**INSTITUTO DE EDUCACION SUPERIOR CIBERTEC**

**DIRECCIÓN ACADÉMICA**

**CARRERA PROFESIONALES**

**CURSO :** 2424 – Pruebas de Software

### NOTA

**PROFESOR** **:** Johan Artemio, Muñoz Cavero

**SEMESTRE** **:** 2024

**CICLO** **:** Sexto

**SECCIÓN :** DCGS2424

**DURACIÓN :** 2 días

**FECHA INICIO** **:** Jueves, 05 de Diciembre desde las 22:00 horas

**FECHA FIN** **:** Sábado 07 de Diciembre a las 21:59 horas

|  |
| --- |
| **ALUMNO (A) : Cesar Alva Valdivia** |

**CASO DE LABORATORIO N° 2 – T2**

**Consideraciones generales:**

* Considerar el orden, la limpieza y la claridad de las respuestas.
* Está permitido el uso o consulta de cuadernos, separatas, libros o cualquier material de la asignatura durante el desarrollo de la evaluación.
* Descargar el proyecto Java (t2.zip).
* Inicie el proyecto con el nombre DCGS2424\_T2\_XXXX (donde XXXX serán sus apellidos y nombres)
* Subir el proyecto al Blackboard

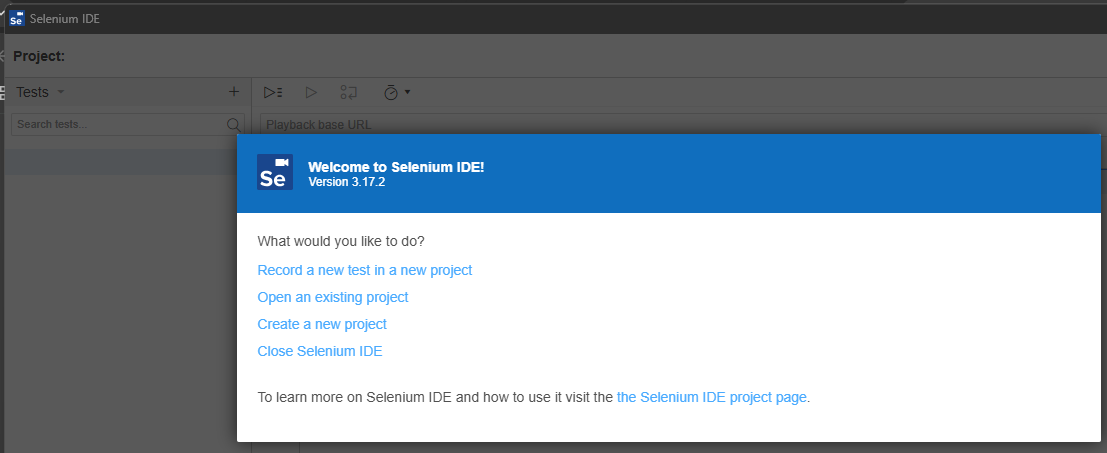
|  |
| --- |
| **LOGRO DE LA EVALUACION:**  Al término de la evaluación, el alumno es capaz de crear y ejecutar pruebas de Desarrollo Guiado por Comportamiento utilizando el lenguaje Gherkin integrado con Cucumber, Serenity y Selenium . |

