

# Ejercicio 1. Preparación del Entorno de Trabajo



Para una buena experiencia en el desarrollo colaborativo de una extensión para Dynamics 365 Business Central, es esencial preparar un buen entorno de trabajo para todos los miembros del equipo, haciendo que todos ellos desarrollen utilizando unas mismas pautas o guías.

En este ejercicio aprenderemos:

- A crear un proyecto a modo de plantilla que posteriormente podremos reutilizar en múltiples proyectos
- A crear un espacio de trabajo que contenga múltiples extensiones AL
- A definir configuraciones compartidas entre los distintos proyectos que conforman un mismo espacio de trabajo
- A crear un proyecto de Test
- A sincronizar nuestro proyecto con un repositorio Git remoto

## Indicaciones

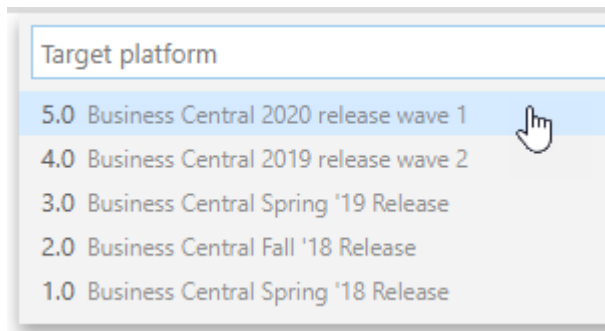
Para completar el ejercicio realiza las siguientes acciones:

1. Clona un repositorio remoto
2. Crea un nuevo proyecto: Extensión principal
3. Crea un segundo proyecto: Extensión de test
4. Crea un espacio de trabajo que incluya las dos extensiones creadas en pasos anteriores
5. Configura el espacio de trabajo para que ambas extensiones se comporten de la misma forma
  - Para que utilicen los mismos símbolos
  - Para que utilicen los distintos analizadores de código
  - Para que utilicen un mismo patrón de nombrado de archivos
  - Etc.
6. Configura el archivo *app.json* de ambas extensiones
7. Haz un commit y sube todos los cambios a un repositorio Git remoto

## Indicaciones paso a paso

Para completar el ejercicio sigue los siguientes pasos en Visual Studio Code:

1. Clonado de un repositorio remoto
  1. Ejecuta el comando *Git: Clone* para clonar el repositorio remoto que encontrarás en [https://ClipScaleUp@dev.azure.com/ClipScaleUp/Proyecto2020/\\_git/Proyecto2020](https://ClipScaleUp@dev.azure.com/ClipScaleUp/Proyecto2020/_git/Proyecto2020)
  2. Guarda el repositorio en *C:/ScaleUp*
2. Creación del proyecto para la Extensión principal
  1. Ejecuta el comando *AL: Go!* para crear un nuevo proyecto y guárdalo en *C:\ScaleUp\Proyecto2020\TusIniciales\MainApp*, seleccionando 5.0 como Target Platform



2. Configura el archivo *launch.json* con los siguientes valores

```
"name": "BC16ES",
"server": "http://localhost",
"serverInstance": "BC160",
"authentication": "Windows",
```

3. Elimina el archivo *HelloWorld.al*

4. Ejecuta el comando *AL: Download Symbols*

5. Publica la extensión

3. Creación del proyecto para la Extensión de test

1. Ejecuta el comando *AL: Go!* para crear un nuevo proyecto y guárdalo en

*C:\ScaleUp\Proyecto2020\TusIniciales\TestApp*, seleccionando 5.0 como Target Platform

2. Elimina el archivo *HelloWorld.al*

3. Edita el archivo *app.json* para añadir dependencias a:

- La extensión principal
- Las extensiones de test

```
{
  "id": "6d25a5ba-89ef-4d9b-8e47-a21dacc99f33",
  "publisher": "ClipPlatform",
  "name": "MainApp",
  "version": "1.0.0.0"
},
{
  "id": "9856ae4f-d1a7-46ef-89bb-6ef056398228",
  "name": "System Application Test Library",
  "publisher": "Microsoft",
  "version": "16.0.0.0"
},
{
  "id": "5d86850b-0d76-4eca-bd7b-951ad998e997",
  "name": "Tests-TestLibraries",
  "publisher": "Microsoft",
  "version": "16.0.0.0"
}
```

