

Pruebas de aceptación

Pruebas de aceptación

Antoni R. Toapanta, Milena V. Maldonado y Alexander D. Guacán
Departamento de Ciencias de la Computación
Universidad de las Fuerzas Armadas ESPE
14569: Ingeniería de Requisitos de Software
Ing. Jenny Ruiz
25 de Julio del 2024

Pruebas de aceptación

Resumen

Las pruebas de aceptación verifican que el software cumple con los requisitos del usuario y funciona correctamente. En el caso del formulario de contacto en NeoStudio, se evaluaron tres escenarios: datos correctos, campos incompletos y validación de formato. Mientras que los primeros dos escenarios pasaron satisfactoriamente, la validación de formatos falló, indicando una área crítica a mejorar. Usando Cucumber y Selenium, se confirmaron las funcionalidades correctas en los escenarios aprobados, pero es necesario fortalecer la validación de formatos y realizar pruebas adicionales antes de la implementación final.

Palabras clave: Cucumber, Selenium, pruebas de aceptación.

Pruebas de Aceptación

Las pruebas de aceptación son una fase crítica en el ciclo de vida del desarrollo de software que tiene como objetivo validar que una aplicación cumple con los requisitos y expectativas del usuario final. Estas pruebas se realizan al final del proceso de desarrollo y se centran en verificar si el sistema funciona de acuerdo con los criterios de aceptación previamente definidos. El propósito principal es asegurarse de que el software no solo sea funcional desde una perspectiva técnica, sino que también satisfaga las necesidades y requisitos del negocio y del usuario.

Durante las pruebas de aceptación, se evalúa el comportamiento del sistema en condiciones reales de uso, simulando escenarios típicos y extremos para verificar que todas las funcionalidades trabajen como se espera. Se pueden realizar de manera manual o automatizada, dependiendo de la complejidad y el alcance del proyecto. Los casos de prueba se diseñan en función de los requisitos del cliente, y la aceptación del software se basa en si cumple con estos requisitos sin defectos críticos ni problemas significativos.

Cucumber

Cucumber es una herramienta de prueba de software que facilita la implementación de pruebas basadas en comportamientos (BDD, por sus siglas en inglés). BDD es una metodología de desarrollo de software que promueve la colaboración entre desarrolladores, testers y otros interesados mediante la definición de criterios de aceptación en un lenguaje comprensible para todos los miembros del equipo. Cucumber permite escribir pruebas en un formato de lenguaje natural que puede ser entendido tanto por técnicos como por no técnicos.

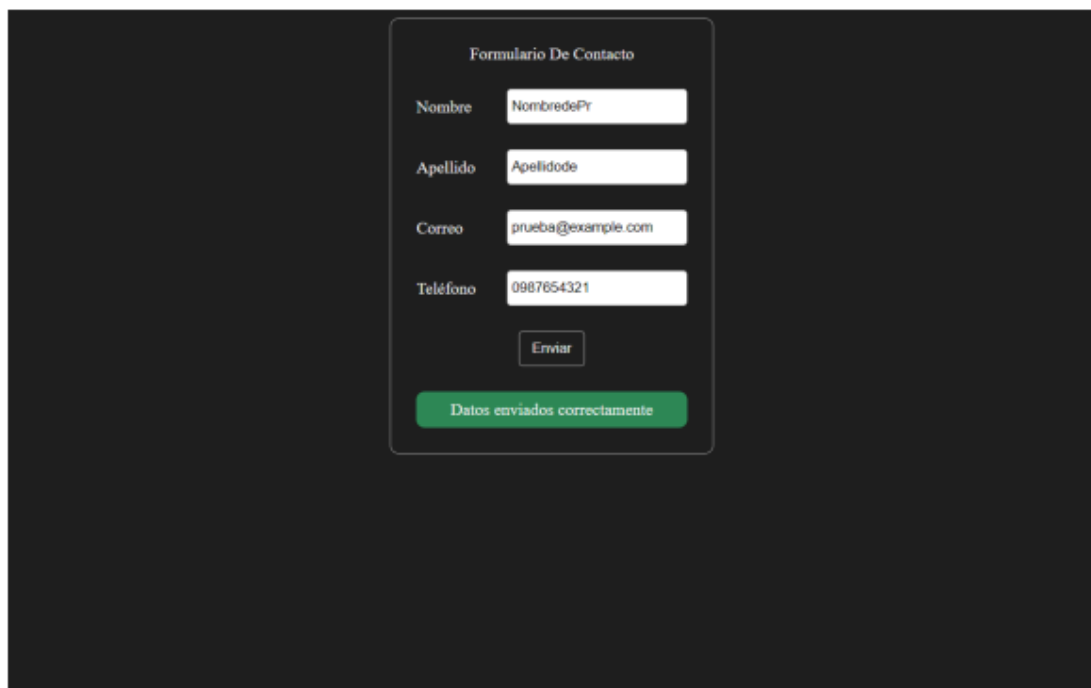
Cucumber utiliza un lenguaje de especificación llamado Gherkin para describir los comportamientos esperados del software en escenarios específicos. Estos escenarios se escriben en un formato legible y estructurado, con frases como "Dado que", "Cuando" y "Entonces", lo que facilita la comprensión de los requisitos y el diseño de pruebas. Por ejemplo, un escenario podría describir cómo un usuario debería ser capaz de iniciar sesión en una aplicación, qué datos debe proporcionar y qué resultados espera obtener.

