

FRAMEWORK PARA LA AUTOMATIZACIÓN DE PRUEBAS DE INTERFAZ DE USUARIO EN EL MÓDULO DE PRODUCTOS Y LISTAS DE PRECIOS DE SALESFORCE

Carlos E. Caballero B.

8 de mayo de 2019

Contenidos

Introducción

- Definición del problema
- Objetivo general
- Objetivos específicos
- Justificación

Conceptos y definiciones

- Automatización
- Criterios de calidad
- Técnicas de prueba

Análisis del Software

- Análisis de valor limite

Casos de Prueba

Desarrollo del framework

Definición del problema

Siendo el módulo de productos y listas de precios componentes participes en la funcionalidad provista por otros módulos de *Salesforce*, un potencial error en este podría tener un efecto en cascada con el potencial de afectar el servicio completo.

Salesforce sigue una política de actualizaciones muy continua, pudiendo existir hasta tres versiones por año, lo que sin una automatización correcta de las pruebas de interfaz de usuario, implicaría un enorme gasto de recursos para la compañía.

Definición del problema

«Garantizar la calidad de los elementos que componen la interfaz de usuario requiere de un proceso de evaluación continuada y eficiente, de forma que los módulos evaluados y sus diferentes versiones contengan la menor cantidad posible de errores.».

Objetivo general

Implementar un *framework* para la automatización de las pruebas de interfaz de usuario en el módulo de Productos y Listas de Precios en *Salesforce*, para garantizar un procedimiento continuo de evaluación y minimizar la cantidad de errores que contiene el software.

Objetivos específicos

- ▶ Formular los casos de prueba necesarios que los módulos de gestión de productos y listas de precios requieran para cubrir los atributos de calidad requeridos.
- ▶ Diseñar e implementar los modelos y bibliotecas de funciones que conforman un *framework* de automatización.
- ▶ Automatizar los casos de prueba de las funciones que componen la interfaz de usuario del módulo de gestión de productos y listas de precios.

Justificación

Con una estrategia de automatización implementada apropiadamente el tiempo de evaluar las nuevas funcionalidades del sistema, además de la evaluación de las funcionalidades antiguas será reducido drásticamente, consiguiendo mejorar la cobertura de las pruebas.

Automatización

Se conoce que en los siguientes escenarios es útil automatizar:

- ▶ Los requerimientos no cambian frecuentemente.
- ▶ Se accede al aplicativo con múltiples y variados usuarios, roles y privilegios.
- ▶ El software es estable respecto a sus pruebas manuales.
- ▶ Se cuenta con el tiempo necesario para automatización en el proyecto.
- ▶ El proyecto sigue o debe seguir estándares estrictos.
- ▶ La complejidad del proyecto es elevada.
- ▶ El proyecto requiere constantes revisiones en algunas de sus características.

Automatización

Los criterios a seguir son:

- ▶ Identificar áreas dentro del software para automatizar.
- ▶ Elegir la herramienta adecuada para la automatización.
- ▶ Escribir las rutinas de prueba.
- ▶ Desarrollar el conjunto de casos de prueba.
- ▶ Ejecutar los casos de prueba.
- ▶ Generar los reportes de resultados.
- ▶ Encontrar los posibles errores o aspectos negativos.

Automatización

Los beneficios de la automatización son:

- ▶ Incremento de la productividad.
- ▶ Ahorro de dinero, en el costo del proyecto.
- ▶ Aumento de la calidad del software.
- ▶ Reducción del tiempo de evaluación del software.
- ▶ Soporte para múltiples aplicativos del software.
- ▶ Aumento de la cobertura de las pruebas.
- ▶ Reducción del trabajo repetitivo.
- ▶ Mejora de la consistencia del producto.

Automatización

Los riesgos de la automatización son:

- ▶ El costo de arranque de la automatización puede llegar a ser muy alto.
- ▶ Debe tenerse en cuenta que la automatización de las pruebas jamás podrá cubrir el 100 % de cobertura del software.
- ▶ No es conveniente automatizar una interfaz de usuario no establecida.
- ▶ Si la aplicación de usuario cambia constantemente, el costo de mantenimiento de las pruebas automatizadas será muy alto.

Capa de interfaz de usuario

Cuando el foco de la evaluación es la interfaz de usuario, se requiere que la mayoría del código y la lógica de negocio este completamente evaluados. El enfoque se centra en simplemente asegurarse de que la propia interfaz de usuario este funcionando correctamente.

Criterios de calidad

Se denomina criterio de calidad a cualquier requerimiento que define lo que el producto debe ser.

