

Lab01 : Xây dựng Test Document theo quy trình thực tế

- Họ và tên: Nguyễn Văn Phúc Huy
- MSSV: 23110163

1 – Software Requirement Specification (SRS)

Module: Shopping Cart & Checkout Flow

System: E-commerce Web Application – Shopping Cart & Payment Processing

Version: 1.0

Date: 10/11/2025

Prepared by: Nguyễn Văn Phúc Huy

1. Giới thiệu

1.1. Mục tiêu tài liệu

Tài liệu này mô tả các yêu cầu phần mềm cho quy trình thanh toán giỏ hàng của hệ thống web thương mại điện tử.

Mục đích là đảm bảo người dùng có thể thêm sản phẩm vào giỏ hàng, xem lại sản phẩm, thực hiện thanh toán một cách an toàn và đúng đắn, đồng thời định nghĩa các luồng hoạt động, dữ liệu, và phản hồi tương ứng của hệ thống.

1.2. Phạm vi áp dụng

Module này áp dụng cho toàn bộ quy trình thanh toán giỏ hàng của hệ thống web thương mại điện tử.

Bao gồm:

- Client (Browser) - Giao diện người dùng trên trình duyệt web
- Server (Payment Processing Module) - Module xử lý thanh toán trên máy chủ
- Shopping Cart Page - Trang hiển thị danh sách sản phẩm
- Error Page - Trang thông báo lỗi

2. Mô tả tổng quan hệ thống

2.1. Sơ đồ luồng (theo hình cung cấp)

Luồng hoạt động gồm 4 giai đoạn:

1. Request – Client yêu cầu tài nguyên được bảo vệ (chọn sản phẩm và gửi yêu cầu “Thêm vào giỏ hàng”).
2. Redirect to shopping cart page – Server kiểm tra trạng thái người dùng và giỏ hàng. Nếu đã đăng nhập → chuyển đến trang giỏ hàng.
3. Checkout Submit – Người dùng gửi form thanh toán (Checkout) với thông tin giao hàng (như là tên, địa chỉ, số điện thoại, ...) và phương thức thanh toán (thẻ tín dụng, ví điện tử, ...).
4. Order Processing (Kiểm tra đơn hàng) – Server kiểm tra thông tin thanh toán, tồn kho, địa chỉ giao hàng:

Nếu hợp lệ → xác nhận đơn hàng, lưu vào cơ sở dữ liệu, redirect đến tài nguyên yêu cầu (Order Success Page).

Nếu thất bại → hiển thị trang lỗi (Error Page) với thông báo chi tiết lỗi.

3. Các tác nhân (Actors)

Actor	Vai trò	Mô tả
Client (Browser)	User Agent	Gửi request, hiển thị trang web, nhận phản hồi từ server
Server	Payment Processing Handler	Kiểm tra trạng thái người dùng, xử lý giỏ hàng, thẩm định th
Shopping Cart Page	Giao diện hiển thị giỏ hàng	Thu thập và hiển thị danh sách sản phẩm, số lượng, giá tiền
Error Page	Trang phản hồi lỗi	Thông báo người dùng khi xác thực thất bại hoặc thanh toá

4. Các chức năng chi tiết

4.1. Request Protected Resource

Mô tả:

Khi người dùng chọn sản phẩm và nhấn nút “Thêm vào giỏ hàng”, hệ thống sẽ xử lý yêu cầu và cập nhật giỏ hàng của người dùng.

Luồng hoạt động:

- Input: HTTP GET request từ client với thông tin sản phẩm (ID sản phẩm, số lượng).
- Server kiểm tra tính hợp lệ (session/token):

- Nếu sản phẩm tồn tại và còn hàng → thêm vào giỏ hàng.
- Nếu sản phẩm không tồn tại hoặc hết hàng → trả về thông báo lỗi.

Kết quả mong đợi:

Sản phẩm được thêm vào giỏ hàng thành công. Người dùng được chuyển hướng đến Shopping Cart Page.

4.2. Redirect to Shopping Cart Page

Mô tả: Server gửi phản hồi HTTP 302 Redirect đến /cart để hiển thị giỏ hàng.

Yêu cầu chức năng:

- Hiển thị danh sách sản phẩm trong giỏ hàng bao gồm:
 - Tên sản phẩm
 - Hình ảnh sản phẩm
 - Giá đơn vị
 - Số lượng
 - Tổng giá từng sản phẩm
 - Hiển thị tổng giá tất cả sản phẩm trong giỏ hàng.
- Có nút "Thanh toán" để tiếp tục quy trình mua hàng.
- Có tùy chọn để cập nhật số lượng hoặc xóa sản phẩm khỏi giỏ hàng.

Ràng buộc:

- Giao diện thân thiện, dễ đọc, responsive trên các thiết bị.
- Hiển thị đầy đủ thông tin sản phẩm để người dùng xem xét trước khi thanh toán.
- Không hiển thị thông tin nhạy cảm trong URL hoặc log.

4.3. Form Submitted (Checkout Information)

Mô tả:

Client gửi thông tin thanh toán và giao hàng đến server qua HTTP POST.

Input:

Thông tin giao hàng:

- Họ và tên (text)
- Địa chỉ giao hàng (text)
- Số điện thoại (text)
- Email (text, optional)
- Phương thức thanh toán:
 - Thẻ tín dụng/ghi nợ
 - Ví điện tử (Momo, ZaloPay, VNPAY, ...)
 - Thanh toán khi nhận hàng (COD)
- Thông tin thanh toán (tùy theo phương thức):

- Số thẻ (masked)
- Ngày hết hạn
- CVV (masked)

Quy tắc xử lý:

- Server nhận form tại endpoint </checkout/submit>.
- Kiểm tra dữ liệu đầu vào:
 - Không được trống (các trường bắt buộc).
 - Định dạng email hợp lệ (nếu có).
 - Số điện thoại hợp lệ (10-11 số).
 - Không chứa ký tự đặc biệt gây lỗi SQL hoặc XSS.
 - Validate thông tin thanh toán theo chuẩn của từng phương thức.

4.4. Order Page

Mô tả: Server kiểm tra thông tin đơn hàng và xử lý thanh toán thông qua các bước xác thực nghiêm ngặt.

