

Lập trình Windows – Books Store Management

TÀI LIỆU KIỂM THỬ



Lập trình Windows
Khoa Công nghệ thông tin
Đại học Khoa học tự nhiên TP HCM

MỤC LỤC

1	Bảng đánh giá thành viên	2
2	Kế hoạch kiểm thử (Test plan)	3
2.1	Mục tiêu kiểm thử	3
2.2	Phạm vi kiểm thử	3
2.2.1	Yêu cầu chức năng	3
2.2.2	Yêu cầu phi chức năng	4
2.3	Các loại kiểm thử	4
2.3.1	Kiểm thử chức năng (Functional Testing)	4
2.3.2	Kiểm thử giao diện người dùng (UI Testing)	5
2.3.3	Kiểm thử khả dụng (Usability Testing)	5
2.3.4	Môi trường kiểm thử	5
2.3.5	Kế hoạch thực hiện	5
2.3.6	Tiêu chí hoàn thành	5
2.3.7	Tài liệu kiểm thử	5
3	Test case	6
3.1	Danh sách các test case	6
3.2	Đặc tả các test case	7
3.2.1.	Test case 1	8
3.2.2.	Test case 2	8
3.2.3.	Test case 3	9
3.2.4.	Test case 4	9
3.2.5.	Test case 5	10
3.2.6.	Test case 6	10
3.2.7.	Test case 7	11
3.2.8.	Test case 8	11

3.2.9.	Test case 9	12
3.2.10.	Test case 10	12
3.2.11.	Test case 11	13
3.2.12.	Test case 12	13
3.2.13.	Test case 13	14
3.2.14.	Test case 14	15
3.2.15.	Test case 15	16
3.2.16.	Test case 16	17
3.2.17	Test case 17	17
3.2.18	Test case 18	18
3.2.19	Test case 19	18
3.2.20	Test case 20	19
3.2.20	Test case 21	20
3.2.21	Test case 22	21
3.2.22	Test case 23	21
3.2.24	Test case 24	22
3.2.25	Test case 25	22
3.2.26	Test case 26	23
3.2.27	Test case 27	23
3.2.28	Test case 28	24
3.2.29	Test case 29	24
3.2.30	Test case 30	25
3.2.31	Test case 31	25
3.2.32	Test case 32	26
3.2.33	Test case 33	27
3.2.34	Test case 34	27
4	Test report.....	28

TÀI LIỆU KIỂM THỬ

1

Bảng đánh giá thành viên

MSSV	Họ Tên	Công việc, trách nhiệm	% đóng góp (tối đa 100%)
22120214	Trương Thị Tú My	Tổng báo cáo, thực hiện test ID 1, 2, 3, 10, 11	100%
22120217	Hoàng Lê Nam	Check báo cáo, thực hiện test ID 7, 8, 9, 13, 14	100%
20120319	Phan Dương Linh	Check báo cáo, thực hiện test ID 4, 5, 6, 12	100%

2 Kế hoạch kiểm thử (Test plan)

2.1 Mục tiêu kiểm thử

Xác định các lỗi, đảm bảo rằng phần mềm đáp ứng đầy đủ các yêu cầu nghiệp vụ, và cải thiện trải nghiệm người dùng.

2.2 Phạm vi kiểm thử

Dựa trên các yêu cầu và thời gian thực hiện milestone 1, kế hoạch kiểm thử được đưa ra phù hợp với giai đoạn này. Dựa trên các tiêu chí công việc thực hiện: 11h làm việc: 10 giờ cho implement code + 1 giờ kiểm thử, viết báo cáo, team work.

2.2.1 Yêu cầu chức năng

- **Về UI (50%):**

- LoginPage: Hiển thị được các trường đăng nhập bao gồm **username**, **password**, **save password**, **login button**.
- DashboardPage: Hiển thị được **Llistview** về **SoldOut** và **BestSeller**.
- OrderPage, StockPage, CustomerPage, ClassificationPage: **1 Listview** hiển thị thông tin.
- AddStockPage, AddOrderPage, UpdateStockPage, UpdateOrderPage, AdminPage: Hiển thị được các **trường thông tin**.
- InvoicePage: **Định dạng 1 hoá đơn**.

- **Về Features (50%)**

- LoginPage: Thực hiện được tính năng **đăng nhập**, **lưu lại mật khẩu** vào **local setting**.
- OrderPage:
 - + Giải quyết tính năng **sort** theo các trường.
 - + 50% phần tính năng **search + filter**.

