

SERENITY BDD con CUCUMBER

7

Usando MyExtra

(Tiempo ejecución 2 horas)

Objetivos:

Agregar al proyecto base utilidades adicionales para lograr interactuar con el GUI de aplicación *EXTRA*.

Crear procesos automatizados para la ejecución de tareas y pruebas desde la interfaz gráfica.

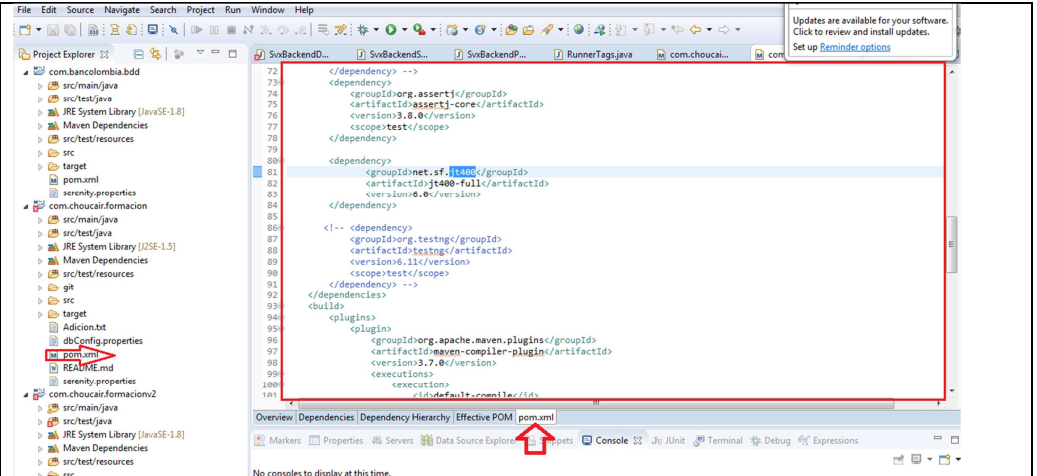
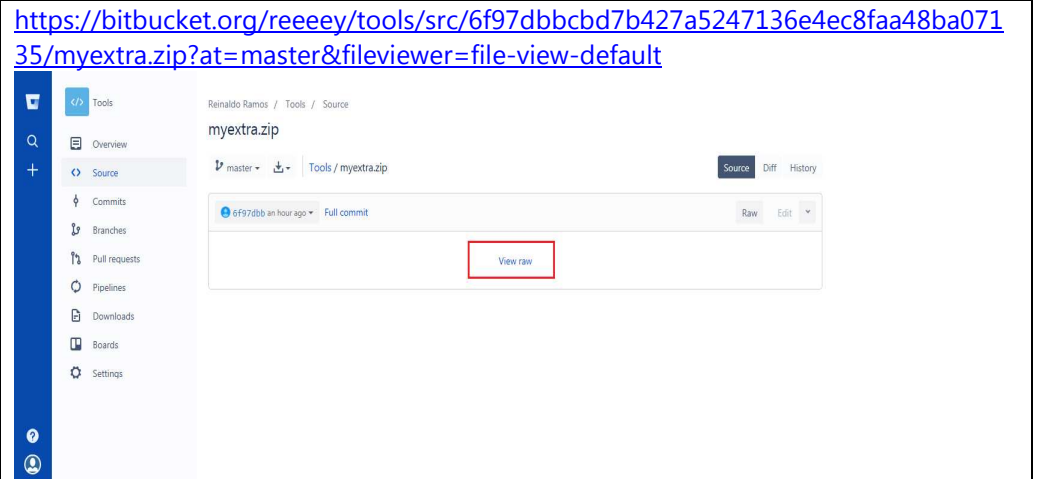
Preparación del Entorno