Các bước xử lý:

1. Kiểm tra tồn kho: Xác minh tất cả sản phẩm trong đơn hàng còn đủ tồn kho. Nếu có sản phẩm hết hàng → chuyển đến Error Page với thông báo chi tiết.
2. Xác thực thông tin thanh toán: Kiểm tra thông tin thẻ tín dụng/ví điện tử với cổng thanh toán (Payment Gateway). Xác minh số dư/hạn mức đủ để thanh toán.
3. Tạo đơn hàng: Nếu tất cả kiểm tra hợp lệ → tạo mã đơn hàng (Order ID). Lưu thông tin đơn hàng vào database. Cập nhật số lượng tồn kho.
4. Xử lý thanh toán: Gọi API của Payment Gateway để xử lý giao dịch. Nếu thanh toán thành công → tạo session/token xác nhận. Nếu thanh toán thất bại → rollback và chuyển hướng đến Error Page.

Ràng buộc bảo mật:

- Thông tin thẻ thanh toán được mã hóa.
- Thông tin giao dịch được ghi log để audit.
- Dữ liệu truyền qua HTTPS.
- Sử dụng tokenization để bảo vệ thông tin thẻ.
- Áp dụng rate limiting để tránh brute force/fraud.

4.5. Redirect to Order Success Page (Success Path)

Mô tả:

Sau khi thanh toán thành công, server chuyển hướng người dùng đến Order Success Page để xác nhận đơn hàng.

Yêu cầu chức năng:

- Hiển thị thông báo thành công: "Đơn hàng của bạn đã được đặt thành công!"
- Hiển thị thông tin đơn hàng:
- Mã đơn hàng (Order ID)
- Danh sách sản phẩm đã mua
- Tổng giá trị đơn hàng
- Phương thức thanh toán
- Thông tin giao hàng
- Thời gian dự kiến giao hàng
- Cung cấp link "Xem chi tiết đơn hàng" hoặc "Tiếp tục mua sắm".
- Gửi email xác nhận đơn hàng đến người dùng.

Ràng buộc:

- Giữ nguyên context và hiển thị đầy đủ thông tin để người dùng có thể tra cứu.
- Cập nhật trạng thái đơn hàng trong hệ thống.
- Xóa giỏ hàng sau khi đơn hàng được xác nhận.

4.6. Error Page (Failure Path)

Mô tả:

Khi có lỗi xảy ra trong quá trình thanh toán (thanh toán thất bại, hết hàng, thông tin không hợp lệ, lỗi hệ thống), server hiển thị Error Page.

Yêu cầu hiển thị:

- Thông báo lỗi thân thiện:
Ví dụ:
 - "Thanh toán thất bại. Vui lòng kiểm tra lại thông tin thẻ."
 - "Sản phẩm [Tên sản phẩm] đã hết hàng. Vui lòng chọn sản phẩm khác."
 - "Thông tin giao hàng không hợp lệ. Vui lòng kiểm tra lại."
- Cung cấp tùy chọn hành động:
- Nút "Thử lại" → quay lại trang Checkout.
- Nút "Quay về giỏ hàng" → chỉnh sửa đơn hàng.
- Link "Liên hệ hỗ trợ" nếu lỗi hệ thống.
- Hiển thị mã lỗi (nếu cần thiết) để hỗ trợ troubleshooting.

Ràng buộc:

- Không tiết lộ chi tiết kỹ thuật hoặc thông tin nhạy cảm (tránh rò rỉ bảo mật).
- Error page không được cache để tránh hiển thị sai thông tin.
- Ghi log lỗi vào hệ thống để phân tích và khắc phục.
- Thông báo lỗi phải đủ rõ ràng để người dùng hiểu và hành động.

5. Yêu cầu phi chức năng (Non-functional Requirements)

Thuộc tính	Mô tả
Hiệu năng (Performance)	Thời gian phản hồi thanh toán < 2 giây
Bảo mật (Security)	Mã hóa HTTPS, session timeout, CSRF protection
Khả dụng (Availability)	Hệ thống thanh toán uptime ≥ 99.5%
Tương thích (Compatibility)	Hỗ trợ Chrome, Firefox, Edge mới nhất
Khả năng mở rộng (Scalability)	Có thể tích hợp OAuth/SSO sau này
Khả năng bảo trì (Maintainability)	Code module thanh toán tách biệt để cập nhật

6. Ràng buộc hệ thống

- Dữ liệu người dùng lưu trong bảng users (username, password_hash, role, status):
 - username: Tên đăng nhập, duy nhất cho từng người dùng.
 - password_hash: Mật khẩu đã mã hóa (bảo mật).
 - role: Vai trò trong hệ thống (vd: user, admin).
 - status: Trạng thái tài khoản (vd: active, locked).
- Server lưu session tại session store (Redis hoặc database).
 - Khi người dùng đăng nhập thành công, server tạo session cho user.
 - Session được lưu tại Redis (ưu tiên tốc độ, thao tác nhanh) hoặc trong database (ưu tiên lưu lâu dài).
 - Mỗi session lưu thông tin: session_id, user_id, thời gian hết hạn, trạng thái.
- Các endpoint:
 - /login: Trang giao diện cho phép người dùng nhập username và password.
 - /auth/login: API xử lý request xác thực khi người dùng submit thông tin đăng nhập.
 - /error: Trang hiển thị thông báo lỗi (sai thông tin, tài khoản bị khóa, ...).
 - /dashboard: Trang tài nguyên chỉ dành cho người dùng đã đăng nhập thành công.

7. Kịch bản chính (Main Scenarios)

Scenario ID	Tên kịch bản	Điều kiện đầu vào	Kết quả mong
SC01	User chưa đăng nhập truy cập tài nguyên	Không có session/token	Redirect đến Login Pa
SC02	User đăng nhập đúng	Nhập đúng username/password	Truy cập thành công t
SC03	User đăng nhập sai	Sai thông tin đăng nhập	Hiển thị Error Page
SC04	User retry sau lỗi	Nhập đúng thông tin sau khi lỗi	Truy cập thành công

8. Logging & Audit

- Ghi log tất cả các nỗ lực đăng nhập với thông tin sau:
 - IP address của client
 - Thời điểm đăng nhập (timestamp)
 - Username được sử dụng
- Cảnh báo (alert) khi có hơn 5 lần đăng nhập sai trong 5 phút với cùng IP hoặc username:
 - Tạo event gửi về hệ thống giám sát hoặc gửi email cho admin
 - Khóa tài khoản hoặc yêu cầu xác thực bổ sung nếu cần thiết
- Log lỗi trong quá trình xác thực được lưu trong file `auth.log`:
 - Lỗi đăng nhập (sai mật khẩu, tài khoản bị khóa, ...)
 - Lỗi hệ thống (kết nối database, hết hạn session, ...)
 - Giữ log an toàn, không lộ thông tin nhạy cảm

9. Kết luận

Tài liệu SRS này mô tả chi tiết toàn bộ yêu cầu chức năng và phi chức năng của **quy trình xác thực người dùng** theo sơ đồ đã cung cấp.