Se definieron los siguientes como criterios de calidad fundamentales para el éxito del producto:

- ▶ Usabilidad.
- ▶ Confiabilidad.
- ▶ Compatibilidad.

Técnicas de prueba

- ▶ Pruebas de aceptación.
- ▶ Pruebas funcionales.
- ▶ Pruebas de dominio.
- ▶ Pruebas negativas.

Análisis del Software

Módulos de Productos

- ▶ Enumerar todas las funciones.
- ▶ Enumerar todas las vistas.
- ▶ Enumerar todos los formularios.
- ▶ Análisis de valores limite.

Módulos de Productos

- ▶ Enumerar todas las funciones.
- ▶ Enumerar todas las vistas.
- ▶ Enumerar todos los formularios.
- ▶ Análisis de valores limite.

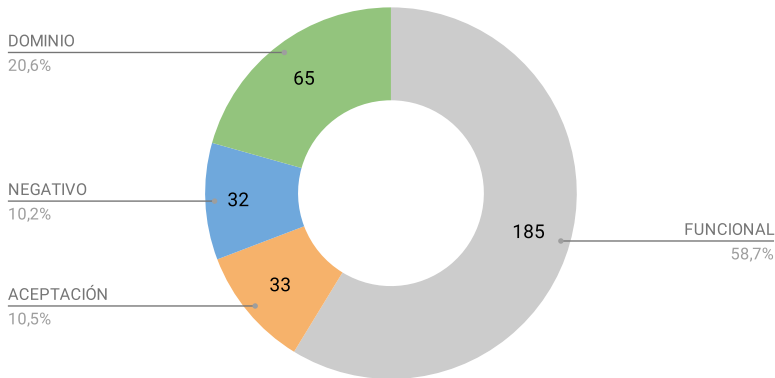
Análisis de valor limite

Variable	Casos Posibles	Casos Inválidos	Límites
Nombre del producto	[1-255] caracteres	0 >255	0 1 255 256
Activo	{verdadero,falso}		
Código de producto	[0-255] caracteres	>255	0 255 256
Familia de productos	ninguno,[lista de valores]		
Programación de cantidades activada	{verdadero,falso}		
Programación de ingresos activada	{verdadero,falso}		
Descripción del producto	[0,4000] caracteres	>4000	0 4000 4001

Casos de Prueba según el tipo de evaluación realizada.

CASOS DE PRUEBA

Según el tipo de evaluación



Casos de Prueba según el tipo de acción a evaluar.

CASOS DE PRUEBA

Según el tipo de acción

LECTURA

5,4%

VALIDACIÓN

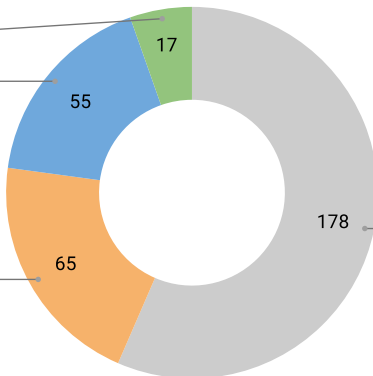
17,5%

ESCRITURA

20,6%

INTERFAZ

56,5%



Page Object

Page Object sirve para mejorar el mantenimiento de las pruebas y reducir la duplicación del código.

Un *Page Object* es una clase orientada a objetos que sirve como la representación de los elementos de una pagina del software a evaluar.

Las pruebas utilizan los métodos de esta clase cuando necesitan interactuar con la interfaz de usuario de la pagina a la que hace referencia.

El beneficio es que si la interfaz de usuario de la pagina cambia, las pruebas en sí no necesitan cambiar, solo el código dentro del *Page Object* necesitará actualizarse.

Diagrama de clases sobre las relaciones entre *Page Objects*.

