

## QA Automation assessment

### Exercices:

1 - There is an application at Factorial Calculator (<http://qainterview.pythonanywhere.com/>) perform the following activities:

a. The app has been seeded with a variety of bugs, find the most you can, describing how you have detected them, classify them by severity.

bug	descripcion	severidad
Numeros mayores a 170	Muestra como resultado "infinito". Esto se debe a que con un numero mayor a 170, alcanza el desbordamiento. No es correcto que se visualice infinito. Deberia implementarse una funcion para calculo de numeros mayores a 170, o en su defecto visualizar otro mensaje.	alta
Numeros menores a 0 y mayores a 989	No devuelve nada, dejando el mensaje del ingreso anterior. La API devuelve " 500 internal server error". Deberia visualizarse otro mensaje.	alta

### Evidencia:

Numeros mayores a 170:

Calculate!

The factorial of 171 is: Infinity

Numeros menores a 0:

Calculate!

The factorial of 2 is: 2

```
-1 (index):65
x POST http://gainterview.pythonanywhere.com/factorial 500 (INTERNAL SERVER ERROR) jquery-1.12.4.js:10254
send @ jquery-1.12.4.js:10254
ajax @ jquery-1.12.4.js:9738
(anonymous) @ (index):79
dispatch @ jquery-1.12.4.js:5226
elemData.handle @ jquery-1.12.4.js:4878
```

Numeros mayores a 989:

Calculate!

The factorial of 2 is: 2

```
990 (index):65
x POST http://gainterview.pythonanywhere.com/factorial 500 (INTERNAL SERVER ERROR) jquery-1.12.4.js:10254
send @ jquery-1.12.4.js:10254
ajax @ jquery-1.12.4.js:9738
(anonymous) @ (index):79
dispatch @ jquery-1.12.4.js:5226
elemData.handle @ jquery-1.12.4.js:4878
```

b. The app also has several workflows which will work well that you should check, please describe them.

Flujo de trabajo normal:

+ Ingresando numeros enteros mayores o iguales a 0 y menores o iguales a 989, arroja como resultado el factorial de dicho número.

+ Ingresando cualquier otro valor distinto a un entero (letras, simbolos, flotante, o dejando vacio), muestra un mensaje "Please enter an integer" y resaltando el campo de ingreso en rojo.

c. Write a locator (CSS selector/XPath) for the red form validation styling

Localizando el elemento de entrada de texto:



Xpath: `//*[@id="number"]`

Localizando el elemento “mensaje de error” :

Please enter an integer

xpath :

`//*[@id="resultDiv"]`

d. Find the console message printed

The greatest factorial calculator!



Calculate!

The factorial of 2 is: 2

2	<a href="#">(index):65</a>
Hello! I am in the done part of the ajax call	<a href="#">(index):80</a>
▼ {answer: 2} ⓘ	<a href="#">(index):81</a>
answer: 2	
▼ __proto__:	
▶ constructor: f Object()	
▶ hasOwnProperty: f hasOwnProperty()	
▶ isPrototypeOf: f isPrototypeOf()	
▶ propertyIsEnumerable: f propertyIsEnumerable()	
▶ toLocaleString: f toLocaleString()	
▶ toString: f toString()	
▶ valueOf: f valueOf()	
▶ __defineGetter__: f __defineGetter__()	
▶ __defineSetter__: f __defineSetter__()	
▶ __lookupGetter__: f __lookupGetter__()	
▶ __lookupSetter__: f __lookupSetter__()	
▶ get __proto__: f __proto__()	
▶ set __proto__: f __proto__()	
>	