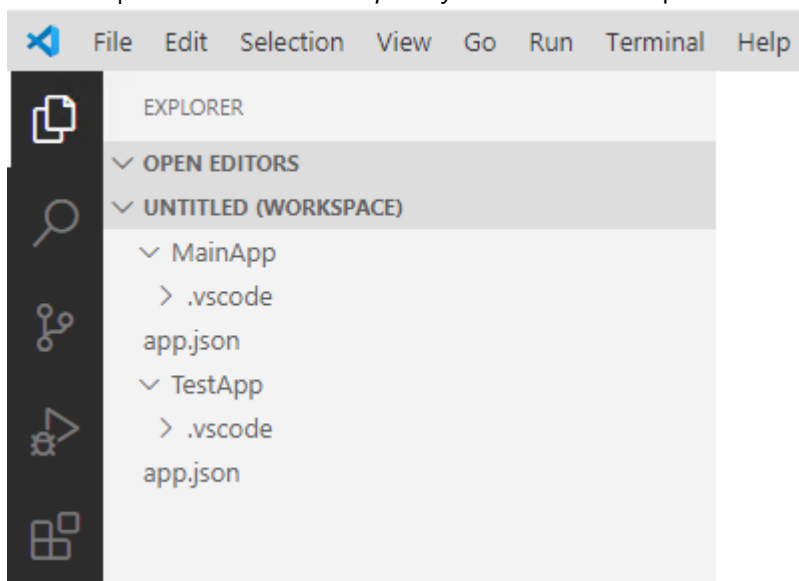
4. Instala las librerías de test en la base de datos (*System Application Test Library*, *Tests-TestLibraries* y *Library Assert*). Para instalarlas se requiere:
  - Tener una licencia de BC16 cargada
  - Tener la dll *MockTest* disponible en el servicio

1. Abre el script `C:\ScaleUp\Testing\InstallTestingBC16.ps1`
2. Ejecuta el script

5. Ejecuta el comando `AL: Download Symbols`

#### 4. Creación de un espacio de trabajo

1. Abre la carpeta de la extensión principal
2. Ve a `File | Add Folder to Workspace` y selecciona la carpeta correspondiente a la extensión de test



3. Ve a `File | Save Workspace As` y guarda el nuevo espacio de trabajo en `C:\ScaleUp\Proyecto2020\TusIniciales`

El sistema habrá creado un archivo `.code-workspace` en el que se hace referencia a las distintas carpetas que conforman el mismo espacio de trabajo

```
{
  "folders": [
    {
      "path": "MainApp"
    },
    {
      "path": "TestApp"
    }
  ],
  "settings": {}
}
```

#### 5. Configuración del espacio de trabajo

1. Abre el archivo `.code-workspace` creado en el punto anterior

2. Añade las siguientes configuraciones a la sección *settings* para activar los distintos analizadores de código

```
"al.enableCodeAnalysis": true,
  "al.codeAnalyzers": [
    "${AppSourceCop}",
    "${CodeCop}",
    "${PerTenantExtensionCop}",
    "${UICop}"
  ]
```

3. Crea un archivo llamado *clip.ruleset.json* con la siguiente información y guárdalo en **\*C:\ScaleUp\Proyecto2020\TusIniciales\***

```
{
  "name": "Clip ruleset",
  "description": "Personalización del analizador de código",
  "rules": [
    {
      "id": "AA0214",
      "action": "None",
      "justification": "Bug"
    }
  ]
}
```

4. Añade la siguiente configuración a la sección *settings* del espacio de trabajo para activar las normas personalizadas del analizador de código

```
"al.ruleSetPath":
"C:/ScaleUp/Proyecto2020/TusIniciales/clip.ruleset.json"
```

5. Añade las siguientes configuraciones a la sección *settings* para activar el renombrado automático de nuestros archivos

```
"CRS.OnSaveAlFileAction": "Rename",
"CRS.RemovePrefixFromFilename": true,
"CRS.FileNamePattern": "<ObjectNameShort>.<ObjectTypeShortPascalCase>.al",
"CRS.FileNamePatternExtensions": "<BaseNameShort>.<ObjectTypeShortPascalCase>.al",
"CRS.ObjectNamePrefix": "Clip"
```

6. Crea un archivo llamado *AppSourceCop.json* con la siguiente información y guárdalo tanto en la carpeta MainApp como en la carpeta TestApp

```
{
  "mandatoryPrefix": "Clip"
}
```

## 6. Configuración del archivo *app.json* de ambas extensiones

1. Actualiza las siguientes propiedades del archivo *app.json* de la extensión principal

```
"publisher": "ClipPlatform",
"brief": "This is our Main Application",
"description": "This is our Main Application",
"privacyStatement": "http://localhost",
"EULA": "http://localhost",
"help": "http://localhost",
"url": "http://localhost",
"logo": "C:/ScaleUp/Image/logo.png",
"idRanges": [
  {
    "from": 50100,
    "to": 50139
  }
],
"features": ["TranslationFile"]
```

2. Actualiza las siguientes propiedades del archivo *app.json* de la extensión de test

```
"publisher": "ClipPlatform",
"brief": "This is our Test Application",
"description": "This is our Test Application",
"privacyStatement": "http://localhost",
"EULA": "http://localhost",
"help": "http://localhost",
"url": "http://localhost",
"logo": "C:/ScaleUp/Image/logo.png",
"idRanges": [
  {
    "from": 50140,
    "to": 50149
  }
],
"features": ["TranslationFile"]
```

## 7. Realización de un commit y sincronización con un repositorio Git remoto

1. Crea un commit con el mensaje "Commit inicial proyecto" que incluya todos los archivos trabajados hasta el momento
2. Sincroniza el repositorio git local con el repositorio git remoto