

# **System Analysis And Design**

## **Graduation Project**

**By: Nancy El-Sherbiny**

**Shahd Yasser**

**Abdelrhman Abdelkader**

**Aziz Mamdouh**

**Sara Atallah**

**Instructor: Eng. Abdelrhman Akl**

**Group: ALX2\_SWD6\_S1, Software Testing, NHA**

**System Analysis & Design**

### **1. Introduction**

#### **1.1 Purpose**

This document analyzes and designs the testing strategy for the JPetStore ecommerce system, ensuring that it meets functional and non-functional

requirements. It follows the ISTQB v4.0 guidelines for Test Analysis and Design.

## 1.2 Scope

The JPetStore platform enables:

- User authentication (registration, login)
- Browsing and searching for products
- Shopping cart and order management
- Secure payment processing
- Order history and tracking
- Administrative management for products and users

## 1.3 Objectives

- Provide a structured approach to test case design.
- Ensure test coverage for all functional and non-functional requirements.
- Implement risk-based prioritization of test cases.

## 2. Test Analysis

### 2.1 Identification of Test Conditions

Test conditions are derived from the test basis, which includes:

- Functional requirements (e.g., FR1: User Registration, FR5: Shopping Cart Management)
  - Non-functional requirements (e.g., NFR1: Performance, NFR2: Security)
  - Business rules and use cases
- Example Test Conditions:**
- **TC1:** Verify successful user registration with valid credentials.
  - **TC2:** Verify error message when registering with an already existing email.
  - **TC3:** Verify that the password complexity rules are enforced.
  - **TC4:** Verify that the page load time is less than 3 seconds for the product search page.
  - **TC5:** Verify that the system is resistant to SQL injection attacks.

## 2.2 Risk-Based Prioritization

Test conditions are prioritized based on:

- **High-risk areas:** Payment processing, authentication, data security.
- **Medium-risk areas:** Product browsing, search functionality.
- **Low-risk areas:** Order history display, UI responsiveness.

### Risk Register:

Risk	Likelihood	Impact	Mitigation Strategy
Payment failures	High	Critical	Implement robust payment validation and retry mechanisms.
SQL Injection	High	Severe	Implement parameterized queries and input validation.
Slow page loads	Medium	Moderate	Optimize database queries and caching.

## 2.3 Evaluation of Test Basis

- Checking completeness and correctness of requirements.
- Identifying ambiguities or inconsistencies.
- Verifying testability of requirements.

## 3. Test Design

### 3.1 Test Case Definition

Each test condition is elaborated into test cases with:

- **Test Case ID** (e.g., TC1, TC2)
- **Test inputs** (e.g., user credentials, product selection)
- **Expected outputs** (e.g., successful login, order confirmation)
- **Preconditions** (e.g., user must be registered before login tests)
- **Postconditions** (e.g., order status updated after checkout)
- **Steps to execute** (e.g., Navigate to login page, enter credentials, click login)

## 3.2 Test Data Identification

Test data is classified as:

- **Valid data:** "username: j2ee, password: j2ee"
- **Invalid data:** "password: short" (for a password too short).
- **Boundary data:** "email address with 255 characters" (if there is a limit).

## 3.3 Test Techniques

### Black-Box Techniques:

- **Equivalence Partitioning:** Valid username and password vs. Invalid username/password.
- **Boundary Value Analysis:** Testing minimum and maximum allowed values for user inputs.
- **Decision Table Testing:** Evaluating combinations of payment success and failure conditions.
- **State Transition Testing:** Validating user account state changes (e.g., from logged out to logged in).

### White-Box Techniques:

- **Statement Testing:** Ensuring all code statements execute at least once.
- **Branch Testing:** Verifying all possible decision branches are tested.

### Experience-Based Techniques:

- **Error Guessing:** Identifying areas prone to common user errors (e.g., incorrect order totals).
- **Exploratory Testing:** Unscripted tests to identify unforeseen defects.

## 4. Test Environment & Execution Plan

### 4.1 Test Environment Setup

- **Operating Systems:** Windows.
- **Browsers:** Chrome.

- **Devices:** Desktop, Mobile.
- **Database:** MySQL.
- **Tools:** Selenium (for automation), JMeter (for performance testing).

## 4.2 Test Execution Schedule

Phase	Start Date	End Date	Deliverables
Test Analysis	02/03/2025	09/03/2025	Identified Test Conditions
Test Design	10/03/2025	14/03/2025	Test Cases & Test Data
Test Implementation	15/03/2025	28/03/2025	Test Scripts & Automation Setup
Test Execution	05/04/2025	18/04/2025	Test Reports & Defect Logs

## 5. Additional Considerations

### 5.1 Traceability Matrix

- A traceability matrix will be used to ensure coverage between requirements, test conditions, and test cases.