e. Figure out the API call being made along with the headers and parameters sent

×	Headers	Preview	Response	Cookies	Timing
▼	General				
	Request URL: http://qainterview.pythonanywhere.com/factorial				
	Request Method: POST				
	Status Code: 🟢 200 OK				
	Remote Address: 35.173.69.207:80				
	Referrer Policy: no-referrer-when-downgrade				
▼	Response Headers	view source			
	Connection: keep-alive				
	Content-Length: 69				
	Content-Type: application/json				
	Date: Sun, 22 Sep 2019 22:53:09 GMT				
	Server: openresty/1.9.15.1				
	X-Clacks-Overhead: GNU Terry Pratchett				
▼	Request Headers	view source			
	Accept: */*				
	Accept-Encoding: gzip, deflate				
	Accept-Language: es-ES,es;q=0.9,gl;q=0.8,en;q=0.7				
	Connection: keep-alive				
	Content-Length: 9				
	Content-Type: application/x-www-form-urlencoded; charset=UTF-8				
	Cookie: _ga=GA1.2.1262661179.1569150927; _gid=GA1.2.1628094824.1569150927				
	Host: qainterview.pythonanywhere.com				
	Origin: http://qainterview.pythonanywhere.com				
	Referer: http://qainterview.pythonanywhere.com/				
	User-Agent: Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36				
	X-Requested-With: XMLHttpRequest				
▼	Form Data	view source	view URL encoded		
	number: 45				

2- Please access the following sample application -  
<http://computerdatabase.herokuapp.com/computers>

a. Create a series of manual test cases that cover the crud operation plus the edge cases.  
Make sure you give detailed instructions for each test case.

### **Feature: buscar una computadora**

**Yo como** interesada de un producto del sitio web  
**Quiero** poder buscar una computadora  
**Para** ver las características de la misma

#### **@Regresion**

**Escenario:** buscar una computadora existente

**given** estoy en el sitio web

**when** ingreso el nombre de una computadora en la barra de búsqueda

**and** hago click en el boton "filter by name"

**then** el producto relacionado a la palabra ingresada debe ser mostrada como resultado

**Escenario:** buscar una computadora que no existe

**given** estoy en el sitio web

**when** ingreso el nombre de una computadora en la barra de búsqueda

**and** hago click en el boton "filter by name"

**and** el producto relacionado a la palabra ingresada no existe

**then** debe mostrar el mensaje "no computer found"

**Escenario:** no ingreso ninguna palabra en la barra de búsqueda

**given** estoy en el sitio web

**when** no ingreso ninguna palabra en la barra de búsqueda

**and** hago click en el boton "filter by name"

**then** debe mostrar el listado de todos los productos existentes

### **Feature: agregar una computadora**

**yo** como usuaria del sitio web  
**quiero** poder agregar una computadora  
**para** ver las características de la misma

