



Proyecto: Sitio Web OrangeHRM Page Factory

Por: Teodoro Calle Lara

Estrategias de Pruebas Automatizadas
Training Automatización de pruebas
SofkaU

Historia de revisiones

Versión	Autor(es)	Descripción	Fecha
1.0	Teodoro Calle	Estrategia de pruebas con Page Factory	Octubre 2021

Tabla de Contenidos

- 1.Introducción
- 2. Alcance
- 3. Análisis de riesgos
- 4. Ambiente y Herramientas de Pruebas
 - 4.1 Herramientas de Pruebas
 - 4.2 Arquitectura del framework de automatización

1.Introducción

En esta Estrategia para la realización de pruebas automatizadas se describe el alcance de las pruebas, el ambiente de pruebas, los recursos necesarios y las herramientas a utilizar para la automatización de las pruebas del sitio web OrangeHRM.

2. Alcance

Se realizarán pruebas de caja negra (automatizadas) a las funcionalidades asignadas para reto de automatización de pruebas:

1. Módulo Recruitment.
2. Crear vacante y subir un archivo

Antes de empezar con la ejecución de pruebas se empezará con un testeo rápido para asegurarnos de las funcionalidades básicas del programa funcionan correctamente, para este caso se verificará que se abra de manera correcta la aplicación y que la información desplegada sea legible, entendible, ordenada y coherente con lo que se esperaría encontrar en un homepage.

Las pruebas se ejecutarán basándose en el $\text{riesgo} = \text{probabilidad} \times \text{impacto}$, y el orden será del mayor riesgo al menor.

La cobertura de las pruebas será en Chrome mediante el sistema operativo Windows 10

3. Análisis de Riesgos

No	Riesgos	Probabilidad de Ocurrencia (1-4)	Impacto (1-4)	Riesgo (Prob*Impacto)
1	Crear la vacante	2	3	6
2	Adjuntar archivo	2	1	2

4. Ambiente y Herramientas de Pruebas

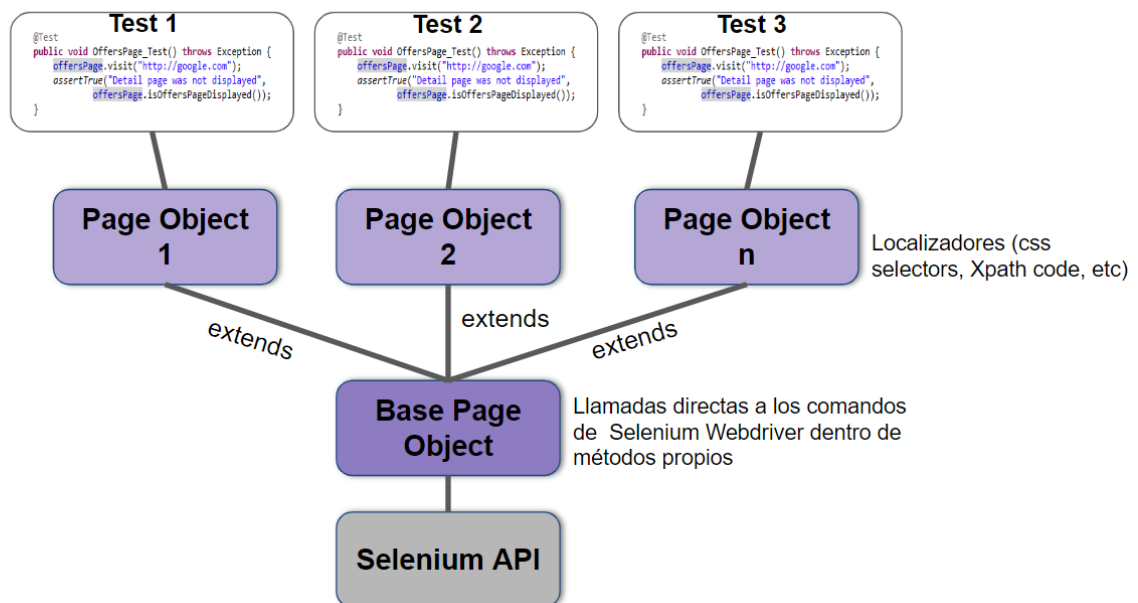
4.1 Herramientas de Pruebas

Herramienta	Función
-------------	---------

Selenium WebDriver	API para automatizar sistemas Web
JUnit testing framework	Ejecución y Reporte de las pruebas
Gradle	Creación de la estructura de proyectos y uso e importación de librerías
Chromedriver	Crea una instancia del navegador Chrome

4.2 Arquitectura del framework de automatización

Arquitectura de Pruebas Automatizadas



Utilizaremos el patrón Page Factory para “mapear” las páginas del sistema a clases “Page” que permitan aislar las acciones de las diferentes páginas y a la vez agrupar todos los webElements de una página y las acciones que se pueden llevar a cabo, en una misma clase. La clase “Base” permite aislar todo el framework de la versión del API de Selenium WD que estemos utilizando. De esta forma si hay algún cambio en los comandos del API no tenemos que cambiar todas las clases sino solo la clase “Base”.

El Page Factory también nos ayudará a concentrar los localizadores en estas clases “Page”, de forma que cuando el sistema cambia y es necesario actualizar el código de los css selectors, xpath o lo que hayamos utilizado para localizar los webElements, solo tenemos que cambiarlo una sola vez en la clase “Page” y los “Tests”, que son el último nivel, no necesitan ningún cambio (a menos que haya cambiado la lógica de

funcionamiento y dentro de los cambios se hayan eliminado o agregado funcionalidades al sistema).