

Proyecto: Sitio Web zonafit.co

Plan de Prueba Automatizadas

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Autor | Descripción | Fecha |
| 1.0 | Camilo Andrés Rodríguez Hoyos | Plan de Prueba | Diciembre 30 de 2021 |

Tabla de Contenidos

[1.Introducción](#_Toc91775411)

[2. Alcance](#_Toc91775412)

[2.1 No está en el alcance](#_Toc91775413)

[3. Roles y Responsabilidades](#_Toc91775414)

[5. Diseño de casos de prueba](#_Toc91775415)

[6. Ambiente y Herramientas de Pruebas](#_Toc91775416)

[6.1 Herramientas de Pruebas](#_Toc91775417)

[6.2 Ambiente de Pruebas](#_Toc91775418)

[7. Criterios de Entrada y Salida](#_Toc91775419)

[7.1 Criterios de Entrada](#_Toc91775420)

[7.2 Criterios de Salida](#_Toc91775421)

[8. Planificación de ejecución de las pruebas](#_Toc91775422)

[9. Reporte de Pruebas](#_Toc91775423)

# 1.Introducción

En este documento se describe el plan para la realización de pruebas automatizadas se describe el alcance de las mismas, el ambiente de pruebas, los recursos necesarios, las herramientas a utilizar y los riesgos de las pruebas del proyecto zonafit.co.

Este documento tiene como principal objetivo generar un plan de pruebas automatizada para la página web reqres.in por medio de RestAssured. Dicho proyecto se trabajará bajo la metodología ágil scrum y contará con 2 Sprints, cada uno de dos semanas.

Se realizarán pruebas de componente para cada historia de usuario, pruebas estáticas y pruebas dinámicas.

Nota:

* Se parte de una base de datos (BD) creada con su respectiva información de catálogos de productos cargada. Se asume que la BD se encuentra funcional y lista para ser integrada con el sitio web.
* Se interpreta la palabra cliente como cualquier persona que ingrese a la pagina web de reqres.in y no solo a las personas que hagan un registro o login.

# 2. Alcance

Se realizarán pruebas automatizadas a las funcionalidades seleccionada durante la planificación de cada sprint usando RestAssured.

**Alcance Sprint 1:** Las pruebas del primer Sprint incluyen la verificación de la siguiente historia de usuario HU001 – Se compran 2 productos por medio de la opción de ofertas.

Criterios de Aceptación 1(CA001):

1. Generar un Get Single<Resource> y un not found.
2. Obtener una respuesta 200 o 404
3. Obtener un body con el item en especifico o sin nada si no se encuentra.

**Alcance Sprint 2:** Las pruebas del primer Sprint incluyen la verificación de la siguiente historia de usuario HU002 – Se compran 2 productos por medio de la opción de Outlet.

Criterios de Aceptación 2(CA002):

1. Validar el botón de Patch
2. Obtener la respuesta 200
3. Obtener un body con el cambio realizado al usuario

## 2.1 No está en el alcance

1. Otras funciones diferentes a las estipuladas en las historias de usuario.
2. base de datos

# 3. Roles y Responsabilidades

|  |  |
| --- | --- |
| **Roles** | **Responsabilidades** |
| Manager de QA | Verificar que lo realizado corresponda. |
| Ingeniero QA de Automatización/ Analista QA | Planificación y monitoreo de las pruebas automatizadas  Reporte de Defectos  Reporte de progreso de las pruebas  Diseño e implementación de las pruebas.  Ejecución de las pruebas automatizadas.  Reporte de resultados de las pruebas. |

