
DataFlex Web Service for Country information

Proyecto: Servicios SOAP (Post)

<http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso>

Estrategia de Prueba Automatizadas

Historia de revisiones

Versión	Autor	Descripción	Fecha
1.0	Juan Esteban Pineda Angel	Creación del documento	Enero 2022

Tabla de Contenidos

[1.Introducción](#)

[2. Alcance](#)

[3. Roles y Responsabilidades](#)

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

[5.1 Herramientas de Pruebas](#)

[5.2 Arquitectura del framework de automatización](#)

[5.3 Ambiente de Pruebas](#)

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

[6.1 Criterios de Entrada](#)

[6.2 Criterios de Salida](#)

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

[7.1 Planificación de las Pruebas de Regresión](#)

[8. Reporte de Pruebas](#)

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, planes de contingencia y el calendario de ejecución de las pruebas de servicios Soap Post usados para obtener el nombre de un país (country Name) y para obtener la ciudad capital de un país (capital city).

Estos servicios inicialmente son evaluados en SoapUI con el fin de explorar su comportamiento ante diferentes estímulos.

Se plantea la selección de dos escenarios en cada servicio uno exitoso y otro con información de búsqueda incorrecta.

2. Alcance

Se realizarán pruebas de caja negra (automatizadas) a las funcionalidades seleccionadas durante el primer sprint el cual tiene una duración de una semana .

Para probar la historia de usuario se plantea en lenguaje Gherkin

Primera historia de usuario

Feature: Nombre de la ciudad capital

Como usuario de un servicio de búsqueda de capitales de países
necesito validar que la funcionalidad búsqueda por código de país para el llenado
de datos que se requieren para comprar productos internacionales

@busquedacapitalcorrecta

Scenario: búsqueda correcta capital

Given que el usuario quiera buscar el indicativo "CO"

When el usuario hace la petición de búsqueda

Then el usuario debería obtener como capital "Bogotá"

@busquedacapitalIncorrecta

Scenario: búsqueda incorrecta capital

Given que el usuario quiera buscar un indicativo que no concuerda "Co"

When el usuario hace la petición de búsqueda de la capital

Then el usuario debería obtener como resultado "Country not found in the database"

Segunda historia de usuario

Feature: Nombre del país

Como usuario de un servicio de búsqueda de nombres de países
necesito validar que la funcionalidad búsqueda por código de país

@busquedaCountryCorrecta

Scenario: búsqueda correcta country

Given que el usuario quiera buscar con el código "CO"

When el usuario realiza la petición de búsqueda

Then el usuario debería obtener como nombre de país "Colombia"

@busquedaCountryIncorrecta

Scenario: búsqueda incorrecta country

Given que el usuario quiera buscar un indicativo que no es válido "Co"

When el usuario hace la petición de búsqueda del país

Then el usuario debería obtener el resultado para este "Country not found in the database"

3. Roles y Responsabilidades

Roles	Responsabilidades
Manager de QA	Planificación y monitoreo de las pruebas automatizadas Reporte de Defectos Reporte de progreso de las pruebas
Ingeniero QA de Automatización/ Analista QA	Diseño e implementación de las pruebas. Ejecución de las pruebas automatizadas. Reporte de resultados de las pruebas.
Product Owner/Stakeholders	Toma de decisiones

4. Riesgos y Planes de Contingencia

S001	Historia de usuario 1	Probabilidad de Ocurrencia (1-5)	Impacto (1-5)	Riesgo
1	Verificar la disponibilidad del servicio búsqueda de capitales.	2	3	6
2	Verificar que se pueda ingresar el código correcto.	3	1	3
3	Verificar la petición	2	2	4
4	Verificar el código de respuesta "200"	1	4	4
5	Verificar el resultado válido.	2	4	4

Tabla 1 Análisis de riesgo Gherkin 1

S002	Historia de usuario 1	Probabilidad de Ocurrencia (1-5)	Impacto (1-5)	Riesgo
1	Verificar la disponibilidad del servicio búsqueda de capitales.	2	3	6
2	Verificar que se pueda ingresar el código incorrecto.	3	1	3
3	Verificar la petición	2	2	4
4	Verificar el código de respuesta "200"	1	4	4
5	Verificar el mensaje de resultado no válido.	2	4	4

Tabla 2 Análisis de riesgo Gherkin 1

S001	Historia de usuario 1	Probabilidad de Ocurrencia (1-5)	Impacto (1-5)	Riesgo
1	Verificar la disponibilidad del servicio búsqueda de países.	2	3	6
2	Verificar que se pueda ingresar el código país correcto.	3	1	3
3	Verificar la petición	2	2	4
4	Verificar el código de respuesta "200"	1	4	4
5	Verificar el resultado válido.	2	4	4

Tabla 1 Análisis de riesgo Gherkin 2

S002	Historia de usuario 1	Probabilidad de Ocurrencia (1-5)	Impacto (1-5)	Riesgo
1	Verificar la disponibilidad del servicio búsqueda de países.	2	3	6
2	Verificar que se pueda ingresar el código incorrecto.	3	1	3
3	Verificar la petición	2	2	4
4	Verificar el código de respuesta "200"	1	4	4
5	Verificar el mensaje de resultado no válido.	2	4	4

