**Test Plan for PetStore**

**1.Test Plan Identifier**:PS-TPV0.2-1024

**2.References:**

1.PetStore\_requerments.dox

2.PetStore\_Test\_strategy\_document\_V01

3.PetStore\_Test\_Case\_Repo\_document\_V01

**3. Introduction:**

This test plan outlines the testing strategy and activities for the latest version of the Pet Store application. In the previous version, feedback indicated the need for updates to several key features. The primary objective of this testing phase is to ensure the functionality and reliability of key features that are within the scope of this release. In this release, we will focus on implementing and thoroughly testing the following features Such as sign in,Add to cart ,checkout and Product category. By executing this test plan, we aim to deliver a robust and user-friendly application that addresses previous feedback and facilitates a smooth shopping experience for our customers

**4. Test Items:**

In the latest verison of petstore V0.2. Things which we are intend to test within this scope of release are,

**Sign Feature**: Verifying that users can successfully create accounts, log in, and manage their profiles securely and sign in as guest.

**Add to Cart Feature**: Ensuring that users can select products and add them to their shopping carts seamlessly.

**Checkout Feature**: Validating the end-to-end process of purchasing items, including payment processing and order confirmation.

**Category Feature**: Testing the navigation and filtering capabilities within product categories to enhance user accessibility and experience

**5. Software Risk Issues:**

The Software Risk issues we can expect during this process like,

**A.** **Integration Issues**: Problems may arise when integrating with third-party services (e.g., payment gateways).

**B. Browser Compatibility**: The application may not function consistently across different web browsers or versions (e.g., Chrome, Firefox, Safari)

.

6.Features To be Tested:

**1. Login Feature-(M)**

* **User Authentication**:
  + Test valid login with correct credentials.
  + Test invalid login attempts (e.g., wrong password, wrong email).
  + Verify the password recovery process (forgot password).
* **Account Creation**:
  + Test successful account registration with valid data.
  + Test validation messages for required fields (e.g., email, password).
  + Verify error messages for duplicate accounts.
* **Session Management**:
  + Ensure users remain logged in during a session.
  + Test session timeout functionality (logout after inactivity).

**2. Add to Cart Feature-(M)**

* **Adding Products**:
  + Test adding a single product to the cart.
  + Test adding multiple quantities of the same product.
  + Verify that different products can be added to the cart.
* **Viewing the Cart**:
  + Ensure users can view their cart with accurate product details.
  + Test that cart updates correctly when items are added or removed.
* **Updating Cart**:
  + Test changing the quantity of items in the cart.
  + Verify the functionality to remove items from the cart.
* **Persistence**:
  + Ensure items remain in the cart after logging out and back in.

**3. Checkout Feature-(H**)

* **Order Process**:
  + Validate the entire checkout process from cart to order confirmation.
  + Test the application of promotional codes and discounts.
* **Payment Processing**:
  + Verify integration with payment gateways.
  + Test valid and invalid payment details (e.g., expired card).
  + Ensure successful transactions are confirmed.
* **Shipping Information**:
  + Test entry and validation of shipping addresses.
  + Verify calculation of shipping costs based on user input.
* **Order Confirmation**:
  + Ensure users receive an order confirmation email with correct details.
  + Validate that order history reflects the completed order.

**4. Category Feature-(L)**

* **Product Navigation**:
  + Test the ability to browse products by categories.
  + Verify that all categories are displayed correctly.
* **Filtering and Sorting**:
  + Ensure users can filter products by various criteria (e.g., price, brand).
  + Test sorting options (e.g., by price, newest).
* **Product Details**:
  + Ensure that clicking on a product within a category leads to its detail page with accurate information.
* **User Experience**:
  + Validate that navigation is intuitive and easy to use.
  + Check for accessibility compliance (e.g., keyboard navigation).

**7.Features Not To Be Tested:**

**Admin Panel Features:**Functionality related to the administration dashboard (e.g., user management, product management) may be excluded if the focus is solely on user-facing features.

