

# Práctico de Test

**GeneXus™ 16**



Setiembre 2019

**Copyright** ♥ GeneXus S.A. 1988-2019.

*All rights reserved. This document may not be reproduced by any means without the express permission of GeneXus S.A. The information contained herein is intended for personal use only.*

**Registered Trademarks:**

*GeneXus is trademark or registered trademark of GeneXus S.A. All other trademarks mentioned herein are the property of their respective owners.*

| CONTENIDO                                                    |    |
|--------------------------------------------------------------|----|
| Contenido                                                    | 2  |
| Objetivo                                                     | 3  |
| Creación de Unit Test                                        | 3  |
| Configuración en Jenkins                                     | 7  |
| 1- Agregar paso para ejecutar las pruebas                    | 7  |
| 2- Agregar paso para publicar los resultados de la ejecución | 9  |
| 3- Commit de los tests y visualización de los resultados     | 10 |

## OBJETIVO

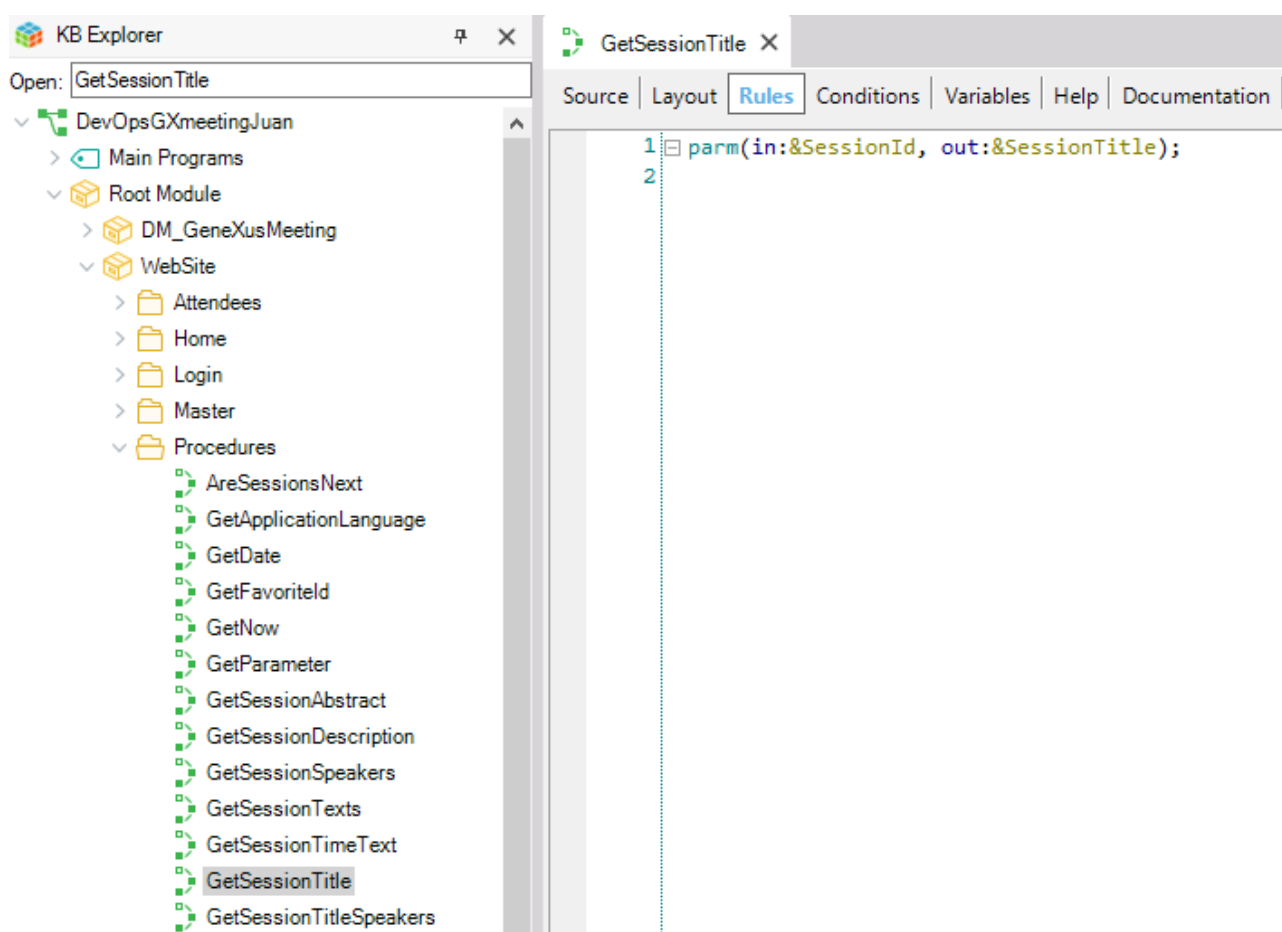
Luego que tenemos configurado y armado todo el proceso de *build* vamos a configurar el proceso de *testing* de nuestra aplicación. Para esto vamos a crear un *unit test* y agregar una tarea en Jenkins para que se corran de forma desatendida las pruebas sobre nuestra aplicación.

