

## Tema 4: Creación de Instaladores

### ¿Qué aprenderás?

---

- Creación de nuevos paquetes de instalación para nuestros proyectos de Visual Studio .NET
- Personalización de los instaladores.

### ¿Sabías que...?

---

- Existen multitud de programas y entornos para crear paquetes de instalación para aplicaciones desarrolladas en Visual Studio .N



## 4.1. Proyectos de Instalación

### 4.1.1. Creación de un proyecto de instalación

Para poder distribuir nuestras aplicaciones a los usuarios no lo debemos hacer grabando el ejecutable y los ficheros necesarios en un DVD, porque esto no solo no es nada elegante, sino que crearía grandes dificultades a algunos usuarios con pocos conocimientos informáticos.

En lugar de eso, lo que debemos hacer es generar un instalador. Entendemos por instalador, un programa que ejecuta un asistente que guía durante la instalación y al que se le pide que tome las decisiones necesarias para la correcta instalación del programa, además de crear los iconos de acceso necesarios en el menú de inicio y el Escritorio.

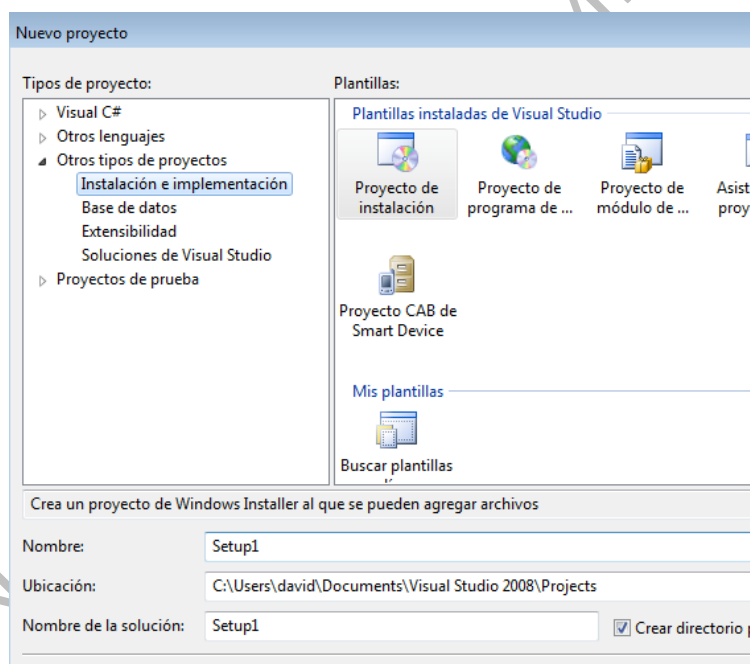


Figura: Nuevo Proyecto de Instalación

Para crear un proyecto de instalación en Visual Studio .NET debemos ejecutar la opción de nuevo proyecto y en la ventana que aparece, en lugar de elegir un lenguaje de programación, lo que vamos a hacer es elegir la opción **Otros tipos de proyectos**, y luego podemos ver la opción **Instalación e implementación**. Dentro de esta opción aparecen distintas posibilidades, entre ellas elegimos **Proyecto de Instalación**. Le ponemos el nombre que deseemos, nosotros lo vamos a llamar Instala; y ya podemos empezar. El aspecto que tiene ahora el Interfaz de Visual Studio .NET es el siguiente:

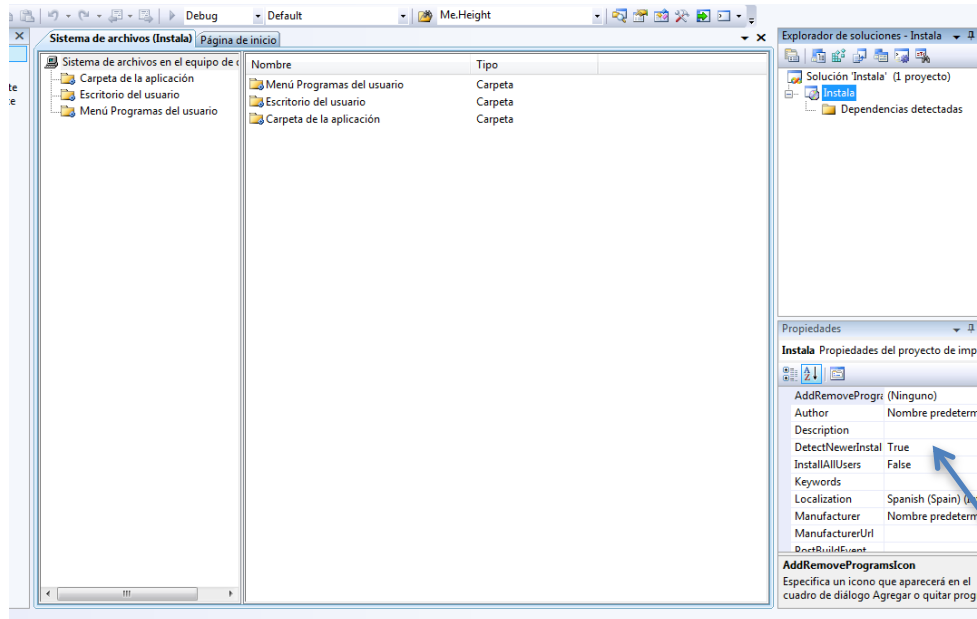


Figura: Interfaz de un proyecto de instalación

#### 4.1.2. Desarrollo de un proyecto de instalación

En el interfaz anterior es importante fijarse en la tira de iconos que hay en la zona superior derecha. Cada uno de ellos representa una de las fases de desarrollo de un proyecto de instalación. Nosotros vamos a trabajar todos los iconos, uno por uno.