### 5.2 Automation Strategy

- **Automated Tests:** Login, checkout, payment validation.
- **Reporting:** Automated test reports generated via Selenium.

### 5.3 Defect Management

- Defects will be reported, tracked, and managed using Jira.
- Defect lifecycle: **Open** → **Assigned** → **Fixed** → **Retested** → **Closed**.

### 5.4 Agile & Continuous Testing

- Shift-left testing approach integrated within Agile sprints.
- CI/CD pipelines will automate test execution and reporting.

## 6. Conclusion

This document provides a structured approach for system analysis and test design, following ISTQB v4.0 principles. By implementing a mix of black-box, white-box, and experience-based techniques, we ensure comprehensive test coverage and risk mitigation for JPetStore.

## 7. Test Cases

These are test cases for website “JPetStore” according to all website requirements

### Test Cases Table:

TC ID	Description	Steps	Test Data	Expected Result
1	Checking that a user with valid credential s can log in successfully to the website.	1. Open URL: <a href="https://jpetstore.aspectran.com/">https://jpetstore.aspectran.com/</a> 2. Enter the valid username. 3. Enter the valid password. 4. Click on the Login button. 5. Observe the redirected page.	Username: [j2ee] Password: [j2ee]	1. Login page should be redirected to the HomePage, showing a welcome message to the user.
2	Checking that a user with invalid credential s cannot log in to	1. Open URL: <a href="https://jpetstore.aspectran.com/">https://jpetstore.aspectran.com/</a> 2. Enter the invalid username. 3. Enter the invalid password. 4. Click on the Login button. 5. Observe the system response.	Username: [invalid] Password: [invalid]	1. System should display an error message: "Invalid username or

TC:I D	Descripti on	Steps	Test Data	Expected Result
	the website.			password. Signon failed."
3	Verify that a user can add a product to the cart.	<ol style="list-style-type: none"> <li>1. Log in with valid credentials.</li> <li>2. Navigate to a product page.</li> <li>3. Click "Add to Cart".</li> <li>4. Observe the cart icon.</li> </ol>	Product: [Bulldog]	<p>1. The product is added to the cart, and the cart icon updates to reflect the number of items.</p> <p>1. The user is redirected to the cart page, displaying the added product(s) with correct details (name, price, quantity, total).</p>
4	Verify that the "Go to Cart" button redirects to the cart page.	<ol style="list-style-type: none"> <li>1. Add a product to the cart.</li> <li>2. Click on the "Go to Cart" button.</li> <li>3. Observe the redirected page.</li> </ol>	Product: [Bulldog]	<p>1. The product is removed from the cart, and the cart</p>
5	Verify that a user can remove a product from the cart.	<ol style="list-style-type: none"> <li>1. Add a product to the cart.</li> <li>2. Go to the cart page.</li> <li>3. Click "Remove" for the product.</li> <li>4. Observe the cart updates.</li> </ol>	Product: [Bulldog]	

TC:ID	Description	Steps	Test Data	Expected Result
6	Verify that a user can search for a product using the search bar.	<ol style="list-style-type: none"> <li>1. Log in with valid credentials.</li> <li>2. Enter a product name in the search bar.</li> <li>3. Click the search button.</li> <li>4. Observe the search results.</li> </ol>	Search Term: [Bulldog]	<p>updates accordingly.</p> <ol style="list-style-type: none"> <li>1. The search results page displays products related to the search term.</li> </ol>
7	Verify that a user can review past orders.	<ol style="list-style-type: none"> <li>1. Log in with valid credentials.</li> <li>2. Navigate to the "My Orders" section.</li> <li>3. Observe the list of past orders.</li> </ol>	N/A	<ol style="list-style-type: none"> <li>1. The user can view a list of past orders with details such as order date, items, and total amount.</li> </ol>
8	Verify that a user can register a new account.	<ol style="list-style-type: none"> <li>1. Open URL: <a href="https://jpetstore.aspectran.com/">https://jpetstore.aspectran.com/</a></li> <li>2. Click on the "Register Now" button.</li> <li>3. Fill in the registration form with valid details.</li> <li>4. Click on the "Submit" button.</li> <li>5. Observe the system response.</li> </ol>	Username: [newuser] Password: [newpassword] Email: [ <a href="mailto:newuser@example.com">newuser@example.com</a> ]	<ol style="list-style-type: none"> <li>1. The user is successfully registered and redirected to the</li> </ol>



TC:I D	Descripti on	Steps	Test Data	Expected Result
9	Verify that a user can update the quantity of a product in the cart.	<ol style="list-style-type: none"> <li>1. Log in with valid credentials.</li> <li>2. Add a product to the cart.</li> <li>3. Go to the cart page.</li> <li>4. Update the quantity of the product.</li> <li>5. Observe the cart updates.</li> </ol>	Product: [Bulldog] Quantity: [2]	homepage . 1. The cart updates to reflect the new quantity and total price.