Una vez que los escenarios están definidos, Cucumber se encarga de ejecutar estas pruebas de aceptación al traducir el lenguaje natural de Gherkin en pasos de prueba automatizados. Los desarrolladores escriben el código necesario para implementar estos pasos, y Cucumber verifica si el comportamiento del software coincide con los criterios descritos en los escenarios. Esto permite detectar errores y garantizar que el software cumpla con los requisitos del negocio desde una perspectiva funcional.

Resultados

Captura de pantalla para el escenario 1

Envio Formulario Correcto



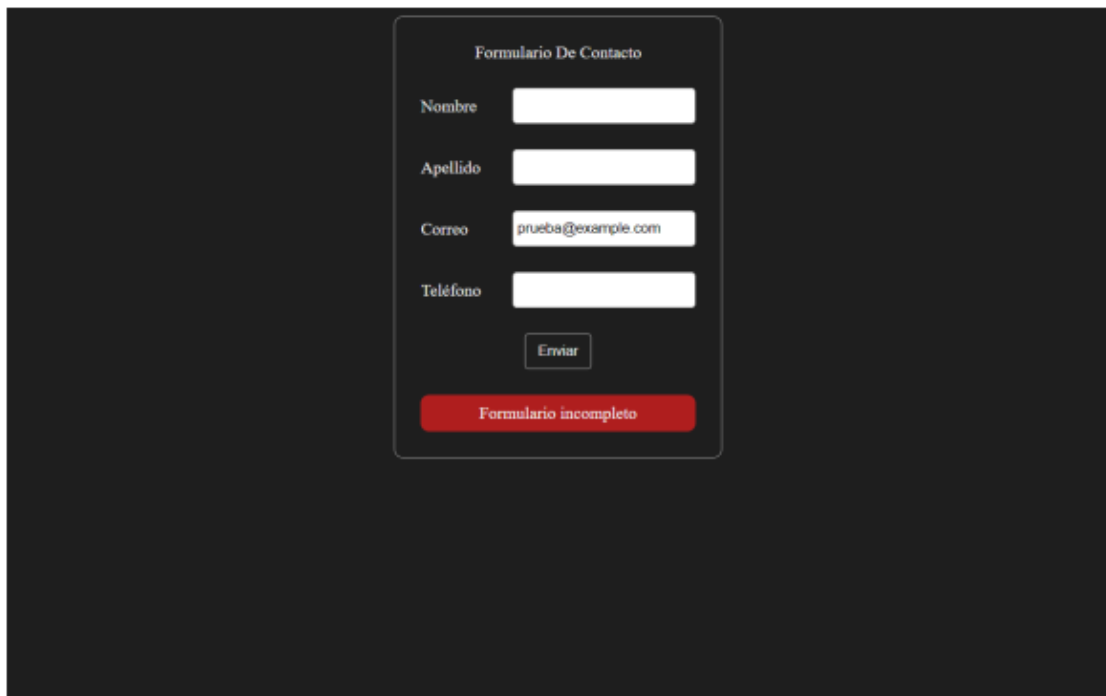
The screenshot shows a dark-themed web interface. In the center, there is a light gray rounded rectangle containing a contact form titled "Formulario De Contacto". The form has four input fields: "Nombre" (containing "NombredesPr"), "Apellido" (containing "Apellidode"), "Correo" (containing "prueba@example.com"), and "Teléfono" (containing "0987654321"). Below these fields is a small "Enviar" button. At the bottom of the form container, a green button displays the message "Datos enviados correctamente".

