanh toán cho sản phẩm đồng thời cũng sẽ được phát cho khách hàng sau khi mua hàng
- Mỗi hóa đơn sẽ gồm 2 phần: mã hóa đơn (được lưu trữ để kiểm tra khi có nhu cầu) và mã đơn hàng (chứa danh sách các sản phẩm có trong hóa đơn đó)

7.5 Lọc sản phẩm

- Lọc sản phẩm sẽ lọc ra 1 loạt sản phẩm có cùng 1 đặc điểm (ví dụ lọc ra loạt sản phẩm có mức giá từ 600.000-800.000vnd hoặc lọc các sản phẩm đến từ 1 hãng nhất định)

7.6 Cập nhật thông tin tài khoản

- Cập nhật thông tin tài khoản cho phép thay đổi các thông tin đã được tạo khi đăng ký tài khoản để nhằm hỗ trợ cho các khách hàng: thay số điện thoại, chuyển nhà, thay email, v.v