

Tutorial: creación de un instalador

Crear nuevo proyecto y escribir “aplicación de Windows”. Elegir Proyecto de paquete de aplicación de Windows.

Ahora hay que elegir la versión mínima y la versión de destino. En este [enlace](#) podéis ampliar la información sobre estas dos versiones.

La **versión de destino** establece la versión de Windows que está destinada nuestro proyecto (y, por lo tanto, el conjunto de API disponibles para la aplicación).

La **versión mínima** establece la versión más antigua de este SO que permitirá ejecutar la aplicación.

Hay que tener en cuenta que esto va a hacer funcionar la aplicación desde la versión más antigua establecida mediante versión mínima y la versión más reciente, es decir, la que hayamos puesto como versión de destino. Por tanto, habrá que tener precaución con las llamadas a la API utilizadas, ya que, si en la versión mínima no está disponible, habrá problemas con la aplicación.

En definitiva, habrá que tener en cuenta las características específicas que hemos utilizado en nuestra aplicación y que puede no estar presentes en algunas versiones más antiguas del sistema operativo.

Para crear el proyecto es posible que Windows nos pida que lo configuremos en modo desarrollador. Lo hacemos desde Sistema > Modo para desarrolladores.

Una vez creado, echamos un vistazo al proyecto de instalación que hemos creado. Veremos un desplegable llamado Dependencias, haciendo *click* con el botón secundario, accedemos a “Agregar referencia del proyecto” y escogemos nuestro proyecto.

En caso de que nuestro proyecto utilizara librerías externas, *dll*, etc., no haría falta escogerlas, ya que cuando la aplicación se compila, si la aplicación las utiliza, las incluye automáticamente.

Al agregarlo, veremos como el nombre de nuestra aplicación se ha incluido en Dependencias/Aplicaciones.

Ahora pulsamos sobre *Package.appxmanifest* y podemos cambiar el nombre de la aplicación, la descripción, la orientación admitida en la aplicación, etc.

La **pestaña de recursos visuales** se utiliza para configurar los recursos visuales asociados con tu aplicación. Estos recursos incluyen íconos, imágenes de mosaicos, logotipos y otros elementos gráficos que se utilizan para representar tu aplicación en la interfaz de usuario del sistema operativo y en la tienda de aplicaciones. Es importante configurar estos recursos para asegurarnos de que nuestra aplicación se presenta de manera efectiva y atractiva en el SO y en la tienda de aplicaciones. Proporcionar versiones de alta calidad y apropiadas de estos elementos permite mejorar la experiencia de usuario y aumentar la visibilidad de tu aplicación.

Como ejemplo, escogeré un origen para un icono, seleccionaré en recursos todos los recursos visuales y todas las escalas, con modo de escalado bicúbico. Marcaremos los dos *checkbox* de aplicar y generamos.

Para obtener iconos, podéis navegar por la web hasta encontrar tiendas gratuitas como, por ejemplo: <https://www.iconfinder.com/search?q=windows%20mobile>. Además, como es obvio, también podremos crear nuestros iconos personalizados mediante diferentes herramientas.

Configuramos el nombre corto y marcamos mostrar nombre en icono mediano, ancho y grande.

Ahora en la pestaña aplicación, establecemos notificaciones de pantalla de bloqueo: distintivo y texto de imagen.

La **pestaña capacidades** sirve para especificar las capacidades o permisos que nuestra aplicación requiere para acceder a ciertos recursos o realizar ciertas acciones en el sistema operativo Windows. Al especificar estas capacidades, nuestra aplicación informa al SO y a los usuarios sobre los recursos a los que necesita acceder.

Estas capacidades son esenciales para garantizar que nuestra aplicación pueda funcionar correctamente y acceder a los recursos necesarios. Además, también son importantes desde el punto de vista de la seguridad y la privacidad, ya que el sistema operativo y los usuarios pueden revisar estas capacidades antes de instalar la aplicación para comprender qué tipo de acceso se solicita. Asegúrate de seleccionar solo las capacidades necesarias para el correcto funcionamiento de la aplicación, ya que solicitar acceso innecesario puede afectar negativamente la aceptación de tu aplicación por parte de los usuarios.

La **pestaña "URI de Contenido"** en el archivo *Package.appxmanifest* de un proyecto de aplicación de Windows, se utiliza para especificar los *URIs* (*Uniform Resource Identifiers*) que tu aplicación tiene permiso de acceder en tiempo de ejecución. Un URI de contenido es una dirección que puede apuntar a recursos en línea o locales que tu aplicación necesita acceder durante su ejecución.