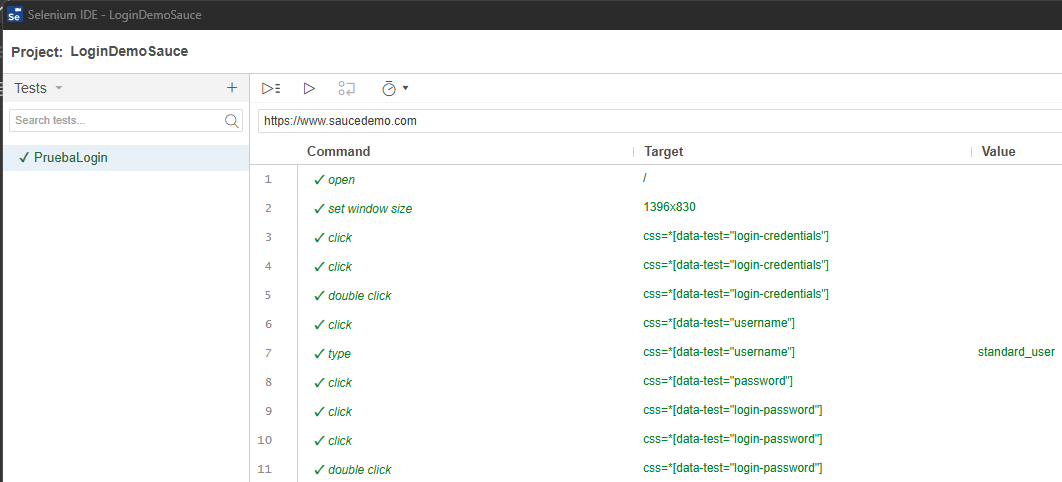
**Consolidado**

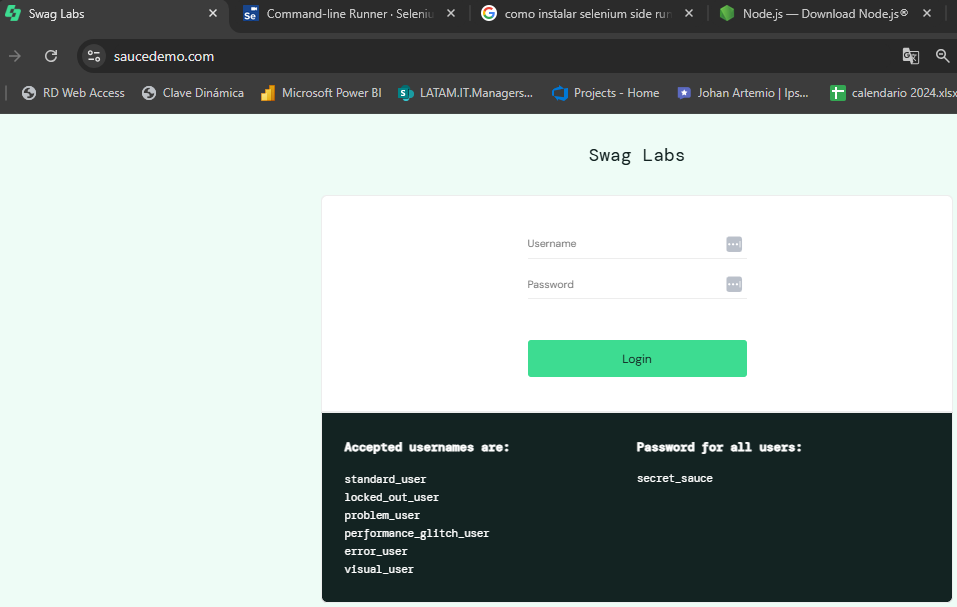
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pregunta** | **Puntaje** | | **Llenar solo en caso de Recalificación justificada** | |
| **Máximo** | **Obtenido** | **Sustento** | **Puntaje** |
| **1** | **20** |  |  |  |
| **Nota Recalificada** | | | |  |

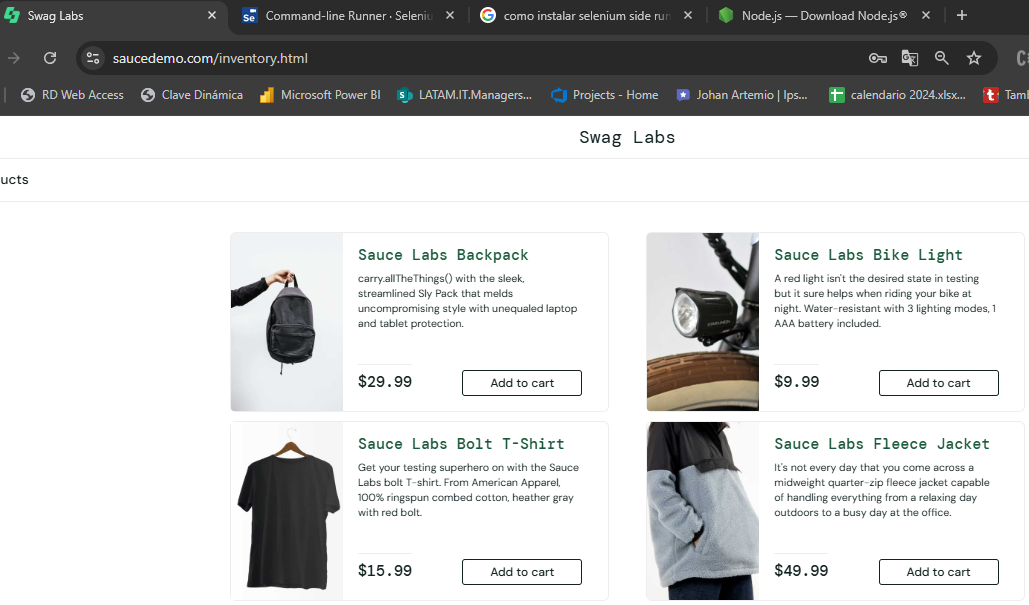
**Pregunta 01 (05 puntos)**

Realizar la grabación desde el login, <https://www.saucedemo.com/> una vez dentro de la plataforma, seleccionar un producto y se envía al carrito de compras, luego terminar la venta (ingresando los datos que nos piden) y por último salimos “Back Home”, utilizar Selenium para el proceso, exportar el resultado en formato JAVA, adicionarlo en un proyecto en IntelliJ Idea, explicar el archivo resultado.









