1. Ambiente y herramientas de pruebas

1.1 Ambiente de pruebas

|  |  |
| --- | --- |
| Navegador(es) | Chrome |
| Sistemas operativos | Windows |

* 1. Herramientas de prueba

|  |  |
| --- | --- |
| Cucumber BDD | metodología de desarrollo la cual se basa en Desarrollo Guiado por Comportamiento |
| Lenguaje gherkin | formato para escribir nuestros casos de prueba de forma simple y clara |
| Selenium webDriver | api la cual para automatizar sistemas web(multiplataforma y multilenguaje ) |
| Jdk | java development kit |
| Gradle | Creación de la estructura del proyecto y uso e importación de librerías |
| Junit Testing framework | ejecución y reporte de pruebas |
| Screenplay | patron de diseño para estructurar nuestras pruebas de software logrando que sean más organizadas y mantenibles a largo plazo |
| Chrome Driver | Controlador para el navegador de Chrome |

serenity BDD - ejecución y reporte de pruebas

Lenguaje Gherkin – formato para escribir nuestros casos de prueba de forma simple y clara

Selenium webDriver – api la cual para automatizar sistemas web (multiplataforma y multilenguaje)

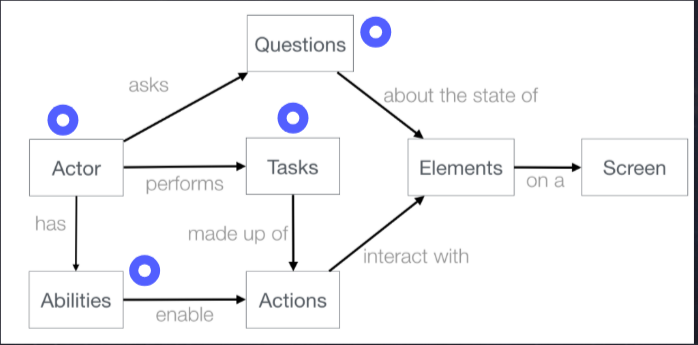
Jdk – java development kit

Gradle - Creación de la estructura del proyecto y uso e importación de librerías

Screenplay – patron de diseño para estructurar nuestras pruebas de software logrando que sean mas organizadas y mantenibles a largo plazo

* 1. Infraestructura

Este plan de pruebas esta pensado única y exclusivamente bajo el patron de diseño screenplay, el cual pasare a explicar mas adelante para tener mas claro como van a suceder las pruebas automatizadas



El patron de diseño screenplay se compone de:

* Actores : son básicamente la base de la automatización ya que sin ellos no podemos interactuar con el sistema
* Abilities : representan las habilidades que tiene el usuario – Aclaración: no son las acciones

Ejemplo: si queremos que una persona corra, tenemos que darle la habilidad de caminar, así mismo funcionan los actores

* Actions: representan las acciones que el actor puede realizar, todas estas acciones están basadas en las habilidades que este cuenta
* Tasks: representan un conjunto de acciones que un actor debe realizar para cumplir con la automatización

Ejemplo: si queremos que una persona gane un maraton(Task) debemos indicarle las acciones que debe realizar, por ejemplo como debe respirar, como trotar, como mantener un ritmo, etc.

* Questions: estas nos ayudan a saber si nuestra automatización se ejecuto tal y como lo describimos en el feature, mediante “preguntas”

Ejemplo: imagina que tienes que sacar 100 en un examen, las questions son aquellas que buscan que calificación sacaste en tu examen y comparan si el resultado es el esperado

* Elementos: representan los diferentes componentes de una pagina tales como botón, campos de una lista, etc.
* Screen: página web

1. Datos

Modulo: Login

* Numero de cedula: 1021394040

Descripción: cedula valida

Valor: 1021394040

* contraseña: 1234765@

Descripción: contraseña valida

Valor: 1234765@

* correo electrónico: juadsalinas111@misena.edu.co

Descripción: correo valido

valor : juadsalinas111@misena.edu.co

* Numero de cedula: 9020393939

Descripción: cedula invalida

Valor: 9020393939

* contraseña: ‘7’7’7’

