Logotipo

Descripción generada automáticamente

Proyecto: Para Bank

Estrategia de Prueba Automatizadas

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Autor(es) | Descripción | Fecha |
| 1.0 | Juan Esteban Cardona | Creación del documento | Marzo 2022 |

Tabla de Contenidos

[1.Introducción](#_knb62690yxb3)

[2. Alcance](#_qlpgkz6qysqk)

[3. Roles y Responsabilidades](#_7b58fkv4on7k)

[5. Ambiente y Herramientas de Pruebas](#_d7hpcbc6w3za)

[5.1 Herramientas de Pruebas](#_3dy6vkm)

[5.2 Arquitectura del framework de automatización](#_4d34og8)

[5.3 Ambiente de Pruebas](#_17dp8vu)

[6. Criterios de Entrada y Salida](#_ye1dj1oecscb)

[6.1 Criterios de Entrada](#_26in1rg)

[6.2 Criterios de Salida](#_lnxbz9)

[7. Planificación de ejecución de las pruebas](#_453gv79mohyv)

[7.1 Planificación de las Pruebas de Regresión](#_1ksv4uv)

[8. Reporte de Pruebas](#_44sinio)

# 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 del proyecto Para Bank.

# 2. Alcance

Se realizarán pruebas de caja negra (automatizadas) a las funcionalidades seleccionadas durante la planificación de cada sprint.

Las funcionalidades para automatizar serán seleccionadas utilizando los criterios donde se evidencia rutas críticas de la aplicación prioritarias para el negocio.

# 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. Riegos y Planes de Contingencia

El análisis de riesgos se hace de analizar los criterios de aceptación, asignándoles una nota cuantitativa a cada uno, se hace un promedio para determinar el valor de criticidad de riesgo de la cada feature asociada a los criterios de aceptación.

El resumen del resultado del análisis que fue creado en conjunto con el Product Owner y los Desarrolladores para determinar el nivel de impacto y ocurrencia del respectivo riesgo asociados a los criterios de aceptación.

|  |  |  |
| --- | --- | --- |
| **Valores de criticidad asociado a la historia de Usuario** | | |
| **HU001** | **HU002** | **HU003** |
| 3,50 | 9,00 | 11 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Riesgos | Probabilidad de Ocurrencia  (1-5) | Impacto  (1-5) | Severidad  (Prob\*Impacto) | Plan de Contingencia |
| HU01CP01 | Falla al enviar el formulario de manera correcta | 2 | 2 | 4 | Envió del Bug a Desarrollo |
| HU01CP02 | Envió del formulario no valido | 3 | 1 | 3 | Envió del Bug a Desarrollo |
| HU02CP01 | Falla en el registro correcto del Usuario | 3 | 3 | 9 | Envió del Bug a Desarrollo |
| HU02CP02 | Permitir envió de registro incorrecto | 3 | 3 | 9 | Envió del Bug a Desarrollo |
| HU03CP01 | Falla en ingreso correcto en el Login | 3 | 4 | 12 | Envió del Bug a Desarrollo |
| HU03CP02 | Ingreso al Login sin cuenta y sin Password | 2 | 5 | 10 | Envió del Bug a Desarrollo |

