ABRIR CUENTA SFP-CFP

1-FEATURE

@AportesVoluntarios @all  
Feature: Aportes Voluntarios  
 Como un usuario registrado  
 Quiero abrir una cuenta de aportes voluntarios y realizar depósitos  
 Para gestión de ahorros  
*#Ejecución exitosa en Azure - Irvin*  
@Escenario1 @SinFinPrevisional @Happy  
 Scenario Outline: "<Caso de Prueba>" El usuario abre una cuenta de aportes voluntarios sin fin previsional de tipo "<tipo fondo>"  
 Given que el usuario está en la página de inicio de sesión  
 When el usuario selecciona el tipo de documento "<tipo documento>"  
 And el usuario ingresa el nombre de usuario "<usuario>"  
 And el usuario ingresa la contraseña "<contraseña>"  
 And el usuario hace clic en el botón de inicio de sesión  
 Then el usuario valida el mensaje de bienvenida  
 And el usuario cierra va ventana modal informativa  
 And el usuario hace clic en el botón Mis trámites del menu  
 And el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Sin Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 Then el usuario valida que la cuenta se haya creado correctamente  
  
 Examples:  
 | Caso de Prueba | tipo documento | usuario | contraseña | tipo fondo |  
 | CP01 | DNI | 40038791 | Cercania2024 | Fondo 1 |  
 | CP02 | DNI | 44189028 | Cercania2024 | Fondo 2 |  
 | CP03 | DNI | 08037123 | Cercania2024 | Fondo 3 |  
*#Ejecución exitosa en Azure - Irvin*  
@Escenario2 @ConFinPrevisional @Happy  
 Scenario Outline: "<Caso de Prueba>" El usuario abre una cuenta de aportes voluntarios con fin previsional de tipo "<tipo fondo>"  
 Given que el usuario está en la página de inicio de sesión  
 When el usuario selecciona el tipo de documento "<tipo documento>"  
 And el usuario ingresa el nombre de usuario "<usuario>"  
 And el usuario ingresa la contraseña "<contraseña>"  
 And el usuario hace clic en el botón de inicio de sesión  
 Then el usuario valida el mensaje de bienvenida  
 And el usuario cierra va ventana modal informativa  
 And el usuario hace clic en el botón Mis trámites del menu  
 And el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Con Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 Then el usuario valida que la cuenta se haya creado correctamente  
  
 Examples:  
 | Caso de Prueba | tipo documento | usuario | contraseña | tipo fondo |  
 | CP04 | CIA | 72777884 | Cercania2024 | Fondo 1 |  
 | CP05 | DNI | 47272121 | Cercania2024 | Fondo 2 |  
 | CP06 | DNI | 32736818 | Cercania2024 | Fondo 3 |  
  
  
*#Ejecución exitosa en Azure - Irvin*  
@Escenario3 @SinFinPrevisional @Alternativo @Happy  
 Scenario Outline: "<Caso de Prueba>" El usuario desde la pagina AFP Integra abre una cuenta de aportes voluntarios sin fin previsional de tipo "<tipo fondo>"  
 Given que el usuario está en la página principal  
 And el usuario acepta el cookis  
 When el usuario hace la búsqueda de Abrir una cuenta de Aportes Aporte Voluntarios  
 And el usuario hace clic en la opción Abrir una cuenta de Aportes Aporte Voluntarios  
 And el usuario hace clic en el botón iniciar trámite  
 When el usuario selecciona el tipo de documento "<tipo documento>"  
 And el usuario ingresa el nombre de usuario "<usuario>"  
 And el usuario ingresa la contraseña "<contraseña>"  
 And el usuario hace clic en el botón de inicio de sesión  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Sin Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 Then el usuario valida que la cuenta se haya creado correctamente  
  
 Examples:  
 | Caso de Prueba | tipo documento | usuario | contraseña | tipo fondo |  
 | CP07 | DNI | 01770753 | Cercania2024 | Fondo 1 |  
 | CP08 | DNI | 71122150 | Cercania2024 | Fondo 2 |  
 | CP09 | DNI | 15862996 | Cercania2024 | Fondo 3 |  
  
 *#Ejecución exitosa en Azure - Irvin #primero pasa el proceso y segunda vez mensaje que ya esta en proceso*  
@Escenario4 @SolicitudProceso @Unhappy  
 Scenario Outline: "<Caso de Prueba>" El usuario crea solicitud de aportes voluntarios y validar que ya tiene solicitud en proceso  
 Given que el usuario está en la página de inicio de sesión  
 When el usuario selecciona el tipo de documento "<tipo documento>"  
 And el usuario ingresa el nombre de usuario "<usuario>"  
 And el usuario ingresa la contraseña "<contraseña>"  
 And el usuario hace clic en el botón de inicio de sesión  
 Then el usuario valida el mensaje de bienvenida  
 And el usuario cierra va ventana modal informativa  
 And el usuario hace clic en el botón Mis trámites del menu  
 And el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Sin Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 And el usuario hace clic en Ir al inicio  
 And el usuario hace clic en el botón Mis trámites del menu  
 And el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Sin Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 Then el usuario valida que tiene solicitud de aportes voluntarios en proceso  
  
 Examples:  
 | Caso de Prueba | tipo documento | usuario | contraseña | tipo fondo |  
 | CP10 | DNI | 01700842 | Cercania2024 | Fondo 1 |  
  
 *#deposito bloqueado*  
@Escenario5 @SinFinPrevisional @Deposito  
 Scenario Outline: "<Caso de Prueba>" El usuario abre una cuenta de aportes voluntarios SFP de tipo "<tipo fondo>" y realiza un deposito único con la entidad bancaria "<banco>"  
 Given que el usuario está en la página de inicio de sesión  
 When el usuario selecciona el tipo de documento "<tipo documento>"  
 And el usuario ingresa el nombre de usuario "<usuario>"  
 And el usuario ingresa la contraseña "<contraseña>"  
 And el usuario hace clic en el botón de inicio de sesión  
 And el usuario cierra va ventana modal informativa  
 And el usuario hace clic en el botón Mis trámites del menu  
 And el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios  
 And el usuario hace clic a Empecemos  
 And el usuario selecciona Abrir Una Cuenta "Con Aportes Voluntarios Sin Fin Previsional"  
 And el usuario hace clic en Continuar  
 And el usuario elige el tipo de fondo "<tipo fondo>"  
 When el usuario hace clic en Abrir Cuenta  
 Then el usuario valida que la cuenta se haya creado correctamente  
 And el usuario hace clic en Realizar depósito  
 And el usuario elige el tipo de deposito "Depósito único"  
 *# And el usuario elige el tipo de deposito "Depósito recurrente"*  
And el usuario elige el banco "<banco>"  
 And el usuario ingresa el monto que desea aportar de "<monto>" soles  
 And el usuario hace clic en Continuar para depositar  
 And el usuario selecciona "<pais>" como país donde vive actualmente y el número de nit "<nit>" si se requiera  
 And el usuario selecciona "<pais operaciones>" como país de origen del dinero que usará para sus operaciones  
 And el usuario selecciona "<origen del dinero>" como origen del dinero  
 And el usuario marca el check de nacionalidad americana si tiene documento fiscal de EEUU "<documento fiscal EEUU>"  
 And el usuario marca el check de Ha sido investigado si tiene algún delito y lo detalla "<detalle delito>"  
 And el usuario marca el check de Acepto que la información es Confidencial "<acepta info confidencial>"  
 And el usuario marca "<acepta mas info>" en el check de Me gustaría recibir más información sobre oportunidades de ahorro  
  