4. Riesgos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CA001** | **Historia de usuario 1** | **Probabilidad de Ocurrencia**  **(1-4)** | **Impacto**  **(1-4)** | **Riesgo** |
| 1 | Validar un Get Single<Resource> y un not found. | 1 | 2 | 2 |
| 2 | Validar una respuesta 200 o 404 | 2 | 2 | 4 |
| 3 | Validar un body con el item en especifico o sin nada si no se encuentra. | 2 | 3 | 6 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CA002** | **Historia de usuario 1** | **Probabilidad de Ocurrencia**  **(1-4)** | **Impacto**  **(1-4)** | **Riesgo** |
| 1 | Validar el botón de Patch | 1 | 2 | 2 |
| 2 | Validar la respuesta 200 | 2 | 3 | 6 |
| 3 | Validar Obtener un body con el cambio realizado al usuario | 2 | 2 | 4 |

# 5. Diseño de casos de prueba

A continuación, se enuncian los diseños de casos de prueba:

Feature:Obtener un recurso y actualizar datos  
 como un usuario registrado del sistema  
 quiere obtener un recurso del sistema y luego hacer un cambio  
 para poder saber si el color que busco se encuentra y luego cambiar mi informacion  
  
 Scenario Outline:Obtener un recurso  
 Given el usuario esta en la pagina de busqueda y desea buscar en la lista de recursos  
 When el usuario busca el recurso que desea y obtiene una respuesta **<respuesta>**  
Then el usuario debera ver la data del recurso como su nombre **<name>** o no ver nada si el recurso no existe  
 Examples:  
 |**respuesta**|**name**|  
 |**HttpStatus.SC\_OK**|**fuchsia rose**|  
 |**HttpStatus.SC\_NOT\_FOUND**|**null**|

Scenario: Actualizar datos  
 Given el usuario esta en la pagina de update de usuarios  
 When el usuario escribe el nombre **"morpheus"** y cambia su cargo **"zion resident"**  
Then el usuario obtiene una respuesta y ve sus nuevos datos

# 6. Ambiente y Herramientas de Pruebas

## 6.1 Herramientas de Pruebas

|  |  |
| --- | --- |
| **Herramienta** | **Función** |
| Rest Assured | Prueba y valida el test services |
| Hamcrest | Es un marco que ayuda a escribir pruebas de software en el lenguaje de programación |
| 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 |
| Cucumber | Gestor de features en leguaje Gherkin |

## 6.2 Ambiente de Pruebas

|  |  |
| --- | --- |
| Navegadores | Chrome, Mozilla Firefox, Edge |
| Sistemas Operativos | Windows |

# 7. Criterios de Entrada y Salida

## 7.1 Criterios de Entrada

* Las funcionalidades deben estar desplegadas en el ambiente de QA y haber sido probadas manualmente.
* El framework de pruebas está instalado y listo para la ejecución
* El ambiente de QA está disponible.

## 7.2 Criterios de Salida

* Ejecución de todos los casos de pruebas automatizados ya mencionados
* Se ha logrado la suficiente cobertura de los requerimientos y funcionalidades bajo pruebas
* Ningún defecto de severidad alta se encuentra abierto.

# 8. Planificación de ejecución de las pruebas

Lista de funcionalidades a ser automatizadas por Sprint

|  |  |  |
| --- | --- | --- |
| Sprint | Funcionalidades | Comentarios |
| 1 | Seleccionar dos tipos de productos desde las ofertas, agregarlos al carro y llenar la información para realizar el pedido | Ya que no se tiene acceso a la base de datos no es posible saber las cantidades de los productos en existencia y que no se realizan compras verdaderas |
| 2 | Seleccionar dos tipos de productos desde el outlet, agregarlos al carro y llenar la información para realizar el pedido | Ya que no se tiene acceso a la base de datos no es posible saber las cantidades de los productos en existencia y que no se realizan compras verdaderas |

Es necesario que las funcionalidades a automatizar se desarrollen, implementen y prueben manualmente para que tengan un nivel determinado de estabilidad cuando comienzan las tareas de automatización.

# 9. Reporte de Pruebas

El Reporte automático de pruebas se obtendrá a través de Cucumber. Este Reporte informará sobre los resultados de la ejecución de la automatización. Incluirá las pruebas que pasaron y las que fallaron, los errores encontrados, la tasa de éxito y el tiempo transcurrido.