Para realizar esta tarea vamos a agregar pasos en la configuración de nuestro proyecto Jenkins en el cual venimos trabajando utilizando DevOps.

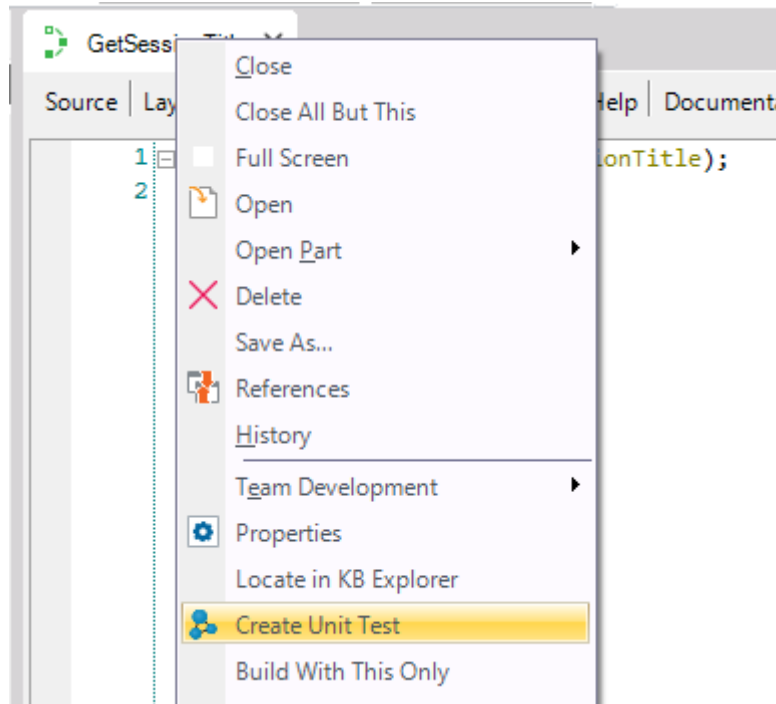
## CREACIÓN DE UNIT TEST

Previo a configurar el proyecto Jenkins vamos a crear una prueba unitaria de un procedimiento de nuestra base de conocimientos.