Respuesta:

Se procedió a usar Selenium IDE y se exporto lenguaje java, a continuación el esquema.

**Pregunta 02 (15 puntos)**

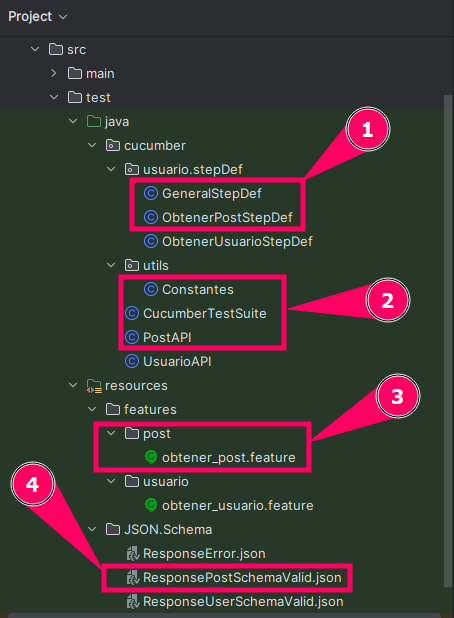
Se solicita la creación de un proyecto Java Utilizando la técnica BDD implemente las pruebas funcionales web con Selenium, Serenity, Gherkin y Cucumber para realizar el logueo y la compra de un producto de la página de demo: <https://gorest.co.in/>

* Explicar el archivo de Pasos a probar, echo con lenguaje Gherkin **“obtener\_post.feature”** que se encuentra en el package **src.test.resorces.features.post**
* Implementar las Pruebas en el archivo **ObtenerPostStepDef.java** el mismo que se ubica en el package **src.test.java.cucumber.usuario.stepdef**, de acuerdo con los pasos a testear alineadas al archivo **obtener\_post.features**
* Generar el reporte con Serenity
* Explicar el resultado obtenido en el reporte Serenity

(\*) Implementar las siguientes funciones del archivo **ObtenerPostStepDef.java** del package **src.test.java.cucumber.usuario.stepdef,** se deja a modo de apoyo las funciones desarrolladas para la orientación del estudiante.

package cucumber.usuario.stepDef;  
  
import cucumber.PostAPI;  
import cucumber.utils.Constantes;  
import io.cucumber.java.en.And;  
import io.cucumber.java.en.Given;  
import io.cucumber.java.en.When;  
import io.cucumber.java.en.Then;  
import io.restassured.module.jsv.JsonSchemaValidator;  
import net.serenitybdd.annotations.Steps;  
import net.serenitybdd.rest.SerenityRest;  
  
import java.io.File;  
  
import static org.hamcrest.Matchers.\*;  
public class ObtenerPostStepDef {  
 @Steps  
 PostAPI postAPI;  
  
 @Given("el post sin parametros")  
 public void postSinParametros()  
 {  
 //Codificar aqui  
 }  
  
 @When("se envia solicitud de post")  
 public void enviarSolicitudPost() {  
 //Codificar aqui  
 }  
  
 @And("validar el post de la lista de esquemas json")  
 public void validarPostJsonEsquema() {  
 //Codificar aqui  
 }  
  
 @Given("el post con Usuario Id {int}")  
 public void postConParametro(Integer usuarioId) {  
 //Codificar aqui  
 }  
  
 @And("validar el Usuario Id del post cual debe ser {int}")  
 public void validarUsuarioIdPost(Integer parameter) {  
 //Codificar aqui  
 }  
  
 @Given("el post con Titulo {string}")  
 public void postConTitulo(String titulo) {  
 //Codificar aqui  
 }  
  
 @And("validar el Titulo del post debe contener {string}")  
 public void validarPostTitulo(String titulo) {  
 //Codificar aqui  
 }  
  
 @Given("el post con Cuerpo {string}")  
 public void postConCuerpo(String cuerpo) {  
 //Codificar aqui  
 }  
  
 @And("validar el Cuerpo del post debe ser {string}")  
 public void validarPostCuerpo(String cuerpo) {  
 //Codificar aqui  
 }  
}

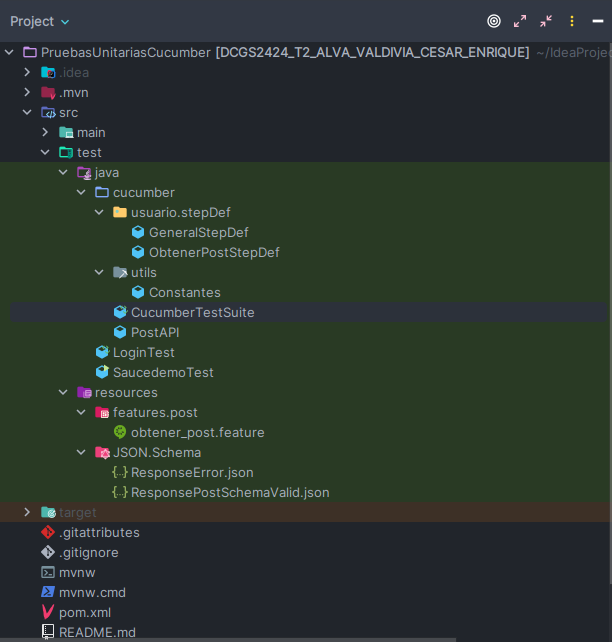
Recuerden que de acuerdo al patrón de diseño visto en clases SCREENPLAY, las funciones a reutilizar ya se encuentran implementadas en el proyecto, el estudiante sólo debe utilizarlas.