Gráfico

Descripción generada automáticamente

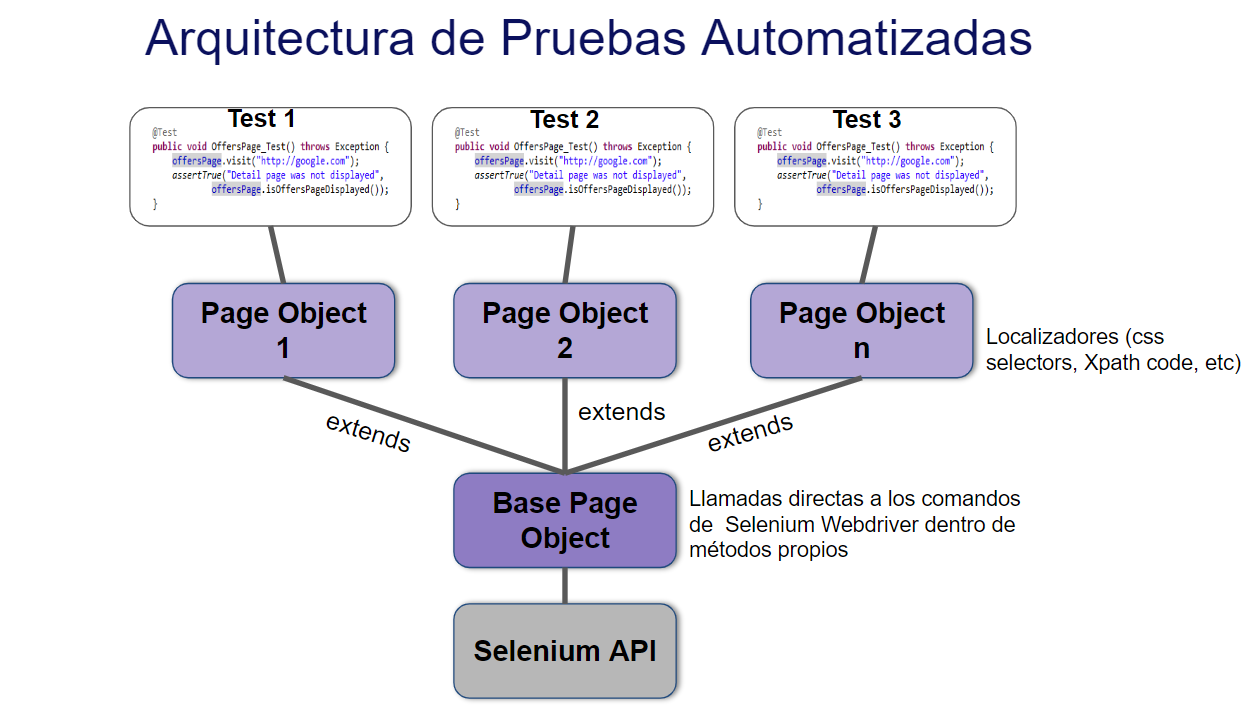
Matriz de Riegos

# 5. Ambiente y Herramientas de Pruebas

## 5.1 Herramientas de Pruebas

|  |  |
| --- | --- |
| Herramienta | Función |
| Selenium WebDriver | API para automatizar sistemas Web |
| Cucumber testing framework | Ejecución y Reporte de las pruebas |
| Gradle | Creación de la estructura de proyectos y uso e importación de librerías |
| Chromedriver | Crea una instancia del navegador Chrome |
| Log4J | Registro de Logs |

## 5.2 Arquitectura del framework de automatización



Utilizaremos el patrón Page Object Model para “mapear” las páginas del sistema a clases “Page” que permitan aislar las acciones de las diferentes páginas y a la vez agrupar todos los webElements de una página y las acciones que se pueden llevar a cabo, en una misma clase.

La clase “Base” permite aislar todo el framework de la versión del API de Selenium WD que estemos utilizando. De esta forma si hay algún cambio en los comandos del API no tenemos que cambiar todas las clases sino solo la clase “Base”.

El Page Object Model también nos ayuda a concentrar los localizadores en estas clases “Page”, de forma que cuando el sistema cambia y es necesario actualizar el código de los css selectors, xpath o lo que hayamos utilizado para localizar los webElements, solo tenemos que cambiarlo una sola vez en la clase “Page” y los “Tests”, que son el último nivel, no necesitan ningún cambio (a menos que haya cambiado la lógica de funcionamiento y dentro de los cambios se hayan eliminado o agregado funcionalidades al sistema).

Adicionalmente se utilizar Page Factory para optimizar la sintaxis al momento de mapear los elementos.

## 5.3 Ambiente de Pruebas

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

# 6. Criterios de Entrada y Salida

## 6.1 Criterios de Entrada

Las funcionalidades deben estar desplegadas en el ambiente de QA y haber sido probadas manualmente.

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.

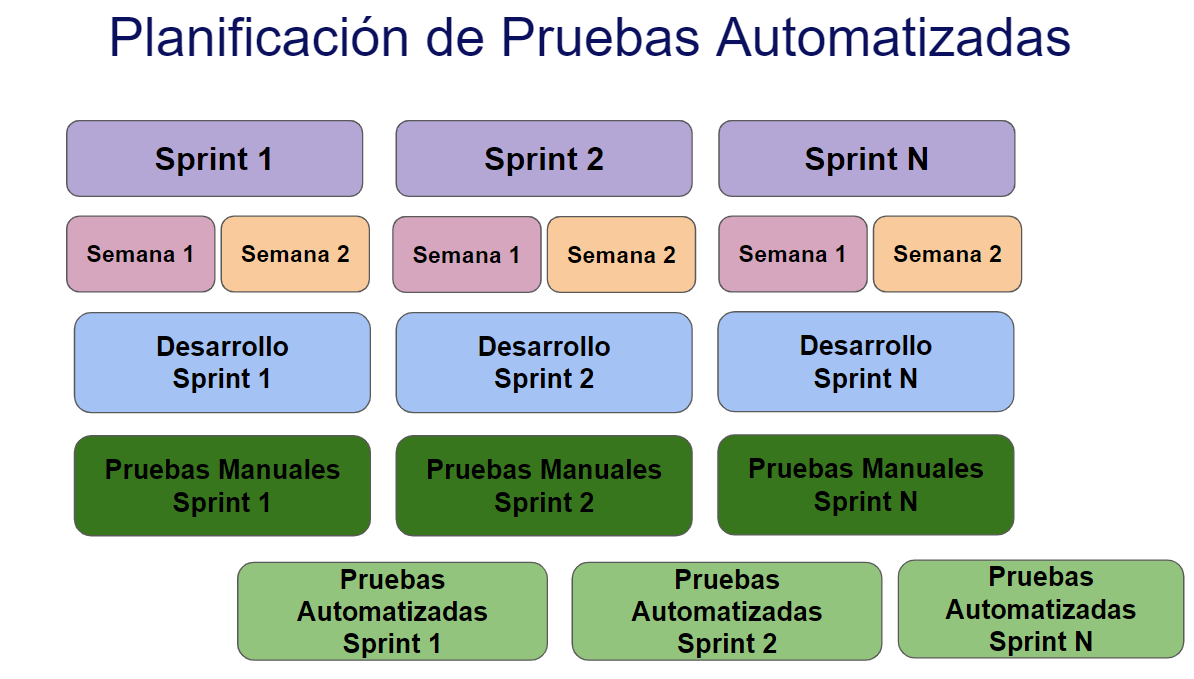
# 7. Planificación de ejecución de las pruebas

Lista de funcionalidades a ser automatizadas por Sprint

|  |  |  |
| --- | --- | --- |
| Sprint número | Funcionalidades | Comentarios |
| 1 | Ingreso a Cuenta de Aplicación |  |
| 2 | Registro de Usuario Nuevo |  |
| 3 | Envió de mensajes a Costumer Care |  |

Las pruebas de automatización normalmente comenzarán en la segunda semana del Sprint (de 2 semanas).

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.



## 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 Cucumber. 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.

Se debe tener en cuenta que para que la prueba de Login de Usuario, este ya debe estar registrado en la base de datos de Para Bank, sino resultara en prueba fallida.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Texto

Descripción generada automáticamente