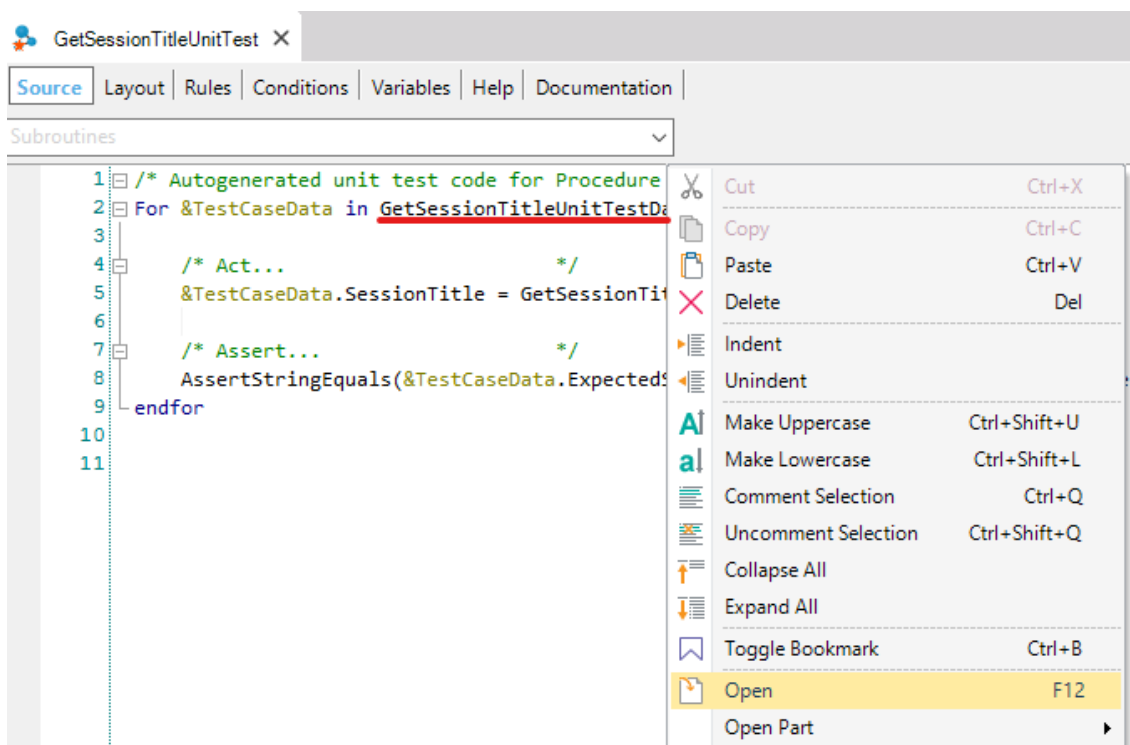
Crearemos la prueba unitaria del procedimiento “*GetSessionTitle*”, el cual recibe como parámetro de entrada “*&SessionId*” y devuelve como parámetro de salida “*&SessionTitle*”.



- Hacer botón derecho sobre el objeto y seleccionar la opción “*Create Unit Test*”.



- Abrir el objeto “*GetSessionTitleUnitTestData*”.



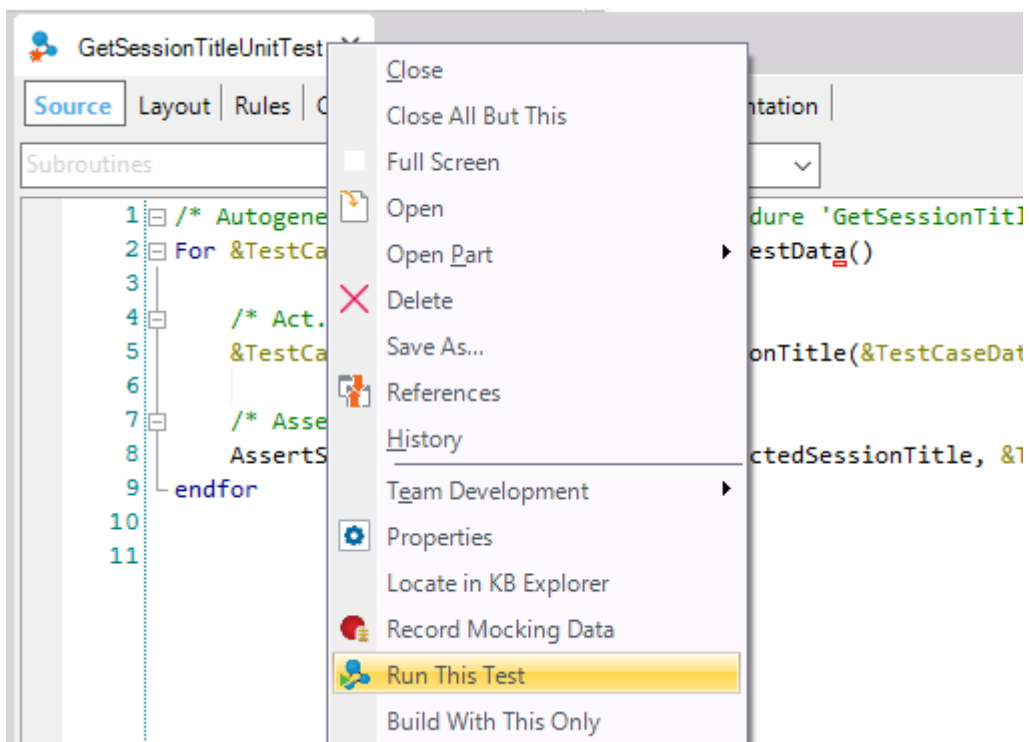
- Copiar los casos de prueba en el Data Provider “*GetSessionTitleUnitTestData*”