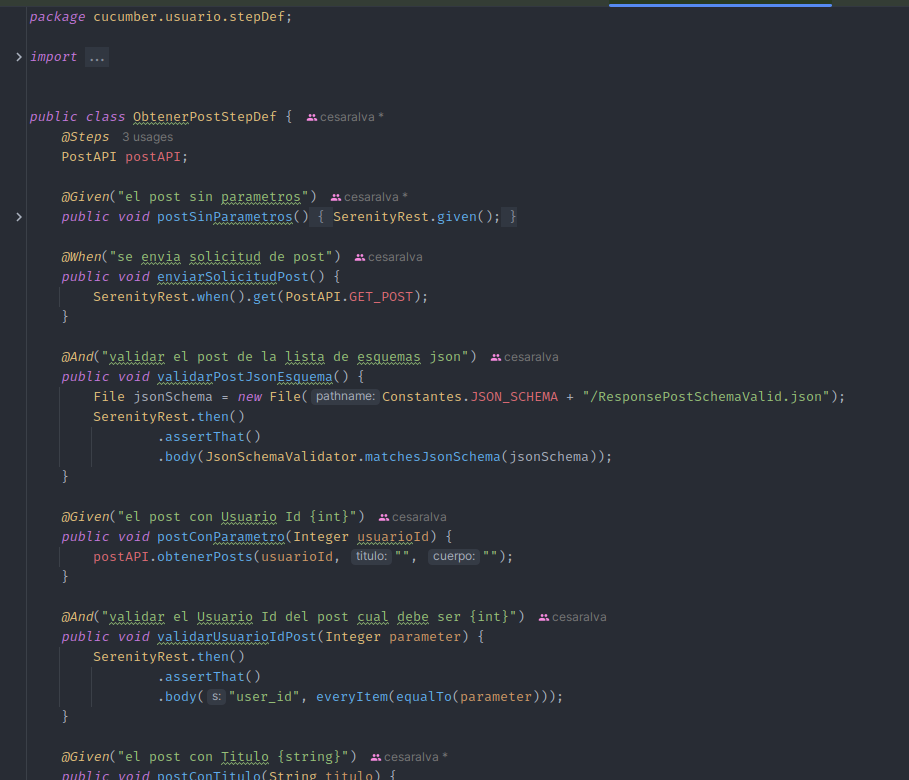
(-) Pantallas referenciales

Respuesta:

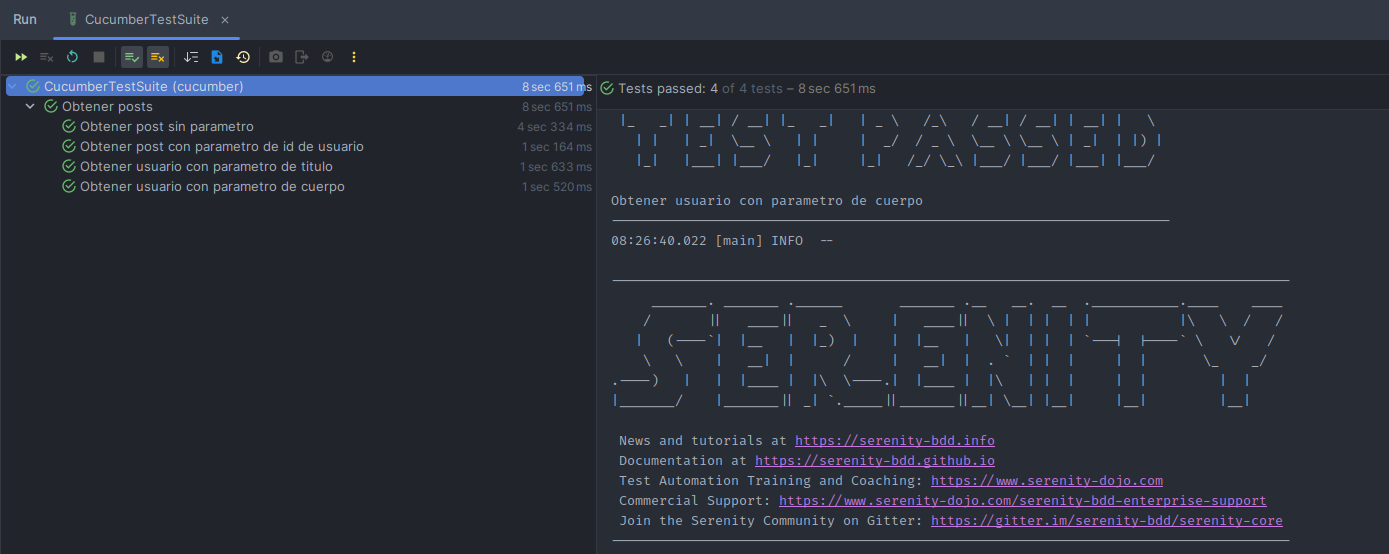
Se genero correctamente el patron Screenplay



