### **TEST PLAN**

Project Name: Demo Web application

### 

### **Introduction**

The aim of this document is to give an overview of the testing approach, strategies, and scope for the Demo Web Shop application. This document includes details like scope of the project, objectives, test schedule and resource allocations, test deliverables and reports.

AUT: https://demowebshop.tricentis.com/

### **1. Test Plan Overview**

* **Project Name**: Demo Web Shop
* **Test Lead**: [Sahil Khan]
* **Date**: [04/02/2025]
* **Prepared by**: [Sahil Khan & Projyoti Suryawanshi]

### **2. Objective**

* Ensure the website works as expected across all features and user journeys (e.g., product search, checkout process, user accounts, etc.).
* Verify security, performance, and usability of the website.
* Validate compatibility across devices and browsers.

**3. Testable features:**

Register Login

Forgot password

Search

Advanced search

Books

Computers

Electronics

Jewelry

Gift Cards

Digital downloads

### **4. Test Scope**

* **In-Scope**:
  + User registration & login functionality
  + Product browsing, searching, and filters
  + Product details, ratings, and reviews
  + Shopping cart and checkout process
  + Payment gateway integration
  + Mobile and desktop compatibility
  + Security testing (SSL, encryption)
  + Performance testing (loading speed)
* **Out-of-Scope**:
  + Backend/Database integration
  + Third-party services (unless specified)

### **5. Test Strategy**

* **Types of Testing**:
  + **Functional Testing**: Verifying core website features (product browsing, cart, checkout)
  + **Usability Testing**: Ensuring ease of navigation and user experience
  + **Security Testing**: SSL/TLS, data encryption, account security
  + **Performance Testing**: Load testing, page speed optimization
  + **Compatibility Testing**: Cross-browser and cross-device testing
  + **Regression Testing**: Ensuring new changes don’t affect existing functionality

### **6. Test Approach**

* **Manual Testing**: For UI/UX and functional testing (e.g., testing the checkout process).
* **Automated Testing**: For regression and performance tests (e.g., load testing, smoke tests).
* **Exploratory Testing**: By testers to uncover any unexpected issues in the user flow.

### **7. Test Environment**

* **Browsers**: Chrome, Firefox, Safari, Edge
* **Devices**: Desktop (Windows, macOS), Mobile (iOS, Android)
* **Operating Systems**: Windows, macOS, iOS, Android
* **Tools**: Selenium (for automation), JMeter (for load testing), BrowserStack (for cross-browser testing)

### **8. Test Schedule**

* **Test Planning**: [Start Date] to [End Date]
* **Test Execution**: [Start Date] to [End Date]
* **Bug Fixing and Re-testing**: [Start Date] to [End Date]

### **9. Defect Reporting**

* **Defect Severity Levels**:
  + **Critical**: Major functionality broken (e.g., cannot add products to cart)
  + **High**: Non-critical functionality broken (e.g., UI elements misaligned)
  + **Medium**: Minor issues (e.g., spelling errors, low-priority bugs)
  + **Low**: Cosmetic issues (e.g., UI color mismatch)
* **Defect Tracking Tool**: JIRA, Trello, etc.

### **10. Risk Assessment**

* **Potential Risks**:
  + Changes in website during testing
  + Time constraints for testing on multiple devices/browsers
  + Integration issues with third-party services
* **Mitigation Plans**:
  + Regular updates from developers
  + Prioritize high-risk areas first
  + Use automated testing for repetitive tasks

### **11. Test Closure**

* **Criteria for Completion**:
  + All critical tests passed
  + No high/medium severity bugs pending
  + Full report generated and shared with stakeholders
* **Final Deliverables**:
  + Test Report
  + Defect Logs
  + Test Case Results