

Reto: Rest Assured

Estrategia de Automatización de Pruebas

Por:

Hansee Jiménez Pérez

**Historia de revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Autor(es) | Descripción | Fecha |
| 1.0 | Hansee Jimenez | Creación del documento | 21 de Marzo 2022 |

Tabla de contenido

[1.Introducción 4](#_Toc98787208)

[2. Alcance 5](#_Toc98787209)

[3. Roles y Responsabilidades 5](#_Toc98787210)

[4. Análisis Riesgos 6](#_Toc98787211)

[5. Ambiente y Herramientas de Pruebas 6](#_Toc98787212)

[5.1 Herramientas de Pruebas 6](#_Toc98787213)

[5.2 Ambiente de Pruebas 6](#_Toc98787214)

[6. Criterios de Entrada y Salida 7](#_Toc98787215)

[6.1 Criterios de Entrada 7](#_Toc98787216)

[6.2 Criterios de Salida 7](#_Toc98787217)

[7. Planificación de ejecución de las pruebas 7](#_Toc98787218)

[8. Reporte de Pruebas 8](#_Toc98787219)

# 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, las herramientas a utilizar, los riesgos y la planificación de ejecución de las pruebas para el **Sitio Web** **REQ | RES**  (https://reqres.in/).

# 2. Alcance

Se realizarán pruebas de caja negra (automatizadas) a dos de los endpoint, de diferente operación Http, de la Api que ofrece la página y verificar que las peticiones solicitadas contengan el estado y la información esperada.

Las funcionalidades a ser automatizadas serán seleccionadas con base a los endpoint elegidos:

* GET - Single User.
* POST – Register Successful.

# 3. Roles y Responsabilidades

|  |  |
| --- | --- |
| Roles | Responsabilidades |
| Coach de QA | Monitoreo de las pruebas automatizadas  Reporte de progreso de las pruebas |
| Estudiante QA de Automatización | Diseño e implementación de las pruebas.  Ejecución de las pruebas automatizadas.  Reporte de resultados de las pruebas. |

# 4. Análisis Riesgos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Riesgos** | **Probabilidad de Ocurrencia**  **(1-5)** | **Impacto**  **(1-5)** | **Criticidad**  **(Prob\*Impacto)** |
| 1 | El usuario se registra de manera exitosa | 1 | 5 | 5 |
| 2 | Se realiza la petición de la información de un usuario | 1 | 3 | 3 |

# 5. Ambiente y Herramientas de Pruebas

## 5.1 Herramientas de Pruebas

|  |  |
| --- | --- |
| Herramienta | Función |
| REST Assured | Testeo y validación de servicios web |
| JUnit Framework | Ejecución de pruebas |
| Gradle | Creación de la estructura de proyectos y uso e importación de librerías |
| Cucumber Framework | Implementar BDD y facilitar la automatización de pruebas |
| SerenityBDD Framework | Generación de reportes y documentación viva |
| Chromedriver | Crea una instancia del navegador Chrome |

## 5.2 Ambiente de Pruebas

|  |  |
| --- | --- |
| **Navegadores** | Chrome |
| **Sistemas Operativos** | Windows |

# 6. Criterios de Entrada y Salida

## 6.1 Criterios de Entrada

La página web de REQ | RES debe estar completamente funcional y se debe poder tener acceso a ella.

Tener todas las herramientas (ambiente) de automatización instaladas y configuradas correctamente.

Tener información previa del comportamiento de la página y sus endpoint con el tipo de información que transfiere.

## 6.2 Criterios de Salida

Ejecución de al menos dos escenarios de prueba.

Reporte de resultados de las pruebas realizadas.

# 7. Planificación de ejecución de las pruebas

Lista de funcionalidades a ser automatizadas por operación Http

|  |  |
| --- | --- |
| Http | Funcionalidad / Escenario |
| GET | Validacion de usuario existente |
| POST | Registro de usuario existoso |

Es necesario que las funcionalidades a automatizar se desarrollen, implementen y prueben manualmente para tener conocimiento de las respuestas e información que se entrega.

# 8. Reporte de Pruebas

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

A continuación se evidencia una captura del reporte generado por SerenityBDD, sin embargo, para poder explorar todos los detalles del reporte se recomienda acceder a GitHub y clonar el repositorio, seguido acceder a la ruta **\target\site\serenity** y abrir en el navegador el archivo **index.html**.

