# Test Plan: CaseKaro E-Commerce Automation

**Project Name:** CaseKaro Checkout Automation

**Document Version:** 1.0

**Date:** February 1, 2026

**Author:** Jayesh Solanke

## 1. Introduction

### 1.1 Objective

The objective of this Test Plan is to define the strategy, scope, and resources required to validate the core user journey on the **CaseKaro** e-commerce platform. The primary focus is verifying the search accuracy (brand isolation), product variant selection (Material types), and cart calculation logic for the "iPhone 16 Pro" product line.

### 1.2 References

* **Requirement Source:** Assignment1B.pdf (QA Intern Assessment)
* **Automation Standard:** Java + Playwright Framework

## 2. Scope of Testing

### 2.1 In-Scope Features

The following functional areas are included in this automation suite:

1. **Navigation:** Homepage to "Mobile Covers" category.
2. **Search Functionality:** \* **Positive Testing:** Search for specific models ("iPhone 16 Pro", "Apple"/”iPhone”).
   * **Negative Testing:** Verification that competitor brands (Samsung, OnePlus, etc.) do not appear in specific search contexts.
3. **Product Interaction:** \* "Quick Add" modal functionality.
   * Selection of specific material variants (Hard, Soft, Glass).
4. **Cart Management:**
   * Adding multiple variants of the same SKU.
   * Validation of Cart Drawer visibility.
   * **Data Integrity:** Verification of price, material label, and product URL in the final cart state.

### 2.2 Out-of-Scope Features

* User Authentication (Login/Signup/Forgot Password).
* Payment Gateway processing and Final Checkout.
* Cross-browser compatibility (Testing limited to Chromium Desktop).
* Mobile viewport responsiveness.
* Third-party integrations (Chat support, Analytics).

## 3. Test Strategy

### 3.1 Tools & Technology Stack

* **Programming Language:** Java (JDK 11 or higher)
* **Automation Framework:** Microsoft Playwright (v1.40+)
* **Build Tool:** Maven
* **Browser:** Chromium (Headless & Headed modes supported)
* **IDE:** IntelliJ IDEA / Eclipse

### 3.2 Automation Approach

The automation script follows a linear execution model designed for stability and speed.

* **Synchronization:** Usage of Playwright's auto-waiting mechanism combined with explicit states (WaitForSelectorState.VISIBLE) for dynamic elements like the Cart Drawer.
* **Locator Strategy:** Priority given to robust locators:
  + *Text-based:* text=Mobile Covers
  + *Filtering:* locator().filter(hasText(...))
  + *CSS Selectors:* For structural elements (e.g., .cart-item).
* **Validation Logic:** \* **Soft Assertions:** Console logging for non-critical data.
  + **Hard Assertions:** Runtime exceptions for critical failures (e.g., Prohibited brand detection).

## 4. Test Environment

* **OS:** Windows 10/11 or macOS.
* **Network:** Stable broadband connection (Required for live site navigation).
* **Test URL:** https://casekaro.com/

## 5. Risks & Mitigation

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk ID** | **Risk Description** | **Probability** | **Mitigation Strategy** |
| **R01** | **Dynamic Overlays:** The "Cart Drawer" overlay may block clicks on the main page. | High | Implemented explicit force: true clicks and wait timeouts to allow overlay animations to complete. |
| **R02** | **Product Availability:** "iPhone 16 Pro" items may go out of stock, changing the UI. | Medium | Script designed to select the *first available* matching item rather than a hardcoded product ID. |
| **R03** | **Network Latency:** Page load speeds may vary, causing ElementNotFound errors. | Medium | Playwright's default 30s timeout is utilized; WaitUntilState.LOAD is used during navigation. |

## 6. Test Deliverables

Upon completion of the testing phase, the following artifacts will be delivered:

1. **Source Code:** CaseKaroTest.java (Clean Java source).
2. **Test Execution Report:** Console logs detailing the "Cart Audit" and price verification.
3. **Defect Report:** (If any failures occur during execution).
4. **Readme File:** Instructions for setting up the Maven environment.

## 7. Entry & Exit Criteria

### 7.1 Entry Criteria

* Test environment (Local Machine) has JDK and Maven installed.
* Website casekaro.com is up and accessible.

### 7.2 Exit Criteria

* 100% of the critical path (Search -> Add to Cart -> Verify) is automated.
* Script executes without TimeoutError or ElementClickIntercepted exceptions.
* Negative validation logic correctly identifies unrelated brands if present.