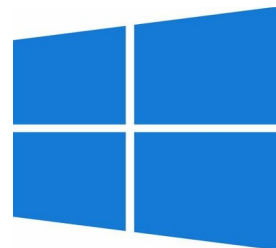
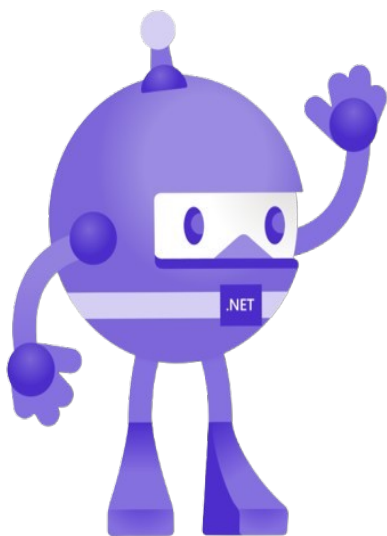


# DESARROLLO DE INTERFACES

Publicar aplicaciones  
desde .NET Maui



Marcos Zahonero

## Índice

<b>Introducción.....</b>	<b>3</b>
<b>Tiempo utilizado.....</b>	<b>4</b>
<b>Material utilizado.....</b>	<b>5</b>
Mi PC de casa.....	5
<b>Objetivos de la práctica.....</b>	<b>6</b>
<b>Desarrollo de la Actividad.....</b>	<b>7</b>
Publicación de aplicaciones en Windows.....	7
Instalar de forma gráfica.....	13
Instalar de forma no gráfica.....	14
Publicación de aplicaciones en Android.....	15
Instalación por APK.....	19
GitHub.....	20
<b>Conclusión.....</b>	<b>21</b>
<b>Problemas y/o sugerencias en la actividad.....</b>	<b>22</b>
1# - Mi extra a la actividad.....	22
2# - No me deja instalar la APK.....	22
<b>Webgrafía.....</b>	<b>23</b>

## **Introducción**

En esta actividad veremos como publicar aplicaciones para Windows desde Visual Studio para .NET Maui con C#, una guía paso a paso de como hacerlo.

## **Tiempo utilizado**

He utilizado 1 hora y media, 1 hora dedicada a hacer la actividad, capturas etc.. y la media hora restante para hacer el documento y un vídeo paso a paso para conseguir algo más visual.

## Material utilizado

- El portátil de clase, con mi ordenador personal de casa.
- Visual Studio

## Mi PC de casa

Componente	Nombre
Sistema Operativo	Windows 11 Pro
Procesador	AMD Ryzen 5 2600X
Tarjeta gráfica	NVIDIA GeForce GTX 1650 4GB
Memoria RAM	16 GB
Placa base	B450M-A PRO MAX
Disco duro 1#	KIOXIA EXCERIA G2 SSD 1 TB
Disco duro 2#	ST1000DM010-2EP102 HDD 1 TB

## Objetivos de la práctica

- Realizar una interfaz con Firebase en MAUI.
- Entender la interfaz de MAUI.

## Desarrollo de la Actividad

### Publicación de aplicaciones en Windows



Vamos a entrar a un proyecto que queramos publicar, en mi caso he creado uno nuevo para que se vea como el “Hello World”, que es la plantilla básica de .NET MAUI y así se publicará más rápidamente, ya que tiene menos cosas por procesar.

Habiendo introducido la actividad vamos a hacer “Clic derecho en la app desde el Explorador de archivos de Visual Studio *(la flecha lo indica)* > **Publicar...**”

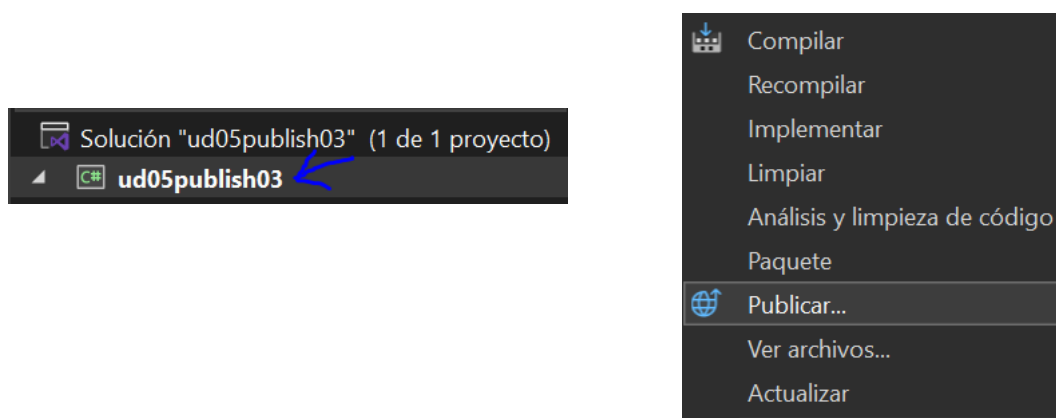


Figura 1: Menu al hacer clic derecho a la solución

Al haberle dado a “**Publicar...**” nos saldrá la siguiente ventana, para elegir el método de distribución, lo dejamos en “**Instalación de prueba**” y habilitamos la casilla de “**Habilitar actualizaciones automáticas**” y le damos a “**Siguiente**”:

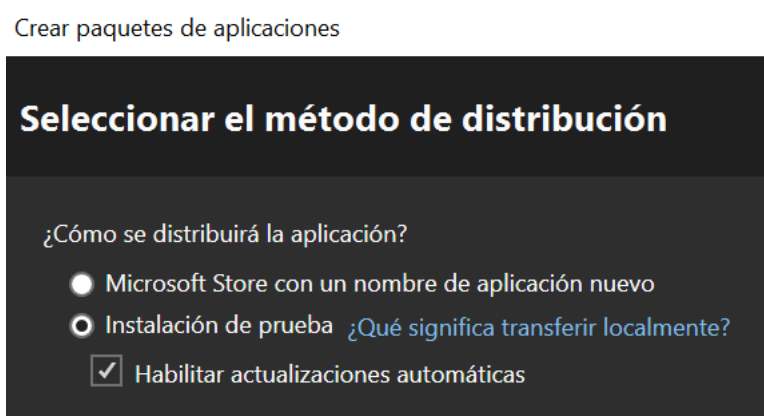


Figura 2: Pantalla para seleccionar el método de distribución

Estamos ahora para seleccionar el método de firma, en nuestro caso lo firmaremos con un certificado, en este caso no deberíamos tener ninguno creado, es por eso que vamos a darle a **“Crear...”**.

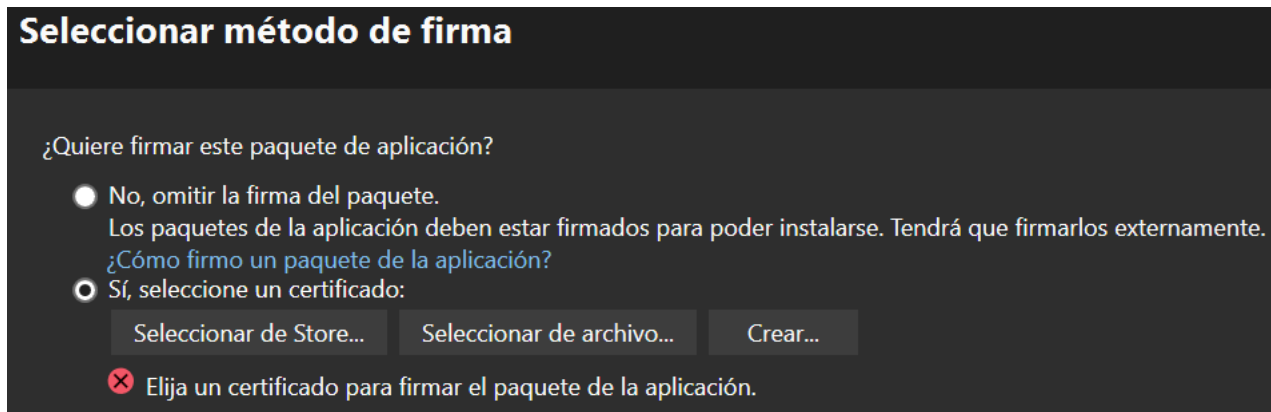


Figura 3: Seleccionar método de firma

En el nombre del certificado podemos poner lo que queramos, pero es mejor intentar asignar un nombre que se parezca al nombre de la actividad para poder identificarlos mejor, después añadimos una contraseña y la confirmamos y le damos a **“Siguiente/Crear”**.

*(Es importante saber la contraseña para futuras pruebas o por si la llega a pedir)*

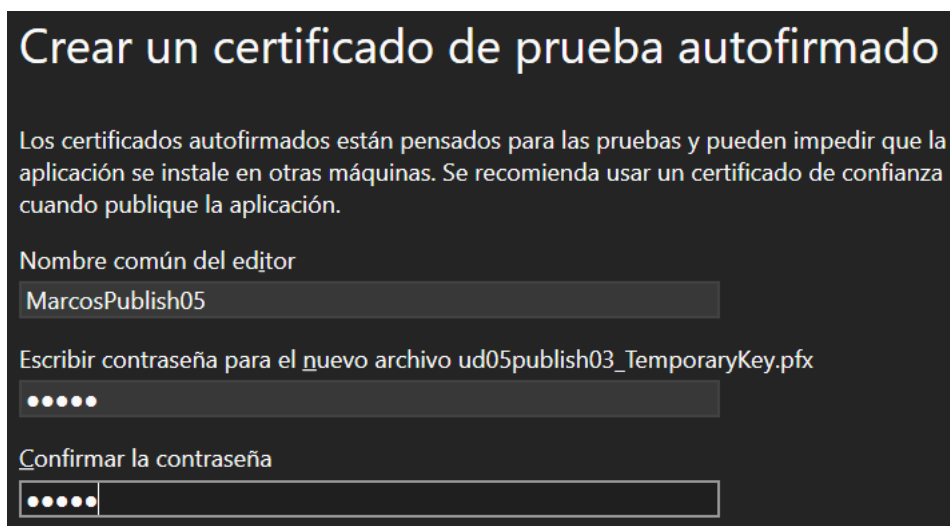
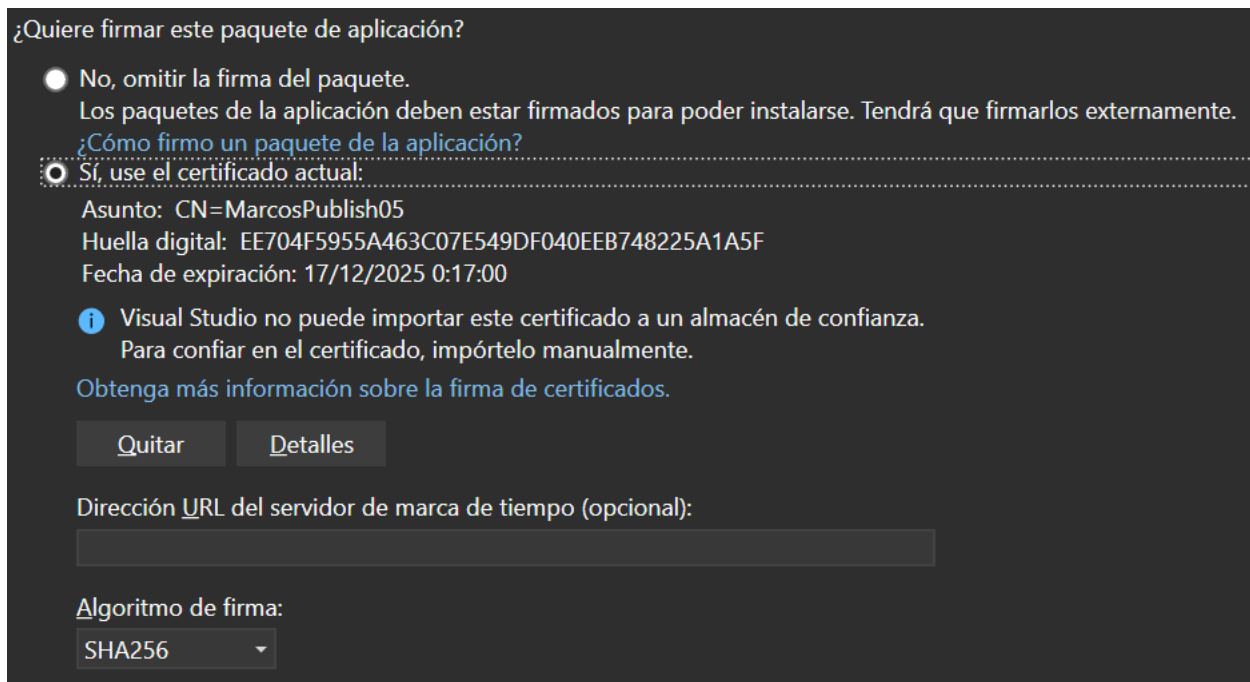