- + Thực hiện được chức năng [add, update, delete order](#).
- StockPage:
 - + Giải quyết tính năng [sort](#) theo các trường.
 - + 50% phần tính năng [search + filter](#).
 - + Thực hiện được chức năng [add, update, delete stock](#).
- AddStockPage: [Hoàn thành 80%](#) tính năng trang thêm thông tin cho sách đủ các trường.
- AddOrderPage: [Hoàn thành 80%](#) tính năng trang thêm thông tin cho order đủ các trường.
- UpdateStockPage: [Hoàn thành 80%](#) tính năng cập nhật lại thông tin cho sách.
- UpdateOrderPage: [Hoàn thành 80%](#) tính năng cập nhật lại thông tin đơn hàng cho khách.
- InvoicePage: [Xuất được hoá đơn định dạng PDF](#).

2.2.2 Yêu cầu phi chức năng

- Về Back-end + Database: [Hoàn thành 20%](#)
 - Kết nối và lấy dữ liệu từ PostgreSQL server
 - Kiểm tra các Exception nhập liệu từ user
- Về Architecture: [Đảm bảo mô hình MVVM](#)

2.3 Các loại kiểm thử

2.3.1 Kiểm thử chức năng (Functional Testing)

- **Kiểm thử đăng nhập:** Kiểm tra tính năng đăng nhập thành công và thất bại.
- **Kiểm thử quản lý sách:** Kiểm tra tính năng thêm, sửa, xóa và tìm kiếm sách.
- **Kiểm thử quản lý đơn hàng:** Kiểm tra tính năng thêm, sửa, xóa và tìm kiếm đơn hàng.
- **Kiểm thử hóa đơn:** Kiểm tra tính chính xác của hóa đơn sau khi hoàn tất đơn hàng.

- **Kiểm thử xuất hoá đơn:** Kiểm tra khả năng xuất hoá đơn ra hệ thống ngoài.

2.3.2 Kiểm thử giao diện người dùng (UI Testing)

- Kiểm tra bố cục và cách sắp xếp các thành phần trên giao diện.
- Đảm bảo tính nhất quán của font chữ, màu sắc và biểu tượng.
- Kiểm tra độ dễ sử dụng và trải nghiệm người dùng.

2.3.3 Kiểm thử khả dụng (Usability Testing)

- Đánh giá tính thân thiện và dễ sử dụng của hệ thống với người dùng cuối.
- Lắng nghe phản hồi từ người dùng để cải thiện trải nghiệm sử dụng.

2.3.4 Môi trường kiểm thử

- Hệ điều hành: [Windows](#)
- Thiết bị: [Laptop](#)

2.3.5 Kế hoạch thực hiện

- **Chuẩn bị tài liệu:** Xác định các trường hợp kiểm thử (test case) dựa trên yêu cầu nghiệp vụ.
- **Thực hiện kiểm thử:** Chạy các trường hợp kiểm thử theo từng loại kiểm thử.
- **Ghi lại kết quả:** Theo dõi lỗi và ghi lại kết quả chi tiết trong tài liệu.
- **Báo cáo:** Tổng hợp kết quả, đưa ra khuyến nghị cải thiện.

2.3.6 Tiêu chí hoàn thành

- Kiểm tra các lỗi cơ bản cho ứng dụng, các lỗi nghiêm trọng gây crash chương trình
- Tổng các trường hợp kiểm thử đảm bảo [95%](#) các tính năng.
- Nhận được kết quả [passed](#) trên [80%](#).

2.3.7 Tài liệu kiểm thử

- **Test Plan:** Kế hoạch chi tiết cho từng đợt kiểm thử.
- **Test Case:** Danh sách các trường hợp kiểm thử.
- **Test Report:** Báo cáo kết quả kiểm thử và thống kê lỗi.

3

Test case

3.1 Danh sách các test case

[Liệt kê tên các test case, các đối tượng test và diễn giải]

Test ID	Tên test case	Đối tượng test	Ý nghĩa
1	LoginSuccess	LoginPage	Kiểm tra việc đăng nhập thành công với thông tin tài khoản chính xác.
2	LoginFail	LoginPage	Kiểm tra phản hồi khi đăng nhập thất bại với thông tin tài khoản không chính xác.
3	SavePassword	LoginPage	Kiểm tra tính năng lưu mật khẩu cho lần đăng nhập sau.
4	AddStock	BookPopupControl + StockPage	Kiểm tra việc thêm sách mới vào kho hàng.
5	UpdateStock	BookPopupControl + StockPage	Kiểm tra khả năng cập nhật thông tin sách trong kho hàng.
6	DeleteStock	StockPage	Kiểm tra việc xóa sách khỏi kho hàng.
7	AddOrder	OrderDetailPage + OrderPage	Kiểm tra tính năng thêm đơn hàng mới vào hệ thống.