Formulario De Contacto	
Nombre	NombredesPr
Apellido	Apellidode
Correo	prueba@example.com
Teléfono	0987654321
<input type="button" value="Enviar"/>	
<input type="button" value="Datos enviados correctamente"/>	

Pruebas de aceptación

Captura de pantalla para el escenario 2

Formulario incompleto



Formulario De Contacto

Nombre

Apellido

Correo

Teléfono

Formulario incompleto

Discusión

Pruebas de Aceptación para el Formulario de Contacto de NeoStudio

Contexto de Pruebas: Las pruebas se llevaron a cabo para validar el comportamiento del formulario de contacto de la página web de diseño interior NeoStudio, utilizando Cucumber con Selenium en JavaScript. Se definieron tres escenarios de prueba:

1. **Datos Correctos:** Verificar que el formulario se envíe correctamente cuando todos los campos están completos y los datos son válidos.
2. **Campos Incompletos:** Evaluar el manejo de la aplicación cuando faltan datos en campos obligatorios.
3. **Formato Incorrecto:** Comprobar la validación de campos con formatos incorrectos (por ejemplo, correos electrónicos mal formateados).

Resultados de las Pruebas:

- **Prueba 1 (Datos Correctos):** Aprobada. Los datos ingresados en el formulario fueron procesados correctamente y se enviaron sin problemas. Esta prueba confirma que la funcionalidad principal del formulario está funcionando como se esperaba.
- **Prueba 2 (Campos Incompletos):** Aprobada. El sistema manejó adecuadamente la ausencia de datos en campos obligatorios, mostrando mensajes de error adecuados y evitando el envío del formulario hasta que se completaran todos los campos requeridos.
- **Prueba 3 (Formato Incorrecto):** Fallida. La validación de campos con formato incorrecto no funcionó correctamente. El formulario permitió el envío de datos incorrectos o mal formateados, lo que indica un problema con las validaciones de formato.

Conclusión

La mayoría de las pruebas realizadas fueron satisfactorias, con dos de los tres escenarios aprobados con éxito. El formulario de contacto demostró una sólida capacidad para manejar datos válidos y gestionar campos incompletos, cumpliendo con los requisitos establecidos y asegurando que los datos se envíen correctamente cuando todos los campos están completos. Esta robustez en la funcionalidad es un indicador positivo de que el formulario está bien diseñado para su propósito principal.

No obstante, la prueba de validación de formatos incorrectos reveló una debilidad significativa en el sistema. La incapacidad del formulario para detectar y manejar datos mal formateados sugiere un área crítica que requiere atención inmediata. Esta deficiencia podría llevar a la aceptación de datos incorrectos o inválidos, lo que podría afectar negativamente la calidad de la

Pruebas de aceptación

información recopilada y la experiencia del usuario. Por lo tanto, es crucial abordar esta falla mejorando la lógica de validación para garantizar que todos los datos ingresados cumplan con los formatos requeridos antes del envío del formulario.

Recomendación

Aunque las pruebas automatizadas con Selenium y Cucumber son útiles, se recomienda revisar manualmente los casos que fallaron para entender mejor el problema y ajustar las pruebas automatizadas según sea necesario.

Documentar detalladamente los errores encontrados y los pasos seguidos para reproducirlos. La información del escenario fallido debe ser incluida en el reporte final para permitir una rápida resolución del problema.

Bibliografías:

Terrera, G. (2014, 15 septiembre). Cucumber y las pruebas de aceptación. TestingBaires.
<https://testingbaires.com/cucumber-y-las-pruebas-de-aceptacion-de-usuario/>