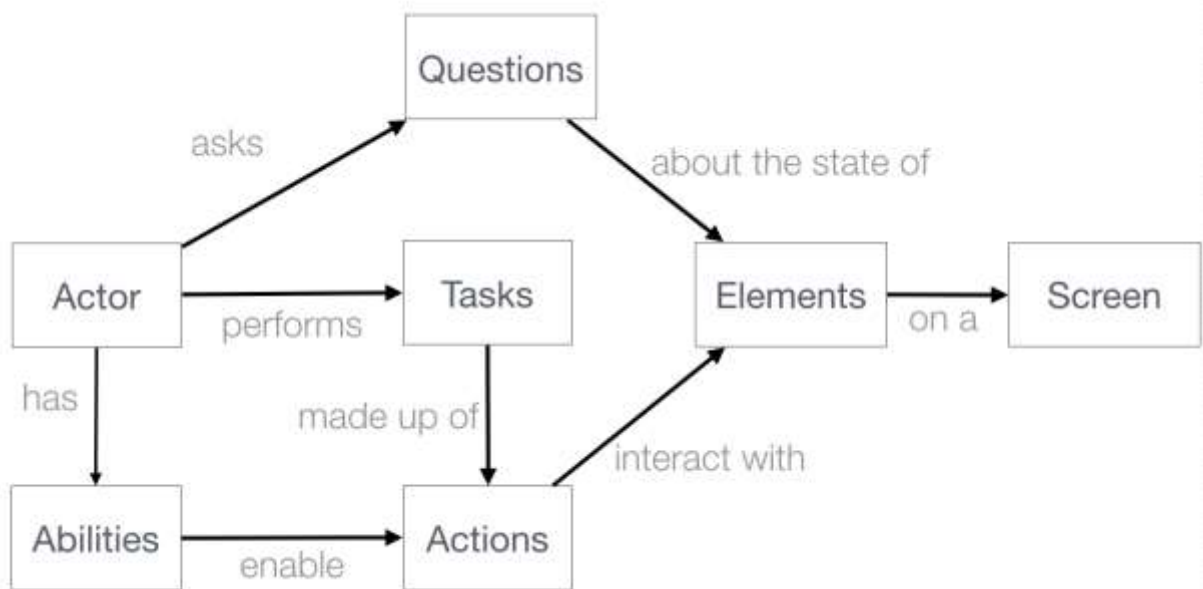
Tabla 2 Análisis de riesgo Gherkin 2

5. Ambiente y Herramientas de Pruebas

5.1 Herramientas de Pruebas

Herramienta	Function
Serenity BDD	API para automatizar peticiones de servicios con patrón ScreenPlay
JUnit testing framework	Ejecución y Reporte de las pruebas
Gradle	Creación de la estructura de proyectos y uso e importación de librerías
Hamcrest	Comparaciones personalizadas
Cucumber	Gestor de Historias de usuario en formato Gherkin
LOG4J	Impresión de mensajes por consola

5.2 Arquitectura del patrón de diseño.



Arquitectura ScreenPlay

Para hacer la verificación de servicios soap se emplea como cliente la aplicación creada en IntelliJ en lenguaje Java, la cual emplea Serenity BDD para la gestión de las peticiones en post de ambos features de Gherkin.

5.3 Ambiente de Pruebas

Resultados	Consola de IntelliJ
Sistemas Operativos	Windows

6. Criterios de Entrada y Salida

6.1 Criterios de Entrada

Los servicios deben de estar desplegados en el dominio de la URL y se debe conocer los mensajes esperados por las peticiones, así como sus códigos de respuesta esperados.

El framework de pruebas está instalado y listo para la ejecución

El ambiente de QA está disponible.

Los defectos críticos encontrados durante las pruebas manuales han sido resueltos y cerrados.

6.2 Criterios de Salida

Ejecución de todos los casos de pruebas automatizados

Se ha logrado la suficiente cobertura de los requerimientos y funcionalidades bajo pruebas

Ningún defecto de severidad alta se encuentra abierto.

7. Planificación de ejecución de las pruebas

Lista de funcionalidades a ser automatizadas por Sprint

Sprint 1	Funcionalidades	Comentarios
1 POST	Post para la búsqueda de capitales	
2 POST	Post para la búsqueda de países	

Las pruebas de automatización normalmente comenzarán finalizando la primera semana del Sprint.

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.

Planificación de Pruebas Automatizadas

7.1 Planificación de las Pruebas de Regresión

Las suites de regresión se ejecutarán al final de cada Sprint (antes de la Revisión del Sprint), al realizarse un cambio o por solicitud de los Clientes, Product Owner y Project Manager.

8. Reporte de Pruebas

El Reporte automático de pruebas se obtendrá a través de Serenity, 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 (documentación viva).