8	UpdateOrder	OrderDetailPage + OrderPage	Kiểm tra khả năng cập nhật thông tin đơn hàng.
9	DeleteOrder	OrderPage	Kiểm tra việc xóa đơn hàng khỏi hệ thống.
10	InvoiceInfomation	InvoicePage + OrderDetailPage	Kiểm tra tính chính xác của thông tin trên hóa đơn dựa trên chi tiết đơn hàng.
11	ExportInvoice	InvoicePage + PDF Platform	Kiểm tra chức năng xuất hóa đơn ra file PDF và tính toàn vẹn của dữ liệu.
12	UI_Layout_Require	Books Store Mangement app	Kiểm tra bố cục giao diện để đảm bảo tính thẩm mỹ và sự dễ sử dụng.
13	MVVM	Direction of App	Kiểm tra tính tuân thủ của kiến trúc MVVM để đảm bảo mã nguồn dễ bảo trì và mở rộng.
14	Data	Psqldao class	Kiểm tra khả năng kết nối và xử lý dữ liệu với cơ sở dữ liệu PostgreSQL.

3.2 Đặc tả các test case

Mỗi testcasse thực hiện tối thiểu 2 lần, những trường hợp nào không đủ bổ sung sau milestone 1

Define: Passed – 100%, Partial Passed – 50%, Failed < 50%

3.2.1. Test case 1

<i>Test case</i>	LoginSuccess
<i>Related Use case</i>	<i>Thao tác đăng nhập đúng username + password</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage</i>
<i>Input Data</i>	<i>Username: user1, Password: 123</i>
<i>Expected Output</i>	<i>Chuyển đến trang Dashboard</i>
<i>Test steps</i>	<i>1. Mở app</i> <i>2. Thực hiện nhập vào 2 Textbox username và password</i> <i>3. Click vào button login</i>
<i>Actual Output</i>	<i>Chuyển đến trang Dashboard</i>
<i>Result</i>	Passed

3.2.2. Test case 2

<i>Test case</i>	LoginSuccess
<i>Related Use case</i>	<i>Thao tác đăng nhập đúng username + password</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage</i>
<i>Input Data</i>	<i>Username: user2, Password: 456</i>
<i>Expected Output</i>	<i>Chuyển đến trang Dashboard</i>
<i>Test steps</i>	<i>1. Mở app</i> <i>2. Thực hiện nhập vào 2 Textbox username và password</i> <i>3. Click vào button login</i>
<i>Actual Output</i>	<i>Chuyển đến trang Dashboard</i>
<i>Result</i>	Passed

3.2.3. Test case 3

Test case	LoginSuccess
Related Use case	Thao tác đăng nhập đúng username + password
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage
Input Data	Username: user3, Password: 789
Expected Output	Chuyển đến trang Dashboard
Test steps	1. Mở app 2. Thực hiện nhập vào 2 Textbox username và password 3. Click vào button login
Actual Output	Chuyển đến trang Dashboard
Result	<i>Passed</i>

3.2.4. Test case 4

Test case	LoginFail
Related Use case	Thao tác đăng nhập sai username + password
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage
Input Data	Username: abc, Password: 456
Expected Output	Thông báo đăng nhập không thành công, không xuất hiện lỗi crash hoặc exit process
Test steps	1. Mở app 2. Thực hiện nhập vào 2 Textbox username và password 3. Click vào button login
Actual Output	Không có phản ứng
Result	<i>Partial passed</i>

3.2.5. Test case 5

<i>Test case</i>	LoginFail
<i>Related Use case</i>	<i>Thao tác đăng nhập sai username + password</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage</i>
<i>Input Data</i>	<i>Username: [empty string], Password: 456</i>
<i>Expected Output</i>	<i>Thông báo đăng nhập không thành công, không xuất hiện lỗi crash hoặc exit process</i>
<i>Test steps</i>	<i>1. Mở app</i> <i>2. Thực hiện nhập vào 2 Textbox username và password</i> <i>3. Click vào button login</i>
<i>Actual Output</i>	<i>Không có phản ứng</i>
<i>Result</i>	<i>Partial passed</i>

3.2.6. Test case 6

<i>Test case</i>	LoginFail
<i>Related Use case</i>	<i>Thao tác đăng nhập sai username + password</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage</i>
<i>Input Data</i>	<i>Username: [empty string], Password: [empty string]</i>
<i>Expected Output</i>	<i>Thông báo đăng nhập không thành công, không xuất hiện lỗi crash hoặc exit process</i>
<i>Test steps</i>	<i>1. Mở app</i> <i>2. Thực hiện nhập vào 2 Textbox username và password</i> <i>3. Click vào button login</i>
<i>Actual Output</i>	<i>Không có phản ứng</i>
<i>Result</i>	<i>Partial passed</i>