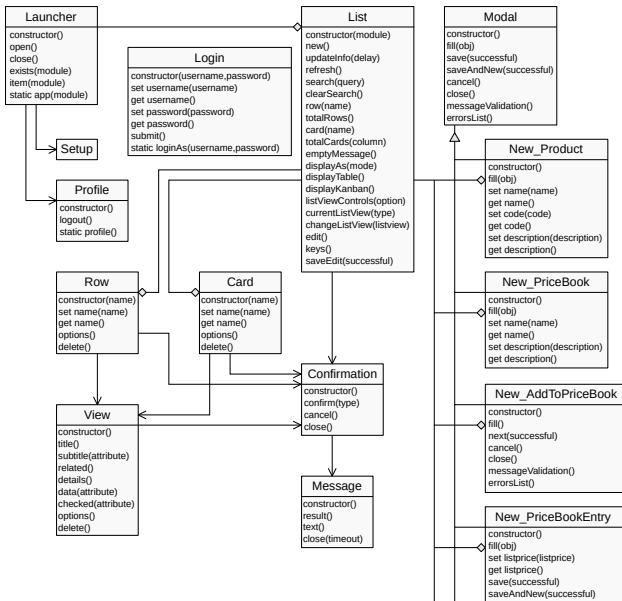
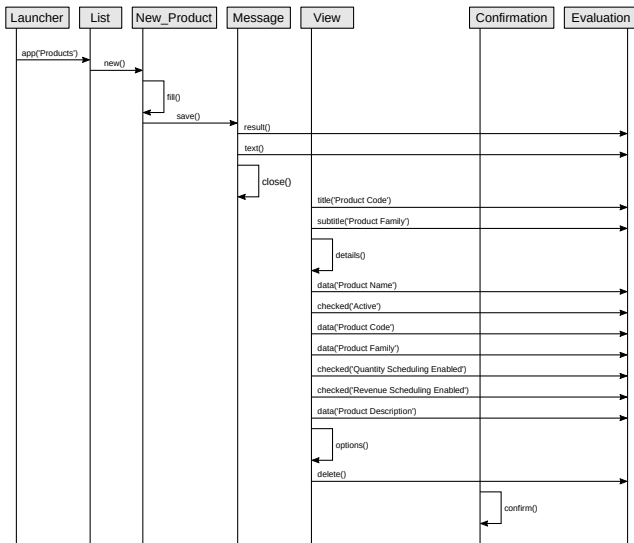


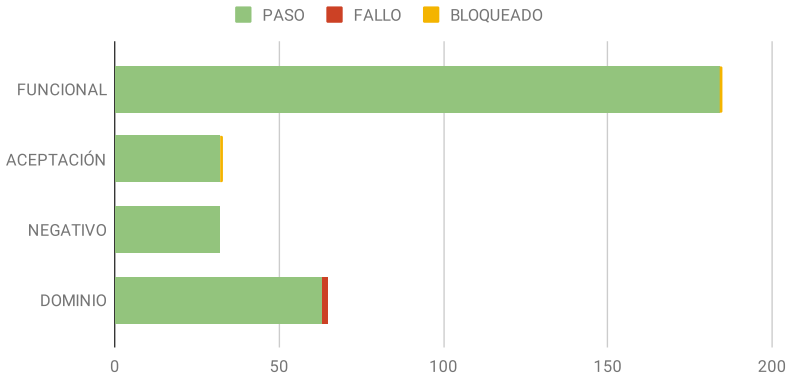
Diagrama de secuencia para el Caso de Prueba A001.



Resultados de las pruebas clasificadas por tipo de evaluación.

RESULTADOS DE LAS PRUEBAS

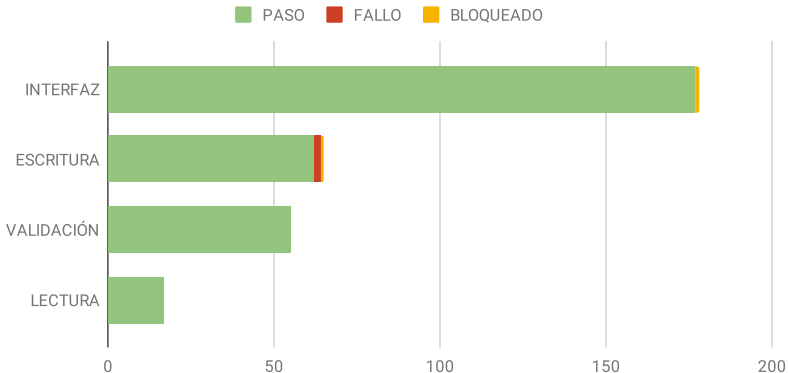
Según el tipo evaluación



Resultados de las pruebas clasificadas por tipo de acción a evaluar.

RESULTADOS DE LAS PRUEBAS

Según el tipo de acción



Casos de prueba que resultaron no exitosos.

Código	Caso de Prueba	Resultado
<i>A007</i>	Precio de producto para una lista de precios es registrado después de accionado el botón «Guardar»	Bloqueado
<i>F041</i>	Mensaje «Se guardó Entrada del catálogo de precios.» se muestra después de modificado un precio	Bloqueado
<i>D017</i>	Formulario «Agregar a lista de precios» permite el registro, cuando en el campo «Lista de precios» se ingresa el valor Nulo	Fallido
<i>D018</i>	Formulario «Agregar a lista de precios» permite el registro, cuando en el campo «Divisa» se ingresa el valor Nulo	Fallido