

AUTOMATIZACION II / MOBILE

EJERCICIO 1

1. Crear un documento PDF con capturas de pantalla realizando paso a paso la automatización de la creación actualización y borrado de una tarea de la aplicación When.do

Solución. -

- a. En la clase *Android* se realiza la definición de características principales de la aplicación mobile *WhenDo*

```
11 public class Android implements IDevice{
12     @Override
13     public AppiumDriver create() {
14         DesiredCapabilities capabilities = new DesiredCapabilities();
15         capabilities.setCapability( capabilityName: "deviceName", value: "V9");
16         capabilities.setCapability( capabilityName: "platformVersion", value: "9");
17         // Paquetes de la aplicacion movile whendo
18         capabilities.setCapability( capabilityName: "appPackage", value: "com.vrproductiveapps.whendo");
19         // Actividad de la aplicacion movile whendo
20         capabilities.setCapability( capabilityName: "appActivity", value: "com.vrproductiveapps.whendo.ui.HomeActivity");
21         capabilities.setCapability( capabilityName: "platformName", value: "Android");
22         AppiumDriver driver= null;
23         try {
24             driver = new AndroidDriver(new URL( spec: "http://127.0.0.1:4723/wd/hub"),capabilities);
25         } catch (MalformedURLException e) {
26             throw new RuntimeException(e);
27         }
28         driver.manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);
29         return driver;
30     }
31 }
```

- b. En la clase *CreateTaskWhenDoTest* ubicado en el paquete *cleanTest*, se desarrolla la automatización de la creación actualización y borrado de una tarea de la aplicación When.do

1. Definición de Variables

```
19 // Definicion de variables
20 String tittle = "Mobile Exercise";
21 String note = "This is the execution of the mobile exercise by xsoto;);";
22 String update = " UPDATE";
23 String messageErrorMainScreen = "Error: the note was not create";
24 String messageErrorTaskScreen = "Error: the task was not create";
25 String messageErrorDeleteTask = "Error: La tarea no fue eliminada";
26 String alertTitle = "Confirm Delete";
27 String messageTaskAdd = "No tasks added";
28 String messageErrorListTask = "Error al listar las tareas";
```

2. Creación

```
32 //Registro de la tarea
33 mainScreen.addButton.click();
34 createNoteForm.tittleTxtBox.setText(tittle);
35 createNoteForm.noteTxtBox.setText(note);
36 createNoteForm.saveButtonBox.click();
```

3. Verificación de la existencia de la tarea

```
38 //Verificacion de la existencia de la tarea
39 Assertions.assertTrue(mainScreen.isNoteDisplayed(tittle), messageErrorMainScreen);
40 Session.getInstnce().getDriver().manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);
```

4. Actualización de la tarea

```

42 //Actualizacion de la tarea
43 mainScreen.labelClicked(tittle);
44 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 10, TimeUnit.SECONDS);
45 createNoteForm.titleTextBox.setText(tittle + update);
46 createNoteForm.saveButtonBox.click();
47 Assertions.assertTrue(mainScreen.isNoteDisplayed( tittle: tittle + update),messageErrorTaskScreen);
48 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);

```

5. Selección de la tarea a ser eliminada

```

50 //Seleccion de la tarea a ser eliminada
51 mainScreen.labelClicked( tittle: tittle + update);
52 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 10, TimeUnit.SECONDS);
53 createNoteForm.deleteButtonBox.click();
54 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);

```

6. Verificación del mensaje de confirmación de la tarea a ser eliminada

```

56 //Verificacion del mensaje de confirmacion de la tarea a ser eliminada
57 Assertions.assertEquals(alertTitle,createNoteForm.alertLabel.getText(),messageErrorDeleteTask);
58 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);

```

7. Eliminación de la tarea

```

60 //Eliminacion de la tarea
61 createNoteForm.deleteAlertButtonBox.click();
62 Session.getInstance().getDriver().manage().timeouts().implicitlyWait( time: 20, TimeUnit.SECONDS);

```

8. Verificación de la tarea eliminada

```

64 //Verificacion de la tarea eliminada
65 Assertions.assertEquals(messageTaskAdd,mainScreen.noTaskLabel.getText(),messageErrorListTask);

```

9. Resultados después de la ejecución del test.

```

Run: CreateTaskWhenDoTest.verifyCreateNewForm (1) x
Tests passed: 1 of 1 test - 1 min 6 sec
Test Results 1 min 6 sec
Starting Gradle Daemon...
Gradle Daemon started in 9 s 45 ms
> Task :compileJava NO-SOURCE
> Task :processResources NO-SOURCE
> Task :classes UP-TO-DATE
> Task :compileTestJava
> Task :processTestResources NO-SOURCE
> Task :testClasses
> Task :test
oct. 16, 2022 9:48:49 P. M. io.appium.java_client.remote.AppiumCommandExecutor$1 lambda$0
INFO: Detected dialect: W3C
BUILD SUCCESSFUL in 2m 12s
2 actionable tasks: 2 executed
21:49:43: Execution finished ':test --tests "CleanTest.CreateTaskWhenDoTest.verifyCreateNewForm"'.

```

2. Subir el código a un repositorio de versionador de código

El código fuente se encuentra versionado en la siguiente dirección

- <https://github.com/xsoto-developer/whenDo>