3.2.7. Test case 7

Test case	SavePassword
Related Use case	Check vào save password
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage
Input Data	Username: user1, Password: 123, Checkbox → checked
Expected Output	Thông tin đăng nhập được tự động load ở lần mở app tiếp theo là của user1
Test steps	1. Mở app 2. Thực hiện nhập vào 2 Textbox username và password 3. Check vào save password 4. Click vào button login
Actual Output	Load được dữ liệu user1 ở lần tiếp theo mở app
Result	Passed

3.2.8. Test case 8

Test case	SavePassword
Related Use case	Check vào save password
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage
Input Data	Username: user2, Password: 456, Checkbox → unchecked
Expected Output	Thông tin đăng nhập được tự động load ở lần mở app tiếp theo là của user1
Test steps	1. Mở app 2. Thực hiện nhập vào 2 Textbox username và password 3. Check vào save password 4. Click vào button login
Actual Output	Load được dữ liệu user1 ở lần tiếp theo mở app
Result	Passed

3.2.9. Test case 9

Test case	SavePassword
Related Use case	Check vào save password
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình LoginPage
Input Data	Username: user3, Password: 789, Checkbox → checked
Expected Output	Thông tin đăng nhập được tự động load ở lần mở app tiếp theo là của user3
Test steps	<ol style="list-style-type: none"> 1. Mở app 2. Thực hiện nhập vào 2 Textbox username và password 3. Check vào save password 4. Click vào button login
Actual Output	Load được dữ liệu user3 ở lần tiếp theo mở app
Result	Passed

3.2.10. Test case 10

Test case	AddStock
Related Use case	Thêm một sản phẩm kho.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl
Input Data	Book name: Walk In The Dark, Author: George Martin Publisher: The Trust, ISBN: 9780003147652, Year: 2002, Classify: Mystery, Selling Price: 12, Purchase Price: 11, Quantities: 10
Expected Output	Sách được thêm vào Stock với đúng các giá trị đã nhập.
Test steps	<ol style="list-style-type: none"> 1. Điều hướng trên trang Stock 2. Nhấn vào nút Add Book (→ Hiển thị Popup) 3. Nhập đầy đủ các trường với Input ở trên

	4. Nhấn nút Save
Actual Output	Quay lại trang Stock, sách vừa nhập đã thêm và hiển thị lên bảng Stock
Result	<i>Passed</i>

3.2.11. Test case 11

Test case	AddStock
Related Use case	Thêm một sản phẩm kho.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl
Input Data	Book name: "", Author: "", Publisher: "", ISBN: "", Year: "", Classify: "", SellingPrice: "", PurchasePrice: "", Quantities: ""
Expected Output	Hiển thị các message lỗi require phía dưới các trường.
Test steps	<ol style="list-style-type: none"> 1. Điều hướng trên trang Stock 2. Nhấn vào nút Add Book (→ Hiển thị Popup) 3. Nhập đầy đủ các trường với Input ở trên 4. Nhấn nút Save
Actual Output	Không chuyển về trang Stock và hiển thị các message lỗi require phía dưới các trường trống
Result	<i>Passed</i>

3.2.12. Test case 12

Test case	AddStock
Related Use case	Thêm một sản phẩm kho.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl
Input Data	Book name: Walk In The Dark, Author: George Martin Publisher: The Trust, ISBN: 123456789, Year: 2002, Classify: Mystery, Selling Price: -10, Purchase Price: 11,

	<i>Quantities: 10</i>
<i>Expected Output</i>	<i>Hiển thị các message lỗi yêu cầu trường ISBN phải đúng 13 số, message lỗi phải lớn hơn 0 đối với Selling Price</i>
<i>Test steps</i>	<ol style="list-style-type: none"> 1. Điều hướng trên trang Stock 2. Nhấn vào nút Add Book (→ Hiển thị Popup) 3. Nhập đầy đủ các trường với Input ở trên 4. Nhấn nút Save
<i>Actual Output</i>	<i>Không chuyển về trang Stock và hiển thị các message lỗi ISBN phải đúng 13 số, Selling Price phải lớn hơn 0</i>
<i>Result</i>	<i>Passed</i>

3.2.13. Test case 13

<i>Test case</i>	UpdateStock
<i>Related Use case</i>	<i>Cập nhật thông tin cho 1 sản phẩm.</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl</i>
<i>Input Data</i>	<p><i>Thông tin ban đầu:</i></p> <p><i>Book name: Walk In The Dark, Author: George Martin</i> <i>Publisher: The Trust, ISBN: 9780003147652, Year: 2002,</i> <i>Classify: Mystery, Selling Price: 12, Purchase Price: 11,</i> <i>Quantities: 10</i></p> <p><i>Sửa:</i></p> <p><i>Book name: Walk In The Dark, Author: George Martin</i> <i>Publisher: The Trust, ISBN: 9780003147652, Year: 2002,</i> <i>Classify: Mystery, Selling Price: 12, Purchase Price: 11,</i> <i>Quantities: 14</i></p>
<i>Expected Output</i>	<i>Quay về trang Stock và cập nhật thuộc tính Quantities của sản phẩm vừa sửa từ 10 → 14.</i>
<i>Test steps</i>	<ol style="list-style-type: none"> 1. Điều hướng trên trang Stock 2. Nhấn vào nút biểu tượng cây bút trong row của sản