  
 Examples:  
 | Caso de Prueba | tipo documento | usuario | contraseña | tipo fondo | banco | monto | pais | nit | pais operaciones | origen del dinero | documento fiscal EEUU | detalle delito | acepta info confidencial | acepta mas info |  
 *# |CP04| CIA | 72777884 | Cercania2024 | Fondo 3 | Yape | 100 | PERÚ | | PERÚ | Otros (especificar) - BBVA, Prestación de servicios - bcp, Venta de propiedad - interbank | | | | |*  
 *# |CP05| CIA | 72777884 | Cercania2024 | Fondo 3 | BCP | 100 | PERÚ | | PERÚ | Otros (especificar) - BBVA, Prestación de servicios - bcp, Venta de propiedad - interbank | | | | |*  
| CP06 | CIA | 72777884 | Cercania2024 | Fondo 3 | Interbank | 100 | ZAMBIA | 9999 | CHILE | Otros (especificar) - BBVA, Prestación de servicios - bcp, Venta de propiedad - interbank | 999999 | delito | Si | Si |  
*# |CP07| CIA | 72777884 | Cercania2024 | Fondo 3 | Scotiabank | 100 | PERÚ | | PERÚ | Otros (especificar) - BBVA, Prestación de servicios - bcp, Venta de propiedad - interbank | | | | |*  
*# |CP08| CIA | 72777884 | Cercania2024 | Fondo 3 | BBVA | 100 | PERÚ | | PERÚ | Otros (especificar) - BBVA, Prestación de servicios - bcp, Venta de propiedad - interbank | | | | |*

STEP DEFINITIONS

package com.sura.web.stepdefinitions;  
  
import io.cucumber.java.en.And;  
import io.cucumber.java.en.Then;  
import io.cucumber.java.en.When;  
import net.serenitybdd.screenplay.waits.WaitUntil;  
import questions.ElementVisible;  
import tasks.common.ClickElement;  
import tasks.common.ClickElementJs;  
import tasks.common.EnterText;  
import tasks.common.WaitFor;  
import tasks.voluntaryContributions.SelectOriginOfMoney;  
import ui.\*;  
  
import static net.serenitybdd.screenplay.GivenWhenThen.*seeThat*;  
import static net.serenitybdd.screenplay.actors.OnStage.*theActorInTheSpotlight*;  
import static net.serenitybdd.screenplay.matchers.WebElementStateMatchers.*isVisible*;  
import static org.hamcrest.Matchers.*is*;  
  
public class VoluntaryContributionsStepDefinitions {  
  
 @And("el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios")  
 public void elusuarioseleccionalaopciónAbreodepositaatucuentadeAportesVoluntarios() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 DashboardPage.*MY\_PROCEDURES\_OPTIONS\_BUTTON*,  
 false,  
 "Abre o deposita a tu cuenta de Aportes Voluntarios"));  
 }  
  
 @And("el usuario hace clic a Empecemos")  
 public void elUsuarioHaceClicAEmpecemos() {  
 // theActorInTheSpotlight().attemptsTo(ClickElement.on(  
 // VoluntaryContributionsPage.LETS\_BEGIN\_BUTTON,  
 // false));  
 *theActorInTheSpotlight*().attemptsTo(  
 WaitUntil.*the*(VoluntaryContributionsPage.*LETS\_BEGIN\_BUTTON*, *isVisible*())  
 .forNoMoreThan(30).seconds(), // Espera dinámica de hasta 15 segundos  
 ClickElement.*on*(VoluntaryContributionsPage.*LETS\_BEGIN\_BUTTON*, false)  
 );  
  
  
 }  
  
 @And("el usuario selecciona Abrir Una Cuenta {string}")  
 public void elUsuarioSeleccionaAbrirUnaCuenta(String tipo\_cuenta) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*SELECT\_VOLUNTARYCONTRIBUTIONTYPE\_BUTTON*,  
 false,  
 tipo\_cuenta));  
 }  
  
 @And("el usuario hace clic en Continuar")  
 public void elUsuarioHaceClicEnContinuar() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CONTINUE\_BUTTON*,  
 true  
 ));  
 }  
  
 @And("el usuario elige el tipo de fondo {string}")  
 public void elUsuarioEligeElTipoDeFondo(String tipo\_fondo) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CHECKBOX\_TYPEFUND*,  
 true,  
 tipo\_fondo));  
 }  
  
 @And("el usuario hace clic en Abrir Cuenta")  
 public void elUsuarioHaceClicEnAbrirCuenta() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*BUTTON\_OPENACCOUNT*,  
 true  
 ));  
 }  
  
 @And("el usuario hace clic en Realizar depósito")  
 public void elUsuarioHaceClicEnRealizarDepósito() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*MAKE\_A\_DEPOSIT\_BUTTON*,  
 false  
 ));  
 }  
  
 @Then("el usuario valida que la cuenta se haya creado correctamente")  
 public void elUsuarioValidaQueLaCuentaSeHayaCreadoCorrectamente() {  
 *theActorInTheSpotlight*().should(*seeThat*(ElementVisible.*isVisible*(  
 VoluntaryContributionsPage.*ACCOUNTCREATED\_LABEL*), *is*(true)));  
 }  
  
 @And("el usuario elige el tipo de deposito {string}")  
 public void elUsuarioEligeElTipoDeDeposito(String tipo\_deposito) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*DEPOSIT\_TYPE\_RADIOBUTTON*,  
 false,  
 tipo\_deposito  
 ));  
 }  
  
 @And("el usuario elige el banco {string}")  
 public void elUsuarioEligeElBanco(String banco) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*BANK\_RADIOBUTTON*,  
 false,  
 banco  
 ));  
 }  
  
 @And("el usuario ingresa el monto que desea aportar de {string} soles")  
 public void elUsuarioIngresaElMontoQueDeseaAportarDeSoles(String monto) {  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(  
 VoluntaryContributionsPage.*AMOUNT\_INPUT*,  
 monto.trim()  
 ));  
 }  
  
 @And("el usuario hace clic en Continuar para depositar")  
 public void elUsuarioHaceClicEnContinuarParaDepositar() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CONTINUE\_BUTTON\_2*,  
 false  
 ));  
 }  
  
