

## Test Plan – Swag Labs (<https://www.saucedemo.com/>)

### 1. Introducción

Este **Test Plan** define el alcance, enfoque, recursos y cronograma de las pruebas para el sitio *Swag Labs* (Sauce Demo), una aplicación de comercio electrónico de demostración utilizada para validación de flujos de usuario recurrentes como inicio de sesión, navegación de catálogo, gestión de carrito y checkout.

### 2. Objetivos del Test Plan

- Verificar que las **funcionalidades críticas** del sitio funcionan conforme a lo esperado (*login*, catálogo, carrito, checkout).
- Detectar defectos funcionales antes de un despliegue/entrega
- Validar la **usabilidad y estabilidad general** de la aplicación bajo flujos típicos de usuario.

### 3. Alcance de Pruebas

- **Autenticación de Usuario:** validaciones de credenciales válidas e inválidas.
- **Inventario & Catálogo:** visualización, sorting básico y detalles de producto.
- **Carrito de Compras:** agregar/eliminar productos, ver UI actualizada.
- **Checkout:** completar formulario de envío y pago, validar pantalla de confirmación.
- **Logout y limpieza de sesión.**

#### 4. Estrategia de Pruebas

- **Prueba Exploratoria Visual:** Dar un Primer Vistazo a la página para ver su alineación visual, estado de botones y responsividad básica
- **Funcional Manual:** ejecutar acciones como usuario final para verificar comportamiento esperado.
- **Automatización:** crear scripts E2E usando Cypress.
- **Smoke Test:** Realizar **conjunto mínimo de pruebas críticas** para verificar que el sistema: **no está roto y es testeable.**
- Regression Test: Realizar un **amplio enfoque de pruebas** que valida que los cambios **no rompieron funcionalidades existentes**

#### 5. Ambientes de Prueba

Entorno	Plataforma	Navegadores
PROD	Desktop	Chrome

#### 6. Datos de Prueba

Caso	Username	Password
Login éxito	standard_user	secret_sauce
Login falla	kendell_rodriguez	secret_sauce

## 7. Criterios de Entrada / Salida

### Entrada

- Sitio accesible (https con carga estable).
- Credenciales de prueba registradas.

### Salida

- Todos los casos críticos con resultado "Passed".
- Todos los defectos registrados con severidad reportados.

## 8. Automatización y Framework Sugerido

- La estrategia de automatización se basa en buenas prácticas de **testing automatizado mantenible**, utilizando el patrón **AAA (Arrange, Act, Assert)** para garantizar claridad, reutilización y legibilidad de los tests.
- **Actualmente utilizamos Page Object Model (POM)** con **Cypress** ampliamente adoptadas en la industria.
- Cada test automatizado sigue la estructura **AAA**:
  - **Arrange**: preparación del estado inicial (datos, navegación, mocks si aplica).
  - **Act**: ejecución de la acción bajo prueba (interacción del usuario).
  - **Assert**: validación del resultado esperado.

## 9. Reportes y Evidencias

- Logs detallados de cada test.
- Capturas de pantallas en fallos.
- Reporte de cobertura por módulo funcional en allure