	<p>phẩm muốn sửa(→ Hiển thị Popup)</p> <p>3. Sửa các thông tin muốn sửa</p> <p>4. Nhấn nút Edit</p>
Actual Output	Chuyển trang về trang Stock, cập nhật thuộc tính Quantities của sản phẩm vừa sửa từ 10 → 14.
Result	Passed

3.2.14. Test case 14

Test case	UpdateStock
Related Use case	Cập nhật thông tin cho 1 sản phẩm.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl
Input Data	<p>Thông tin ban đầu:</p> <p>Book name: Walk In The Dark, Author: George Martin Publisher: The Trust, ISBN: 9780003147652, Year: 2002, Classify: Mystery, Selling Price: 12, Purchase Price: 11, Quantities: 10</p> <p>Sửa:</p> <p>Book name:"", Author: George Martin Publisher: The Trust, ISBN: 9780003147652, Year: 2002, Classify: Mystery, Selling Price: "", Purchase Price: -1, Quantities: 10</p>
Expected Output	Hiển thị message lỗi Title is required, Selling price is required, Purchase price must be greater than 0.
Test steps	<p>1. Điều hướng trên trang Stock</p> <p>2. Nhấn vào nút biểu tượng cây bút trong row của sản phẩm muốn sửa(→ Hiển thị Popup)</p> <p>3. Sửa các thông tin muốn sửa</p> <p>4. Nhấn nút Edit</p>
Actual Output	Không chuyển trang về trang Stock, hiển thị message lỗi

	<i>Title is required, Selling price is required, Purchase price must be greater than 0.</i>
<i>Result</i>	<i>Passed</i>

3.2.15. Test case 15

<i>Test case</i>	UpdateStock
<i>Related Use case</i>	<i>Cập nhật thông tin cho 1 sản phẩm.</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình BookPopupControl</i>
<i>Input Data</i>	<p><i>Thông tin ban đầu:</i></p> <p><i>Book name: Walk In The Dark, Author: George Martin</i> <i>Publisher: The Trust, ISBN: 9780003147652, Year: 2002,</i> <i>Classify: Mystery, Selling Price: 12, Purchase Price: 11,</i> <i>Quantities: 10</i></p> <p><i>Sửa:</i></p> <p><i>Book name: Walk In The Dark, Author, Author: George Martin, Publisher: The Trust, ISBN: 9780003147652, Year: 2002, Classify: Mystery, Selling Price: 10, Purchase Price: 9, Quantities: 0</i></p>
<i>Expected Output</i>	<i>Hiển thị message lỗi Quantities must be greater than 1.</i>
<i>Test steps</i>	<ol style="list-style-type: none"> <i>1. Điều hướng trên trang Stock</i> <i>2. Nhấn vào nút biểu tượng cây bút trong row của sản phẩm muốn sửa(→ Hiển thị Popup)</i> <i>3. Sửa các thông tin muốn sửa</i> <i>4. Nhấn nút Edit</i>
<i>Actual Output</i>	<i>Chuyển về trang Stock, cập nhật là Quantities của sản phẩm vừa nhập là 0</i>
<i>Result</i>	<i>Failed</i>

3.2.16. Test case 16

Test case	DeleteStock
Related Use case	Xóa đi một sản phẩm tra khỏi Stock
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình StockPage
Input Data	
Expected Output	Xóa sản phẩm vừa xóa ra khỏi danh sách Stock
Test steps	1. Điều hướng trên trang Stock 2. Nhấn vào nút biểu tượng thùng rác trong row của sản phẩm muốn xóa.
Actual Output	Xóa đi sản phẩm vừa chọn xóa và cập nhật lại bảng Stock
Result	Passed

3.2.17 Test case 17

Test case	AddOrder
Related Use case	Thêm đơn hàng mới với tên khách hàng trống
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: <blank> Purchase Date: 11/2/2024 – Saturday 12:03 (<default>) Coupon: <blank> Books: Animal Farm Payment Method: Momo Delivery: <check>
Expected Output	Không tạo được đơn hàng cụ thể sẽ thông báo chưa nhập tên khách hàng.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn +Add Order trên. 2. Nhập Input như trên.