Descripción: contraseña invalida

Valor: 1234765@

* correo electrónico: juads11@

Descripción: correo invalido

valor : juads11@

Modulo de Account :

* nombre: juan

Descripción: nombre valido

valor : juan

* apellidos: salinas

Descripción: apellidos valido

valor : salinas

* genero: masculino

Descripción: genero valido

valor : masculino

* pais: colombia

Descripción: pais valido

Valor: colombia

* Departamento: Distrito capital de Bogotá

Descripción: Departamento valido

Valor: :Distrito capital de Bogotá

* ciudad: bogota

Descripción: ciudad valida

Valor: bogota

* código postal: 110101

Descripción: código postal valido

Valor: 110101

* contraseña actual: pan23402@

Descripción: contraseña actual valida

Valor: pan23402@

* contraseña nueva: ebli8@

Descripción: contraseña nueva valida

Valor: ebli8@

Modulo: Bonos de regalo

* precio bonos de regalo: 50000

Descripción: precio bonos de regalo valido

Valor: 50000

* precio bonos de regalo: 500000

Descripción: precio bonos de regalo invalido

Valor: 500000

Modulo: PQRS

* nombre Usuario: juanD

Descripción: nombre de usuario valido

Valor: juanD

* clave: jsnsena123

Descripción: clave valida

Valor: jsnsena123

Modulo: Zapatos

* talla: 30

Descripción: talla invalida

Valor: 30

* talla: 40

Descripción: talla valida

Valor: 30

* color: morado

Descripción: color invalido

Valor: morado

* talla: verde aguamarina

Descripción: color valido

Valor: verde aguamarina

Modulo: Cinturones

* talla: 2

Descripción: talla invalida

Valor: 2

* talla:10

Descripción: talla valida

Valor: 10

* color: morado

Descripción: color invalido

Valor: morado

* talla: verde aguamarina

Descripción: color valido

Valor: verde aguamarina

1. Supuestos

* Supuestos de infraestructura
  + Se supone que el entorno de desarrollo se encontrara configurado correctamente bajo las especificaciones necesarias
* Supuestos de datos
  + Se supone que los datos de prueba no cambiaran en ningún momento durante la ejecución de las pruebas
* Supuestos de acceso
  + Se supone que los usuarios tendrán habilitada cada parte del sistema para la correcta ejecución de las pruebas
  + Se supone que antes de cada automatización dependiendo del caso se ajustara el usuario para el reajuste de permisos
* Supuestos de condiciones previas
  + Se supone que previamente antes de la ejecución de las pruebas se tendrán todos las herramientas instaladas y configuradas según especificaciones
* Supuestos de compatibilidad
  + Se supone que todas las herramientas que se van a utilizar para la ejecución de las pruebas sean compatibles

1. Limitaciones

* Limitaciones de recursos
  + Se cuenta con un equipo de prueba reducido lo que podría resultar en una cobertura reducida para las pruebas
* Limitaciones de acceso a datos
  + Los datos de prueba podrían no ser utilizados debido a las restricciones de privacidad, además con datos generados automáticamente podríamos no reflejar completamente datos reales
* Limitaciones de compatibilidad
  + Las herramientas utilizadas para las pruebas podrían no ser compatibles entre ellas
* Limitaciones de rendimiento
  + Debido al poco personal con el que se cuenta a la hora de realizar las pruebas, podríamos no probar completamente el sistema con grandes volúmenes de datos a la vez y de esta manera ser ineficientes a la hora de probar el rendimiento del sistema

1. Reportes

Al finalizar cada prueba con la ayuda de serenity podemos generar reportes los cuales confirmar los resultados obtenidos en la automatización, además si queremos podemos ver mas detalladamente como sucedió la automatización y buscar bugs en el caso que sea pertinente

