**Proceso de inmersión GreenSQA para Automatización de pruebas**

**Responsable: Alexander Moreano Ramirez**

Haces parte del equipo GreenSQA para el cliente Latam: https://www.latamairlines.com/co/es, en donde quieren realizar una prueba de concepto para una implementación Shift Right del servicio de aseguramiento de calidad y te realizan unas preguntas y unas solicitudes para los siguientes

**módulos: Modulo Diseño:**

¿Cómo plantearías los casos de pruebas unitarias para un componente de la página de Latam y qué herramienta utilizaría para generar dichos casos? (Caja Blanca)

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Prueba** | **Descripción** | **Datos de Entrada** | **Resultado Esperado** |
| **Caso 1 (Ida y Regreso)** | Ingresar ciudad de origen y destino válidas, fecha de ida y regreso válidas | Ciudad de origen: [Origen]  Ciudad de destino: [Destino]  Fecha de ida: [FechaIda]  Fecha de regreso: [FechaRegreso] | Mostrar resultados de vuelos de ida y vuelta. |
| **Caso 2 (Ida y Regreso con Fecha de Regreso Inválida)** | Ingresar ciudad de origen y destino válidas, fecha de ida válida y fecha de regreso inválida | Ciudad de origen: [Origen]  Ciudad de destino: [Destino]  Fecha de ida: [FechaIda] Fecha de regreso: [FechaRegreso] | Mostrar resultados de vuelos de ida y mensaje de error para la fecha de regreso. |
| **Caso 3 (Ida y Regreso sin Fecha de Regreso)** | Ingresar ciudad de origen y destino válidas, fecha de ida válida, pero sin fecha de regreso | Ciudad de origen: [Origen]  Ciudad de destino: [Destino]  Fecha de ida: [FechaIda]  Fecha de regreso: (vacío) | Mostrar resultados de vuelos de ida y mensaje que indique que se necesita una fecha de regreso. |
| **Caso 4 (Ida y Regreso con Fecha de Regreso en el Pasado)** | Ingresar ciudad de origen y destino válidas, fecha de ida válida y fecha de regreso en el pasado | Ciudad de origen: [Origen]  Ciudad de destino: [Destino]  Fecha de ida: [FechaIda]  Fecha de regreso: [FechaRegreso] | Mostrar un mensaje de error indicando que la fecha de regreso es inválida. |
| **Caso 5 (Ida y Negativo - Destino Inválido)** | Ingresar ciudad de origen válida y ciudad de destino inválida, fecha de ida y regreso válidas | Ciudad de origen: [Origen]  Ciudad de destino: [Destino]  Fecha de ida: [FechaIda]  Fecha de regreso: [FechaRegreso] | Mostrar un mensaje de error indicando que la ciudad de destino es inválida. |

Herramienta para implementar las pruebas unitarias:

Unittest: Para aplicaciones en Python.

Jest: Para web desarrolladas en REACT

¿Cómo plantearías los casos de pruebas de integración para un componente de la página de Latam y qué herramienta utilizaría para generar dichos casos? (Caja Blanca)

Además, el cliente te solicita el diseño de 3 casos de pruebas de UI de la página de Latam. Utiliza la metodología de diseño de pruebas apropiada para este tipo de pruebas, considerando que posteriormente podrían solicitar que sean automatizadas.

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de Prueba de UI** | **Descripción** | **Pasos a Seguir** | **Resultado Esperado** |
| Caso de Prueba 1: Búsqueda de Vuelos | Verificar que el usuario pueda realizar una búsqueda de vuelos de ida y regreso en la página de inicio. | 1. Abrir el navegador y navegar a la página de inicio de Latam.  2. Ingresar una ciudad de origen válida en el campo correspondiente.  3. Ingresar una ciudad de destino válida en el campo correspondiente.  4. Seleccionar fechas de ida y regreso válidas en los campos de fecha.  5. Hacer clic en el botón de "Buscar Vuelos". | Se deben mostrar los resultados de vuelos disponibles para las ciudades y fechas especificadas. |
| Caso de Prueba 2: Filtrar Resultados de Búsqueda | Verificar que el usuario pueda aplicar filtros a los resultados de búsqueda de vuelos. | 1. Realizar una búsqueda de vuelos según el Caso de Prueba 1.  2. En la página de resultados, aplicar un filtro de "Clase de Vuelo" seleccionando una opción (por ejemplo, "Económica").  3. Aplicar un filtro de "Aerolínea" seleccionando una aerolínea específica. | Los resultados de la búsqueda deben actualizarse y mostrar solo los vuelos que cumplen con los filtros seleccionados. |
| Caso de Prueba 3: Reservar un Vuelo | Verificar que el usuario pueda seleccionar y reservar un vuelo. | 1. Realizar una búsqueda de vuelos según el Caso de Prueba 1.  2. En la página de resultados, seleccionar un vuelo específico.  3. Ingresar los detalles del pasajero (nombre, apellido, etc.).  4. Hacer clic en el botón de "Reservar". | El usuario debe ser redirigido a una página de confirmación de reserva que muestra los detalles del vuelo y del pasajero. |