  
 @And("el usuario selecciona {string} como país donde vive actualmente y el número de nit {string} si se requiera")  
 public void elUsuarioSeleccionaComoPaísDondeViveActualmenteYElNúmeroDeNitSiSeRequiera(String pais, String nit) {  
  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRY\_SELECT*,false));  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRY\_ITEM*,false, pais));  
  
 if (!nit.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*IDENTITYTRIBUTTE\_INPUT*, nit.trim()));  
 }  
  
 }  
  
 @And("el usuario selecciona {string} como país de origen del dinero que usará para sus operaciones")  
 public void elUsuarioSeleccionaComoPaísDeOrigenDelDineroQueUsaráParaSusOperaciones(String pais) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRYOPERATIONS\_SELECT*,false));  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRYOPERATIONS\_ITEM*,false, pais));  
 }  
  
 @And("el usuario selecciona {string} como origen del dinero")  
 public void elUsuarioSeleccionaComoOrigenDelDinero(String valor\_origen) {  
 *theActorInTheSpotlight*().attemptsTo(SelectOriginOfMoney.*withValues*(valor\_origen));  
 }  
  
 @And("el usuario marca el check de nacionalidad americana si tiene documento fiscal de EEUU {string}")  
 public void elUsuarioMarcaElCheckDeNacionalidadAmericanaSiTieneDocumentoFiscalDeEEUU(String documento\_eeuu) {  
 if (!documento\_eeuu.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*AMERICAN\_NATIONALITY\_CHECK*,false));  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*ID\_NATIONALITY\_EEUU\_INPUT*, documento\_eeuu));  
 }  
 }  
  
 @And("el usuario marca el check de Ha sido investigado si tiene algún delito y lo detalla {string}")  
 public void elUsuarioMarcaElCheckDeHaSidoInvestigadoSiTieneAlgúnDelitoYLoDetalla(String delitos) {  
 if (!delitos.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*IS\_INVESTIGATED\_CHECK*,false));  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*INVESTIGATED\_DETAILS\_INPUT*, delitos));  
 }  
 }  
  
 @And("el usuario marca el check de Acepto que la información es Confidencial {string}")  
 public void elUsuarioMarcaElCheckDeAceptoQueLaInformaciónEsConfidencial(String acepta\_info\_conf) {  
 if (acepta\_info\_conf.trim().equalsIgnoreCase("SI")){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*CONFIDENTIAL\_INFO\_CHECK*,false));  
 }  
 }  
  
 @And("el usuario marca {string} en el check de Me gustaría recibir más información sobre oportunidades de ahorro")  
 public void elUsuarioMarcaEnElCheckDeMeGustaríaRecibirMásInformaciónSobreOportunidadesDeAhorro(String mas\_informacion) {  
 if (mas\_informacion.trim().equalsIgnoreCase("SI")){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*MORE\_INFO\_CHECK*,false));  
 }  
 }  
  
 @When("el usuario hace la búsqueda de Abrir una cuenta de Aportes Aporte Voluntarios")  
 public void elUsuarioHaceLaBúsquedaDeAbrirUnaCuentaDeAportesAporteVoluntarios() {  
 *theActorInTheSpotlight*().attemptsTo(WaitFor.*seconds*(3));  
 *theActorInTheSpotlight*().attemptsTo(ClickElementJs.*on*(HomePage.*BUSQUEDA\_ABRIR\_CUENTA*));  
 }  
  
 @And("el usuario hace clic en la opción Abrir una cuenta de Aportes Aporte Voluntarios")  
 public void elUsuarioHaceClicEnLaOpciónAbrirUnaCuentaDeAportesAporteVoluntarios() {  
 *theActorInTheSpotlight*().attemptsTo(WaitFor.*seconds*(3));  
 *theActorInTheSpotlight*().attemptsTo(ClickElementJs.*on*(HomePage.*BUTTON\_ABRIR\_CUENTA*));  
  
 }  
  
 @And("el usuario hace clic en Ir al inicio")  
 public void elUsuarioHaceClicEnIrAlInicio() {  
 *theActorInTheSpotlight*().attemptsTo(WaitFor.*seconds*(10));  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*BUTTON\_IR\_INICIO*, true));  
  
 }  
  
 @Then("el usuario valida que tiene solicitud de aportes voluntarios en proceso")  
 public void elUsuarioValidaQueTieneSolicitudDeAportesVoluntariosEnProceso() {  
 *theActorInTheSpotlight*().should(*seeThat*(ElementVisible.*isVisible*(  
 VoluntaryContributionsPage.*MESSAGE\_PROCESS\_SOLICITUD*), *is*(true)));  
 }  
}

UI

package ui;  
  
import net.serenitybdd.screenplay.targets.Target;  
import org.openqa.selenium.By;  
  
public class VoluntaryContributionsPage {  
 public static final Target *LETS\_BEGIN\_BUTTON* = Target.*the*("Empecemos")  
 .locatedBy("//div[text()='Empecemos']/parent::button");  
  
 public static final Target *SELECT\_VOLUNTARYCONTRIBUTIONTYPE\_BUTTON* = Target.*the*("Selecciona el tipo {0}")  
 .locatedBy("//div[contains(@class,'i-hide-mobile')]//p[text()='{0}']/parent::div/following-sibling::button");  
  
 public static final Target *CONTINUE\_BUTTON* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *CHECKBOX\_TYPEFUND* = Target.*the*("Tipo de fondo {0}")  
 .locatedBy("//h3[contains(text(),'{0}')]");  
  
 public static final Target *BUTTON\_OPENACCOUNT* = Target.*the*("Abrir Cuenta")  
 .locatedBy("//div[text()='Abrir Cuenta']/parent::button");  
  
 public static final Target *ACCOUNTCREATED\_LABEL* = Target.*the*("Cuenta creada")  
 .locatedBy("(//p[text()='Cuenta creada'])[1]");  
  
 public static final Target *MAKE\_A\_DEPOSIT\_BUTTON* = Target.*the*("Realizar deposito")  
 .locatedBy("(//div[text()='Realizar depósito']/parent::button)[1]");  
  
 public static final Target *DEPOSIT\_TYPE\_RADIOBUTTON* = Target.*the*("Tipo de deposito {0}")  
 .locatedBy("//div[text()='{0}']/parent::div/parent::label");  
  
 public static final Target *BANK\_RADIOBUTTON* = Target.*the*("Selecciona el banco {0}")  
 .locatedBy("//input[@type='radio' and @value='{0}']/parent::label");  
  
 public static final Target *AMOUNT\_INPUT* = Target.*the*("Monto")  
 .located(By.*id*("inputAmount"));  
  
 public static final Target *CONTINUE\_BUTTON\_2* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *COUNTRY\_SELECT* = Target.*the*("País")  
 .located(By.*id*("sltCountry"));  
  