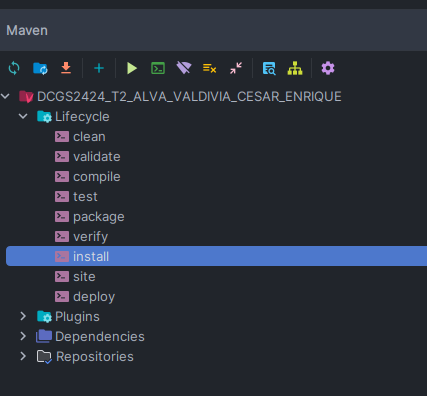
ObtenerPostStepDef



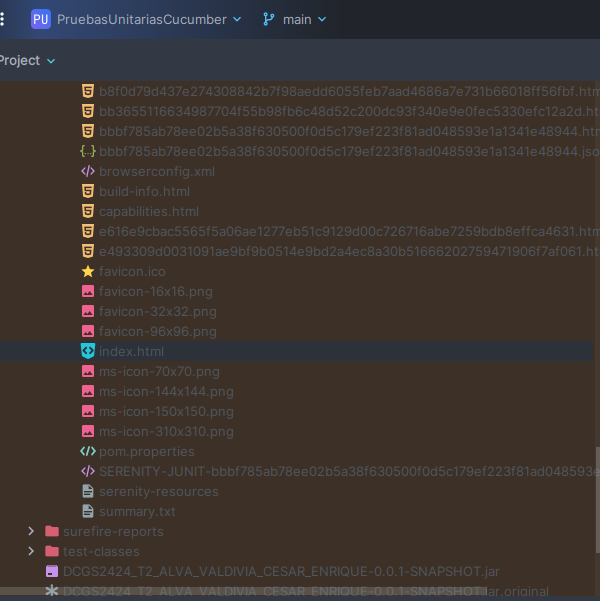
Correcto Funcionamiento del Test:



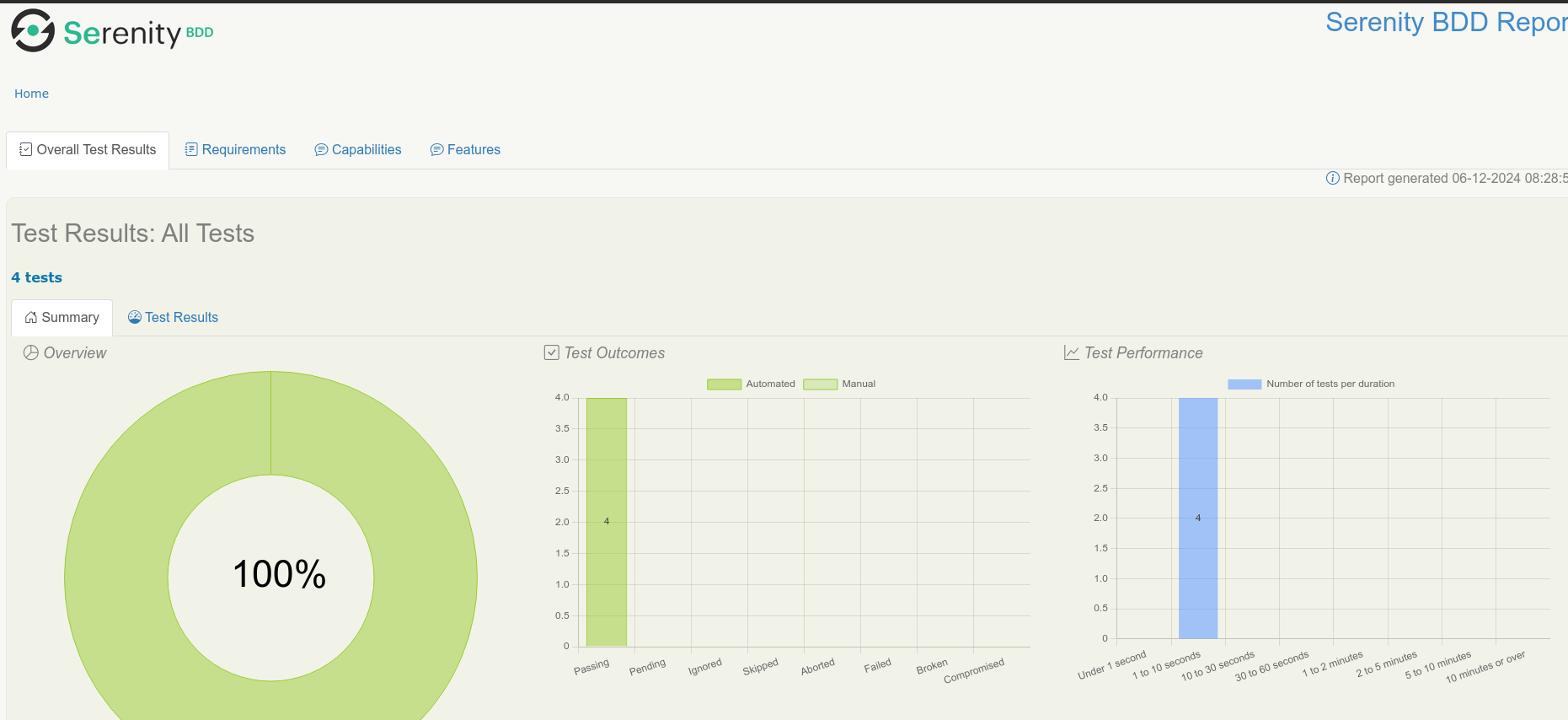
Se procede a instalar en Maven el reporte de Serenity