Cuando defines *URIs* de contenido en esta pestaña, estás declarando explícitamente los recursos externos o locales que tu aplicación podría necesitar cargar o acceder dinámicamente durante la ejecución. Esto podría incluir recursos web, servicios, imágenes, o cualquier otro tipo de recurso identificado por un URI.

Algunos escenarios comunes en los que podrías usar *URIs* de contenido:

- **Recursos web:** Si tu aplicación necesita cargar datos o recursos desde servidores en línea, deberías especificar esos *URIs* aquí.
- **Servicios REST:** Si tu aplicación se comunica con servicios web mediante *REST*, podrías declarar los *URIs* de esos servicios en esta sección.
- **Imágenes o archivos locales:** Si tienes recursos locales como imágenes, archivos de datos, o cualquier otro tipo de archivo que necesitas cargar dinámicamente, puedes especificar sus *URIs* aquí.

Es importante destacar que, al especificar *URIs* en esta sección, estás informando al sistema operativo sobre los recursos a los que tu aplicación tiene permiso de acceder. Esto también puede ser relevante para cuestiones de seguridad y privacidad.

Recuerda que el uso de *URIs* de contenido debe alinearse con las políticas de seguridad y privacidad de tu aplicación, así como con los requisitos funcionales específicos de tu proyecto.

La **pestaña de "Empaquetado"** permite configurar diversos aspectos relacionados con el empaquetado y la distribución de nuestra aplicación. Algunos de los elementos que podemos configurar en esta pestaña son:

- **Nombre del paquete:** Especifica el nombre único del paquete de la aplicación. Este nombre es importante para la identificación única de tu aplicación en el sistema y en la tienda de aplicaciones.
- **Ubicación de la carpeta de salida (*Output location*):** Indica la ubicación donde se generará el paquete de la aplicación después de la compilación. Puedes configurar si deseas que el paquete se genere en una carpeta específica.
- **Generación automática del paquete de aplicación (*Generate app bundle*):** Determina si el paquete de la aplicación se generará automáticamente como un paquete de aplicaciones (App Bundle) o no. Los paquetes de aplicaciones permiten que se instale en dispositivos solo el código y los recursos específicos necesarios para su arquitectura y configuración regional.
- **Versión:** establece la versión del paquete de la aplicación. Esta información es importante para la gestión de versiones y actualizaciones de la aplicación.
- **Configuración del certificado:** permite configurar y seleccionar el certificado que se utilizará para firmar tu aplicación. La firma digital es esencial para la seguridad y la autenticación de la aplicación.

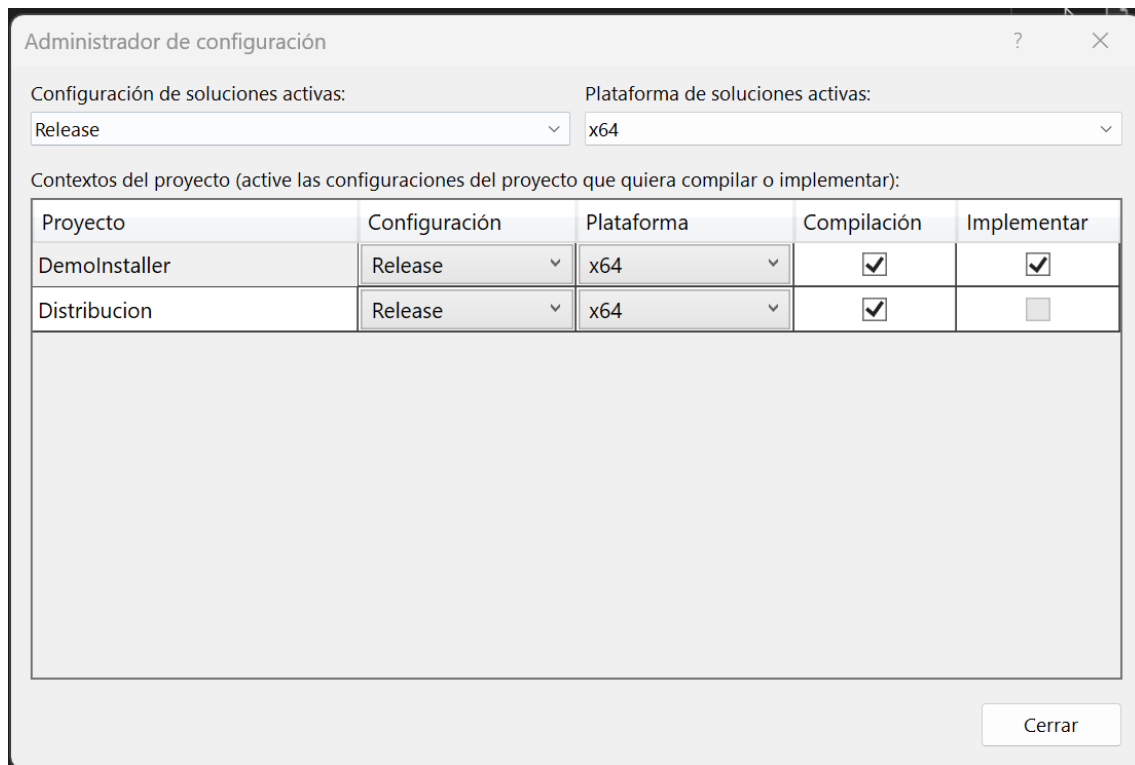
Configurar adecuadamente la pestaña de "Empaquetado" es fundamental para garantizar que tu aplicación se empaquete correctamente, tenga la información de versión adecuada, esté firmada digitalmente. Además, estas configuraciones son importantes al preparar tu aplicación para su distribución a través de la Microsoft Store o mediante otros medios.

