# **Project Documentation for JpetStore Automated Testing**

## **Introduction**

In an era where online shopping has become the norm, maintaining a seamless and reliable user experience is paramount. This project aims to automate the testing processes for the JpetStore, an online pet shopping portal. By implementing automated test scripts using Selenium, TestNG, Cucumber, and API testing we strive to ensure that users can register, login, search for pets, add pets to the cart, update the cart, and make payments without issues.

## **Project Title**

**Automated Testing for JpetStore**

## **Objective**

This project aims to develop comprehensive automated test scripts to validate the functionality of the JpetStore portal. This includes UI and API testing to ensure the site operates smoothly, delivering a dependable and user-friendly experience for all customers.

## **Project Components**

1. **Selenium**: For automating the web UI.
2. **TestNG**: For managing and executing tests.
3. **Cucumber**: For writing feature files and creating BDD (Behavior-Driven Development) scenarios.
4. **RestAssured**: For API(Application Programming Interface) testing.
5. **Postman**: For manual API testing and documentation.
6. **Maven**: For project build management.
7. **Extent-Reports**: For generating test reports.

## 

## 

## **Project Requirements**

### **Test Case Design and Test Case Requirements**

* **Test Cases for Registration**
  + Verify that a user can successfully register with valid details.
  + Validate error messages for invalid input fields.
* **Test Cases for Login**
  + Verify successful login with valid credentials.
  + Validate error messages for incorrect credentials.
* **Test Cases for Searching Pets**
  + Verify that pets can be searched by name or category.
* **Test Cases for Cart Management**
  + Verify adding pets to the cart.
  + Validate updating the quantity of pets in the cart.
  + Verify removing pets from the cart.
* **Test Cases for Payment**
  + Verify that payment can be successfully made.
  + Validate error messages for payment failures.

### **Test Case Automation Framework Development**

**1. Framework Architecture**

Implemented a hybrid automation testing framework that combines Data-Driven and Behaviour-Driven Development (BDD) approaches.

Using Page Object Model (POM) design pattern to maintain the test object repository.

**2. Selenium, TestNG, and Cucumber Integration**

Integrated Selenium WebDriver for web automation.

Using TestNG for test case management, execution, and reporting.

Implemented Cucumber for BDD test execution.

**3. Data-Driven Testing**

Implement data-driven testing to separate test logic from data, enabling multiple data sets to validate functionality.

### **API Automation Using Postman and RestAssured**

1. **Manual API Testing with Postman**:
   * Document all API endpoints.
   * Verify responses and status codes manually.
2. **Automated API Testing with RestAssured**:
   * Write test scripts to validate API responses.
   * Implement tests for all critical API endpoints.

### 

### **Test Execution**

* Run the automated test suite using TestNG.
* Execute tests for different browsers and devices to ensure cross-browser compatibility.

### 

### **Reporting**

* Generate detailed test reports using ExtentReports.
* Include screenshots for failed test cases.
* Summarize test execution results.

### **Documentation**

* Create detailed documentation for the entire testing process.
* Include instructions for setting up and running the tests.
* Provide a summary of test results and identified issues.

## **Project Deliverables**

1. **Automated Test Scripts**: Selenium scripts for UI testing and RestAssured scripts for API testing.
2. **Test Reports**: Detailed execution reports using Extent-Reports.
3. **Documentation**: Comprehensive documentation covering setup, execution, and results.

## **Evaluation Criteria**

1. **Coverage**: Percentage of functional requirements covered by automated tests.
2. **Accuracy**: Correctness of the automated test results.
3. **Performance**: Efficiency and speed of test execution.
4. **Maintainability**: Ease of updating test scripts for new features or changes.

## 

## 

## **Project Timeline**

* **Day 1**: Requirement gathering and test case design.
* **Day 2**: Setting up the automation framework.
* **Day 3**: Writing and executing UI test scripts.
* **Day 4**: Writing and executing API test scripts.
* **Day 5**: Integration and reporting.
* **Day 6**: Final review and project submission.

## **Project Submission**

* Submit the complete project including all test scripts, reports, and documentation.
* Provide a presentation summarizing the project objectives, approach, results, and conclusions.

## **Conclusion**

By implementing automated testing for the JpetStore, we aim to ensure a reliable and seamless user experience, ultimately contributing to user satisfaction and business success. This project will not only enhance the quality of the JpetStore portal but also set a benchmark for future automated testing endeavors.