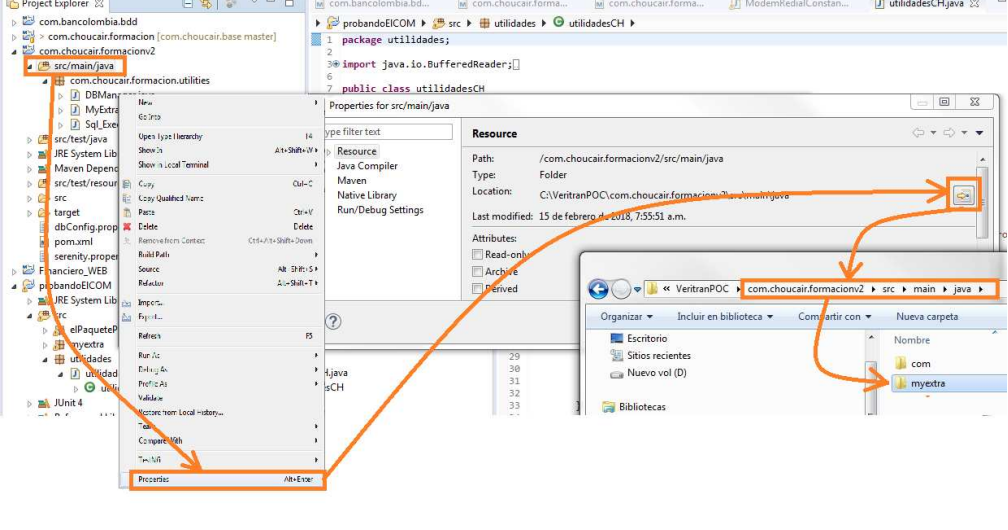
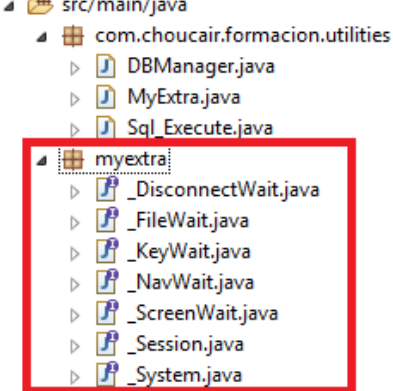
Configuraciones:

Se requiere la instalación de la librería COM4J, la cual nos va a permitir la interoperatividad a través de COM.

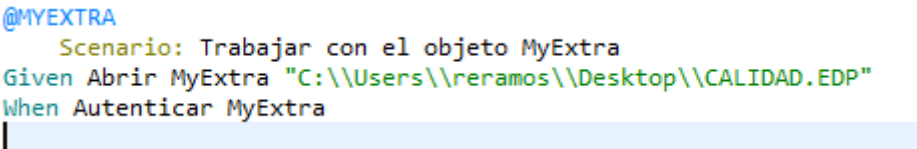
COM : (Component Object Model) es una plataforma de Microsoft® para componentes software. Se usa para permitir comunicación entre aplicaciones y creación dinámica de objetos en cualquier lenguaje de programación que soporte esa tecnología.

Archivo pom.xml, como vimos anteriormente, en los proyectos maven se cuenta con un archivo “pom.xml” el cual contiene datos de configuración de nuestro proyecto, como dependencias con otros jar, tipos de informes que queremos en la página web de nuestro proyecto, etc.. dentro de este archivo encontrará una sección que inicia como “<dependencies>” y finaliza con “</dependencies>”, en medio de estas se debe agregar la fracción de xml que maven requiere para descargar el **jar** que necesitamos para nuestro proyecto.

<p>Editar el archivo pom.xml</p> <p>Doble clic al archivo pom.xml > clic a la pestaña pom.xml</p>	
<p>Agregar la dependencia</p> <p>Al interior de las etiquetas <dependencies> </dependencies></p> <p>Agregue el siguiente xml agregará el jar para la conexión a COM.</p> <p>Luego actualice el proyecto: Clic derecho proyecto > Maven > Update Project</p>	<pre><dependency> <groupId>org.jvnet.com4j</groupId> <artifactId>com4j</artifactId> <version>2.1</version> </dependency></pre>
<p>Descargar paquete MyExtra, desde bitbucket.</p> <p>clic en el link “View raw” para descargar</p>	<p>https://bitbucket.org/reeeey/tools/src/6f97dbbcbd7b427a5247136e4ec8faa48ba07135/myextra.zip?at=master&fileviewer=file-view-default</p> 

<p>Agregar el paquete MyExtra al proyecto, puede ser en la ruta src/main/java</p> <p>Descomprimir el archivo descargado "myextra.zip" en la ruta seleccionada.</p>	
<p>Verificar que el paquete se haya agregado</p>	
<p>Crear la clase ProcessUtility</p> <p>Agregué una clase al paquete de utilities con el nombre <i>ProcessUtility</i>.</p> <p>Edite la clase y agregue el código adjunto.</p> <p>Clase genérica que permite la ejecución de una aplicación, adicional realiza la cancelación de cualquier proceso.</p>	<pre>package com.choucair.formacion.utilities; import java.io.BufferedReader; import java.io.IOException; import java.io.InputStreamReader; public class ProcessUtility { // command used to get list of running task private static final String ListaDeTareas = "tasklist"; // command used to kill a task private static final String asesinoDeTareas = "taskkill /IM "; public static boolean isProcessRunning(String serviceName) { try { Process pro = Runtime.getRuntime().exec(ListaDeTareas); BufferedReader reader = new BufferedReader(new InputStreamReader(pro.getInputStream())); String line; while ((line = reader.readLine()) != null) { // System.out.println(line); if (line.startsWith(serviceName)) { return true; } } catch (IOException e) { e.printStackTrace(); } return false; } public static void killProcess(String serviceName) { try { Runtime.getRuntime().exec(asesinoDeTareas + serviceName); System.out.println(serviceName+" killed successfully!"); } catch (IOException e) { e.printStackTrace(); } } } }</pre>