#### **@Regresion**

**Escenario:** agregar una nueva computadora

**given** estoy en el sitio web

**when** hago click en el boton "add a new computer"

**and** en el campo "computer name" ingreso el nombre

**and** en el campo "introduced date" ingreso una fecha

**and** en el campo "discontinued date" ingreso una fecha

**and** en el listado "company" selecciono una opcion

**and** hago click en el boton "create this computer"

**then** debe mostrar el mensaje "Done! Computer has been created"

**Escenario:** no ingreso valor en el campo obligatorio

**given** estoy en el sitio web

**when** hago click en el boton "add a new computer"

**and** dejo vacio el campo "computer name"

**and** en el campo "introduced date" ingreso una

**and** en el campo "discontinued date" ingreso una fecha

**and** en el listado "company" selecciono una opcion

**and** hago click en el boton "create this computer"

**then** debe mostrar el mensaje "Requerid" en el campo "computer name"

**Escenario:** cancelar antes de agregar la computadora

**given** estoy en el sitio web

**when** hago click en el boton "add a new computer"

**and** hago click en el boton "cancel"

**then** debe redireccionarme a la pagina de busqueda

**And** mostrar el listado de computadoras.

**Escenario:** formato inválido de fecha

**given** estoy en el sitio web

**when** hago click en el boton "add a new computer"

**and** en el campo "computer name" ingreso el nombre de una computadora

**and** en el campo "introduced date" ingreso cualquier valor distinto al formato de fecha 'yyyy-mm-dd'

**and** en el campo "discontinued date" ingreso una fecha

**and** en el listado "company" selecciono una opcion

**and** hago click en el boton "create this computer"

**then** debe mostrar el mensaje "Date ('yyyy-mm-dd')" en el campo "introduced date"

**Escenario:** fecha "discuntinued date" menor que la fecha "introduced date"

**given** estoy en el sitio web

**when** hago click en el boton "add a new computer"

**and** en el campo "computer name" ingreso el nombre de una computadora

**and** en el campo "introduced date" ingreso una fecha

**and** en el campo "discontinued date" ingreso una fecha menor o igual a la fecha ingresada en "introduced date"

**and** en el listbox "company" selecciono la opcion "<company>"

**and** hago click en el boton "create this computer"

**then** debe mostrar el mensaje "Discontinued date must be greater than Introduced date"

## **Feature: editar una computadora**

**yo** como usuaria del sitio web

**quiero** poder editar una computadora

**para** visualizar las características actualizadas

### **@Regresion**

**Escenario:** editar una computadora existente

**given** estoy en el sitio web

**when** selecciono una computadora del listado

**and** en el campo "computer name" cambio el nombre existente

**and** en el campo "introduced date" cambio fecha  
**and** en el campo "discontinued date" cambio la fecha  
**and** en el listado "company" selecciono una opcion  
**and** hago click en el boton "save this computer"  
**then** debe mostrar el mensaje "Done! Computer has been update"

**Escenario:** no ingreso valor en el campo obligatorio

**given** estoy en el sitio web  
**when** selecciono una computadora del listado  
**and** en el campo "computer name" cambio el nombre existente por campo vacio  
**and** en el campo "introduced date" cambio fecha  
**and** en el campo "discontinued date" cambio la fecha  
**and** en el listbox "company" selecciono una opcion  
**and** hago click en el boton "save this computer"  
**then** debe mostrar el mensaje en rojo "Requerid" en el campo "computer name"

**Escenario:** cancelar antes de editar la computadora

**given** estoy en el sitio web  
**when** selecciono una computadora del listado  
**and** hago click en el boton "cancel"  
**then** debe redireccionarme a la pagina principal  
**And** mostrar el listado de computadoras.

**Escenario:** formato invalido de fecha

**given** estoy en el sitio web  
**when** selecciono una computadora del listado  
**and** en el campo "computer name" cambio el nombre existente  
**and** en el campo "introduced date" cambio fecha por cualquier valor con formato distinto a 'yyyy-mm-dd'  
**and** en el campo "discontinued date" cambio la fecha  
**and** en el listado "company" selecciono una opcion  
**and** hago click en el boton "save this computer"  
**then** debe mostrar el mensaje "Date ('yyyy-MM-dd')" en el campo "introduced date"

**Escenario:** editar la fecha "discontinued date" con una fecha menor que la fecha "introduced date"

**given** estoy en el sitio web  
**when** selecciono una computadora del listado  
**and** en el campo "computer name" cambio el nombre existente  
**and** en el campo "introduced date" cambio la fecha  
**and** en el campo "discontinued date" cambio la fecha por una menor o igual a la fecha "introduced date"  
**and** en el listbox "company" selecciono la opcion "<company>"  
**and** hago click en el boton "save this computer"  
**then** debe mostrar el mensaje "Discontinued date must be greater than introduced date"

## **Feature: eliminar una computadora**

**yo** como usuaria del sitio web  
**quiero** poder eliminar una computadora  
**para** visualizar las computadoras actualizadas

### @Regresion

**Escenario:** eliminar una computadora seleccionada

**given** estoy en el sitio web

**when** selecciono una computadora del listado

**and** hago click en el boton "delete this computer"

**then** mostrar el mensaje "Done! Computer has been deleted"

b. Write scripts that would automate the manual test cases that you see fit to be included in a regression test set. Please use any of below programming languages:

- Javascript
- Java (Please avoid frameworks that only record test cases.)