**8 Approach:**

**1. Test Design**

* **Test Case Development**: Create detailed test cases for each feature, specifying input conditions, expected results, and execution steps.
* **Test Data Preparation**: Prepare necessary test data for various scenarios, including valid, invalid, and edge cases.

**2. Test Levels**

* **Unit Testing**: Conduct unit tests for individual components (e.g., login logic, cart functionality) to ensure each piece functions correctly.
* **Integration Testing**: Test interactions between components (e.g., the login process leading to cart functionality) and external systems (e.g., payment gateways).
* **System Testing**: Perform end-to-end testing of the complete application to validate the entire user journey from login to checkout.
* **User Acceptance Testing (UAT)**: Engage end-users to validate the application’s functionality and usability in real-world scenarios.

**3. Testing Techniques**

* **Functional Testing**: Verify that each feature behaves as expected based on defined requirements.
* **Boundary Value Testing**: Test edge cases, such as maximum and minimum input values (e.g., password length).
* **Negative Testing**: Intentionally input invalid data to ensure the application handles errors gracefully (e.g., invalid email during login).
* **Exploratory Testing**: Allow testers to explore the application freely to uncover unexpected issues.

**4. Test Environments**

* **Development Environment**: Initial testing in the development environment to catch issues early.
* **Staging Environment**: Conduct tests in a staging environment that mirrors the production setup to validate real-world scenarios.

**5. Automation Strategy**

* **Identify Candidates for Automation**: Determine which repetitive tests (e.g., regression tests) can be automated for efficiency.
* **Automation Tools**: Select appropriate automation tools (e.g., Selenium for web testing) to implement automated tests.

**6. Performance Testing**

* **Load Testing**: Simulate multiple users during the checkout process to assess the application’s performance under load.
* **Stress Testing**: Determine the application’s breaking point by pushing beyond normal usage limits.

**7. Security Testing**

* **Vulnerability Scanning**: Use tools to identify security vulnerabilities, particularly in the login and payment processes.
* **Penetration Testing**: Conduct tests to simulate attacks and ensure that the application can withstand malicious attempts.

**8. Reporting and Metrics**

* **Defect Tracking**: Implement a system for tracking defects found during testing.
* **Test Coverage Metrics**: Monitor the percentage of features and requirements covered by tests to ensure thorough validation.

**9 Item Pass/Fail Criteria**

**1. Login Feature**

* **Pass Criteria**:
  + Users can log in with valid credentials and access their account.
  + Invalid credentials return appropriate error messages (e.g., "Invalid username or password").
  + Password recovery functionality sends a reset email within a specified time frame.
  + Session timeout occurs after the predefined period of inactivity.
* **Fail Criteria**:
  + Users are unable to log in with valid credentials.
  + Error messages are unclear or misleading.
  + Password recovery fails to send an email or takes too long.
  + Session remains active beyond the defined timeout period.

**2. Add to Cart Feature**

* **Pass Criteria**:
  + Users can successfully add items to the cart, with the cart reflecting accurate product details (name, quantity, price).
  + Updating item quantities in the cart functions correctly and updates the total price.
  + Users can remove items from the cart without errors.
  + Cart items persist correctly after logging out and back in.
* **Fail Criteria**:
  + Items cannot be added to the cart or show incorrect details.
  + Cart updates fail or reflect incorrect totals after quantity changes.
  + Items cannot be removed from the cart or cause errors when removed.
  + Cart does not retain items after a user logs out.

**3. Checkout Feature**

* **Pass Criteria**:
  + Users can complete the checkout process with valid payment and shipping information, receiving an order confirmation.
  + All entered information is validated correctly, with appropriate error messages for invalid input.
  + Payment processing occurs successfully without errors.
  + Users receive confirmation emails with accurate order details.
* **Fail Criteria**:
  + Checkout fails to process with valid information, displaying errors without clear guidance.
  + Invalid inputs are accepted without appropriate validation messages.
  + Payment fails without a clear reason or error message.
  + Order confirmation emails are not sent or contain incorrect information.