Đây là cơ sở để phát triển các tài liệu tiếp theo như **Test Plan**, **Test Case**, và **Security Validation Report**.

2 – Test Plan

Module: Shopping Cart & Checkout Flow

System: E-commerce Web Application – Shopping Cart & Payment Processing

Version: 1.0

Date: 10/11/2025

Prepared by: Nguyễn Vạn Phúc Huy

1. Giới thiệu

1.1. Mục tiêu

Mục tiêu của Test Plan là xác minh tính đúng đắn, bảo mật và ổn định của quy trình giỏ hàng và thanh toán.

Hệ thống phải đảm bảo:

- Người dùng có thể thêm sản phẩm vào giỏ hàng thành công.
- Giỏ hàng hiển thị đúng thông tin (tên, giá, số lượng, tổng tiền).
- Khi checkout, người dùng có thể nhập thông tin giao hàng và phương thức thanh toán.
- Khi thanh toán thành công, hệ thống tạo đơn hàng và chuyển đến trang xác nhận.
- Khi thanh toán thất bại (hết hàng, không đủ tiền, dữ liệu sai), hệ thống trả về trang lỗi (Error Page).

1.2. Module phạm vi

Bao gồm:

- **Client (Browser):** Gửi yêu cầu thêm sản phẩm, xem giỏ hàng, nhập form thanh toán, nhận phản hồi điều hướng.
- **Shopping Cart Page:** Giao diện hiển thị danh sách sản phẩm đã chọn, hỗ trợ chỉnh sửa số lượng, xóa sản phẩm, nút thanh toán.
- **Server (Shopping Cart & Checkout Module):** Kiểm tra tồn kho, xác thực dữ liệu, xử lý thanh toán, điều hướng kết quả.
- **Error Page / Redirect Logic:** Hiển thị kết quả đơn hàng hoặc thông báo lỗi (hết hàng, thanh toán thất bại, dữ liệu không hợp lệ).

2. Phạm vi kiểm thử (Scope)

Mục	Mô tả
Trong phạm vi	Toàn bộ quy trình từ: yêu cầu thêm sản phẩm → xem giỏ hàng → checkout → nhập thông tin → - Chức năng quản lý tài khoản (profile, login, sign-up) - Reset mật khẩu, xác thực 2 lớp
Ngoài phạm vi	- Tích điểm hoặc chương trình loyalty - Chức năng quản trị (admin panel) - Load testing và stress testing - Tích hợp sâu với payment gateway thực

3. Mục tiêu kiểm thử (Objectives)

- Xác nhận hệ thống thêm sản phẩm vào giỏ hàng đúng (tên, giá, số lượng).
- Đảm bảo hiển thị giỏ hàng chính xác (danh sách, subtotal, totals).
- Kiểm tra chuyển đến trang Order với form checkout hoàn chỉnh.
- Kiểm tra luồng thanh toán thành công (tạo đơn hàng, chuyển Success Page).
- Kiểm tra luồng thanh toán thất bại (hết hàng, thanh toán lỗi, dữ liệu sai).
- Xác minh xác thực dữ liệu đầu vào (không bỏ trống, định dạng email/phone hợp lệ).
- Đảm bảo dữ liệu thanh toán được bảo vệ (không lộ card data, truyền qua HTTPS).
- Kiểm tra thông báo lỗi thân thiện và không lộ chi tiết kỹ thuật.

4. Chiến lược kiểm thử (Test Strategy)

Loại kiểm thử	Mô tả
Functional Testing	Kiểm tra từng chức năng (add product, view cart, checkout, payment processing)
UI Testing	Kiểm tra form, nút, layout, responsiveness trên Shopping Cart Page và Order Page
Data Validation	Kiểm tra xác thực dữ liệu đầu vào (required fields, email format, phone format)
Negative Testing	Nhập sai thông tin, bỏ trống, ký tự đặc biệt, sản phẩm hết hàng, thanh toán không đủ tiền
Security Testing	Kiểm tra session/token, mã hóa HTTPS, không lộ card data, SQL injection prevention
Usability Testing	Đánh giá thông báo lỗi, mô tả hành động, bước retry, flow dễ hiểu
Regression Testing	Đảm bảo fix bug không phá vỡ luồng chính

5. Các thành phần cần kiểm thử (Test Items)

- URL và tài nguyên Shopping Cart Page và Order Page.
- Shopping Cart Page UI (danh sách sản phẩm, form chỉnh sửa, nút Thanh toán).
- Order Page UI (form giao hàng, phương thức thanh toán, nút submit).
- Server-side Shopping Cart & Payment Module (kiểm tra dữ liệu, tồn kho, xử lý thanh toán).
- Redirect logic (HTTP 302 hoặc chuyển trang đúng khi hợp lệ/lỗi).
- Order Success Page (hiển thị Order ID, tổng tiền, thông báo thành công).
- Error Page (hiển thị lỗi hết hàng, thanh toán thất bại, dữ liệu không hợp lệ).
- Email notification system (gửi xác nhận đơn hàng).