	3. Nhấn Create Order
Actual Output	Không tạo được đơn hàng cụ thể sẽ thông báo chưa nhập tên khách hàng.
Result	Passed

3.2.18 Test case 18

Test case	AddOrder
Related Use case	Thêm đơn hàng vào danh sách với tên sản phẩm trống
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: Ricky LTC Purchase Date: 11/2/2024 – Saturday 12:08 (<default>) Coupon: <blank> Books: <blank> Payment Method: Momo Delivery: <check>
Expected Output	Không tạo được đơn hàng cụ thể sẽ thông báo chưa nhập tên sản phẩm.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn +Add Order 2. Nhập Input như trên. 3. Nhấn Create Order
Actual Output	Không tạo được đơn hàng cụ thể sẽ thông báo chưa nhập tên sản phẩm.
Result	Passed

3.2.19 Test case 19

Test case	AddOrder
Related Use case	Thêm đơn hàng vào danh sách với lựa chọn thanh toán cash

Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: Ricky LTC Purchase Date: 11/2/2024 – Saturday 12:15 (<default>) Coupon: 10% Books: Animal Farm Payment Method: Cash Delivery: <check>
Expected Output	Tạo được đơn hàng và chuyển trang Invoice.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn +Add Order trên. 2. Nhập Input như trên. 3. Nhấn Create Order
Actual Output	Tạo được đơn hàng và chuyển trang Invoice.
Result	Passed

3.2.20 Test case 20


Test case	AddOrder
Related Use case	Thêm đơn hàng vào danh sách và chọn ngày đặt hàng
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: <blank> Purchase Date: 11/11/2024 – Saturday 12:03 Coupon: <blank> Books: Animal Farm Payment Method: Momo Delivery: <check>
Expected Output	Tạo được đơn hàng và chuyển trang Invoice.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn +Add Order trên. 2. Nhập Input như trên.

	3. Nhấn Create Order
Actual Output	Tạo được đơn hàng và chuyển trang Invoice.
Result	Passed


3.2.20 Test case 21

Test case	AddOrder
Related Use case	Thêm đơn hàng vào danh sách với số lượng đơn hàng lớn hơn số lượng trang có thể hiển thị
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: <blank> Purchase Date: 11/11/2024 – Saturday 12:03 Coupon: <blank> Books: Animal Farm Payment Method: Momo Delivery: <check>
Expected Output	Tạo được đơn hàng và các đơn hàng được tạo quá giới hạn trang sẽ được đưa vào trang mới + Tổng số trang tăng lên.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage và trang Invoice được thông qua. Step: 1. Nhấn +Add Order 2. Nhập Input như trên. 3. Nhấn Create Order 4. Thực hiện lại các thao tác khi đủ số lượng cần có.
Actual Output	Tạo được đơn hàng và các đơn hàng được tạo quá giới hạn trang sẽ được đưa vào trang mới + Tổng số trang tăng lên.
Result	Passed

3.2.21 Test case 22



Test case	UpdateOrder
Related Use case	Kiểm tra danh sách cập nhật khi cập nhật tên khách hàng của đơn hàng.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: "Charlie Brown" => "Jonh Brown"
Expected Output	Đơn hàng được cập nhật và trật tự trang được thiết lập lại.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi Đơn hàng cần sửa. 2. Tại ô Customer Name nhập lại input như trên.
Actual Output	Đơn hàng được cập nhật và trật tự trang được thiết lập lại.
Result	Passed

3.2.22 Test case 23

Test case	UpdateOrder
Related Use case	Kiểm tra danh sách cập nhật khi xóa hoàn toàn tên khách hàng của đơn hàng.
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	Customer Name: "Charlie Brown" => ""
Expected Output	Không cập nhật được đơn hàng cụ thể sẽ thông báo tên khách hàng trống.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi Đơn hàng cần sửa. 2. Tại ô Customer Name xóa tên khách hàng.
Actual Output	Không cập nhật được đơn hàng cụ thể sẽ thông báo tên


	<i>khách hàng trống.</i>
Result	<i>Passed</i>

3.2.24 Test case 24

Test case	UpdateOrder
Related Use case	<i>Kiểm tra danh sách cập nhật khi xóa sản phẩm của đơn hàng.</i>
Context	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage</i>
Input Data	<i><blank></i>
Expected Output	<i>Cập nhật nếu đơn hàng còn sản phẩm. Không cập nhật được đơn hàng nếu đơn hàng không còn sản phẩm nào, thông báo sản phẩm trống.</i>
Test steps	<i>Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi Đơn hàng cần sửa. 2. Tại cửa sổ sản phẩm nhấn vào biểu tượng  bên dưới từng sản phẩm để xóa .</i>
Actual Output	<i>Cập nhật nếu đơn hàng còn sản phẩm. Không cập nhật được đơn hàng nếu đơn hàng không còn sản phẩm nào, thông báo sản phẩm trống.</i>
Result	<i>Passed</i>

