TEST AUTOMATION PLAN - WALMART

Author: Venkatramana Reddy Araveeti

Version 1.0 | Date: Jan 12th, 2019

Revision History

Version Number	Revision Date	Summary of Changes	Modified by	Reviewers
1.0	Jan 12, 2018	Draft Initial	Venkatramana Reddy Araveeti	

Prepared By

Name	Role on Project	Start Date	Date Artifact Complete	Approved By
Venkatramana Reddy Araveeti	Senior Test Automation Specialist	Jan 12, 2019		

Table of Contents

1.1 Document Purpose

This Test Plan document is a high-level document that sets standards and define the approach to achieve automation testing objectives for Walmart ecommerce Items Page.

The components of the Test Plan include:

- Document Purpose
- Project Overview
- Automation Testing Objective
- High Level in Scope/Out of Scope Items
- Automation Approach
- Automation Tool Stack
- Testing Milestones
- Execution & Management
- Key Challenges
- Test Cases and Traceability
- Test Data Management
- Deliverables
- Dependencies
- Defect Severity & Priority Standards
- Metrics and Reporting
- Appendix

1.2 Project Overview

The project will be adding new features to add the different Items and release are planned accordingly. Each sprint will be of 3 weeks.

1.3 Automation Testing Objective

The Objective of the Test Automation is to increase test coverage for Items, maximize device support, and save time and money by quickly validating the Test cases.

Overall Automation Testing Objectives:

- Design the Test Automation scripts to perform E2E cycle that can be run after every build. These scripts will also be run across multiple browsers and multiple Operating systems as well
- Test Automation suites can also be run on Dev Environments to ensure the Development Environment are also working fine before they push to QA Environments
- Integrate the Test Automation suites with CI/CD Pipeline and create the smoke, Sanity, narrow Regression and Regression Jobs and Schedule the Each Job as and when needed.
- Configure the Test Automation Smoke and Sanity Jobs as the Downstream Jobs to Development Deployments jobs and ensure these jobs will be run immediately as soon as the deployments are done.
- Track All the Automation Test cases, and the Test Artifacts for every execution.
- Also design a Few API Automation cases and verify the Basic APIs are working as and when the Deployments are done
- Track and provide ROI usage statistics when requested

1.4 Test Case Selection Criteria

Automating every kind of functional test case is neither possible, nor required. Good candidates for automation will be selected based on following criteria:

Ref ID	Feature Selection Criteria
1	Repetitive
2	Stable features
3	Business Scenarios
4	Data-driven tests
5	Complex Test Flows

Complexity of test cases will be determined based on criteria mentioned below:

S. No	Test Case Complexity	No. Of Actions	No. Of Verifications
1	Simple	< 5	< 5
2	Medium	> 5 < 15	> 5 < 10
3	Complex	> 15 < 25	> 10 < 15

1.5 High-Level in Scope/Out of Scope Items

1.5.1 In Scope Items

Below is a list of high-level product Items and items and its functionalities to be covered

Ref ID	Functionality
1	Product Main Details: - Name - Description - Category - Color - Quantity - Price - Rating
2	Related Products Widget
3	Specification Section
4	Ratings & Review Section
5	Q & A Section
6	Add Product to Favorite list / my list
7	Widgets - Pickup - Shipping - Buy In-Store

8	Shopping Cart - Customer can buy chosen product - Customer can continue shopping after selecting a product - Customer delete the items from the Cart - Shopping Cart displays right product details - Customer can edit the product specifications selected
9	Product Availability - Product Available in Stock - Product not available in Stock
10	Return policy of the Product - Return in store - Return by mail

1.5.2 In Scope Devices

Ref ID	Devices
1	Laptop/Desktop
2	iPhone XR
3	iPhone X
4	Nexus
5	Windows Phone
7	Blackberry
6	Tablets - iPad Mini Samsung Galaxy Tab S4
	Platforms
1	Windows 10
2	Mac OS
3	Android
4	iOS

1.5.3 In Scope Browsers

Ref ID	Browsers
1	Chrome
2	Firefox
3	Safari - on Mac OS
4	IE
5	Edge

1.5.4 Out of Scope Items

Below is a list of high-level 'Out of Scope' items that **WILL NOT** considered for Test Automation