Figura 4: Pantalla para asignar nombre del certificado y contraseña



Ahora estaremos de vuelta en esta pantalla, pero ahora tenemos un certificado así que podremos firmar este paquete de aplicación, le damos **“Siguiente”**.




¿Quiere firmar este paquete de aplicación?

☐ No, omitir la firma del paquete.  
Los paquetes de la aplicación deben estar firmados para poder instalarse. Tendrá que firmarlos externamente.

¿Cómo firmo un paquete de la aplicación?

☒ Sí, use el certificado actual:

Asunto: CN=MarcosPublish05  
Huella digital: EE704F5955A463C07E549DF040EEB748225A1A5F  
Fecha de expiración: 17/12/2025 0:17:00

 Visual Studio no puede importar este certificado a un almacén de confianza.  
Para confiar en el certificado, impórtelo manualmente.

[Obtenga más información sobre la firma de certificados.](#)

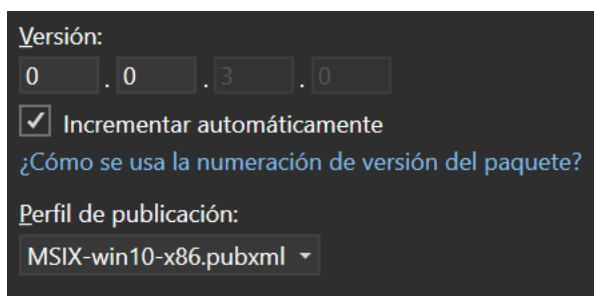
Dirección URL del servidor de marca de tiempo (opcional):

Algoritmo de firma:

Figura 5: Firmamos el paquete de aplicación

Esta parte es algo importante de entender y es que esto de la versión es el encargado de marcar el número de la versión que al usuario le puede llegar a aparecer o a ti mismo para guiarte que versión llevas de la aplicación, en mi caso lo deje incremental ya que no necesito llevar un control de versiones excesivo con esta prueba pero es algo bastante útil.

En cuanto el perfil de publicación tenemos que darle y nos aparece la opción de **“New...”** le damos y seguimos los pasos y después volveremos a esta ventana y le damos a **“Siguiente”**.



Versión:

0 . 0 . 3 . 0

☒ Incrementar automáticamente

[¿Cómo se usa la numeración de versión del paquete?](#)

Perfil de publicación:

Ya estamos a un paso de publicar nuestra aplicación en Windows, ahora en el recuadro de “Ubicación del instalador” debemos seleccionar la ruta para dejar la aplicación con su instalador, esto es similar a [Android Studio](#) por si ya has trabajado en ese programa para que sepas que esto funciona de forma similar.

En mi caso lo dejo en la misma carpeta del proyecto y luego sigo esto (05publish03 es mi nombre de proyecto):

ud05publish03\bin\Debug\net8.0-windows10.0.19041.0\win10-x64\

De ahí ya creará la carpeta **AppPackages**, que ahora veremos más sobre eso, en cuanto a la opción de abajo de “¿Cómo publico mi aplicación?” yo lo deje en Cada 0 días, para evitar sobreescribir la información y también porque no necesitaba que se actualice ya que no iban a haber cambios.