3.2.25 Test case 25


Test case	DeleteOrder
Related Use case	<i>Kiểm tra danh sách cập nhật khi xóa đơn hàng cuối trang</i>
Context	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage</i>
Input Data	<i><blank></i>
Expected Output	<i>Đơn hàng bị xóa và trật tự trang được thiết lập lại.</i>

Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi Đơn hàng cần xóa.
Actual Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại.
Result	Passed

3.2.26 Test case 26

Test case	DeleteOrder
Related Use case	Kiểm tra danh sách cập nhật khi xóa đơn hàng đầu trang
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	<blank>
Expected Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi đơn hàng cần xóa.
Actual Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại.
Result	Passed

3.2.27 Test case 27


Test case	DeleteOrder
Related Use case	Kiểm tra danh sách cập nhật khi xóa đơn hàng giữa trang
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	<blank>
Expected Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại.
Test steps	Điều kiện: Đăng nhập được app và vào được OrderPage Step: 1. Nhấn biểu tượng  trên mỗi đơn hàng cần xóa.
Actual Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại.

Result	<i>Passed</i>
--------	---------------

3.2.28 Test case 28

Test case	InvoiceInformation
Related Use case	Thực hiện tạo invoice dựa trên order
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderDetailPage + InvoicePage
Input Data	Nhập đủ và chính xác các trường thông tin của OrderDetailPage
Expected Output	Các trường thông tin vừa nhập hiển thị đúng trong InvoicePage
Test steps	<ol style="list-style-type: none"> 1. Nhập thông tin khi thêm Order 2. Nhấn nút tạo bill 3. Xác thực độ chính xác của các trường thông tin
Actual Output	Hiển thị trùng khớp
Result	<i>Passed</i>

3.2.29 Test case 29

Test case	DeleteOrder
Related Use case	Kiểm tra danh sách cập nhật khi xóa hết đơn hàng của 1 trang
Context	Thực hiện trên phần mềm Books Store Management + Trên màn hình OrderPage
Input Data	<blank>
Expected Output	Đơn hàng bị xóa và trật tự trang được thiết lập lại + Tổng số trang được tính toán lại.
Test steps	<p>Điều kiện: Đăng nhập được app và vào được OrderPage</p> <p>Step: 1. Nhấn biểu tượng  trên mỗi Order cần xóa.</p> <p>2. Lặp lại bước 1 khi đủ số lượng cần xóa.</p>

<i>Actual Output</i>	<i>Đơn hàng bị xóa và trật tự trang được thiết lập lại + Tổng số trang được tính toán lại.</i>
<i>Result</i>	<i>Passed</i>

3.2.30 Test case 30

<i>Test case</i>	ExportInvoice
<i>Related Use case</i>	<i>Thực hiện in InvoicePage</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management + Trên màn hình InvoicePage</i>
<i>Input Data</i>	<i>Không có data, check thao tác</i>
<i>Expected Output</i>	<i>Tạo được bản PDF chứa thông tin Invoice</i>
<i>Test steps</i>	<i>1. Nhấn nút export tại InvoicePage 2. Wait và check bản PDF được tạo tại hệ thống ngoài</i>
<i>Actual Output</i>	<i>Xuất hiện một trang PDF mở bằng Foxit Reader</i>
<i>Result</i>	<i>Passed</i>

3.2.31 Test case 31

<i>Test case</i>	UI_Layout_Require
<i>Related Use case</i>	<i>Đáp ứng nhu cầu thị giác của user</i>
<i>Context</i>	<i>Thực hiện trên phần mềm Books Store Management</i>
<i>Input Data</i>	<i>Không có data, thực hiện các thao tác check UI</i>
<i>Expected Output</i>	<i>Check khả năng hoạt động của Navigation, check consistent các Listview, Color chủ đạo là tông màu tối, cách bố trí Layout, các hiển thị Controller</i>
<i>Test steps</i>	<i>1. Mở app, thực hiện đăng nhập thành công 2. Thực hiện từng tag trên Navigation quan sát chuyển hướng và xem yêu cầu bố cục, màu sắc từng Page 3. Thực hiện nhấn các nút kiểm tra flow</i>

<i>Actual Output</i>	<i>Các trang đáp ứng Layout chính của design, flow đúng Color các Page: Classification, Customer, Admin chưa đáp ứng đúng màu sắc chủ đạo Một số tính năng nút bấm chưa hoàn thiện</i>
<i>Result</i>	<i>Partial Passed</i>