### Casos de prueba automatizados incluidos en la regresion:

Feature 1: Buscar una computadora

**Escenario:** buscar una computadora existente

Feature 2: Agregar una computadora

**Escenario:** agregar una nueva computadora

Feature 3: Editar una computadora

**Escenario:** editar una computadora existente

Feature 4: Eliminar una computadora

**Escenario:** eliminar una computadora seleccionada

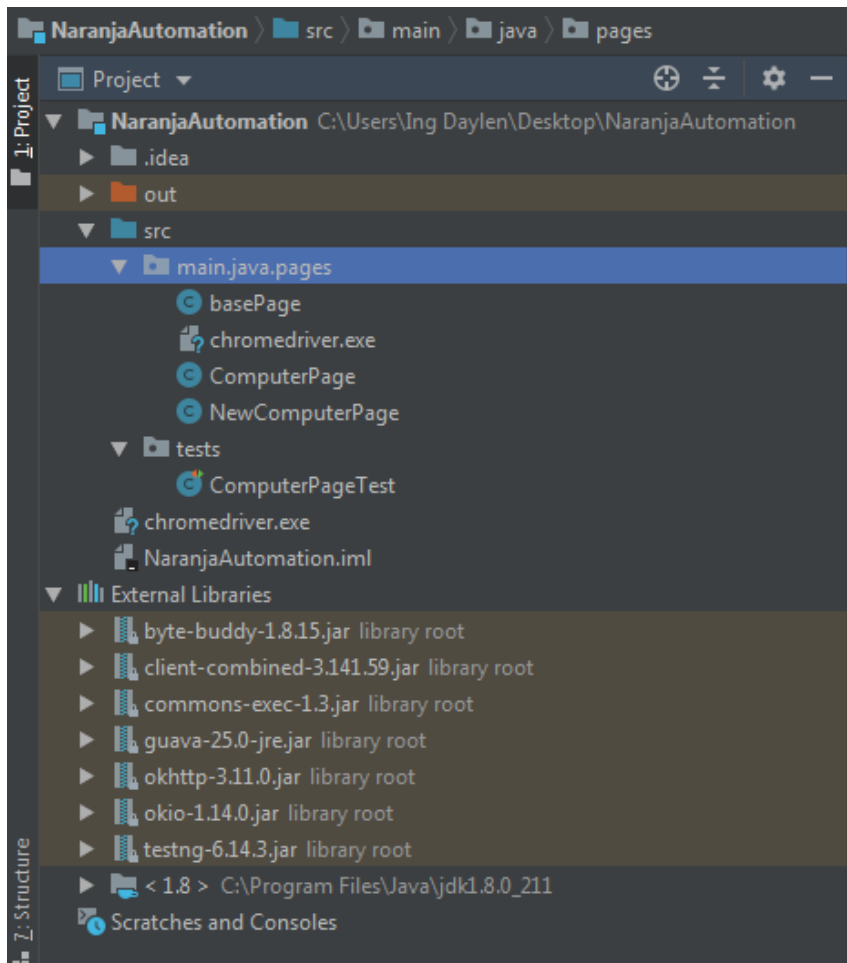
Framework utilizado:

Para la automatizacion de las pruebas se utilizò Selenium WebDriver, mediante el patron Page Object Model

Como framework de ejecucion de pruebas se utilizò TestNG.