**Modulo Arquitectura:**

Una vez que se tiene los diseños de pruebas, un compañero tuyo realizó la implementación de las pruebas unitarias e integración. Ahora el cliente tiene las siguientes preguntas:

1. ¿Dónde debo dejar la implementación de estas pruebas desarrolladas, en el código de desarrollo o un proyecto aparte?

Para pruebas implementación de pruebas unitarias e integración, también conocido integración continua debe ir en el mismo repositorio del código, pero en una carpeta especificado como test en el código del desarrollo, Porque es necesario ya que facilita una colaboración entre el código desarrollado y el grupo de trabajo.

Ya para pruebas de UA se puede hacer en un proyecto independiente esto aplica si tiene avanzado el proyecto y hay despliegue del proyecto desarrollado para entrar a interactuar con la aplicación.

1. ¿Cuáles son los patrones de arquitectura de pruebas que implementarías para los diseños de pruebas de UI que planteaste anteriormente?

Patrones que utilizaría para las pruebas y que interprete lenguaje natural para facilitar las pruebas serian

BDD-Cucumber o SpecFlow

Screenplay Pattern-Serenity

1. Además, el cliente te solicita realizar un diagrama implementando y señalando los anteriores patrones de arquitectura. Tomar uno de los casos planteados en la primera pregunta. (Puedes utilizar cualquier notación que te sientas cómodo).

**Funcionalidad: Búsqueda de Vuelos**

**Escenario:** Búsqueda exitosa de vuelos de ida y regreso

Dado que el usuario se encuentra en la página de inicio de Latam

Cuando el usuario ingresa la ciudad de origen "Santiago" y la ciudad de destino "Lima"

Y selecciona la fecha de ida "2023-09-15" y la fecha de regreso "2023-09-20"

Y hace clic en el botón "Buscar Vuelos"

Entonces se muestran resultados de vuelos disponibles para las fechas y destinos especificados

**Escenario:** Aplicar filtro de clase de vuelo

Dado que el usuario se encuentra en la página de resultados de búsqueda

Cuando el usuario selecciona la opción "Económica" en el filtro de clase de vuelo

Entonces los resultados se actualizan para mostrar vuelos de clase económica

**Escenario:** Reservar un vuelo

Dado que el usuario se encuentra en la página de resultados de búsqueda

Cuando el usuario selecciona un vuelo específico

Y completa los detalles del pasajero con nombre "Alexander" y otros datos válidos

Y hace clic en el botón "Reservar"

Entonces se redirige a una página de confirmación de reserva con los detalles del vuelo y del pasajero

**Modulo Automatización:**

Acabas de iniciar tu primer sprint en donde tienes las siguientes actividades a desarrollar:

1. Generar la automatización de pruebas de acuerdo con los lineamientos de arquitectura que planteaste en el punto anterior, implementando los 3 casos de pruebas de UI que planteaste en el punto de diseño.

Repositorio:

**Modulo DevOps (CD/CI):**

Una vez desarrolladas las pruebas unitarias, de integración y de UI el cliente te hace las siguientes preguntas:

1. ¿Cuáles herramientas utilizarías para la integración continua y el despliegue continuos de estos casos de pruebas?

Herramientas para:

Integración continua: Jenkins-Cucumber/unittest

Despliegue continuo: Docker y AWS pero para pruebas no lo he utilizado.

1. Plantea la infraestructura de ejecución para las pruebas anteriormente desarrolladas, en lo posible un diagrama. 3. ¿En cuál pipeline debería correr cada una de las pruebas?

4. ¿Hay algún tipo de pruebas que se pueda agregar extra y en cual pipeline debería ir?