Una vez configurado, guardamos las modificaciones del fichero *Package.appxmanifest*, recompilamos y con el botón secundario sobre el proyecto de empaquetado y pulsamos sobre publicar y elegimos la opción de crear paquete de instalación. Elegimos instalación de prueba, elegimos el certificado que hemos creado y pulsamos siguiente.

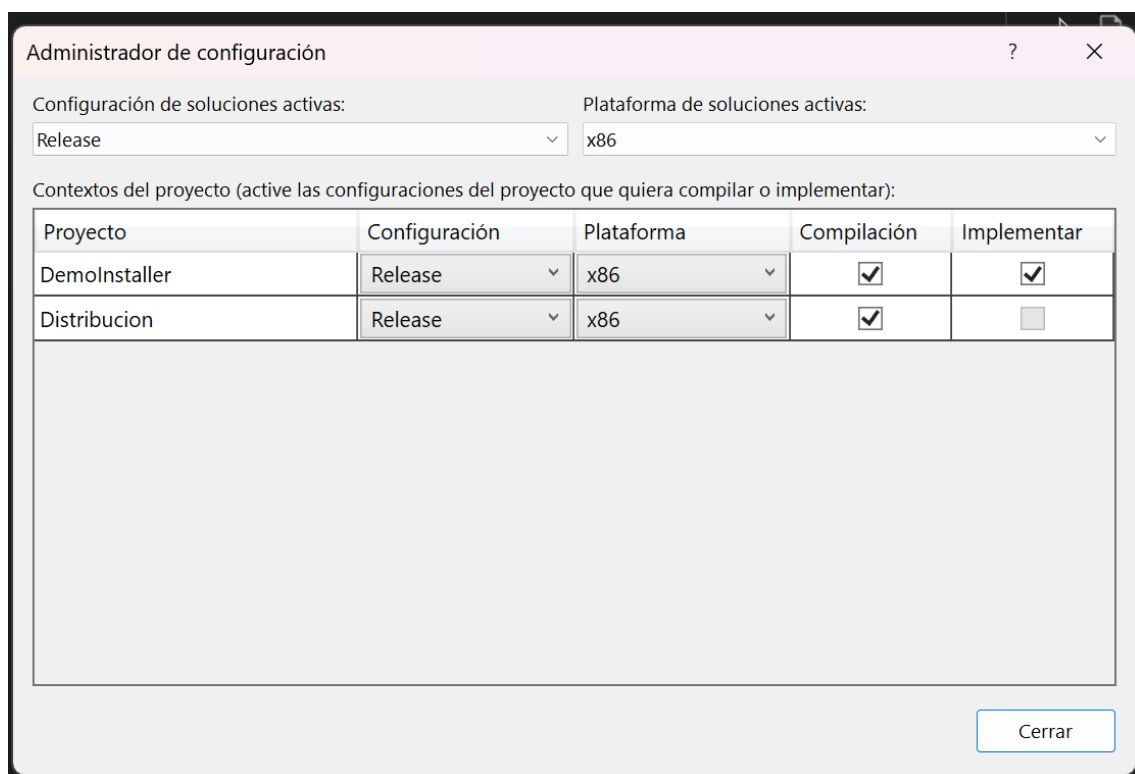
Ahora vemos que podemos elegir la arquitectura de la aplicación, etc. Pero lo dejamos como está y pulsamos siguiente. Creamos el instalador.