Estructura del proyecto:





## Còdigo:

### Clase basePage:

```
package main.java.pages;
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import java.util.concurrent.TimeUnit;

public class basePage{
    private WebDriver driver;
    public basePage(WebDriver driver)
    {
        this.driver= driver;
    }

    public WebDriver chromeDriverConection() {
        System.setProperty("webdriver.chrome.driver", "chromedriver.exe");
        driver = new ChromeDriver();
        return driver;
    }

    public WebElement findElement(By locator)
    {
        return driver.findElement(locator);
    }
}
```

```

public List<WebElement> findElements(By locator)
{
    return driver.findElements(locator);
}

public String getText(WebElement element)
{
    return element.getText();
}

public String getText (By locator)
{
    return driver.findElement(locator).getText();
}

public void writeText(Keys text, By locator)
{
    driver.findElement(locator).sendKeys(text);
}

public void writeText(String text, By locator)
{
    driver.findElement(locator).sendKeys(text);
}

public void clearText (By locator){
    driver.findElement(locator).clear();
}

public void click(By locator)
{
    driver.findElement(locator).click();
}

public Boolean isDisplayed(By locator)
{
    try
    {
        return driver.findElement(locator).isDisplayed();
    }
    catch (org.openqa.selenium.NoSuchElementException e )
    {
        return false;
    }
}

public void waitTime(int seconds )
{
    driver.manage().timeouts().implicitlyWait(seconds, TimeUnit.SECONDS);
}

public void openURL(String url)
{
    driver.get(url);
}

```

## Clase ComputerPage

```
package main.java.pages;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import java.util.List;
public class ComputerPage extends basePage{

    public ComputerPage(WebDriver driver) {
        super(driver);
    }

    By searchLocator = By.id("searchbox");
    By searchButtonLocator = By.id ("searchsubmit");
    By messageLocator = By.xpath("//*[@id=\"main\"] /h1");

    public void search( ) {
clearText(searchLocator);
        writeText("TOSHIBA", searchLocator);
        click(searchButtonLocator);
        waitTime(10);
    }

    public String messageText()
    {
        return getText(messageLocator);
    }
}
```

## Clase NewComputerPage

```
package main.java.pages;
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.Select;

public class NewComputerPage extends basePage{

    public NewComputerPage(WebDriver driver) {
        super(driver);
    }
private WebDriver driver;

    By addButtonLocator = By.id ("add");
    By titlePage = By.xpath ("//*[@id=\"main\"] /h1");
    By computerNameLocator = By.id ("name");
    By introducedDateLocator = By.id ("introduced");
    By discontinuedDateLocator = By.id ("discontinued");
    By dropdownListCompanyLocator= By.name("company");
    By createButtonLocator= By.xpath("//*[@id=\"main\"] /form/div/input");
    By messAlert = By.xpath("//*[@id=\"main\"] /div[1]");
```

```

String url="http://computer-database.herokuapp.com/computers";

By elementTableLocator= By.xpath( "//*[@id=\"main\"] /table/tbody/tr/td[1]/a");
By saveButtonLocator= By.xpath("//*[@id=\"main\"] /form[1]/div/input");

By deleteButtonLocator = By.xpath("//*[@id=\"main\"] /form[2]/input");


    public void returnToHomePage()
    {
        openURL(url);
    }

    public void addComputer(){
        click(addButtonLocator);
        waitTime(20);

        if (isDisplayed(titlePage) ){
            writeText("TOSHIBA", computerNameLocator);
            writeText("1998-12-12", introducedDateLocator);
            writeText("1999-10-11", discontinuedDateLocator);
            selectDropDownListCompany();
            click(createButtonLocator);

        }
        getText(messAlert);
    }


    public String messageText()
    {
        return getText(messAlert);
    }

    public String getText()
    {
        return driver.findElement(messAlert).getText();
    }
    public void editComputer()

    {
        if (isDisplayed(titlePage)) {
            click(elementTableLocator);
            clearText(discontinuedDateLocator);

            writeText("2019-10-11", discontinuedDateLocator);
            click(saveButtonLocator);
            waitTime(10);

        }
    }

    public String selectDropDownListCompany()
    {
        Select selectList= new Select(findElement(dropDownListCompanyLocator));
        selectList.selectByVisibleText("Apple Inc.");
        return getText(selectList.getFirstSelectedOption());
    }

```

```

public void deleteComputer()

{
    if (isDisplayed(titlePage)){
        click(elementTableLocator);
        click(deleteButtonLocator);
        waitTime(16);
    }
}
}