6. Giả định và phụ thuộc (Assumptions & Dependencies)

Giả định	Phụ thuộc
Dữ liệu sản phẩm, người dùng tồn tại trong DB	Database hoạt động chuẩn, có seed data test sẵn
HTTPS đã được cấu hình	Hạ tầng test có SSL certificate hợp lệ
Payment Gateway sandbox sẵn sàng	API Payment Gateway phải response đúng giờ, test cards hoạt động
Trình duyệt hỗ trợ session/cookie	Server có quản lý session đúng chuẩn

7. Môi trường kiểm thử (Test Environment)

Thành phần	Cấu hình
Client	Chrome 118+, Firefox 120+, Safari 16+
Server	Apache / NodeJS / Django (staging environment)
Database	MySQL 8.0 (tables: products, shopping_carts, orders, users)
Payment Gateway	Stripe / PayPal Sandbox environment
OS	Windows 11 / macOS / Ubuntu 22.04
Network	Localhost hoặc môi trường staging
Tools	Postman, Selenium IDE, Chrome DevTools, Burp Suite

8. Kế hoạch tài nguyên & nhân sự (Resources & Roles)

Vai trò	Tên	Trách nhiệm
Test Lead	Nguyễn A	Lập kế hoạch, giám sát tiến độ, review defect, đánh giá kết quả
Tester 1	Nguyễn A	Viết & chạy test case chức năng (Functional, Negative Testing)
Tester 2	Nguyễn A	Kiểm tra UI, ghi log, tạo test data, Security Testing
Dev Support	Nguyễn A	Fix lỗi phát hiện, cập nhật environment, hỗ trợ debug

9. Tiêu chí bắt đầu & kết thúc (Entry & Exit Criteria)

Loại	Tiêu chí
Entry Criteria	<ul style="list-style-type: none">- Module Shopping Cart được build thành công- Database users, products, orders đã được tạo- Môi trường test sẵn sàng (server chạy)- Payment Gateway sandbox configured- Test case document đã được review- Test data đã được chuẩn bị
Exit Criteria	<ul style="list-style-type: none">- 100% test case đã được chạy- ≥95% test case Passed- Không còn lỗi Critical hoặc Major- Defect report hoàn tất- Tất cả test evidence được lưu trữ- Test Summary Report đã approved

10. Quản lý rủi ro (Risk Management)

Mức độ	Rủi ro	Hậu quả	
Cao	Server timeout / session lỗi khi checkout	Người dùng không thể hoàn tất đơn hàng	Kiểm tra retr
Cao	Payment Gateway sandbox down	Không thể test thanh toán	Liên hệ prov
Trung bình	Dữ liệu inventory không cập nhật real-time	Cho phép order sản phẩm hết hàng	Kiểm tra syn
Trung bình	Error Page không hiển thị đúng	Người dùng bị kẹt trong flow	Test UI bằng
Thấp	Kết nối DB gián đoạn	Delay tạm thời, retry	Setup conne

11. Lịch trình (Test Schedule)

Hoạt động	Người phụ trách	Ngày bắt đầu	Ngày kết thúc	Ước tính
Viết Test Case	QA Team	11/11/2025	12/11/2025	1 ngày
Chuẩn bị Test Data	Tester 2	12/11/2025	13/11/2025	1 ngày
Thực hiện kiểm thử Functional	Tester 1, 2	14/11/2025	16/11/2025	3 ngày
Thực hiện kiểm thử Security/Validation	Tester 1	17/11/2025	18/11/2025	2 ngày
Ghi Log & Báo cáo	Test Lead	19/11/2025	20/11/2025	2 ngày
Review & Kết thúc	Dev & QA	21/11/2025	22/11/2025	2 ngày

12. Tiêu chí Pass/Fail cho từng Test Case

Loại test	Pass Criteria	
Functional	Hệ thống phản hồi đúng input (sản phẩm được thêm, hiển thị đúng, order được tạo)	Sản phẩm
UI	Form hiển thị đúng, nút hoạt động, layout không bị vỡ, responsive trên mobile	UI lệch, nút
Payment	Thanh toán thành công / lỗi được xử lý đúng theo scenario	Thanh toán
Validation	Dữ liệu sai bị từ chối, thông báo lỗi hiển thị rõ ràng	Dữ liệu sai
Security	Dữ liệu gửi qua HTTPS, session hợp lệ, không lộ password/card	Lộ dữ liệu,
Error Handling	Error Page hiển thị, thông báo thân thiện, không lộ chi tiết kỹ thuật	Error Page

13. Báo cáo kết quả (Deliverables)

- Test Plan Document
- Test Case Specification
- Test Data Sheet
- Test Execution Log
- Defect Report
- Test Summary Report

14. Quy trình quản lý lỗi (Defect Management Process)

1. Ghi nhận lỗi – Tester ghi nhận trong Jira / Trello với:
 - Test Case ID, Severity (Critical/High/Medium/Low)
 - Description, Steps to reproduce, Expected vs Actual Result
 - Screenshot / Log attachment
2. Xác nhận lỗi – Dev team xác nhận lỗi, gán Priority, Assignee
3. Gán trạng thái – Chuyển "In Progress" khi Dev bắt đầu fix
4. Fix & Build – Dev sửa lỗi, tạo build mới
5. Retest – Tester chạy lại test case, xác minh lỗi đã fix
6. Kết thúc – Khi đạt yêu cầu → chuyển "Closed"
7. Regression Test – Chạy lại toàn bộ suite liên quan để đảm bảo không tạo bug mới

15. Kết luận

Tài liệu Test Plan này xác định cách thức kiểm thử, phạm vi, lịch trình, tài nguyên và tiêu chí thành công cho module Shopping Cart & Checkout Flow.

Đây là nền tảng để thực hiện các tài liệu tiếp theo như Test Case Specification, Test Execution Log* và Test Summary Report.

3 – Test Case Specification

Module: Shopping Cart & Checkout Flow

System: E-commerce Web Application – Shopping Cart & Payment Processing

Version: 1.0

Date: 10/11/2025

Prepared by: Nguyễn Văn Phúc Huy

1. Giới thiệu

Mục đích:

Kiểm thử tính đúng đắn, bảo mật, xác thực dữ liệu và khả năng xử lý lỗi của quy trình giỏ hàng, checkout và thanh toán trực tuyến.

Phạm vi:

Bao gồm các bước từ yêu cầu thêm sản phẩm → xem giỏ hàng → chuyển đến trang Order → nhập thông tin → xử lý thanh toán → chuyển hướng đến Success/Error Page.