Ahora es posible que de un error. Pinchamos en Compilar > Administración de configuración y elegimos la configuración de soluciones activas: *Release* y elegimos en plataforma de soluciones activas: x64.

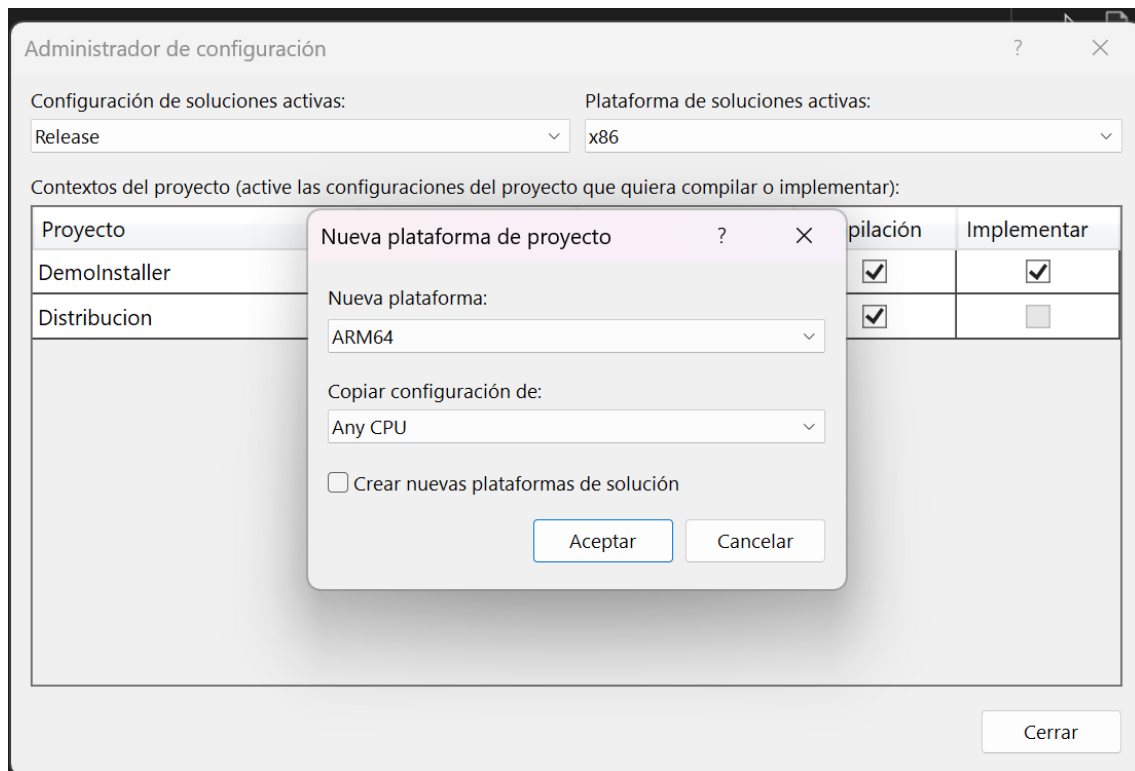
Ahora nos aseguramos de que tanto la aplicación como el proyecto de distribución estén configurados con plataforma x64.



Hacemos lo mismo seleccionando plataforma de soluciones activa x86:



Para poder seleccionar la plataforma x64 y x86 en nuestro proyecto, pulsaremos sobre añadir una nueva plataforma y seleccionamos la plataforma en la pestaña de arriba y no marcamos el *check*, aceptamos:



Ahora vamos de nuevo a publicar y finalmente tras uso instantes nuestra aplicación estará empaquetada.

Ahora vamos al directorio donde hemos creado el instalador y tendremos un `index.html`, que muestra la vista de la tienda de la Microsoft store y nos permite instalar la aplicación, así como ver sus datos. Tenemos también un fichero `.appinstaller` que es el instalador de nuestra aplicación. También tenemos un directorio `test` que incluye el paquete `.msixbundle` y un certificado público, `.cer`. Este paquete también nos permitirá instalar nuestra aplicación.

Si intentamos instalar la aplicación no funcionará, ya que el autor no es de confianza. Tendremos que instalar el certificado que hemos visto en la carpeta `test` para el equipo local e introducirlo en el almacén de entidades de certificación raíz de confianza. Una vez lo hemos instalado, nos dejará instalar nuestra aplicación, ya que el sistema podrá confiar en el creador de la aplicación, al tener el certificado instalado.