GetSessionTitleUnitTestSDT

```
{  
    SessionId = 4448  
    ExpectedSessionTitle = "Tips & trends for smart devices"  
    ErrorMsgSessionTitle = ""  
}  
  
GetSessionTitleUnitTestSDT  
  
{  
    SessionId = 4619  
    ExpectedSessionTitle = "GX28 Mobile: GeneXus 16 en su máxima potencia"  
    ErrorMsgSessionTitle = ""  
}  
  
GetSessionTitleUnitTestSDT  
  
{  
    SessionId = 4597  
    ExpectedSessionTitle = "Next-Gen Trends: NOSQL and Serverless Apps in the Cloud"  
    ErrorMsgSessionTitle = ""  
}  
  
GetSessionTitleUnitTestSDT  
  
{  
    SessionId = 4568  
    ExpectedSessionTitle = "El estado del Mapa GeneXus"  
    ErrorMsgSessionTitle = ""  
}  
  
GetSessionTitleUnitTestSDT
```

```
{
    SessionId = 4591
    ExpectedSessionTitle = "Café con Startups"
    ErrorMsgSessionTitle = ""
}
```

- Seleccionar la opción “Run This Test”



- En el panel “Tests Results” puede ver los resultados de ejecutar el test.

Tests Results

Tests list

| Name                                                                                                                                             | Started at | Elapsed |
|--------------------------------------------------------------------------------------------------------------------------------------------------|------------|---------|
| Passed (1)                                                                                                                                       |            |         |
| <input checked="" type="checkbox"/>  Web Site.GetSessionTitl... | 00:38:40   | 133 ms  |

**Execution detail**

Export

Clear Results

 [Web Site.GetSessionTitleUnit Test](#)  
Started at: Wednesday, September 25, 2019 12:38:40 AM  
Elapsed time: 00:00:00.133

|                                                                                     | Expected               | Obtained                        | Error message |   |
|-------------------------------------------------------------------------------------|------------------------|---------------------------------|---------------|---|
|   | Tips & trends for s... | Tips & trends for smart devices |               | + |
|  | GX28 Mobile: Gen...    | GX28 Mobile: GeneXus 16 en...   |               | + |
|  | Next-Gen Trends: ...   | Next-Gen Trends: NOSQL an...    |               | + |
|  | El estado del Map...   | El estado del Mapa GeneXus      |               | + |
|  | Café con Startups      | Café con Startups               |               | + |

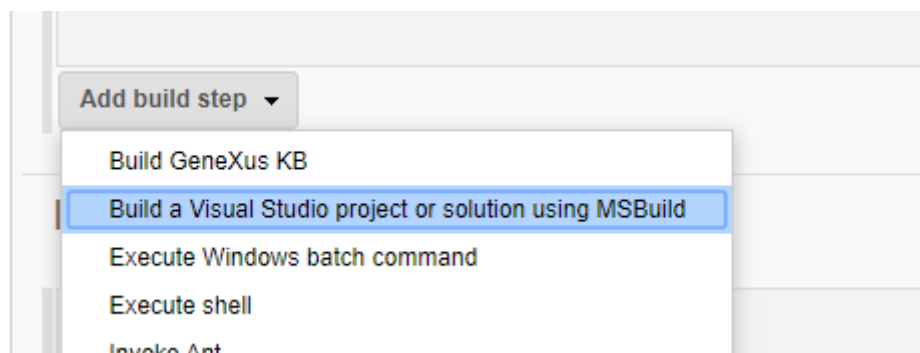
## CONFIGURACIÓN EN JENKINS

Ahora sí, vamos a configurar las tareas de *test* en Jenkins que ejecutarán las pruebas unitarias que creamos cuando hagamos *commit*.