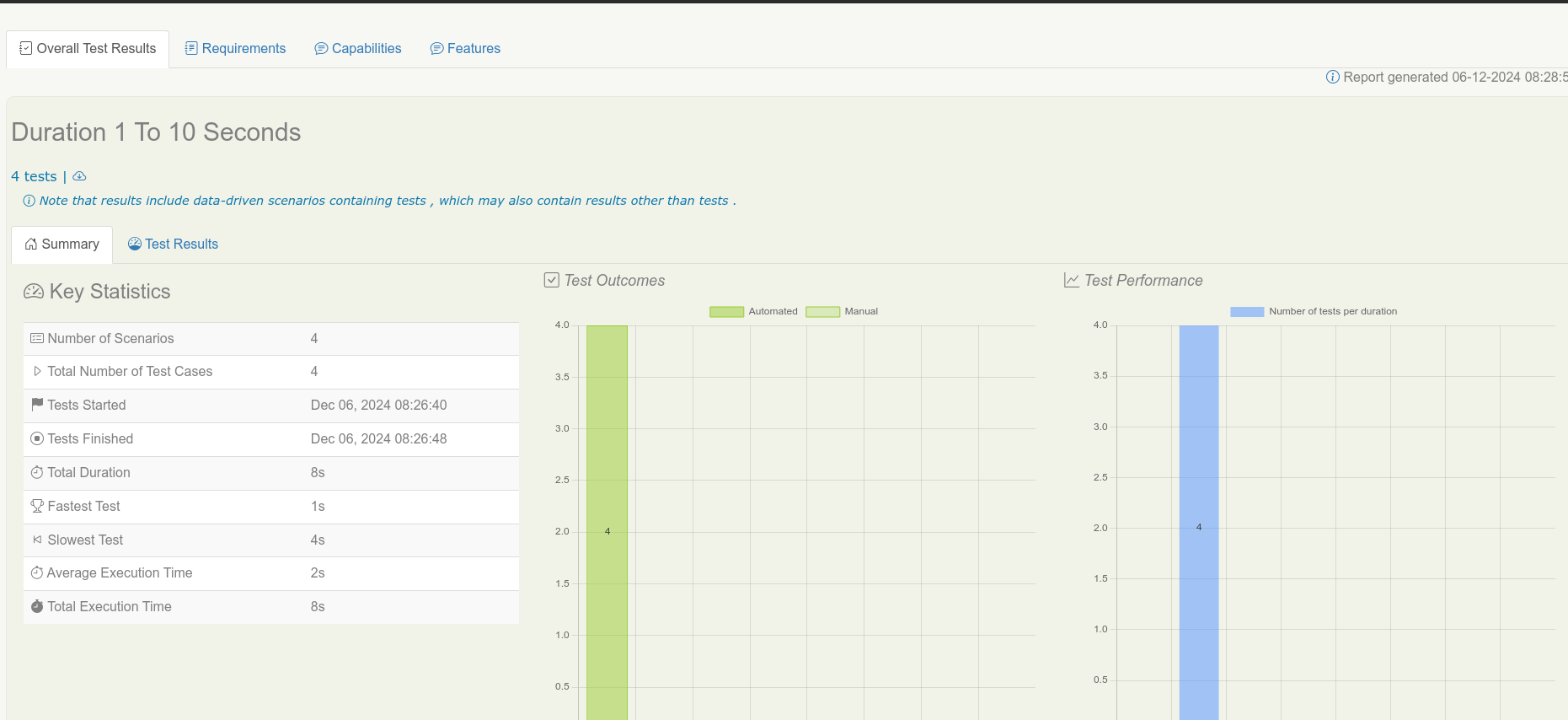
Reporte creado:



Reporte de serinity al 100%

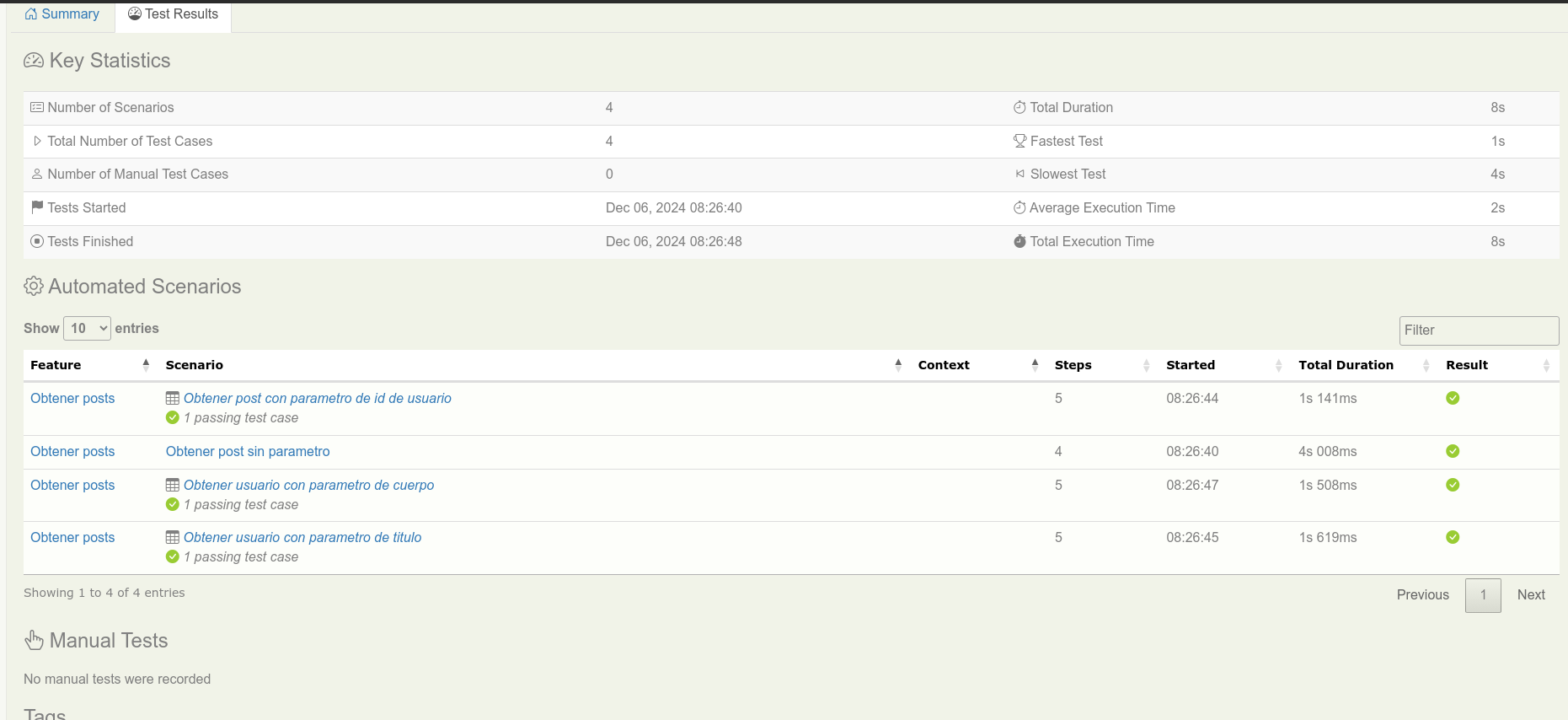


Verificación del test con la duracion de 1 a 10 segundos, se puede verr que en el key statistics hrepresenta la taza de Outcomes como Performance llegando al 100% de factibilidad.