 public static final Target *IDENTITYTRIBUTTE\_INPUT* = Target.*the*("Número de identificación tributaria (NIT)")  
 .located(By.*id*("inpIdentityTributte"));  
  
 public static final Target *COUNTRYOPERATIONS\_SELECT* = Target.*the*("País de operaciones")  
 .located(By.*id*("sltCountryOperations"));  
  
 public static final Target *AMERICAN\_NATIONALITY\_CHECK* = Target.*the*("Check nacionalidad estadounidense")  
 .locatedBy("//span[contains(text(),'Tienes nacionalidad estadounidense o eres ciudadano(a)')]");  
  
 public static final Target *ID\_NATIONALITY\_EEUU\_INPUT* = Target.*the*("Número de documento fiscal de EEUU")  
 .located(By.*id*("inpHasNationalityEEUU"));  
  
 public static final Target *IS\_INVESTIGATED\_CHECK* = Target.*the*("Check ha sido investigado o procesado")  
 .locatedBy("//span[contains(text(),'Ha sido investigado o procesado')]");  
  
 public static final Target *INVESTIGATED\_DETAILS\_INPUT* = Target.*the*("Detallar delito y estado de la investigación")  
 .located(By.*id*("inpIsInvestigated"));  
  
 public static final Target *CONFIDENTIAL\_INFO\_CHECK* = Target.*the*("Check acepta información es CONFIDENCIAL")  
 .locatedBy("//span[contains(text(),'Acepto que la información brindada tiene carácter de Declaración Jurada CONFIDENCIAL')]");  
  
 public static final Target *MORE\_INFO\_CHECK* = Target.*the*("Check recibir más información")  
 .locatedBy("//span[contains(text(),'Me gustaría recibir información sobre otras oportunidades')]");  
  
 public static final Target *CONTINUE\_BUTTON\_3* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *GENERATE\_TICKET\_BUTTON* = Target.*the*("Generar ticket")  
 .locatedBy("//div[text()='Generar ticket']/parent::button");  
  
 public static final Target *GENERATED\_YOUR\_TICKET\_LABEL* = Target.*the*("¡Listo! Hemos generado tu ticket")  
 .locatedBy("//p[text()='¡Listo! Hemos generado tu ticket']");  
  
 public static final Target *UNDERSTOOD\_BUTTON* = Target.*the*("Entendido")  
 .locatedBy("//div[text()='Entiendo']/parent::button");  
  
 public static final Target *COUNTRY\_ITEM* = Target.*the*("País {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountry']//li[text()='{0}']");  
  
 public static final Target *COUNTRYOPERATIONS\_ITEM* = Target.*the*("País de operaciones {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountryOperations']//li[text()='{0}']");  
  
 public static final Target *ORIGIN\_SELECT* = Target.*the*("Origen del dinero desplegable número {0}")  
 .locatedBy("(//input[@id='sltOrigin'])[{0}]");  
  
 public static final Target *ORIGINOPTIONS\_ITEM* = Target.*the*("Tipo de origen {0} para para el desplegable número {1}")  
 .locatedBy("(//div[@id='dropdown\_options\_sltOrigin']//li[text()='{0}'])[{1}]");  
  
 public static final Target *ADD\_OTHER\_BUTTON* = Target.*the*("Agregar otro origen del dinero")  
 .locatedBy("//div[text()='Agregar otro']/parent::button");  
  
 public static final Target *BANK\_INPUT* = Target.*the*("Entidad financiera {0}")  
 .locatedBy("(//input[@id='inpEntity'])[{0}]");  
  
 public static final Target *BUTTON\_IR\_INICIO* = Target.*the*("Botón ir a inicio")  
 .locatedBy("(//span[text()='Ir al Inicio'])[1]");  
  
 public static final Target *MESSAGE\_PROCESS\_SOLICITUD* = Target.*the*("Mensaje ya tiene solicitud proceso")  
 .locatedBy("(//div[contains(text(),'ya tienes una solicitud de apertura')])[1]");  
  
}

ESCENARIO 5

PASO 1

@Given("que el usuario está en la página de inicio de sesión")  
public void queElUsuarioEstáEnLaPáginaDeInicioDeSesión() {  
 *theActorCalled*("user").wasAbleTo(OpenUrl.*loginAFP*());  
}

package tasks.login;  
  
import net.serenitybdd.annotations.Step;  
import net.serenitybdd.model.environment.ConfiguredEnvironment;  
import net.serenitybdd.model.environment.EnvironmentSpecificConfiguration;  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Task;  
import net.serenitybdd.screenplay.actions.Open;  
import net.thucydides.model.util.EnvironmentVariables;  
import static net.serenitybdd.screenplay.Tasks.*instrumented*;  
  
public class OpenUrl implements Task {  
  
 private final String url;  
 private final EnvironmentVariables environmentVariables;  
  
 public OpenUrl(String url) {  
 this.url = url;  
 this.environmentVariables = ConfiguredEnvironment.*getEnvironmentVariables*();  
 }  
  
 public static Task loginAFP() {  
 String url = "login.url";  
 return *instrumented*(OpenUrl.class, url);  
 }  
  
 @Override  
 @Step("{0} Inicia la página #url")  
 public <T extends Actor> void performAs(T actor) {  
  
 String targetUrl = EnvironmentSpecificConfiguration.*from*(environmentVariables).getProperty(url);  
  
 if (targetUrl == null) {  
 throw new IllegalArgumentException("URL not found in environment variables: " + url);  
 } else {  
 actor.attemptsTo(Open.*url*(targetUrl));  
 }  
 }  
}

PASO2

@And("el usuario selecciona el tipo de documento {string}")  
public void elUsuarioSeleccionaElTipoDeDocumento(String typeDocument) {  
 *theActorInTheSpotlight*().attemptsTo(  
 WaitFor.*seconds*(5),  
 ClickElement.*on*(LoginPage.*DOCUMENT\_TYPE\_DROPDOWN*,false),  
 WaitFor.*seconds*(5),  
 ClickElement.*on*(LoginPage.*CLICK\_OPTION\_TYPE\_DROPDOWN*,false, typeDocument)  
 );  
}

public static final Target *DOCUMENT\_TYPE\_DROPDOWN* = Target.*the*("Mensaje por usuario incorrecto")  
 .locatedBy("//input[@id='sltDocumentType']");

public static final Target *CLICK\_OPTION\_TYPE\_DROPDOWN* = Target.*the*("Click drop down option")  
 .locatedBy("//li[@data-keyprop='{0}']");