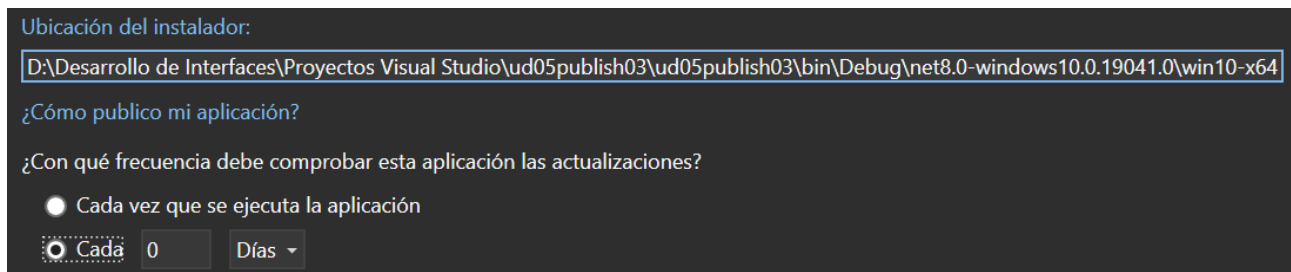
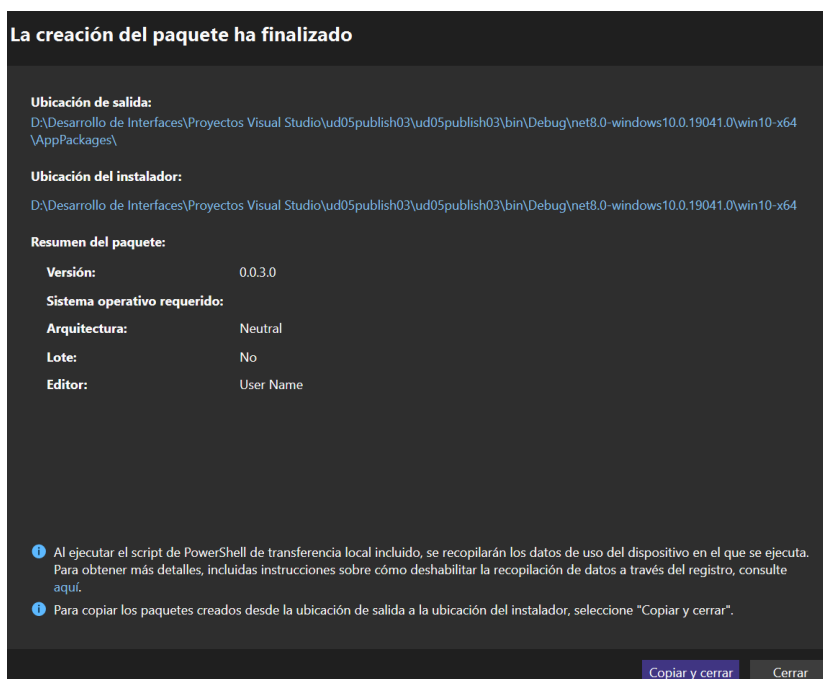
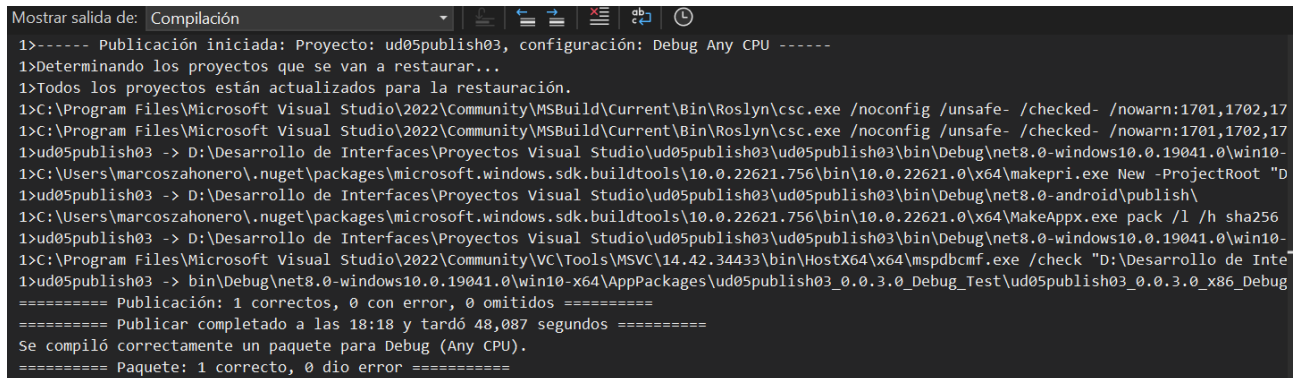


Figura 6: Definir ubicación del instalador y cada cuanto se actualiza.

Al finalizar la publicación nos aparecerá esto, con su ruta de instalación para evitar confusiones y una pequeña descripción de abajo:



Desde la salida de la consola con el modo de **Compilación** veremos el proceso paso a paso como va funcionando por si estamos con una aplicación muy pesada tardará más y esto nos ayudará un poco a ver que no se ha atascado y que simplemente va a tardar un poco más.



```

Mostrar salida de: Compilación
1>----- Publicación iniciada: Proyecto: ud05publish03, configuración: Debug Any CPU -----
1>Determinando los proyectos que se van a restaurar...
1>Todos los proyectos están actualizados para la restauración.
1>C:\Program Files\Microsoft Visual Studio\2022\Community\MSBuild\Current\Bin\Roslyn\csc.exe /noconfig /unsafe- /checked- /nowarn:1701,1702,17
1>C:\Program Files\Microsoft Visual Studio\2022\Community\MSBuild\Current\Bin\Roslyn\csc.exe /noconfig /unsafe- /checked- /nowarn:1701,1702,17
1>ud05publish03 -> D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-windows10.0.19041.0\win10-
1>C:\Users\marcoszahonero\.nuget\packages\microsoft.windows.sdk.buildtools\10.0.22621.756\bin\10.0.22621.0\x64\makepri.exe New -ProjectRoot "D
1>ud05publish03 -> D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-android\publish\
1>C:\Users\marcoszahonero\.nuget\packages\microsoft.windows.sdk.buildtools\10.0.22621.756\bin\10.0.22621.0\x64\MakeAppx.exe pack /l /h sha256
1>ud05publish03 -> D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-windows10.0.19041.0\win10-
1>C:\Program Files\Microsoft Visual Studio\2022\Community\VC\Tools\MSVC\14.42.34433\bin\HostX64\x64\mspdbsrv.exe /check "D:\Desarrollo de Inte
1>ud05publish03 -> bin\Debug\net8.0-windows10.0.19041.0\win10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\ud05publish03_0.0.3.0_x86_Debug
===== Publicación: 1 correctos, 0 con error, 0 omitidos =====
===== Publicar completado a las 18:18 y tardó 48,087 segundos =====
Se compiló correctamente un paquete para Debug (Any CPU).
===== Paquete: 1 correcto, 0 dio error =====
  
```

Figura 7: Salida de compilación de la publicación de la aplicación.

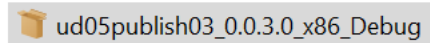
Vamos a la carpeta que estaba comentando con anterioridad, la carpeta de **AppPackages**, que contiene todas las versiones que vayamos publicando, yo al final acabe en la 0.0.3.0 porque tuve algunas pruebas/errores, pero eso es lo de menos, vemos el contenido de la carpeta que nos ayudará a instalarlo en un equipo distinto de **2 formas distintas** ahora las veremos con detalle.

ud05publish03 > ud05publish03 > bin > Debug > net8.0-windows10.0.19041.0 > win10-x64 > AppPackages > ud05publish03_0.0.3.0_Debug_Test				
Nombre	Fecha de modificación	Tipo	Tamaño	
Add-AppDevPackage.resources	16/12/2024 18:18	Carpeta de archivos		
Dependencies	16/12/2024 18:18	Carpeta de archivos		
Add-AppDevPackage	26/06/2024 21:15	Script de Windows P...	37 KB	
Install	26/06/2024 21:15	Script de Windows P...	14 KB	
ud05publish03_0.0.3.0_x86_Debug	16/12/2024 18:18	Certificado de seguri...	1 KB	
ud05publish03_0.0.3.0_x86_Debug	16/12/2024 18:18	Archivo MSIX	42.758 KB	
ud05publish03_0.0.3.0_x86_Debug.msixsym	16/12/2024 18:18	Archivo MSIXSYM	29 KB	

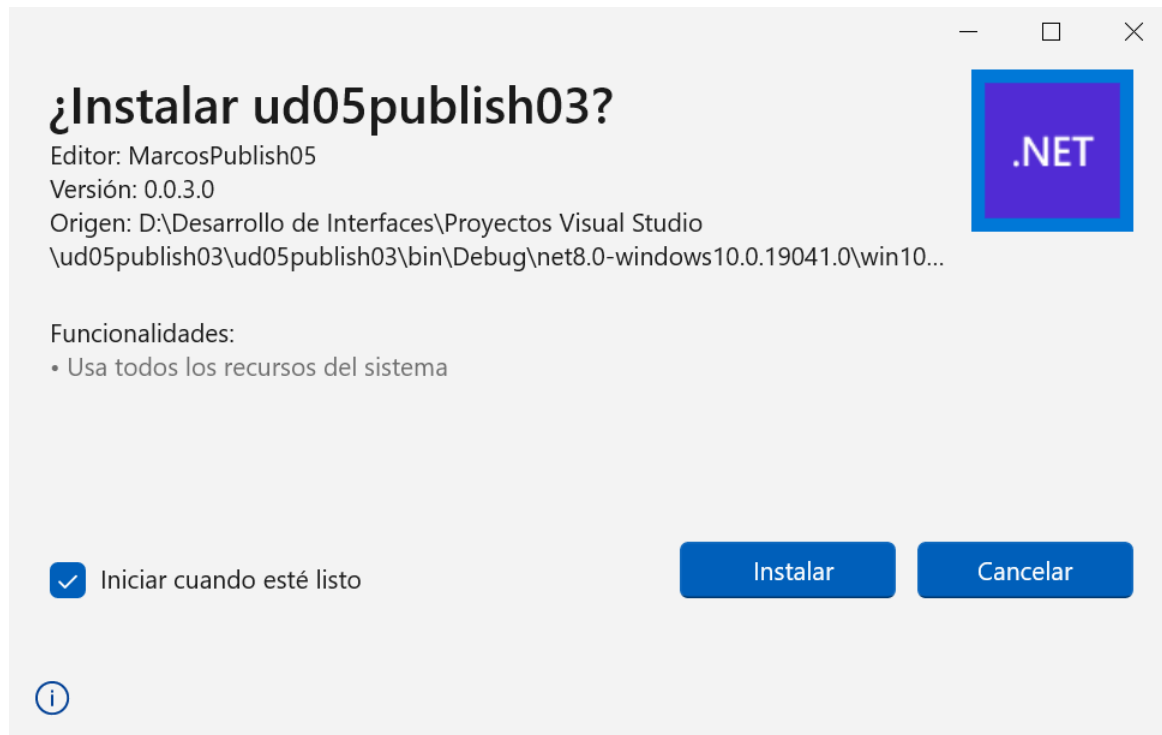
Figura 8: Explorador de archivos (AppPackages)

## Instalar de forma gráfica

Le damos al archivo “**ud05publish03\_0.0.3.0\_x86\_Debug**”

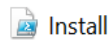


Y nos saldrá esta ventana y le daremos a “**Instalar**”, se nos instalará y ya esta.