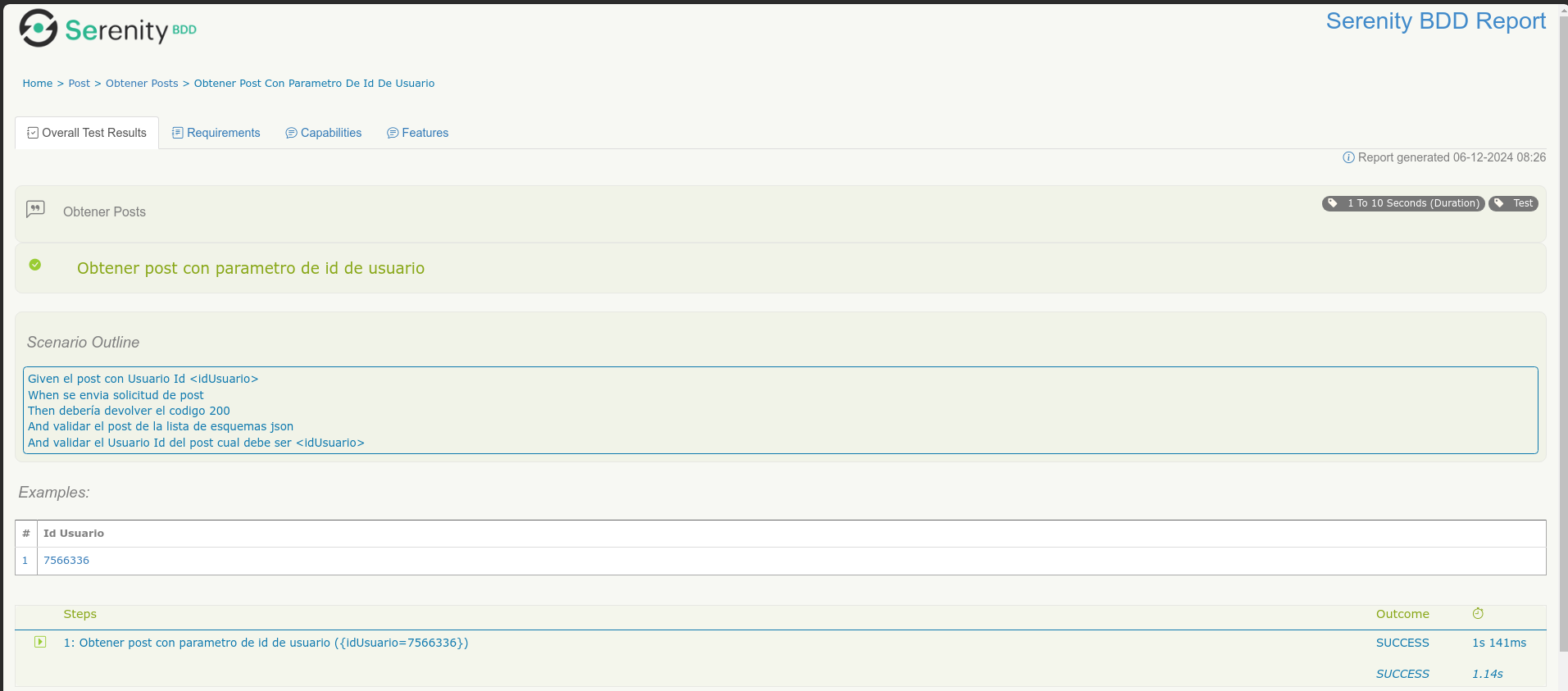


Resultado del test:

La automatizacion que se puede mostrar en el cuadro representa a las peticiones que se realizaron en el test siendo este un escenario necesario para validar la solicitud y pasar la prueba.



Ingresamos a un resultado de la búsqueda en el Overal, podemos verificar en el ejemplo que en la busqueda del usuario encontramos ID como se muestra en la captura del reporte.



Rubrica de la evaluación:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Puntaje** | **Pregunta** | **Excelente** | **Muy Bueno** | **Bueno** | **Regular** | **Deficiente** |
| **20 puntos** | **1** | **Graba correctamente el proyecto utilizando Selenium y exportando test en lenguaje Java, luego inserta el archivo en el proyecto.** | **Graba correctamente el proyecto utilizando Selenium y exportando test en lenguaje Java** | **Graba algún test del proyecto utilizando Selenium y exporta algún lenguaje.** | **Graba algún test del proyecto utilizando Selenium** | **No implementa ninguna consulta solicitada.** |
| **05 puntos** | **04 puntos** | **03 puntos** | **02 puntos** | **0 puntos** |
| **2** | **Implementa correctamente el proyecto Java utilizando el lenguaje Gherkin integrado con Cucumber, Serenity con todos los test solicitados, genera el reporte Serenity y explica de manera adecuada el resultado obtenido.** | **Implementa correctamente el proyecto Java utilizando el lenguaje Gherkin integrado con Cucumber, Serenity con todos los test solicitados, genera el reporte con Serenity.** | **Implementa correctamente el proyecto Java utilizando el lenguaje Gherkin integrado con Cucumber, Serenity incluyendo algunos de los test solicitados, no genera el reporte Serenity.** | **Implementa el proyecto Java utilizando el lenguaje Gherkin integrado con Cucumber, Serenity incluyendo alguno de los test solicitados** | **No implementa ninguna consulta solicitada.** |
|  | **15 puntos** | **11 puntos** | **07 puntos** | **03 puntos** | **0 puntos** |