PASO 3

@And("el usuario ingresa el nombre de usuario {string}")  
public void elUsuarioIngresaElNombreDeUsuario(String user) {  
 *theActorInTheSpotlight*().attemptsTo(InputUser.*withData*(user));  
}

package tasks.login;  
  
import net.serenitybdd.annotations.Step;  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Performable;  
import net.serenitybdd.screenplay.Task;  
import net.serenitybdd.screenplay.Tasks;  
import net.serenitybdd.screenplay.actions.Enter;  
import net.serenitybdd.screenplay.matchers.WebElementStateMatchers;  
import net.serenitybdd.screenplay.waits.WaitUntil;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import ui.CookiesPage;  
  
public class InputUser implements Task {  
  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(InputUser.class);  
 private final String enterUser;  
  
 public InputUser(String enterUser) {  
 this.enterUser = enterUser;  
 }  
  
 public static Performable withData(String enterUser) {  
 return Tasks.*instrumented*(InputUser.class, enterUser);  
 }  
  
 @Override  
 @Step("{0} ingresa usuario")  
 public <T extends Actor> void performAs(T actor) {  
 long endTime = System.*currentTimeMillis*() + 10000; // 10 segundos  
 boolean entered = false;  
  
 while (System.*currentTimeMillis*() < endTime && !entered) {  
 try {  
  
 if (CookiesPage.*USERNAME\_FIELD*.resolveFor(actor).isVisible()) {  
 actor.attemptsTo(  
 WaitUntil.*the*(CookiesPage.*USERNAME\_FIELD*, WebElementStateMatchers.*isClickable*()).forNoMoreThan(15).seconds(),  
 Enter.*theValue*(enterUser).into(CookiesPage.*USERNAME\_FIELD*)  
 );  
 entered = true;  
 } else {  
 *LOGGER*.info("El campo de entrada de usuario no está visible inicialmente.");  
 }  
 } catch (Exception e) {  
 *LOGGER*.warn("Intento fallido de ingresar el usuario. Reintentando...", e);  
 try {  
 Thread.*sleep*(500); // Espera medio segundo antes de reintentar  
 } catch (InterruptedException ie) {  
 Thread.*currentThread*().interrupt();  
 *LOGGER*.error("El hilo fue interrumpido", ie);  
 }  
 }  
 }  
  
 if (!entered) {  
 *LOGGER*.error("No se pudo ingresar el usuario después de 10 segundos.");  
 }  
 }  
}

public static final Target *USERNAME\_FIELD* = Target.*the*("Enter Username")  
 .located(org.openqa.selenium.By.*id* ("inpDocumentNumber"));

PASO4

@And("el usuario ingresa la contraseña {string}")  
public void elUsuarioIngresaLaContraseña(String password) {  
 *theActorInTheSpotlight*().attemptsTo(InputPassword.*withData*(password));  
}

package tasks.login;  
  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Task;  
import net.serenitybdd.screenplay.actions.Enter;  
import net.serenitybdd.screenplay.matchers.WebElementStateMatchers;  
import net.serenitybdd.screenplay.waits.WaitUntil;  
import ui.CookiesPage;  
  
public class InputPassword implements Task {  
  
 private final String enterPassword;  
  
 public InputPassword(String enterPassword) {  
 this.enterPassword = enterPassword;  
 }  
  
 public static InputPassword withData(String enterPassword) {  
 return new InputPassword(enterPassword);  
 }  
  
 @Override  
 public <T extends Actor> void performAs(T actor) {  
 actor.attemptsTo(  
 WaitUntil.*the*(CookiesPage.*PASSWORD\_FIELD*, WebElementStateMatchers.*isVisible*()).forNoMoreThan(15).seconds(),  
 Enter.*theValue*(enterPassword).into(CookiesPage.*PASSWORD\_FIELD*));  
 }  
}

public static final Target *PASSWORD\_FIELD* = Target.*the*("Enter Password")  
 .located(org.openqa.selenium.By.*id*("inpPassword"));

PASO5

@And("el usuario hace clic en el botón de inicio de sesión")  
public void elUsuarioHaceClicEnElBotónDeInicioDeSesión() {  
 *theActorInTheSpotlight*().attemptsTo(  
 ClickElement.*on*(LoginPage.*LOGIN\_BUTTON*,false)  
 );  
}

public static final Target *LOGIN\_BUTTON* = Target.*the*("Click on the login button")  
 .located(By.*xpath*("//button[@type='submit']"));

PASO6

@And("el usuario cierra va ventana modal informativa")  
public void el\_usuario\_cierra\_va\_ventana\_modal\_informativa() {  
 *theActorInTheSpotlight*().attemptsTo(WaitFor.*seconds*(10));  
 *theActorInTheSpotlight*().attemptsTo(CloseModalPresent.*close*());  
}

package tasks.dashboard;  
  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Task;  
import net.serenitybdd.screenplay.actions.Click;  
import net.serenitybdd.screenplay.actions.Scroll;  
import net.serenitybdd.screenplay.conditions.Check;  
import ui.DashboardPage;  
  
public class CloseModalPresent implements Task {  
 @Override  
 public <T extends Actor> void performAs(T actor) {  
 actor.attemptsTo(  
 Check.*whether*(DashboardPage.*MODAL\_BUTTON*.resolveFor(actor).isVisible())  
 .andIfSo(  
 Scroll.*to*(DashboardPage.*MODAL\_BUTTON*),  
 Click.*on*(DashboardPage.*MODAL\_BUTTON*)  
 )  
 );  
 }  
  
 // Mé*todo estático para facilitar la creación de la tarea*  
public static CloseModalPresent close() {  
 return new CloseModalPresent();  
 }  
}

public static final Target *MODAL\_BUTTON* = Target.*the*("Botón del modal")  
 .locatedBy("//div[contains(@class,'IsOpen')]//button");

PASO7

@And("el usuario hace clic en el botón Mis trámites del menu")  
public void elUsuarioIngresaElNombreDeUsuario() {  
 *theActorInTheSpotlight*().attemptsTo(new ButtonMyProcedures());  
}

package tasks.changeFund;  
  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Task;  
import net.serenitybdd.screenplay.actions.Click;  
import net.serenitybdd.screenplay.waits.WaitUntil;  
import ui.DashboardPage;  
  
import static net.serenitybdd.screenplay.matchers.WebElementStateMatchers.*isVisible*;  
  
public class ButtonMyProcedures implements Task {  
  
 @Override  
 public <T extends Actor> void performAs(T actor) {  
  
 actor.attemptsTo(  
 WaitUntil.*the*(DashboardPage.*MY\_PROCEDURES\_BUTTON*, *isVisible*()).forNoMoreThan(15).seconds(),  
 Click.*on*(DashboardPage.*MY\_PROCEDURES\_BUTTON*));  
 }  
}

public static final Target *MY\_PROCEDURES\_BUTTON* = Target.*the*("Mis trámites")  
 .locatedBy("//a[@href='/mis-tramites']");