<p>Implementar Pageobject</p> <p>-Crear la clase "IseriesMyExtraPage"</p> <p>-Crear el siguiente método genérico: <i>iniciar_Extra = el cual estará encargado de:</i></p> <p><i>*Cerrar la aplicación Extra en el evento de encontrarse activa previamente.</i></p> <p><i>*Abrir la aplicación Extra, requiere como parámetro la ruta del archivo de sesión .EDP.</i></p> <p><i>*Establece el objeto System "sSystem"</i></p> <p><i>*Establece el objeto Sesión "Sess0"</i></p> <p><i>*Establece el objeto Pantalla "Screen"</i></p>	<pre> import java.io.IOException; import com.choucair.formacion.utilities.ProcessUtility; import myextra.*; import net.serenitybdd.core.pages.PageObject; import static org.hamcrest.MatcherAssert.assertThat; import static org.hamcrest.Matchers.*; public class IseriesMyExtraPage extends PageObject{ public _System sSystem=null; public Screen Screen=null; public _Session Sess0=null; public int g_HostSettleTime = 3000; // <u>milisegundos</u> public void iniciar_Extra(String rutaCalidad) throws InterruptedException { try { if(ProcessUtility.isProcessRunning("EXTRA.exe")) MyExtra_init.cerrarMyExtraAbierto(); MyExtra_init.abrirArchivoDeMyExtra(rutaCalidad); Thread.sleep(5000); sSystem= MyExtra_init.crearSistema(); Sess0=MyExtra_init.crearSesion(sSystem); Screen=MyExtra_init.crearPantalla(Sess0); sSystem.timeoutValue(g_HostSettleTime); }catch (IOException e) { // <u>TODO Auto-generated catch block</u> e.printStackTrace(); } } } </pre>
<p>Implementar Pageobject</p> <p>-Agregar el siguiente método genérico a la clase "IseriesMyExtraPage".</p> <p><i>Autenticar_Extra = el cual estará encargado de:</i></p> <p><i>*Ingresar el usuario y la contraseña</i></p> <p><i>*Verifica que no se presente algún mensaje de validación de inicio de sesión</i></p> <p><i>*Verifica que se presente el label "Ubicarse:" del menú principal.</i></p> <p><i>*Si no se presenta el label, presiona hasta 5 veces la tecla <Enter></i></p> <p><i>Nota: el método requiere los parámetros de entrada: Usuario y Clave.</i></p>	<pre> public void Autenticar_Extra(String strUsuario, String strClave) { try { String TextoPantalla = ""; Screen.putString(strUsuario, 6, 53, null); Screen.putString(strClave, 7, 53, null); Screen.sendKeys("<Enter>"); Screen.waitHostQuiet(g_HostSettleTime); Thread.sleep(2000); TextoPantalla = Screen.getString(24, 1, 49, Screen); //capturar texto <u>validación de clave</u> assertThat(TextoPantalla, is(not("La información de inicio de sesión no es correcta"))); int i = 0; while (true) { i++; //verifica label para identificar si se logró la autenticación exitosa Thread.sleep(2000); TextoPantalla = Screen.getString(5, 7, 9, Screen); if (!TextoPantalla.equals("Ubicarse:")){ Screen.sendKeys("<Enter>"); Screen.waitHostQuiet(g_HostSettleTime); } else { break; } if (i==5) { break;} } assertThat(TextoPantalla,is("Ubicarse:")); } catch (Exception e) { // <u>TODO Auto-generated catch block</u> System.out.println(e.getMessage()); } } </pre>