2. Cấu trúc Test Case

Mục	Mô tả
Test Case ID	Định danh duy nhất (TC01, TC02...)
Test Objective	Mục tiêu kiểm thử
Pre-Condition	Điều kiện trước khi thực thi
Test Steps	Các bước thực hiện chi tiết
Test Data	Input cụ thể
Expected Result	Kết quả mong đợi
Actual Result	Kết quả thực tế (sau khi test)
Status	Pass / Fail
Priority	High / Medium / Low

3. Danh sách Test Case chi tiết

TC01 – Thêm sản phẩm vào giỏ hàng (Single Product)

- **Mục tiêu:** Xác minh hệ thống thêm một sản phẩm vào giỏ hàng khi người dùng nhấn "Thêm vào giỏ".
- **Tiền điều kiện:**
 - Người dùng đang trên trang chi tiết sản phẩm
 - Sản phẩm có sẵn hàng (stock > 0)
 - Giỏ hàng trống hoặc có sản phẩm khác
- **Các bước thực hiện:**
 1. Mở chi tiết sản phẩm (VD: Áo thun – 100.000 VND)
 2. Chọn số lượng = 1
 3. Nhấn nút "Thêm vào giỏ hàng"
 4. Chờ phản hồi từ server
- **Dữ liệu kiểm thử:**
 - Product ID: P001
 - Product Name: Áo thun
 - Price: 100,000 VND
 - Quantity: 1

- **Kết quả mong đợi:**
 - Thông báo thành công: "Sản phẩm đã được thêm vào giỏ hàng"
 - Số lượng trong giỏ tăng lên (hiển thị badge: 1)
 - Dữ liệu được lưu trong session/database
- **Mức ưu tiên: High**

TC02 – Xem giỏ hàng với 1 sản phẩm

- **Mục tiêu:** Xác minh trang Shopping Cart hiển thị đúng thông tin sản phẩm, giá, số lượng.
- **Tiền điều kiện:**
 - Người dùng đã thêm sản phẩm vào giỏ (TC01 passed)
- **Các bước thực hiện:**
 1. Nhấn nút "Xem giỏ hàng" hoặc vào </cart/view>
 2. Kiểm tra hiển thị sản phẩm
 3. Kiểm tra giá và tổng tiền
- **Dữ liệu kiểm thử:**
 - Expected product: Áo thun
 - Expected quantity: 1
 - Expected unit price: 100,000 VND
- **Kết quả mong đợi:**
 - Hiển thị bảng với các cột: Sản phẩm | Số lượng | Đơn giá | Thành tiền
 - Sản phẩm đúng: "Áo thun" – Qty: 1 – Đơn giá: 100,000 VND – Thành tiền: 100,000 VND
 - Subtotal = 100,000 VND
 - Có nút "Tiếp tục mua sắm" và "Thanh toán"
- **Mức ưu tiên: High**

TC03 – Thêm nhiều sản phẩm vào giỏ hàng

- **Mục tiêu:** Kiểm tra giỏ hàng có thể chứa nhiều sản phẩm khác nhau.
- **Tiền điều kiện:**
 - Giỏ hàng rỗng
 - Có ít nhất 2 sản phẩm khác nhau có sẵn hàng
- **Các bước thực hiện:**
 1. Thêm Áo thun (100,000 VND) × 1
 2. Thêm Quần jean (250,000 VND) × 2
 3. Vào giỏ hàng

- **Dữ liệu kiểm thử:**

- Product 1: Áo thun | Qty: 1 | Price: 100,000 VND
- Product 2: Quần jean | Qty: 2 | Price: 250,000 VND (mỗi cái)

- **Kết quả mong đợi:**

- Giỏ hàng hiển thị 2 dòng sản phẩm
- Thành tiền Áo thun: 100,000 VND
- Thành tiền Quần jean: 500,000 VND ($250,000 \times 2$)
- Subtotal: 600,000 VND

- **Mức ưu tiên: High**

TC04 – Chỉnh sửa số lượng sản phẩm trong giỏ

- **Mục tiêu:** Kiểm tra người dùng có thể thay đổi số lượng sản phẩm trực tiếp trong giỏ hàng.

- **Tiền điều kiện:**

- Giỏ hàng có sản phẩm (VD: Áo thun \times 1)

- **Các bước thực hiện:**

1. Mở giỏ hàng
2. Tìm sản phẩm "Áo thun"
3. Thay đổi số lượng từ 1 \rightarrow 3 (nhấn nút + hoặc nhập trực tiếp)
4. Quan sát cập nhật giá

- **Dữ liệu kiểm thử:**

- Product: Áo thun | Current Qty: 1 | Unit Price: 100,000 VND
- New Qty: 3

- **Kết quả mong đợi:**

- Số lượng được cập nhật: 3
- Thành tiền được tính lại: 300,000 VND ($100,000 \times 3$)
- Subtotal được cập nhật tương ứng

- **Mức ưu tiên: High**

TC05 – Xóa sản phẩm khỏi giỏ hàng

- **Mục tiêu:** Kiểm tra người dùng có thể xóa sản phẩm khỏi giỏ hàng.

- **Tiền điều kiện:**

- Giỏ hàng có ít nhất 1 sản phẩm

- **Các bước thực hiện:**

1. Mở giỏ hàng
2. Nhấn nút "Xóa" (x) trên sản phẩm "Áo thun"

3. Xác nhận xóa

- **Dữ liệu kiểm thử:**

- Product to remove: Áo thun

- **Kết quả mong đợi:**

- Sản phẩm được xóa khỏi giỏ hàng
- Subtotal được cập nhật (hoặc giỏ hàng rỗng nếu không còn sản phẩm)
- Hiển thị thông báo "Sản phẩm đã được xóa"

- **Mức ưu tiên: Medium**

TC06 – Tiến hành thanh toán (Checkout Navigation)

- **Mục tiêu:** Xác minh hệ thống chuyển đến trang Order Page khi người dùng nhấn "Thanh toán".

- **Tiền điều kiện:**

- Giỏ hàng có sản phẩm (VD: Áo thun × 1 = 100,000 VND)

- **Các bước thực hiện:**

1. Mở giỏ hàng
2. Nhấn nút "Thanh toán"
3. Kiểm tra trang Order Page hiển thị

- **Dữ liệu kiểm thử:**

- Không cần

- **Kết quả mong đợi:**

- Chuyển đến trang </order/create>
- Hiển thị form với các trường:
 - Tên người nhận
 - Địa chỉ giao hàng (Thành phố, Quận, Phường, Số nhà)
 - Số điện thoại
 - Email
 - Phương thức thanh toán (Credit card, eWallet, COD)
- Hiển thị Order Summary (danh sách sản phẩm, tổng tiền)

- **Mức ưu tiên: High**

TC07 – Điền thông tin giao hàng hợp lệ

- **Mục tiêu:** Kiểm tra form đầu vào chấp nhận thông tin giao hàng hợp lệ.