PASO8

@And("el usuario selecciona la opción Abre o deposita a tu cuenta de Aportes Voluntarios")  
public void elusuarioseleccionalaopciónAbreodepositaatucuentadeAportesVoluntarios() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 DashboardPage.*MY\_PROCEDURES\_OPTIONS\_BUTTON*,  
 false,  
 "Abre o deposita a tu cuenta de Aportes Voluntarios"));  
}

public static final Target *MY\_PROCEDURES\_OPTIONS\_BUTTON* = Target.*the*("Mis trámites opción: {0}")  
 .locatedBy("//div[text()='{0}']/../..//button[not(@disabled)]");

PASO9

@And("el usuario hace clic a Empecemos")  
public void elUsuarioHaceClicAEmpecemos() {  
 // theActorInTheSpotlight().attemptsTo(ClickElement.on(  
 // VoluntaryContributionsPage.LETS\_BEGIN\_BUTTON,  
 // false));  
 *theActorInTheSpotlight*().attemptsTo(  
 WaitUntil.*the*(VoluntaryContributionsPage.*LETS\_BEGIN\_BUTTON*, *isVisible*())  
 .forNoMoreThan(30).seconds(), // Espera dinámica de hasta 15 segundos  
 ClickElement.*on*(VoluntaryContributionsPage.*LETS\_BEGIN\_BUTTON*, false)  
 );  
  
  
}

public static final Target *LETS\_BEGIN\_BUTTON* = Target.*the*("Empecemos")  
 .locatedBy("//div[text()='Empecemos']/parent::button");

PASO 10

@And("el usuario selecciona Abrir Una Cuenta {string}")  
public void elUsuarioSeleccionaAbrirUnaCuenta(String tipo\_cuenta) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*SELECT\_VOLUNTARYCONTRIBUTIONTYPE\_BUTTON*,  
 false,  
 tipo\_cuenta));  
}

public static final Target *SELECT\_VOLUNTARYCONTRIBUTIONTYPE\_BUTTON* = Target.*the*("Selecciona el tipo {0}")  
 .locatedBy("//div[contains(@class,'i-hide-mobile')]//p[text()='{0}']/parent::div/following-sibling::button");

PASO 11

@And("el usuario hace clic en Continuar")  
public void elUsuarioHaceClicEnContinuar() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CONTINUE\_BUTTON*,  
 true  
 ));  
}

public static final Target *CONTINUE\_BUTTON* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");

PASO 12

@And("el usuario elige el tipo de fondo {string}")  
public void elUsuarioEligeElTipoDeFondo(String tipo\_fondo) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CHECKBOX\_TYPEFUND*,  
 true,  
 tipo\_fondo));  
}

public static final Target *CHECKBOX\_TYPEFUND* = Target.*the*("Tipo de fondo {0}")  
 .locatedBy("//h3[contains(text(),'{0}')]");

PASO 13

@And("el usuario hace clic en Abrir Cuenta")  
public void elUsuarioHaceClicEnAbrirCuenta() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*BUTTON\_OPENACCOUNT*,  
 true  
 ));  
}

public static final Target *BUTTON\_OPENACCOUNT* = Target.*the*("Abrir Cuenta")  
 .locatedBy("//div[text()='Abrir Cuenta']/parent::button");

PASO 14

@Then("el usuario valida que la cuenta se haya creado correctamente")  
public void elUsuarioValidaQueLaCuentaSeHayaCreadoCorrectamente() {  
 *theActorInTheSpotlight*().should(*seeThat*(ElementVisible.*isVisible*(  
 VoluntaryContributionsPage.*ACCOUNTCREATED\_LABEL*), *is*(true)));  
}

public static final Target *ACCOUNTCREATED\_LABEL* = Target.*the*("Cuenta creada")  
 .locatedBy("(//p[text()='Cuenta creada'])[1]");

PASO 15

@And("el usuario hace clic en Realizar depósito")  
public void elUsuarioHaceClicEnRealizarDepósito() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*MAKE\_A\_DEPOSIT\_BUTTON*,  
 false  
 ));  
}

public static final Target *MAKE\_A\_DEPOSIT\_BUTTON* = Target.*the*("Realizar deposito")  
 .locatedBy("(//div[text()='Realizar depósito']/parent::button)[1]");

PASO 16

@And("el usuario elige el tipo de deposito {string}")  
public void elUsuarioEligeElTipoDeDeposito(String tipo\_deposito) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*DEPOSIT\_TYPE\_RADIOBUTTON*,  
 false,  
 tipo\_deposito  
 ));  
}

public static final Target *DEPOSIT\_TYPE\_RADIOBUTTON* = Target.*the*("Tipo de deposito {0}")  
 .locatedBy("//div[text()='{0}']/parent::div/parent::label");

PASO 17 OPCIONAL

@And("el usuario elige el tipo de deposito {string}")  
public void elUsuarioEligeElTipoDeDeposito(String tipo\_deposito) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*DEPOSIT\_TYPE\_RADIOBUTTON*,  
 false,  
 tipo\_deposito  
 ));  
}

public static final Target *DEPOSIT\_TYPE\_RADIOBUTTON* = Target.*the*("Tipo de deposito {0}")  
 .locatedBy("//div[text()='{0}']/parent::div/parent::label");

PASO 18

@And("el usuario elige el banco {string}")  
public void elUsuarioEligeElBanco(String banco) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*BANK\_RADIOBUTTON*,  
 false,  
 banco  
 ));  
}

public static final Target *BANK\_RADIOBUTTON* = Target.*the*("Selecciona el banco {0}")  
 .locatedBy("//input[@type='radio' and @value='{0}']/parent::label");

PASO 19

@And("el usuario ingresa el monto que desea aportar de {string} soles")  
public void elUsuarioIngresaElMontoQueDeseaAportarDeSoles(String monto) {  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(  
 VoluntaryContributionsPage.*AMOUNT\_INPUT*,  
 monto.trim()  
 ));  
}

public static final Target *AMOUNT\_INPUT* = Target.*the*("Monto")  
 .located(By.*id*("inputAmount"));

PASO 20

@And("el usuario hace clic en Continuar para depositar")  
public void elUsuarioHaceClicEnContinuarParaDepositar() {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(  
 VoluntaryContributionsPage.*CONTINUE\_BUTTON\_2*,  
 false  
 ));  
}

public static final Target *CONTINUE\_BUTTON\_2* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");

PASO 21

@And("el usuario selecciona {string} como país donde vive actualmente y el número de nit {string} si se requiera")  
public void elUsuarioSeleccionaComoPaísDondeViveActualmenteYElNúmeroDeNitSiSeRequiera(String pais, String nit) {  
  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRY\_SELECT*,false));  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRY\_ITEM*,false, pais));  
  
 if (!nit.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*IDENTITYTRIBUTTE\_INPUT*, nit.trim()));  
 }  
  