Ref ID	Feature Type
1	Usability Testing
2	Human intervention is required
3	Static Text Validation
4	Look & Feel Scenarios
5	Single-use tests
6	Unstable features

1.6 Automation Approach

Selenium, Appium, Rest Assured and Testng Driven Framework

Test Automation Framework has been build using Selenium, Appium, Rest Assured and TestNG Framework. Will be using the Page Object Model with Page Factory as the maintenance and reusability of the Test Code is high. This Framework has also Integrated with Jenkins and will be able to create the Jenkins Jobs and run the Executions on Dev, QA, PAT, BAT, UAT and also Prod Environments

1.7 Automation Tool Stack

Automation Framework has been implemented using the below open source tools that includes:

Selenium WebDriver (For Web)

Appium (Mobile Web and Native Apps)

Eclipse / intelliJ Idea (Editors)

Maven – Dependency management

TestNG – as a Testing Framework

Log4j – to log the Results

Allure Report – Reporting

Properties files – Providing the Environments Configurations

Excel files – Provide the Test Data

Java– to Develop the Test Automation Scripts

Jenkins – Continuous Integration

Rest Assured – for API Automation

Git – Code Repository

1.8 Testing Milestones

Phase	Target Start Date	Target End Date	Comments
1.0			

1.9 Execution & Management of Automation Results

1. The automation team is responsible for running all the test scripts on all the specified browsers, Devices and platforms.

2. The automation team will provide the Test Artifacts and issues to the QA manual team.

3. Manual team will create the Defects and include a Keyword "Automation" to track the Defects Identified by Test Automation

4. Test Automation Team will be responsible for Updating the Test scripts and maintain accordingly as and when changed to the Application flows

1.10 Key Challenges

- **Device diversity:** Multiple platforms and browsers, rendering differences, mobile devices with varied application runtimes.
- **Platform challenges:** Frequent changes and releases.
- **Resource Availability:** Resources required to complete the Coverage
- **Environment Availability:** Downtime of the Application will be affected drastically on the productivity

1.11 Test Cases and Traceability

Candidate test cases will be identified for Automation based on selection criteria mentioned above in section 1.3 and will be tracked using an excel sheet & JIRA/Confluence, rally or Create a Traceability matrix for the Requirements Coverage and use any Code coverage tools

1.12 Test Data Management

Test Data Requirement:
Test data will be prepared before Test Automation execution. This data will be used to execute the flows that require some special data conditions.

Automation Data for Scripts:
Test Automation data will be entered in the Excel file and we will retrieve the data from Excel files programmatically. This will be very much helpful for Data Driven Approach

1.13 Deliverables

- Test Data
- Environment Configurations
- Automated test scripts
- Documentation on the Framework set up and usage
- Test Results and other Artifacts

1.14 Dependencies

1.14.1 Support Required from DEV

The main objective of Test Automation is to speed delivery & get high ROI. This can only be achieved by writing reliable test scripts. For automation team to write stable & reliable scripts, it is very important to have applications designed in the way that support Automation. Dev team can help meet this objective by providing support in following way:

- Add unique id/name to the page elements
- Avoid using frequent Deployments
- Notify the Test Automation Team for Changes or Any Deployments
- Work with Test Automation team to Configure the Jenkins Jobs with Development Deployment Jobs
- Provide the POJO Classes to the API Test Automation Team.
- Provide the Details on the Authentication for any Services
- Maintain a Document for all the APIs, Environments Details

1.14.2 Availability of Hardware Environment for cross-platform & cross-browser testing

Need sufficient number of functioning devices available for automation team to execute the automated scripts on real devices.

1.15 Defect Severity & Priority Standards

This will be managed by Manual Team

1.16 Metrics and Reporting

The objective of the test reporting process is to measure, assess and report to key stakeholders the progress and effectiveness of the testing activities. This activity will occur during strategy development, planning, preparation, execution, and closure phases of testing.

Standardized reporting and metrics include the following:

- Execution Reports with failed/passed scenarios details
- ROI from execution of Automated Scripts
- Test Artifacts
- No of Defects Identified by Automation Runn

1.17 Appendix

1.17.1 Glossary of Terms

1.17.2 Related Documentation References