## Instalar de forma no gráfica

Le damos al archivo “**Install.ps1**”



Al iniciarlo nos saldrá que se van a instalar las dependencias que en este caso es el certificado de firma.

```
Windows PowerShell
Paquete encontrado: D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-win
dows10.0.19041.0\win10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\ud05publish03_0.0.3.0_x86_Debug.msix
Certificado encontrado: D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0
-windows10.0.19041.0\win10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\ud05publish03_0.0.3.0_x86_Debug.cer

Antes de instalar esta aplicación, es necesario realizar lo siguiente:
- Instale el certificado de firma
Se requieren credenciales de administrador para continuar. Acepte la solicitud de UAC e indique su contraseña de adminis
trador si así se le solicita.
Presione Entrar para continuar...: 
```

Nos ponemos a instalar el certificado, solo tenemos que poner “**S**” y le damos ENTER.

```
Instalando certificado...

Está a punto de instalar un certificado digital en el almacén de certificados de personas de confianza de su equipo.
Esta operación entraña un riesgo de seguridad importante, por lo que solo deberá realizarla si confía en el emisor de
este certificado digital.

Cuando termine de usar la aplicación, deberá quitar manualmente el certificado digital asociado. Aquí encontrará
instrucciones para realizar esta operación: http://go.microsoft.com/fwlink/?LinkId=243053

¿Está seguro de que desea continuar?

[S] Sí [N] No [?] Ayuda (el valor predeterminado es "N"):
```

Se instala todo el resto, le dejamos descargarse, al acabar ya estará todo instalado.

```
Paquete encontrado: D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-win
dows10.0.19041.0\win10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\ud05publish03_0.0.3.0_x86_Debug.msix

Progreso de la operación de implementación: D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publi
Processing
[

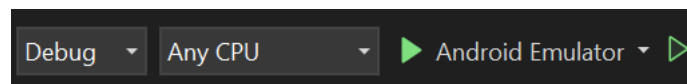
Se requieren credenciales de administrador para continuar. Acepte la solicitud de UAC e indique su contraseña de adminis
trador si así se le solicita.
Presione Entrar para continuar...:
El certificado se ha instalado correctamente.

Instalando la aplicación...
Paquetes de dependencia encontrados:
D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-windows10.0.19041.0\win
10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\Dependencies\x86\Microsoft.WindowsAppRuntime.1.5.msix
D:\Desarrollo de Interfaces\Proyectos Visual Studio\ud05publish03\ud05publish03\bin\Debug\net8.0-windows10.0.19041.0\win
10-x64\AppPackages\ud05publish03_0.0.3.0_Debug_Test\Dependencies\x64\Microsoft.WindowsAppRuntime.1.5.msix
```

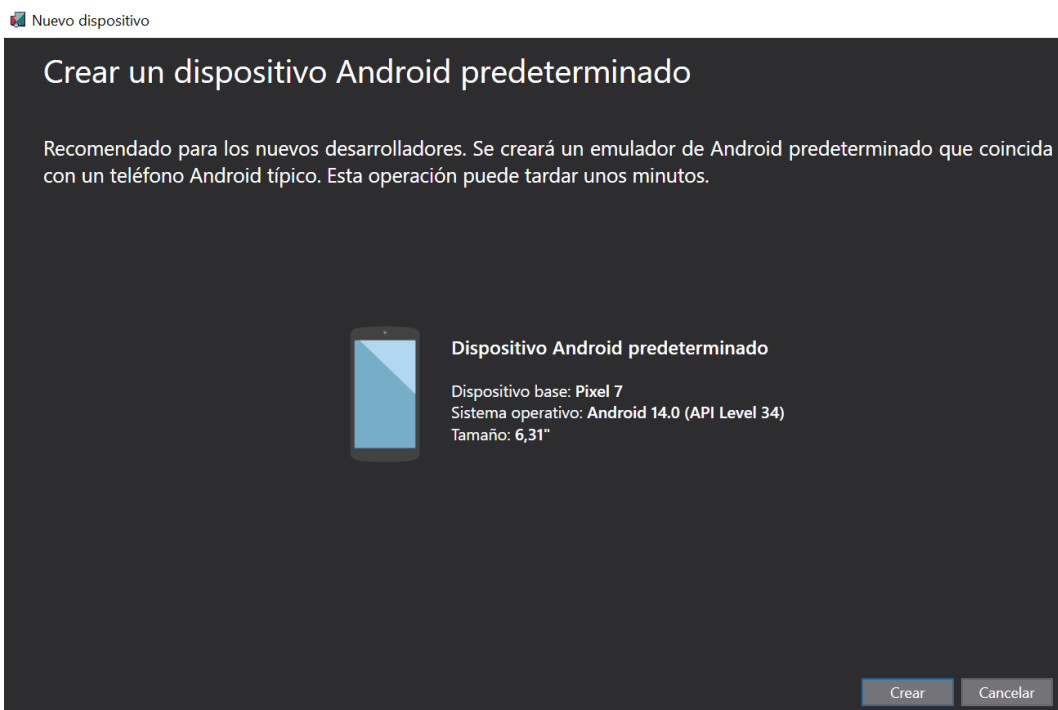
## Publicación de aplicaciones en Android



Cambiamos en la parte superior desplegando varias opciones y seleccionamos la versión de android, y saldrá así:



Y le damos al “**Play**” y nos saldrá esta ventana para establecer nuestro emulador de Android, le damos a “**Crear**”.