- **Tiền điều kiện:**

- Đang ở trang Order Page

- **Các bước thực hiện:**

1. Điền tên: "Nguyễn Văn A"
2. Điền địa chỉ: "Thành phố HCM, Quận 1, Phường Bến Nghé, Số 123"
3. Điền số điện thoại: "0932123456"
4. Điền email: "user@example.com"
5. Chọn phương thức: "Thẻ tín dụng"

- **Dữ liệu kiểm thử:**

- Tên: Nguyễn Văn A
- Địa chỉ: Thành phố HCM, Quận 1, Phường Bến Nghé, Số 123
- Số điện thoại: 0932123456
- Email: user@example.com
- Payment method: Credit Card

- **Kết quả mong đợi:**

- Tất cả trường đều chấp nhận dữ liệu
- Không hiển thị thông báo lỗi
- Form có thể submit

- **Mức ưu tiên: High**

TC08 – Thanh toán thành công

- **Mục tiêu:** Xác minh hệ thống xử lý thanh toán thành công và tạo đơn hàng.

- **Tiền điều kiện:**

- Đã điền đủ thông tin giao hàng (TC07 passed)
- Payment Gateway sandbox sẵn sàng

- **Các bước thực hiện:**

1. Nhấn nút "Xác nhận thanh toán"
2. Chờ server xử lý (≤ 5 giây)
3. Kiểm tra phản hồi

- **Dữ liệu kiểm thử:**

- Card Number: 4242 4242 4242 4242 (Stripe test card)
- Expiry: 12/25
- CVC: 123

- **Kết quả mong đợi:**

- Chuyển đến trang </order/success>
- Hiển thị:
 - Order ID: ORD-20251110-001 (ví dụ)
 - Thông báo: "Đơn hàng của bạn đã được tạo thành công"

- Tổng tiền: 100,000 VND
- Trạng thái: "Chờ xác nhận"
- Nút "Xem chi tiết" hoặc "Tiếp tục mua sắm"
- Email xác nhận được gửi tới user@example.com
- **Mức ưu tiên: High**

TC09 – Thanh toán thất bại (Insufficient Funds)

- **Mục tiêu:** Kiểm tra hệ thống xử lý khi thanh toán thất bại do không đủ tiền.
- **Tiền điều kiện:**
 - Đã điền đủ thông tin giao hàng
- **Các bước thực hiện:**
 1. Điền thẻ tín dụng không đủ tiền
 2. Nhấn "Xác nhận thanh toán"
 3. Chờ server xử lý
- **Dữ liệu kiểm thử:**
 - Card: 4000 0000 0000 0002 (Stripe test – declined card)
 - Expiry: 12/25
 - CVC: 123
- **Kết quả mong đợi:**
 - Chuyển đến trang /error
 - Hiện thị thông báo: "Thanh toán thất bại. Vui lòng kiểm tra lại thông tin thẻ"
 - Nút "Quay lại" hoặc "Thử lại"
 - Đơn hàng KHÔNG được tạo
 - KHÔNG lộ chi tiết: "Insufficient funds" hay "Card declined due to..."
- **Mức ưu tiên: High**

TC10 – Thanh toán thất bại (Out of Stock)

- **Mục tiêu:** Kiểm tra hệ thống khi sản phẩm hết hàng trong quá trình checkout.
- **Tiền điều kiện:**
 - Giỏ hàng có sản phẩm (Áo thun × 1)
 - Sản phẩm bị hết hàng từ sau khi thêm vào giỏ
- **Các bước thực hiện:**
 1. (Scenario: Sản phẩm bị hết hàng trên server)
 2. Điền thông tin giao hàng đầy đủ
 3. Nhấn "Xác nhận thanh toán"
- **Dữ liệu kiểm thử:**

- Product: Áo thun | Cart Qty: 1 | Server Stock: 0
- **Kết quả mong đợi:**
 - Hiển thị Error Page
 - Thông báo: "Sản phẩm 'Áo thun' không còn sẵn hàng"
 - Nút "Quay lại giỏ hàng" để cập nhật
 - Đơn hàng KHÔNG được tạo
- **Mức ưu tiên: High**

TC11 – Xác thực dữ liệu: Bỏ trống trường bắt buộc

- **Mục tiêu:** Kiểm tra form xác thực khi bỏ trống các trường bắt buộc.
- **Tiền điều kiện:**
 - Đang ở trang Order Page
- **Các bước thực hiện:**
 1. Để trống trường "Tên người nhận"
 2. Nhấn "Xác nhận thanh toán"
- **Dữ liệu kiểm thử:**
 - Tên: (trống)
 - Địa chỉ: "123 Main St"
 - Số điện thoại: "0932123456"
 - Email: "user@example.com"
- **Kết quả mong đợi:**
 - Hiển thị lỗi: "Vui lòng nhập tên người nhận"
 - Form KHÔNG được submit
 - Trường bị highlight (đỏ hoặc lỗi tương tự)
- **Mức ưu tiên: High**

TC12 – Xác thực dữ liệu: Số điện thoại không hợp lệ

- **Mục tiêu:** Kiểm tra form từ chối số điện thoại sai định dạng.
- **Tiền điều kiện:**
 - Đang ở trang Order Page
- **Các bước thực hiện:**
 1. Điền số điện thoại: "123" (quá ngắn)
 2. Nhấn "Xác nhận thanh toán"
- **Dữ liệu kiểm thử:**
 - Số điện thoại: "123"

- **Kết quả mong đợi:**
 - Hiện thị lỗi: "Số điện thoại phải có 10-11 chữ số"
 - Form KHÔNG được submit
- **Mức ưu tiên: Medium**

TC13 – Xác thực dữ liệu: Email không hợp lệ

- **Mục tiêu:** Kiểm tra form từ chối email sai định dạng.
- **Tiền điều kiện:**
 - Đang ở trang Order Page
- **Các bước thực hiện:**
 1. Điền email: "notanemail" (không có @)
 2. Nhấn "Xác nhận thanh toán"
- **Dữ liệu kiểm thử:**
 - Email: "notanemail"
- **Kết quả mong đợi:**
 - Hiện thị lỗi: "Email không hợp lệ"
 - Form KHÔNG được submit
- **Mức ưu tiên: Medium**

TC14 – Kiểm tra bảo mật: Dữ liệu thanh toán không được hiển thị

- **Mục tiêu:** Đảm bảo thông tin thẻ tín dụng không được lưu trữ hoặc hiển thị trên client.
- **Tiền điều kiện:**
 - Đã thanh toán thành công (TC08)
- **Các bước thực hiện:**
 1. Mở trang Order Success Page
 2. Mở Chrome DevTools (F12)
 3. Kiểm tra Local Storage, Session Storage, Cookies
 4. Tìm kiếm "4242" hoặc số thẻ
- **Dữ liệu kiểm thử:**
 - Card: 4242 4242 4242 4242
- **Kết quả mong đợi:**
 - KHÔNG có thông tin thẻ trong Local Storage / Session Storage
 - KHÔNG có số thẻ trong response body
 - Chỉ lưu các ký tự cuối (VD: "****4242") nếu cần hiển thị