}

public static final Target *COUNTRY\_SELECT* = Target.*the*("País")  
 .located(By.*id*("sltCountry"));

public static final Target *COUNTRY\_ITEM* = Target.*the*("País {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountry']//li[text()='{0}']");

public static final Target *IDENTITYTRIBUTTE\_INPUT* = Target.*the*("Número de identificación tributaria (NIT)")  
 .located(By.*id*("inpIdentityTributte"));

PASO 22

@And("el usuario selecciona {string} como país de origen del dinero que usará para sus operaciones")  
public void elUsuarioSeleccionaComoPaísDeOrigenDelDineroQueUsaráParaSusOperaciones(String pais) {  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRYOPERATIONS\_SELECT*,false));  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*COUNTRYOPERATIONS\_ITEM*,false, pais));  
}

public static final Target *COUNTRYOPERATIONS\_SELECT* = Target.*the*("País de operaciones")  
 .located(By.*id*("sltCountryOperations"));

public static final Target *COUNTRYOPERATIONS\_ITEM* = Target.*the*("País de operaciones {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountryOperations']//li[text()='{0}']");

PASO 23

@And("el usuario selecciona {string} como origen del dinero")  
public void elUsuarioSeleccionaComoOrigenDelDinero(String valor\_origen) {  
 *theActorInTheSpotlight*().attemptsTo(SelectOriginOfMoney.*withValues*(valor\_origen));  
}

package tasks.voluntaryContributions;  
  
import net.serenitybdd.screenplay.Actor;  
import net.serenitybdd.screenplay.Task;  
import tasks.common.ClickElement;  
import tasks.common.EnterText;  
import ui.VoluntaryContributionsPage;  
  
public class SelectOriginOfMoney implements Task {  
  
 private final String valorOrigen;  
  
 public SelectOriginOfMoney(String valorOrigen) {  
 this.valorOrigen = valorOrigen;  
 }  
  
 @Override  
 public <T extends Actor> void performAs(T actor) {  
 String[] origenes = valorOrigen.split(",");  
  
 for (int i = 0; i < origenes.length; i++) {  
 String[] partes = origenes[i].split("-");  
 String tipoOrigen = partes[0].trim();  
 String banco = partes[1].trim();  
  
 // Seleccionar el tipo de origen  
 actor.attemptsTo(  
 ClickElement.*on*(VoluntaryContributionsPage.*ORIGIN\_SELECT*, true, String.*valueOf*(i + 1)),  
 ClickElement.*on*(VoluntaryContributionsPage.*ORIGINOPTIONS\_ITEM*, true, tipoOrigen, String.*valueOf*(i + 1))  
 );  
  
 // Ingresar el banco  
 actor.attemptsTo(  
 EnterText.*into*(VoluntaryContributionsPage.*BANK\_INPUT*.of(String.*valueOf*(i + 1)), banco)  
 );  
  
 // Hacer clic en "Agregar otro" si es necesario  
 if (i < origenes.length - 1) {  
 actor.attemptsTo(  
 ClickElement.*on*(VoluntaryContributionsPage.*ADD\_OTHER\_BUTTON*, true)  
 );  
 }  
 }  
 }  
  
 public static SelectOriginOfMoney withValues(String valorOrigen) {  
 return new SelectOriginOfMoney(valorOrigen);  
 }  
}

public static final Target *ORIGIN\_SELECT* = Target.*the*("Origen del dinero desplegable número {0}")  
 .locatedBy("(//input[@id='sltOrigin'])[{0}]");

public static final Target *ORIGINOPTIONS\_ITEM* = Target.*the*("Tipo de origen {0} para para el desplegable número {1}")  
 .locatedBy("(//div[@id='dropdown\_options\_sltOrigin']//li[text()='{0}'])[{1}]");

public static final Target *BANK\_INPUT* = Target.*the*("Entidad financiera {0}")  
 .locatedBy("(//input[@id='inpEntity'])[{0}]");

public static final Target *ADD\_OTHER\_BUTTON* = Target.*the*("Agregar otro origen del dinero")  
 .locatedBy("//div[text()='Agregar otro']/parent::button");

PASO 24

@And("el usuario marca el check de nacionalidad americana si tiene documento fiscal de EEUU {string}")  
public void elUsuarioMarcaElCheckDeNacionalidadAmericanaSiTieneDocumentoFiscalDeEEUU(String documento\_eeuu) {  
 if (!documento\_eeuu.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*AMERICAN\_NATIONALITY\_CHECK*,false));  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*ID\_NATIONALITY\_EEUU\_INPUT*, documento\_eeuu));  
 }  
}

public static final Target *AMERICAN\_NATIONALITY\_CHECK* = Target.*the*("Check nacionalidad estadounidense")  
 .locatedBy("//span[contains(text(),'Tienes nacionalidad estadounidense o eres ciudadano(a)')]");

public static final Target *ID\_NATIONALITY\_EEUU\_INPUT* = Target.*the*("Número de documento fiscal de EEUU")  
 .located(By.*id*("inpHasNationalityEEUU"));

PASO 25

@And("el usuario marca el check de Ha sido investigado si tiene algún delito y lo detalla {string}")  
public void elUsuarioMarcaElCheckDeHaSidoInvestigadoSiTieneAlgúnDelitoYLoDetalla(String delitos) {  
 if (!delitos.trim().isEmpty()){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*IS\_INVESTIGATED\_CHECK*,false));  
 *theActorInTheSpotlight*().attemptsTo(EnterText.*into*(VoluntaryContributionsPage.*INVESTIGATED\_DETAILS\_INPUT*, delitos));  
 }  
}

public static final Target *IS\_INVESTIGATED\_CHECK* = Target.*the*("Check ha sido investigado o procesado")  
 .locatedBy("//span[contains(text(),'Ha sido investigado o procesado')]");

public static final Target *INVESTIGATED\_DETAILS\_INPUT* = Target.*the*("Detallar delito y estado de la investigación")  
 .located(By.*id*("inpIsInvestigated"));

PASO 26

@And("el usuario marca el check de Acepto que la información es Confidencial {string}")  
public void elUsuarioMarcaElCheckDeAceptoQueLaInformaciónEsConfidencial(String acepta\_info\_conf) {  
 if (acepta\_info\_conf.trim().equalsIgnoreCase("SI")){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*CONFIDENTIAL\_INFO\_CHECK*,false));  
 }  
}

public static final Target *CONFIDENTIAL\_INFO\_CHECK* = Target.*the*("Check acepta información es CONFIDENCIAL")  
 .locatedBy("//span[contains(text(),'Acepto que la información brindada tiene carácter de Declaración Jurada CONFIDENCIAL')]");

