Date: Jun 04, 2025

Y Test Plan

Project Name: Snapdeal Automation Testing

Prepared By: Nikhil Damale

Section 2 Table of Contents

- 1. Introduction
- 2. Objective
- 3. Scope
- 4. Testable Features
- 5. Testing Approach
- 6. Test Environment
- 7. Roles and Responsibilities
- 8. Test Schedule
- 9. Test Deliverables
- 10. Entry & Exit Criteria
- 11. Tools
- 12. Risks and Mitigation Plans
- 13. Approvals

1. <a>¶ Introduction

This test plan describes the testing strategy for Snapdeal.com, a major Indian e-commerce platform. The purpose is to ensure the core functionalities such as product browsing, filtering, cart management, and user account operations work seamlessly across supported platforms using Cypress automation and the Page Object Model (POM) design.

AUT (Application Under Test): https://www.snapdeal.com

2. 🎯 Objective

To design and implement a maintainable and scalable test automation framework for Snapdeal using Cypress and POM that validates:

- Product search and filter features
- Category browsing
- Cart operations
- Checkout (simulated)

- User login/signup flows
- Core UI functionalities

This aims to catch bugs early and deliver a seamless user experience.

3. TScope

In Scope:

- Homepage and navigation links
- Product search with valid/invalid inputs
- Product detail view validations
- Cart operations (add/remove/update)
- Login, signup, and profile features
- Wishlist, address management
- Simulated checkout (no real payments)

Out of Scope:

- Payment gateway testing
- Backend/API/database testing
- Mobile app testing
- Security and performance/load testing

4. 🔽 Testable Features

- Homepage loading and search bar functionality
- Navigation bar and category menus
- Product filters (brand, price, delivery type)
- Product detail validation (image, price, availability)
- Add to cart and remove from cart
- Cart quantity change and price recalculation
- Login/signup functionality
- Wishlist and profile updates
- Checkout flow with delivery details
- Validation messages (errors, confirmations)

5. 🧪 Testing Approach

Testing Types:

- Functional Testing
- UI Testing
- Regression Testing
- Usability Testing

Methodologies:

- Black Box Testing
- Exploratory Testing

The POM structure ensures reusable, modular scripts. Cypress handles asynchronous flows with built-in wait mechanisms and retry-ability.

6. 星 Test Environment

Environment Component	Details	
OS	Windows 11, macOS	
Browsers	Chrome, Firefox, Edge	
Network	Broadband, 4G/5G	
Devices	Desktop and laptop	
Test Data	Managed with Cypress fixtures	
Automation Framework	Cypress with POM	

7. Roles and Responsibilities

Role	Responsibility
Test Engineer	Write and execute test cases, report bugs
Developer	Fix bugs and support QA
Test Manager	Define test strategies, monitor progress
Business Analyst	Ensure requirement coverage
Project Manager	Align testing with deadlines

8. Test Schedule

Task	Start Date	End Date
Test Plan Creation	04-Jun-2025	05-Jun-2025
Scenario Creation	05-Jun-2025	06-Jun-2025
Test Case Development	05-Jun-2025	06-Jun-2025
Test Execution	06-Jun-2025	07-Jun-2025
Bug Reporting	07-Jun-2025	07-Jun-2025
Summary Report	07-Jun-2025	07-Jun-2025

9. 📂 Test Deliverabl<u>es</u>

Deliverable	Description	Date
Test Plan	Testing strategy and scope documentation	08-Jun-2025
Test Scenarios	All major test workflows	09-Jun-2025
Test Cases	Functional and regression case scripts	10-Jun-2025
Bug Report	Issues with description and severity	13-Jun-2025
Summary Report	Final testing insights, coverage, suggestions	14-Jun-2025

10. 🚪 Entry & Exit Criteria

Entry Criteria:

- Functional and UI requirements finalized
- Environment setup complete
- Approved test plan and scenarios

Exit Criteria:

- All test cases executed
- Critical bugs resolved
- Summary report submitted and approved

11. **%** Tools

- Cypress (Test Automation)
- JavaScript (Scripting Language)
- Visual Studio Code (IDE)
- Git & GitHub (Version Control)
- Cypress Fixtures (Test Data)

- XMind (Mind Maps)
- Cypress Dashboard (Reports & Logs)

12. 🚹 Risks and Mitigation Plans

Risk Mitigation Strategy

UI changes Use resilient locators and abstraction in POM

Unclear requirements Frequent sync with stakeholders

Test data inconsistencies Use fixture files and mock data

Automation failures Run CI checks frequently, use retry logic

Time limitation Prioritize key user flows

13. **Approvals**

Testing will only proceed to execution after the following are approved:

- Test Plan
- Test Scenarios
- Final Summary Report