Se nos descargará un móvil, en este caso el Pixel 7.



Habiendo dejado descargar el emulador del móvil vamos al Explorador de archivos de Visual Studio y le damos “**Clic derecho > Propiedades**”.



Vamos al apartado de “**Android > Opciones**” y desactivamos la opción de “**Debug & net8.0-android34.0**”

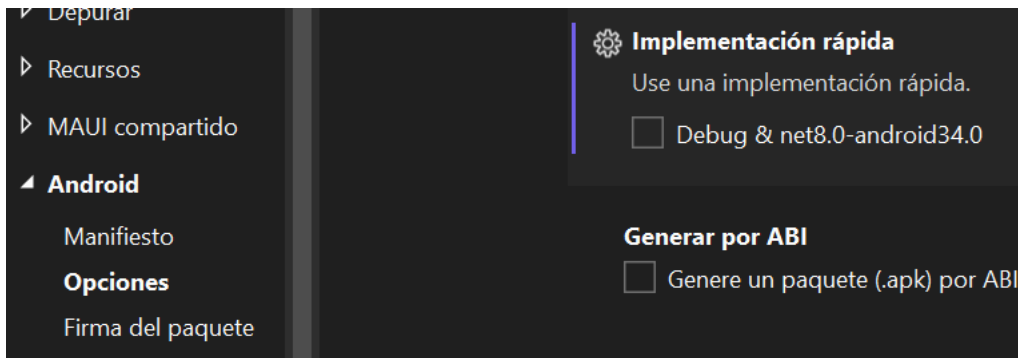
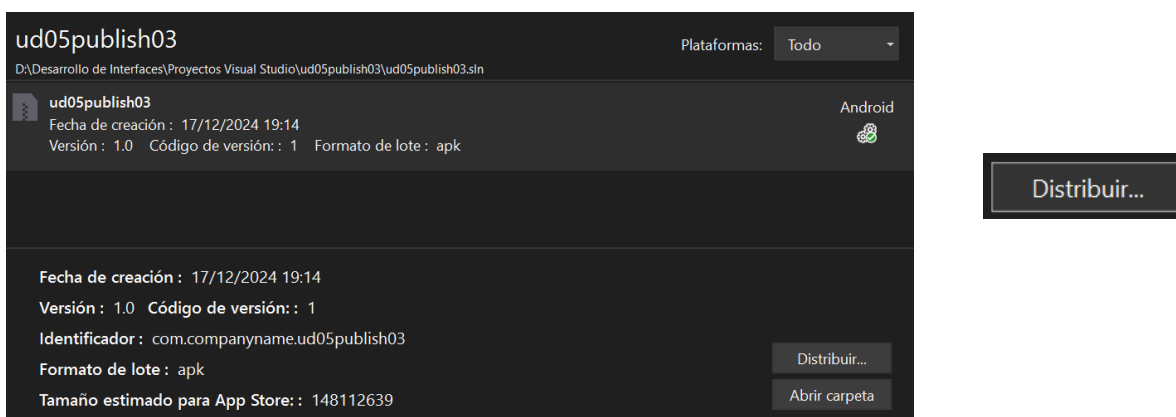
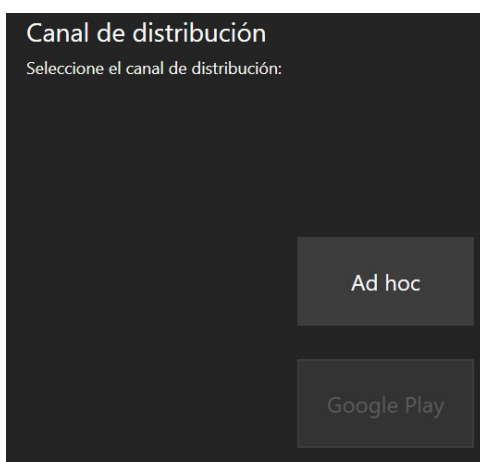


Figura 9: Propiedades del proyecto

Volvemos a esta pantalla y nos saldrá el tick debajo en Android, le damos a “**Distribuir...**”



Seleccionamos el canal de distribución “**Ad hoc**”.



Ahora le damos al + y nos saldrá una ventana.

### Distribuir - Identidad de firma

#### Detalles de la aplicación

ud05publish03  
Fecha de creación: 17/12/2024  
Versión: 1.0

Seleccionar canal  
Ad hoc

Identidad de firma

#### Identidad de firma

Nombre

+ - Importar...

Rellenamos los datos y lo creamos:

### Crear almacén de claves Android

Alias\* jsmgrs

Contraseña\* ..... Confirmar\* .....