PASO 27

@And("el usuario marca {string} en el check de Me gustaría recibir más información sobre oportunidades de ahorro")  
public void elUsuarioMarcaEnElCheckDeMeGustaríaRecibirMásInformaciónSobreOportunidadesDeAhorro(String mas\_informacion) {  
 if (mas\_informacion.trim().equalsIgnoreCase("SI")){  
 *theActorInTheSpotlight*().attemptsTo(ClickElement.*on*(VoluntaryContributionsPage.*MORE\_INFO\_CHECK*,false));  
 }  
}

public static final Target *MORE\_INFO\_CHECK* = Target.*the*("Check recibir más información")  
 .locatedBy("//span[contains(text(),'Me gustaría recibir información sobre otras oportunidades')]");

Adicional UI de abrir cuenta

package ui;  
  
import net.serenitybdd.screenplay.targets.Target;  
import org.openqa.selenium.By;  
  
public class VoluntaryContributionsPage {  
 public static final Target *LETS\_BEGIN\_BUTTON* = Target.*the*("Empecemos")  
 .locatedBy("//div[text()='Empecemos']/parent::button");  
  
 public static final Target *SELECT\_VOLUNTARYCONTRIBUTIONTYPE\_BUTTON* = Target.*the*("Selecciona el tipo {0}")  
 .locatedBy("//div[contains(@class,'i-hide-mobile')]//p[text()='{0}']/parent::div/following-sibling::button");  
  
 public static final Target *CONTINUE\_BUTTON* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *CHECKBOX\_TYPEFUND* = Target.*the*("Tipo de fondo {0}")  
 .locatedBy("//h3[contains(text(),'{0}')]");  
  
 public static final Target *BUTTON\_OPENACCOUNT* = Target.*the*("Abrir Cuenta")  
 .locatedBy("//div[text()='Abrir Cuenta']/parent::button");  
  
 public static final Target *ACCOUNTCREATED\_LABEL* = Target.*the*("Cuenta creada")  
 .locatedBy("(//p[text()='Cuenta creada'])[1]");  
  
 public static final Target *MAKE\_A\_DEPOSIT\_BUTTON* = Target.*the*("Realizar deposito")  
 .locatedBy("(//div[text()='Realizar depósito']/parent::button)[1]");  
  
 public static final Target *DEPOSIT\_TYPE\_RADIOBUTTON* = Target.*the*("Tipo de deposito {0}")  
 .locatedBy("//div[text()='{0}']/parent::div/parent::label");  
  
 public static final Target *BANK\_RADIOBUTTON* = Target.*the*("Selecciona el banco {0}")  
 .locatedBy("//input[@type='radio' and @value='{0}']/parent::label");  
  
 public static final Target *AMOUNT\_INPUT* = Target.*the*("Monto")  
 .located(By.*id*("inputAmount"));  
  
 public static final Target *CONTINUE\_BUTTON\_2* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *COUNTRY\_SELECT* = Target.*the*("País")  
 .located(By.*id*("sltCountry"));  
  
 public static final Target *IDENTITYTRIBUTTE\_INPUT* = Target.*the*("Número de identificación tributaria (NIT)")  
 .located(By.*id*("inpIdentityTributte"));  
  
 public static final Target *COUNTRYOPERATIONS\_SELECT* = Target.*the*("País de operaciones")  
 .located(By.*id*("sltCountryOperations"));  
  
 public static final Target *AMERICAN\_NATIONALITY\_CHECK* = Target.*the*("Check nacionalidad estadounidense")  
 .locatedBy("//span[contains(text(),'Tienes nacionalidad estadounidense o eres ciudadano(a)')]");  
  
 public static final Target *ID\_NATIONALITY\_EEUU\_INPUT* = Target.*the*("Número de documento fiscal de EEUU")  
 .located(By.*id*("inpHasNationalityEEUU"));  
  
 public static final Target *IS\_INVESTIGATED\_CHECK* = Target.*the*("Check ha sido investigado o procesado")  
 .locatedBy("//span[contains(text(),'Ha sido investigado o procesado')]");  
  
 public static final Target *INVESTIGATED\_DETAILS\_INPUT* = Target.*the*("Detallar delito y estado de la investigación")  
 .located(By.*id*("inpIsInvestigated"));  
  
 public static final Target *CONFIDENTIAL\_INFO\_CHECK* = Target.*the*("Check acepta información es CONFIDENCIAL")  
 .locatedBy("//span[contains(text(),'Acepto que la información brindada tiene carácter de Declaración Jurada CONFIDENCIAL')]");  
  
 public static final Target *MORE\_INFO\_CHECK* = Target.*the*("Check recibir más información")  
 .locatedBy("//span[contains(text(),'Me gustaría recibir información sobre otras oportunidades')]");  
  
 public static final Target *CONTINUE\_BUTTON\_3* = Target.*the*("Continuar")  
 .locatedBy("//div[text()='Continuar']/parent::button");  
  
 public static final Target *GENERATE\_TICKET\_BUTTON* = Target.*the*("Generar ticket")  
 .locatedBy("//div[text()='Generar ticket']/parent::button");  
  
 public static final Target *GENERATED\_YOUR\_TICKET\_LABEL* = Target.*the*("¡Listo! Hemos generado tu ticket")  
 .locatedBy("//p[text()='¡Listo! Hemos generado tu ticket']");  
  
 public static final Target *UNDERSTOOD\_BUTTON* = Target.*the*("Entendido")  
 .locatedBy("//div[text()='Entiendo']/parent::button");  
  
 public static final Target *COUNTRY\_ITEM* = Target.*the*("País {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountry']//li[text()='{0}']");  
  
 public static final Target *COUNTRYOPERATIONS\_ITEM* = Target.*the*("País de operaciones {0}")  
 .locatedBy("//div[@id='dropdown\_options\_sltCountryOperations']//li[text()='{0}']");  
  
 public static final Target *ORIGIN\_SELECT* = Target.*the*("Origen del dinero desplegable número {0}")  
 .locatedBy("(//input[@id='sltOrigin'])[{0}]");  
  
 public static final Target *ORIGINOPTIONS\_ITEM* = Target.*the*("Tipo de origen {0} para para el desplegable número {1}")  
 .locatedBy("(//div[@id='dropdown\_options\_sltOrigin']//li[text()='{0}'])[{1}]");  
  
 public static final Target *ADD\_OTHER\_BUTTON* = Target.*the*("Agregar otro origen del dinero")  
 .locatedBy("//div[text()='Agregar otro']/parent::button");  
  
 public static final Target *BANK\_INPUT* = Target.*the*("Entidad financiera {0}")  
 .locatedBy("(//input[@id='inpEntity'])[{0}]");  
  
 public static final Target *BUTTON\_IR\_INICIO* = Target.*the*("Botón ir a inicio")  
 .locatedBy("(//span[text()='Ir al Inicio'])[1]");  
  
 public static final Target *MESSAGE\_PROCESS\_SOLICITUD* = Target.*the*("Mensaje ya tiene solicitud proceso")  
 .locatedBy("(//div[contains(text(),'ya tienes una solicitud de apertura')])[1]");  
  
}