

E- Commerce en Automation Practice

Plan de Pruebas

Versión	Fecha	Autor	Descripción	
1.0	07/11/2020	Yunuen Parra	Versión inicial	

Contenido

Introducción	3
Alcance de las pruebas	4
Tipos de pruebas	4
Riesgos	5
Proceso de pruebas	5
Planeación	6
Definiciones y abreviaciones	6

1. Introducción

Este documento describe el plan de pruebas para el e-Commerce de automation practice, por medio de diferentes flujos, plasmando los resultados de las mismas en reportes.

Las pruebas son necesarias porque son parte del proyecto final del curso de Testcafe con Javascript impartido por QA Minds.

1.1 Objetivo general

Probar el sitio *e-Commerce* de *automation practice*, usando la herramienta *TestCafe* con *JavaScript*.

1.1.1 Objetivos específicos

- Probar los módulos *Home, My account, Products*
- Realizar pruebas *End To End* para hacer una compra
- Generar *scripts*
- Generar reportes
- Generar listado de casos de prueba
- Generar reporte de bugs
- Subir el proyecto a *Git*
- Identificar riesgos del proyecto

1.2 Artefactos de pruebas

- Plan de pruebas
- Casos de prueba
- Reporte de bugs
- Scripts en *Visual Studio Code*
- Reporte de bugs
- Resultados de ejecución en reportes en HTML

1.3 Recursos

Para realizar estas pruebas se necesitarán los siguientes recursos instalados:

- Testcafe(es necesario tener instalado el node)
- Visual Studio Code
- Generador de reportes

2. Alcance de las pruebas

2.1. Dentro del alcance

Se ejecutarán pruebas funcionales y automáticas en el *e-Commerce* de *automation practice*, con la siguiente estructura:

- Crear el POM de los selectores de los módulos de *Home*, *My Account*, *Products* y *Shopping Cart*.
- Crear los *scripts* de las pruebas de los módulos *Home*, *My Account* y *Products*
- Crear *scripts* de pruebas *End To End* de diferentes flujos de compra
- Las pruebas se ejecutarán desde la consola
- Generar reportes en *HTML* de los resultados de las pruebas de los diferentes módulos y de los flujos de compra

2.2 Fuera del alcance

Errores que salgan en navegadores diferentes a Chrome y Firefox

3. Tipos de pruebas

3.1 Pruebas Manuales

- Definición

Son llevadas a cabo por personas que interactúan con el software, ejecutando diferentes escenarios de pruebas y obtener los resultados

- Metodología

Ejecutar pruebas manuales en la página para obtener los escenarios de pruebas para los casos de prueba y las pruebas automatizadas.

3.2 Pruebas Automatizadas

- Definición

Son pruebas que nos ayudan a validar el funcionamiento de una aplicación, a base de scripts creados con una herramienta externa que es diferente a la usada para crear la aplicación. Es una prueba manual codificada.

- Metodología

Visual Studio Code:

1. Crear clases para los selectores de cada módulo

2. Crear clases para los test de cada módulo y una clase para los flujos de compra, comenzando cada script de prueba con la palabra test y agregando una descripción.

Consola:

1. Ejecutar las clases que contienen los tests y generar los reportes

4. Riesgos

1. La página deja de funcionar por hacer tantas peticiones a la misma.

El impacto de este riesgo es muy alto debido a que no se pueden ejecutar las pruebas automáticas.

2. Actualización en la página

Si se actualiza o se eliminan los selectores de la página, los scripts ya codificados pueden fallar al momento de la revisión y esto puede impactar en el resultados de las pruebas.

5. Proceso de pruebas

Se muestra la lista de casos de prueba que se crearon para el proyecto. Para ver los pasos de prueba es necesario consultar el documento Casos de prueba - Proyecto final testcafe.

No	#Identificador	Descripción
1	CP_Home1	Abrir la página de automation practice y se muestre el Home
2	CP_Home2	Revisar elementos de Home
3	CP_Home3	Realizar la búsqueda de un producto
4	CP_Home4	Seleccionar tab Best Sellers
5	CP_Home5	Inscribirse al Newsletter
6	CP_Home6	Seleccionar el carrito de compras
7	CP_MyAccount1	Ingresar un correo no válido para crear una cuenta
8	CP_MyAccount2	Ingresar un correo válido para crear una cuenta
9	CP_MyAccount 3	Validar campos obligatorios del formulario Create an account
10	CP_MyAccount4	Ingresar campos obligatorios para crear una cuenta
11	CP_MyAccount5	Crear una wishlist

12	CP_Products1	Ver los elementos de la categoría Dresses
13	CP_Products2	Ver los productos de la categoría Summer Dresses
14	CP_Products3	Seleccionar un vestido y revisar sus elementos
15	CP_Products4	Agregar un producto al carrito
16	CP_Products5	Agregar un producto al wishlist sin tener iniciada la sesión
17	CP_EndToEnd1	Realizar una compra con el método de pago por banco con una nueva cuenta
18	CP_EndToEnd2	Realizar una compra con el método de pago por banco con correo ya registrado
19	CP_EndToEnd3	Iniciar sesión y luego realizar una compra con el método de pago por cheque
20	CP_EndToEnd4	Iniciar sesión y realizar una compra donde el producto se agregue a la wishList

6. Planeación

Tarea	Inicio	Fin
Crear plan de pruebas	01/11/2020	08/11/2020
Crear casos de prueba	02/11/2020	03/11/2020
Crear reporte de bugs	04/11/2020	07/11/2020
Generar scripts	04/11/2020	07/11/2020
Ejecutar pruebas	07/11/2020	07/11/2020

7. Definiciones y abreviaciones

TestCafé: Herramienta de node.js para probar aplicaciones web de manera automática.

End To End: Pruebas de extremo a extremo(E2E), que implica probar un flujo de trabajo de una aplicación de principio a fin. El objetivo principal es replicar escenarios de usuarios reales para que el sistema pueda validarse para la integración y la integridad de los datos.

Visual Studio Code: Editor de código gratuito, que proporciona una utilidad para descargar y gestionar extensiones con las que se puede personalizar la herramienta.

Git: Sistema de control de versiones