Validez 30 (Años)

<b>Especifique al menos uno de los siguientes valores</b> \*

Nombre completo

Unidad organizativa:

Organización:

Ciudad o localidad Valencia

Estado o provincia

Código de país: (2 dígitos)



Volveremos donde antes y nos saldrá ya la firma:



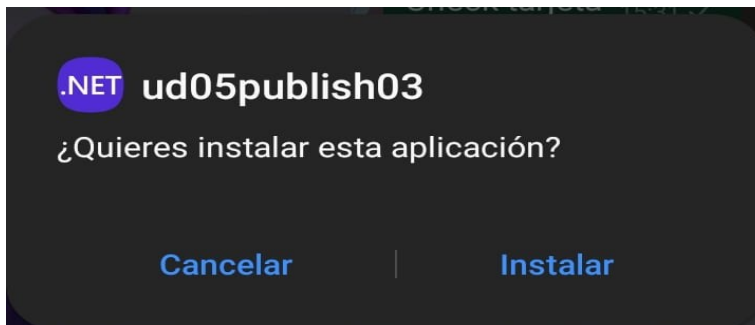
Ahora le damos a “**Guardar como**” y nos dejará guardar la APK en un sitio, y en mi caso lo deje en una carpeta llamada “apksMovil”:

quipo > MZ DISCO (D:) > Desarrollo de Interfaces > Proyectos Visual Studio > apksMovil

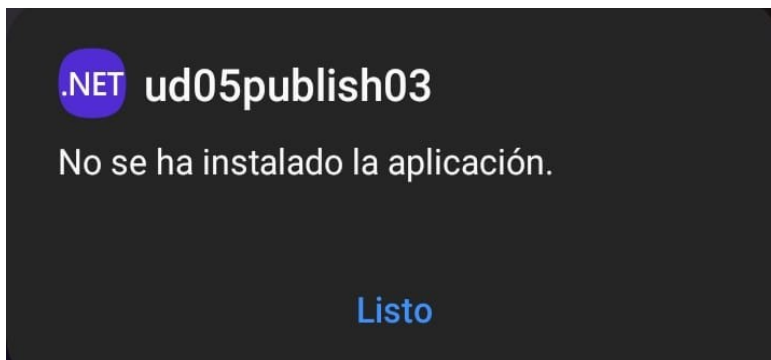
Nombre	Fecha de modificación	Tipo
com.companyname.ud05publish03.apk	17/12/2024 19:21	Archivo

## Instalación por APK

Nos pasamos la APK al móvil ya que en este caso habían problemas con el emulador de Visual Studio, por configuraciones del portátil y para evitar problemas la opción más sencilla es instalarlo en tu propio móvil, en caso de que tengas iOS te recomiendo un móvil de verdad.



Ha ocurrido un error al instalarlo en mi caso, tal vez sea por mi versión de Android, por la versión del .NET ya que lo hice con la versión 8.0, y podría ser que solo se podía con la versión 9.0.



## GitHub

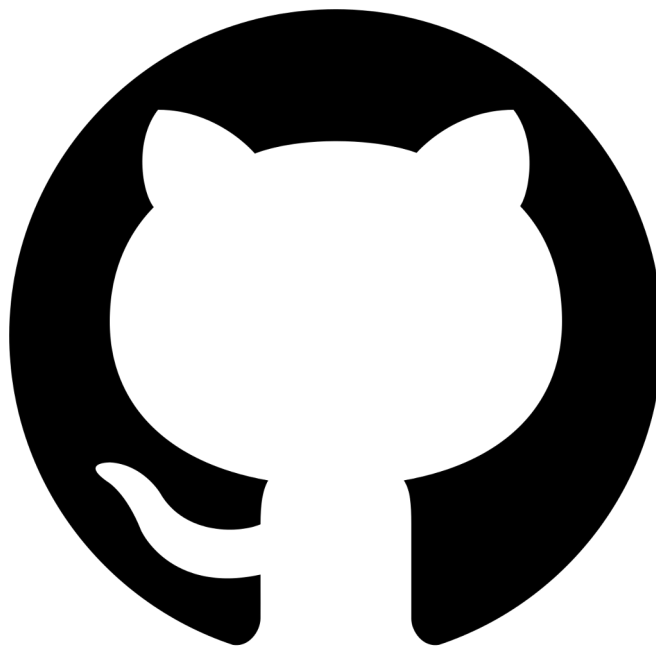


**He cambiado el nombre del repositorio así que si has estado tenido problemas con mis actividades al corregir será por eso, en esta actividad está corregido para el resto del curso.**



En esta actividad he añadido GitHub para añadir el proyecto completo, junto a una descripción en el [README.md](#) para añadir una breve explicación y preview de cada actividad.

**Enlace:** [Maek0s/2DAM\\_DesarrolloInterfaces](#)



## **Conclusión**

En conclusión, las guías que seguí para hacerlo no estaban actualizadas actualmente y tuve que improvisar un poco al respecto, pero al final se saco investigando y probando un poco.

## Problemas y/o sugerencias en la actividad

### 1# - Esta vez no hay extra pero...

Igual te suena un poco lo que voy a hacer a otro año, pero en estas fechas es para estar con la familia, descansar y también disfrutar las vacaciones así que esta vez me tomo la libertad de no hacer videos enseñando el proceso, pero...

**¡Feliz Navidad!**



## **2# - No me deja instalar la APK**

Como he dicho en la actividad a la hora de probar la APK he tenido problemas a la hora de instalarlo, así que lo que he deducido es básicamente problemas de la configuración del proyecto o de mi móvil por su propia versión, pero el procedimiento esta bien hecho.

## Webgrafía

- <https://visualstudio.microsoft.com/es/vs/community/>
- <https://dotnet.microsoft.com/es-es/apps/maui>