```

## Clase ComputerPageTest

```

package tests;
import main.java.pages.ComputerPage;
import main.java.pages.NewComputerPage;
import org.openqa.selenium.WebDriver;

import org.testng.annotations.BeforeClass;
import org.testng.annotations.AfterClass;
import org.testng.annotations.Test;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertTrue;

public class ComputerPageTest {

    private WebDriver driver;
    ComputerPage computerPage;
    NewComputerPage newComputerPage;

    @BeforeClass
    public void setUp() {

        computerPage= new ComputerPage(driver);
        driver= computerPage.chromeDriverConnection();
        computerPage.openURL("http://computer-database.herokuapp.com/computers");

        //driver.manage().timeouts().implicitlyWait(15,SECONDS);
        driver.manage().window().maximize();
    }

    @Test (priority = 3)
    //1) buscar computadora y
    public void findComputer() throws InterruptedException {
        computerPage.openURL("http://computer-database.herokuapp.com/computers");

        computerPage= new ComputerPage(driver);
        computerPage.search();
        assertEquals(computerPage.messageText(), "One computer found");
    }

    //2) agregar computadora
    @Test (priority = 1)
    public void addComputer() throws InterruptedException {

        newComputerPage = new NewComputerPage(driver);
        newComputerPage.addComputer();
    }
}

```

```

        assertTrue(newComputerPage.messageText().contains("has been created"));
    }
    //3) editar computadora
    @Test (priority = 2)
    public void editComputer() throws InterruptedException {

        newComputerPage = new NewComputerPage(driver);
        computerPage= new ComputerPage(driver);
        computerPage.search();
        newComputerPage.editComputer();
        assertTrue(newComputerPage.messageText().contains("has been updated"));

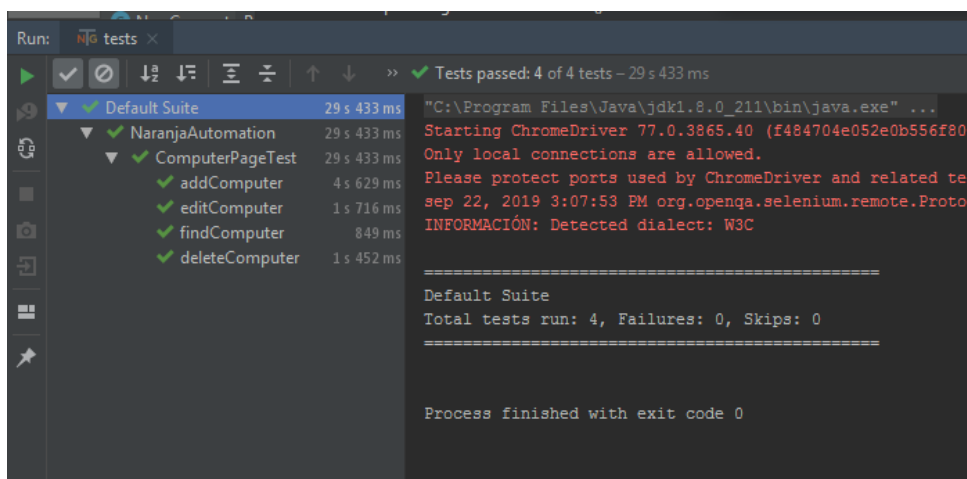
    }
    //4) eliminar computadora

    @Test (priority = 4)
    public void deleteComputer() throws InterruptedException {
        newComputerPage = new NewComputerPage(driver);
        computerPage.search();
        newComputerPage.deleteComputer();
        assertEquals(newComputerPage.messageText(),"Done! Computer has been
deleted");
    }

    @AfterClass
    public void tearDown() {
        driver.quit();
    }
}

```

Resultado de la ejecucion del suite de test:



c. Push the file containing the manual test cases and the automation project to GitHub

<https://github.com/daidaragona/ProyectSelenium>