**4. Category Feature**

* **Pass Criteria**:
  + Users can navigate through product categories without issues, accessing all relevant products.
  + Filtering and sorting options work correctly, providing expected results.
  + Clicking on a product leads to the correct product detail page with accurate information.
* **Fail Criteria**:
  + Navigation between categories leads to errors or incorrect pages.
  + Filtering or sorting does not work as intended, showing irrelevant results.
  + Product detail pages are broken or contain incorrect information.

**10. Suspension Criteria And Resumption Requirement**

**Suspension Criteria**

Testing may be suspended under the following conditions:

1. **Critical Defects Found**
   1. A critical defect is identified that prevents further testing (e.g., inability to log in, major payment processing failures).
2. **Environmental Issues**
   1. Problems with the testing environment, such as server downtime or configuration errors, impede the ability to execute tests effectively.
3. **Lack of Test Data**
   1. Inability to access necessary test data or a failure to prepare test data that impacts multiple tests.
4. **Incomplete Requirements**
   1. Significant changes to requirements or missing functionalities that affect the scope of testing.
5. **Resource Availability**
   1. Key resources (e.g., testers, test environments) are unavailable for an extended period, hindering the testing process.

**Resumption Requirements**

Testing can resume when the following conditions are met:

1. **Defect Resolution**
   1. Critical defects identified during the suspension have been addressed and verified by the development team.
2. **Environmental Stability**
   1. The testing environment is restored to a functional state, allowing tests to be conducted without issues.
3. **Test Data Availability**
   1. Required test data has been prepared and made available for testing, ensuring comprehensive coverage.
4. **Updated Requirements**
   1. All relevant requirements are clarified, documented, and approved, enabling accurate testing.
5. **Resource Availability**
   1. All necessary resources, including testers and tools, are back in place and ready to proceed.

**11. Test Deliverables**

1.Acceptance test plan

2.System/Integration test plan

3.Unit test plans/turnover documentation

4.Screen prototypes

**12. Remaining Test Tasks**

Making the appllicaton compatible with different versions of the Browser

**13. Environmental Needs:**

We need to Test Our Application in vairous different environments such as Pc,Phones and differnet opearting system to test whether our fetaure are functioning properly

**14. Staffing and Training Needs:**

1. Training on the application/system.

2. Training for any test tools to be used.

**15 Responsiblity**

**Test Manager/Lead:** Oversee the entire testing process and ensure alignment with project goals.

**Test Analysts/Testers:** Design and document test cases based on requirements and specifications

**Quality Assurance (QA) Team:** Ensure that testing processes comply with quality standards and best practices.

**16.Schedule:**

The testing schedule from October to November includes several key tasks. First, the test planning phase will take place from October 1 to 5, where the test plan document will be finalized and resources identified. Next, test case development will occur from October 6 to 12, focusing on creating test cases for the login, add to cart, checkout, and category features. Test data will be prepared from October 13 to 15, followed by setting up the test environment from October 16 to 18. Test execution will take place from October 19 to November 2, where the test cases will be executed and defects logged. Retesting and regression testing will occur from November 3 to 7, followed by User Acceptance Testing (UAT) from November 8 to 12. Finally, test reporting and closure activities will take place from November 13 to 17, summarizing the results and documenting lessons learned

**17. Planning Risks and Contingencies:**

In planning for risks and contingencies for the test plan, it’s essential to identify potential issues that could impact the testing process, such as delays in requirements gathering, resource availability, or critical defects found during testing. To mitigate these risks, establish contingency plans, including flexible scheduling to accommodate unexpected delays, maintaining a pool of backup resources, and prioritizing testing tasks based on critical functionalities. Regular risk assessments should be conducted throughout the testing phase to ensure timely identification and resolution of issues. Additionally, fostering open communication among team members and stakeholders will help ensure that everyone is aware of potential risks and can contribute to effective solutions as needed.

**18.Approvals:**

**Test Manager**: [Name]

**Project Manager**: [Name]

**Quality Assurance Lead**: [Name]