Para esto es necesario agregar un paso en nuestro *pipeline*, que realice la ejecución de las pruebas unitarias. Siga los siguientes puntos para lograrlo:

### 1- AGREGAR PASO PARA EJECUTAR LAS PRUEBAS

En el proyecto MsBuild “*Integration*”, seleccionar el menú “*Configure*”, ir a la sección “*Build*” de las pestañas del proyecto y seleccionar la opción “*Add build step*” de tipo “*Build Visual Studio project or solution using MsBuild*”.



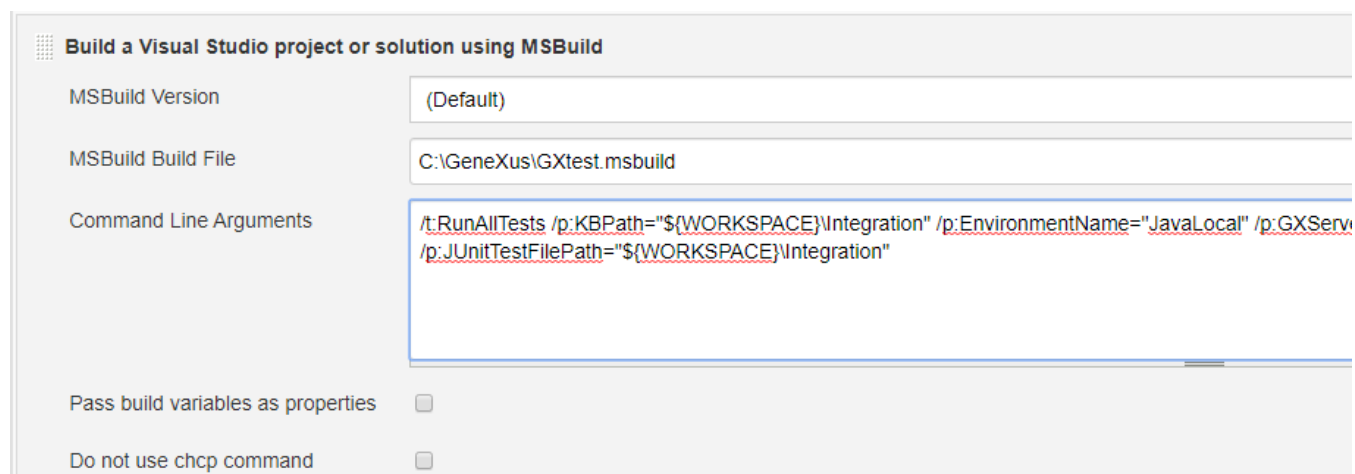
Completar con los siguientes datos:

### MSBuild Build File

“C:\GeneXus\GXtest.msbuild”