3.2.32 Test case 32

<i>Test case</i>	MVVM
<i>Related Use case</i>	<i>Yêu cầu phi chức năng trong việc tổ chức phần mềm</i>
<i>Context</i>	<i>Check trong thư mục tổ chức app</i>
<i>Input Data</i>	<i>Không có data, thực hiện kiểm tra cách tổ chức thư mục</i>
<i>Expected Output</i>	<i>Các file được tổ chức theo mô hình MVVM trong đó có ba thư mục chính Models, ViewModels, Views Models: Chứa thông tin đối tượng ViewModels: Chứa các logic xử lý giữa Models à Views Views: Chứa các màn hình UI, phần code-behind không chứa quá nhiều các logic xử lý</i>
<i>Test steps</i>	<i>1. Truy cập vào thư mục chứa ứng dụng 2. Check 3 thư mục chính Models, ViewModels, Views 3. Check ViewModel của các View tương ứng → Xem phần code-behind</i>
<i>Actual Output</i>	<i>Tổ chức theo MVVM Tổ chức thư mục theo Views, ViewModels, Models rõ ràng Có sử dụng lớp RelayCommand để thực hiện mô hình MVVM Chưa thực sự chuyển giao mọi Logic cho lớp ViewModel LoginPage thực hiện đúng yêu cầu, các Page còn lại để logic nhiều ở code-behind</i>
<i>Result</i>	<i>Partial Passed</i>

3.2.33 Test case 33

<i>Test case</i>	<i>Data</i>
<i>Related Use case</i>	<i>Yêu cầu phi chức năng trong việc lấy dữ liệu từ Postgres SQL, start database server</i>
<i>Context</i>	<i>Các ListView hiển thị trên app</i>
<i>Input Data</i>	<i>Không có data, thực hiện kiểm tra việc load data từ database</i>
<i>Expected Output</i>	<i>Các ListView load được database tạo trong Postgres Port: 5432, Password: 1234, Name: mybookstore</i>
<i>Test steps</i>	<i>1. Start database server, mở app 2. Check thông tin hiển thị các ListView OrderPage + BookPage</i>
<i>Actual Output</i>	<i>Hiển thị đúng các trường thông tin trong database</i>
<i>Result</i>	<i>Passed</i>

3.2.34 Test case 34

<i>Test case</i>	<i>Data</i>
<i>Related Use case</i>	<i>Yêu cầu phi chức năng trong việc lấy dữ liệu từ Postgres SQL, không start dataserver</i>
<i>Context</i>	<i>Các ListView hiển thị trên app</i>
<i>Input Data</i>	<i>Không có data, thực hiện kiểm tra việc load data từ database</i>
<i>Expected Output</i>	<i>Thông báo yêu cầu start database</i>
<i>Test steps</i>	<i>1. Disconnect database server, mở app 2. Check thông báo mở database server</i>
<i>Actual Output</i>	<i>Error</i>
<i>Result</i>	<i>Failed</i>

4

Test report

Test ID	Tên test case	Test case liên quan	Kết quả failed	Lỗi nghiêm trọng	Người thực hiện
1	LoginSuccess	1, 2, 3	No	No	Trương Thị Tú My
2	LoginFail	4, 5, 6	4, 5, 6	Yes → Cần tiến hành điều chỉnh → Vì trải nghiệm người dùng không tốt khi không nhận biết được trình trạng lỗi đăng nhập.	Trương Thị Tú My
3	SavePassword	7, 8, 9	No	No	Trương Thị Tú My
4	AddStock	10, 11, 12	No	No	Hoàng Lê Nam
5	UpdateStock	13, 14, 15	15		Hoàng Lê Nam
6	DeleteStock	16	No	No	Hoàng Lê Nam
7	AddOrder	17, 18, 19, 20, 21	No	No	Phan Dương Linh
8	UpdateOrder	22, 23, 24	No	No	Phan Dương Linh

9	DeleteOrder	25, 26, 27	No	No	Phan Dương Linh
10	InvoiceInfomation	29	No	No	Trương Thị Tú My
11	ExportInvoice	30	No	No	
12	UI_Layout_Require	31	31	No → Cần điều chỉnh Layout, màu sắc phù hợp với bản design trước đó ở các page: Classification, Customer, Admin.	Phan Dương Linh
13	MVVM	32	32	Yes → Cần hiểu rõ mô hình MVVM để tổ chức đúng, nắm về cách hoạt động của lớp ICommand, Command, RelayCommand để tổ chức Logic giữa View và Model, mọi xử lý chuyển qua cho ViewModel, không để ở code-behind.	Hoàng Lê Nam
14	Data	33, 34	34	Yes → Chương trình crash khi khi không start database, xử lý thông báo cho người dùng.	Hoàng Lê Nam

→ Các trường hợp Error sẽ hoàn thành hiệu chỉnh milestone 2.