<p>Implementar los Steps -Crear la clase steps "IseriesMyExtraSteps"</p> <p>-Agregar los pasos correspondientes:</p> <p>Abrir_Extra: recibe como parámetro la ruta del archivo .EDP</p> <p>Autenticar_Extra: recibe los parámetros usuario y clave Verifica que exista una sesión activa como prerequisite para ejecutar la autenticación.</p>	<pre>import com.choucair.formacion.pageobjects.IseriesMyExtraPage; import net.thucydides.core.annotations.Step; public class IseriesMyExtraSteps { IseriesMyExtraPage iseriesMyExtraPage; @Step public void Abrir_Extra(String rutaCalidad) throws InterruptedException{ iseriesMyExtraPage.iniciar_Extra(rutaCalidad); } @Step public void Autenticar_Extra(String strUsuario, String strClave) throws InterruptedException{ if (iseriesMyExtraPage.Sess0 != null) { iseriesMyExtraPage.Autenticar_Extra(strUsuario, strClave); } else { System.out.println("No es posible ejecutar no se cuenta con una sesion activa"); } } }</pre>
<p>Crear feature y agregar un escenario.</p> <p>Para efectos del ejemplo vamos a agregar un Scenariio: en la feature "backend_consultas.feature" ya existente.</p> <p>Los pasos a implementar serán: *Abrir la aplicación MyExtra *Autenticar en MyExtra</p>	
<p>Implementar en definition -Crear la clase definition "IseriesMyExtraDefinition" -Agregar los métodos propuestos para cada línea gherkin.</p> <p>Nota: En el evento del método "autenticar_MyExtra", se implementa la lectura del archivo properties para extraer el usuario y la clave del usuario AS400.</p> <p>Recuerde reemplazar lo resaltado con la ruta física en su máquina del archivo.</p>	<pre>import java.io.FileReader; import java.util.Properties; import com.choucair.formacion.steps.IseriesMyExtraSteps; import cucumber.api.java.en.Given; import cucumber.api.java.en.When; import net.thucydides.core.annotations.Steps; public class IseriesMyExtraDefinition { private String user; private String password; @Steps IseriesMyExtraSteps iseriesMyExtraSteps; @Given("^Abrir MyExtra \"([^\"]*)\"\$") public void abrir_MyExtra(String rutaCalidad) throws Throwable { iseriesMyExtraSteps.Abrir_Extra(rutaCalidad); } @When("^Autenticar MyExtra\$") public void autenticar_MyExtra() throws Throwable { Properties prop = new Properties(); prop.load(new FileReader("C:/VeritransPOC/com.choucair.formacionv2/dbConfig.properties")); this.user = prop.getProperty("db.user"); this.password = prop.getProperty("db.password"); iseriesMyExtraSteps.Autenticar_Extra(user, password); } }</pre>
<p>El archivo dbconfig.properties contiene información necesaria para la conexión. Los campos db.user y db.password hacen referencia al usuario y clave para ingresar a iseries, reemplácelos con la información de su usuario.</p>	<pre>db.driver=com.ibm.as400.access.AS400JDBCdriver db.url=jdbc:as400:10.9.2.221 db.user=CXXXXXXX db.password=????????</pre>

Resumen de Métodos más utilizados

Sendkeys	<p>Envía pulsaciones de teclas al host, incluidas las teclas de función. Las pulsaciones de teclas aparecen en la sesión como si fueran ingresadas manualmente por un usuario.</p> <p>Sintaxis: <i>object.SendKeys(String)</i></p> <p>Nota: Para pausar su aplicación mientras el host procesa las pulsaciones de teclas transmitidas, use un método de espera, como WaitForCursor.</p> <p>Ejemplo: Screen.SendKeys("hola mundo"<Enter>); Screen.SendKeys(<pf3>);</p> <p>Otras teclas:</p> <table><tr><td>BackSpace</td><td>Retroceder borrando</td></tr><tr><td>FieldExit</td><td>Salida de Campo</td></tr><tr><td>Pf1 ...pf12</td><td>Teclas de funciones</td></tr><tr><td>Up, Down, Left, Right</td><td>Movimientos Cursor</td></tr><tr><td>Rolldown, Rollup</td><td>Movimientos pagina</td></tr><tr><td>Tab</td><td>Tabulador</td></tr><tr><td>Delete</td><td>Borrar carácter</td></tr><tr><td>Beginline</td><td>Inicio de línea</td></tr></table>	BackSpace	Retroceder borrando	FieldExit	Salida de Campo	Pf1 ...pf12	Teclas de funciones	Up, Down, Left, Right	Movimientos Cursor	Rolldown, Rollup	Movimientos pagina	Tab	Tabulador	Delete	Borrar carácter	Beginline	Inicio de línea
BackSpace	Retroceder borrando																
FieldExit	Salida de Campo																
Pf1 ...pf12	Teclas de funciones																
Up, Down, Left, Right	Movimientos Cursor																
Rolldown, Rollup	Movimientos pagina																
Tab	Tabulador																
Delete	Borrar carácter																
Beginline	Inicio de línea																
PutString	<p>Coloca texto en la ubicación especificada en la pantalla.</p> <p>Sintaxis: <i>object.PutString(String, Row, Col, [Page])</i></p> <p>Page es opcional</p>																
MoveTo	<p>Mueve el cursor a una ubicación especificada</p> <p>Sintaxis: <i>object.MoveTo(Row, Col, [Page])</i></p> <p>Page es opcional</p>																
getString	<p>Retorna el texto de una ubicación en la pantalla</p> <p>Sintaxis: <i>Rc = object.GetString(Row, Col, Length, [Page])</i></p>																
Para mayor información : http://docs.attachmate.com/extra/x-treme/apis/com/																	

Taller

Crear un metodo que permita seleccionar de forma dinámica una opción en pantalla, se debe recibir por parámetro el Tipo Código.