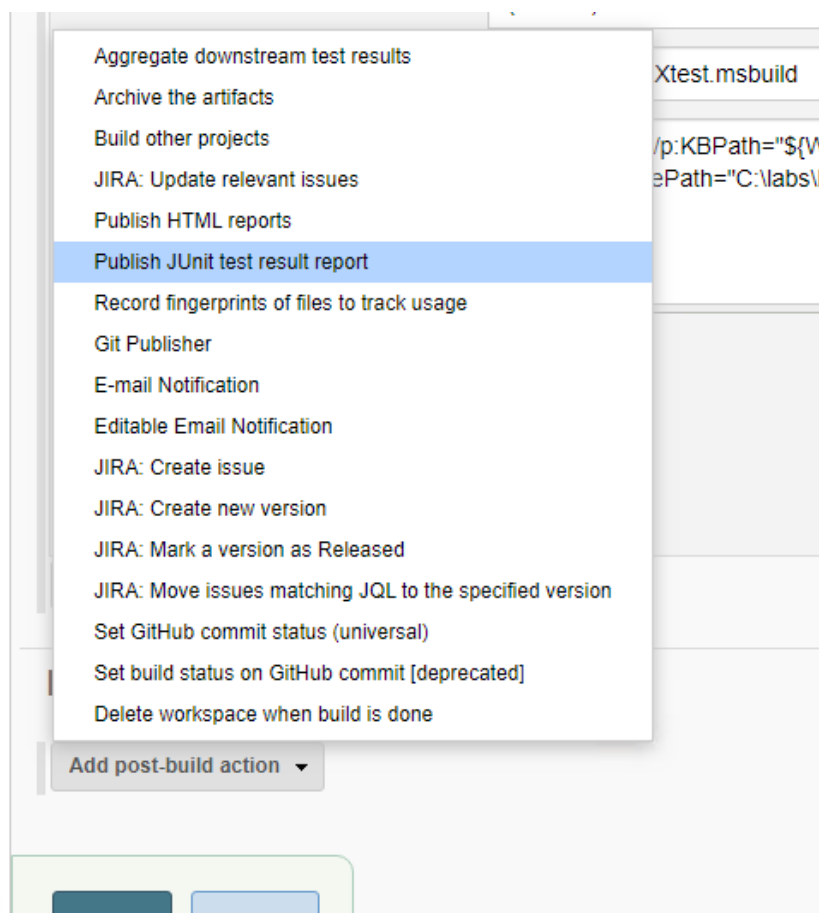
### Command Line Arguments

```
/t:RunAllTests /p:KBPath="${WORKSPACE}\Integration" /p:EnvironmentName="JavaLocal"  
/p:GXServerUser="local\builder" /p:GXServerPass="builder123"  
/p:JUnitTestFilePath="${WORKSPACE}\Integration"
```



## 2- AGREGAR PASO PARA PUBLICAR LOS RESULTADOS DE LA EJECUCIÓN

Ir a la sección del proyecto “*Post-build Actions*”, expandir la opción “*Add post-build action*” y luego seleccionar “*Publish JUnit test result report*”



Completar con el valor “`${WORKSPACE}\Integration\TestResults*.xml`” en el campo “*Test report XMLs*” y salvar los cambios.

### Post-build Actions

Publish JUnit test result report

Test report XMLs

[Fileset 'includes'](#) setting that specifies the generated raw XML report files, such as 'myproject/target/test-reports/\*.xml'. Basedir of the fileset is [the workspace root](#).

☐ Retain long standard output/error

Health report amplification factor

1% failing tests scores as 99% health. 5% failing tests scores as 95% health

Allow empty results

☐ Do not fail the build on empty test results

Add post-build action ▼

Save

Apply

### 3- COMMIT DE LOS TESTS Y VISUALIZACIÓN DE LOS RESULTADOS

Finalmente hacer *commit* de los *tests* y ver los resultados de la ejecución del pipeline.

Al finalizar la ejecución del pipeline se pueden ver los resultados de los tests en la sección “*Test Result*” de ese pipeline.

The screenshot shows the Jenkins web interface in a browser window. The address bar indicates the URL is `localhost:8082/job/Integration/9/testReport/`. The Jenkins logo and name are at the top left, with a search bar and user information 'Bob The Builder | log out' on the top right. A breadcrumb trail shows 'Jenkins > Integration > #9 > Test Results'. On the left, a sidebar contains links to various project views, with 'Test Result' highlighted. The main content area is titled 'Test Result' and shows '0 failures (-1)' with a blue progress bar. To the right of the bar, it says '1 tests (±0)', 'Took 0.5 sec.', and has a link to 'add description'. Below this is a section titled 'All Tests' containing a table with test results.

| Package | Duration | Fail (diff) | Skip (diff) | Pass (diff) | Total (diff) |
|---------|----------|-------------|-------------|-------------|--------------|
| (root)  | 0.5 sec  | 0 -1        | 0           | 1 +1        | 1            |

**GeneXus™ 16**

**GXtest**