- **Mức ưu tiên: High** (Security)

TC15 – Kiểm tra bảo mật: Dữ liệu truyền qua HTTPS

- **Mục tiêu:** Xác minh tất cả request/response thanh toán được mã hóa qua HTTPS.
- **Tiền điều kiện:**
 - Đang ở trang Order Page
- **Các bước thực hiện:**
 1. Mở Chrome DevTools → Network tab
 2. Điền thông tin thẻ tín dụng
 3. Nhấn "Xác nhận thanh toán"
 4. Kiểm tra Protocol của request
- **Dữ liệu kiểm thử:**
 - Không cần
- **Kết quả mong đợi:**
 - Tất cả request đến [/order/create](#) và payment endpoint là **HTTPS**
 - Lock icon hiển thị trên URL bar
 - Response không chứa thông tin thẻ plaintext
- **Mức ưu tiên: High** (Security)

4. Dữ liệu kiểm thử (Test Data)

Test Case	Product	Quantity	Phone	Email	Card	Expected Result
TC01	P001 (100K)	1	-	-	-	Added successfully
TC03	P001 + P002	1 + 2	-	-	-	Multiple items in cart
TC04	P001 (100K)	3 (edited)	-	-	-	Subtotal updated to 300K
TC08	P001 (100K)	1	0932123456	user@ex.com	4242..4242	Payment success
TC09	P001 (100K)	1	0932123456	user@ex.com	4000..0002	Payment failed
TC11	P001 (100K)	1	0932123456	-	-	Validation error
TC12	P001 (100K)	1	123	user@ex.com	-	Invalid phone
TC13	P001 (100K)	1	0932123456	notanemail	-	Invalid email

5. Kết quả mong đợi (tổng quan)

Luồng	Kết quả mong đợi
Add product	Thêm thành công, badge tăng, dữ liệu lưu
View cart	Hiển thị đúng sản phẩm, giá, tổng tiền
Edit quantity	Cập nhật số lượng, tính toán lại subtotal
Delete product	Xóa khỏi giỏ, subtotal cập nhật
Navigate to order	Chuyển đến Order Page, form đầy đủ

Luồng	Kết quả mong đợi
Fill shipping info	Chấp nhận dữ liệu hợp lệ
Payment success	Tạo Order, chuyển Success Page, gửi email
Payment failed	Hiển thị Error Page, không tạo Order
Out of stock	Thông báo lỗi, yêu cầu cập nhật giỏ
Missing data	Thông báo "Vui lòng nhập đầy đủ"
Invalid phone	Thông báo format lỗi
Invalid email	Thông báo format lỗi
Card data security	Không hiển thị card, chỉ "****4242"
HTTPS encryption	Tắt cả request HTTPS, lock icon hiển thị

6. Ghi nhận thực thi (Execution Log)

Test Case	Ngày chạy	Người thực hiện	Kết quả thực tế	Trạng thái	Ghi chú
TC01–TC05	14/11/2025	Tester 1	Đạt đúng mong đợi	Pass	Shopping cart operations ok
TC06–TC08	14/11/2025	Tester 2	Đúng mong đợi	Pass	Checkout & success payment ok
TC09–TC10	15/11/2025	Tester 1	Đúng mong đợi	Pass	Error handling ok
TC11–TC13	15/11/2025	Tester 2	Đúng mong đợi	Pass	Data validation ok
TC14–TC15	16/11/2025	Tester 1	Đúng mong đợi	Pass	Security checks ok

7. Tổng kết

- Tổng test case: 15
- Pass: 15
- Fail: 0
- Critical Defect: 0
- Kết luận: Module Shopping Cart & Checkout Flow hoạt động đúng theo yêu cầu SRS.

4 – Test Suite

Module: Shopping Cart & Checkout Flow

System: E-commerce Web Application – Shopping Cart & Payment Processing

Version: 1.0

Date: 10/11/2025

Prepared by: QA Team

1. Mục tiêu (Purpose)

Tài liệu Test Suite này xác định:

- Cách nhóm các Test Case theo luồng kiểm thử (functional flow).
- Thứ tự thực hiện.
- Mối liên hệ giữa các Test Case.

- Mục tiêu, điều kiện và kết quả mong đợi cho từng nhóm.

Mục đích là để đảm bảo quá trình kiểm thử diễn ra có tổ chức, dễ dàng lặp lại và đánh giá toàn bộ luồng Shopping Cart & Checkout từ đầu đến cuối.

2. Phạm vi (Scope)

Áp dụng cho toàn bộ module giỏ hàng, thanh toán, và xử lý đơn hàng gồm:

- Kiểm tra request thêm sản phẩm vào giỏ
- Kiểm tra redirect đến trang giỏ hàng
- Kiểm tra submit form thanh toán
- Kiểm tra xử lý đơn hàng (thành công/thất bại)
- Xử lý lỗi và retry

3. Cấu trúc tổng quan của Test Suite

Test Suite ID	Tên Test Suite	Mục tiêu chính
TS01	Request & Add to Cart Suite	Kiểm tra hành vi của hệ thống khi người dùng thêm sản phẩm
TS02	Redirect to Cart Page Suite	Kiểm thử chuyển hướng đến trang giỏ hàng và hiển thị dữ liệu
TS03	Checkout Submit & Validation Suite	Đảm bảo hệ thống xác thực form thanh toán với dữ liệu đúng.
TS04	Order Processing & Payment Suite	Kiểm thử các kịch bản xử lý đơn hàng, thanh toán, và lỗi

4. Mô tả chi tiết từng Test Suite

TS01 – Request & Add to Cart Suite

Mục tiêu:

Đảm bảo hệ thống xử lý đúng yêu cầu thêm sản phẩm vào giỏ hàng từ phía client, kiểm tra tồn kho, và lưu dữ liệu.

Điều kiện khởi tạo:

- Sản phẩm tồn tại trong database với stock > 0
- Giỏ hàng rỗng hoặc có sản phẩm khác
- Server hoạt động bình thường

Danh sách Test Case:

- TC01 – Thêm sản phẩm vào giỏ hàng (Single Product)
- TC02 – Xem giỏ hàng với 1 sản phẩm
- TC03 – Thêm nhiều sản phẩm vào giỏ hàng
- TC04 – Chỉnh sửa số lượng sản phẩm trong giỏ
- TC05 – Xóa sản phẩm khỏi giỏ hàng

Kết quả mong đợi:

- Sản phẩm được thêm vào giỏ hàng thành công
- Dữ liệu được lưu trong session/database

- Hiển thị thông báo thành công cho người dùng
- Giỏ hàng hiển thị đúng số lượng, giá cả

Tiêu chí Pass:

- 100% Test Case Pass
- Tất cả sản phẩm được thêm đúng, không mất dữ liệu
- Subtotal tính toán chính xác

TS02 – Redirect to Cart Page Suite

Mục tiêu:

Đảm bảo hệ thống chuyển hướng đúng đến trang giỏ hàng sau khi thêm sản phẩm, và hiển thị đầy đủ thông tin.

Điều kiện khởi tạo:

- Đã thêm ít nhất 1 sản phẩm vào giỏ hàng (TS01 passed)
- Giỏ hàng page đã được load

Danh sách Test Case:

- TC06 – Tiến hành thanh toán (Redirect to Order Page)
- TC07 – Kiểm tra form checkout hiển thị đầy đủ

Kết quả mong đợi:

- Khi nhấn "Thanh toán" → hệ thống redirect đến </order/create>
- Order Page hiển thị form đầy đủ với các trường: tên, địa chỉ, số điện thoại, email, phương thức thanh toán
- Order Summary hiển thị đúng danh sách sản phẩm và tổng tiền

Tiêu chí Pass:

- 100% Test Case Pass
- Redirect hoạt động trong < 2 giây
- Form hiển thị chính xác, không bị vỡ bố cục

TS03 – Checkout Submit & Validation Suite

Mục tiêu:

Kiểm thử khả năng xác thực dữ liệu form thanh toán, xử lý dữ liệu sai và bỏ trống.

Điều kiện khởi tạo:

- Đã vào trang Order Page (TS02 passed)
- Form thanh toán sẵn sàng để điền

Danh sách Test Case:

- TC08 – Điền thông tin giao hàng hợp lệ
- TC09 – Xác thực dữ liệu: Bỏ trống trường bắt buộc

- TC10 – Xác thực dữ liệu: Số điện thoại không hợp lệ
- TC11 – Xác thực dữ liệu: Email không hợp lệ

Kết quả mong đợi:

- Form chấp nhận dữ liệu hợp lệ và có thể submit
- Form từ chối dữ liệu không hợp lệ (trống, định dạng sai)
- Hiện thị thông báo lỗi rõ ràng cho người dùng
- Không gửi dữ liệu lên server nếu validation thất bại

Tiêu chí Pass:

- 100% Test Case Pass
- Không có lỗi validation
- Thông báo lỗi thân thiện, không tiết lộ logic server

TS04 – Order Processing & Payment Suite

Mục tiêu:

Kiểm thử khả năng xử lý đơn hàng, thanh toán, và xử lý các kịch bản lỗi thanh toán, hết hàng.

Điều kiện khởi tạo:

- Đã điền đầy đủ thông tin thanh toán (TS03 passed)
- Payment Gateway sandbox sẵn sàng
- Server có khả năng xử lý payment callbacks

Danh sách Test Case:

- TC12 – Thanh toán thành công
- TC13 – Thanh toán thất bại (Insufficient Funds)
- TC14 – Thanh toán thất bại (Out of Stock)
- TC15 – Kiểm tra bảo mật: Dữ liệu thẻ không được lưu trữ trên client

Kết quả mong đợi:

- Thanh toán thành công → Tạo Order, chuyển đến </order/success>
- Thanh toán thất bại → Chuyển đến /error với thông báo lỗi
- Hết hàng → Thông báo lỗi "Out of stock", không tạo order
- Dữ liệu thẻ tín dụng không lộ trong storage, logs, hoặc response

Tiêu chí Pass:

- 100% Test Case Pass
- Không có lỗi bảo mật hoặc rò rỉ dữ liệu thanh toán
- Tất cả payment responses được xử lý đúng cách

5. Trình tự thực thi Test Suite

Thứ tự	Test Suite	Lý do chạy
1	TS01 – Request & Add to Cart	Thiết lập giỏ hàng và xác minh add product
2	TS02 – Redirect to Cart Page	Kiểm tra chuyển hướng và hiển thị giỏ hàng
3	TS03 – Checkout & Validation	Kiểm tra xác thực form thanh toán
4	TS04 – Payment & Processing	Kiểm tra xử lý thanh toán và bảo mật

6. Dữ liệu dùng chung giữa các Suite

Tên dữ liệu	Mục đích sử dụng	Ghi chú
Product 1 (Áo thun 100K)	Thêm vào giỏ, test add product	Dữ liệu test chính
Product 2 (Quần jean 250K)	Test multiple items	Dữ liệu test thứ 2
Valid Shipping Info	Test checkout form	Dữ liệu giao hàng hợp lệ
4242 4242 4242 4242	Test thanh toán thành công	Stripe test card
4000 0000 0000 0002	Test thanh toán thất bại	Stripe declined card
user@example.com	Test email validation	Email hợp lệ
notanemail	Test email validation	Email không hợp lệ
0932123456	Test phone validation	Phone hợp lệ
123	Test phone validation	Phone quá ngắn

7. Kết quả mong đợi tổng hợp

Suite	Kết quả mong đợi	Trạng thái cuối
TS01	Sản phẩm được thêm vào giỏ, dữ liệu lưu đúng	✅ Pass
TS02	Redirect đúng, form checkout hiển thị đầy đủ	✅ Pass
TS03	Form validation hoạt động, thông báo lỗi rõ ràng	✅ Pass
TS04	Thanh toán xử lý đúng, bảo mật an toàn	✅ Pass

8. Tổng kết

- **Tổng số Test Suite:** 4
- **Tổng số Test Case:** 15
- **Phạm vi bao phủ:** 100% luồng Shopping Cart & Checkout (4 bước: Request → Redirect → Submit → Processing)
- **Kết luận:** ✅ Module Shopping Cart & Checkout hoạt động ổn định, đạt tiêu chuẩn QA cơ bản.