Es evidente, que cuando desarrollamos un proyecto de instalación cabe la posibilidad que no necesitemos todas la fases, aunque resulta útil examinarlas todas, de izquierda a derecha, e ir ignorando las que no necesitemos.

#### 4.1.3. Fase 1: Propiedades

En esta fase podemos definir propiedades generales de nuestro proyecto, como el nombre del archivo que ejecuta el instalador, los requisitos que debe cumplir el usuario para poder ejecutar nuestra aplicación o la posibilidad de aplicar compresión al proyecto de instalación para que este ocupe menos espacio en el disco.



La ventana de propiedades tiene el aspecto siguiente:

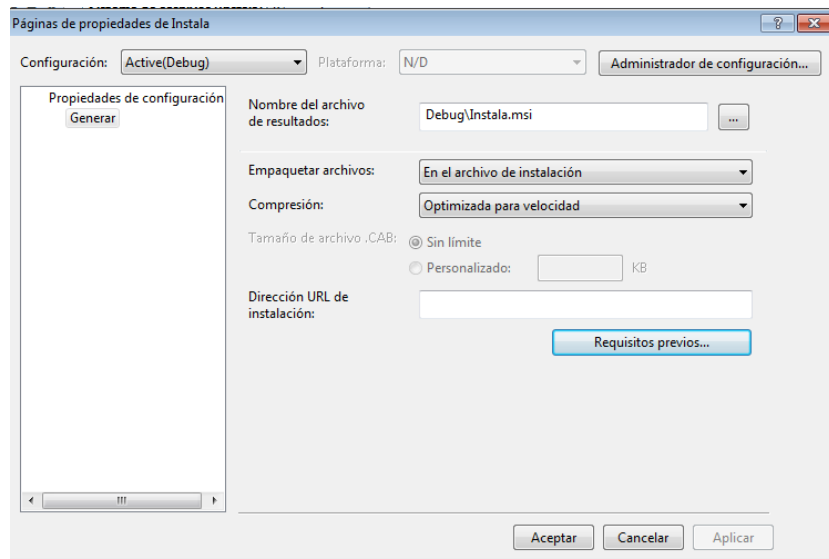


Figura: Propiedades generales de un proyecto de instalación

Podemos identificar claramente la caja de texto que contiene el nombre del archivo del instalador, así como un desplegable que nos permite ajustar las opciones de compresión y un botón para acceder a la pantalla con los requisitos de instalación. Si accedemos a la pantalla de requisitos podemos ver una ventana como la siguiente:

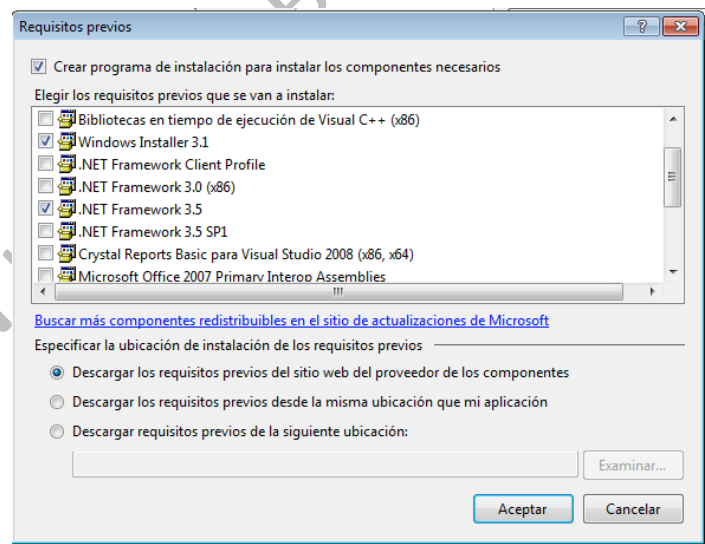


Figura: Requisitos de instalación

En esta ventana, no solamente podemos establecer dichos requisitos, sino que también podemos determinar cómo se procederá en el caso que dichos requisitos no se cumplieran.



#### 4.1.4. Fase 2: Sistema de Archivos

Esta fase es una de las más importantes. Aquí debemos elegir qué archivos queremos que se instalen en el ordenador del usuario cliente, así como la jerarquía de directorios que se deben crear. También debemos crear los iconos de acceso directo a nuestra aplicación. El interfaz presenta una estructura con 3 carpetas llamadas:

- Carpeta de la aplicación: Aquí hay que colocar todos los archivos que queremos copiar en la máquina cliente.
- Escritorio del Usuario: Aquí hay que crear los iconos de acceso directo que queremos poner en el Escritorio del usuario cliente.
- Menú Programas del Usuario: Es como la carpeta anterior, pero con el Menú de inicio.

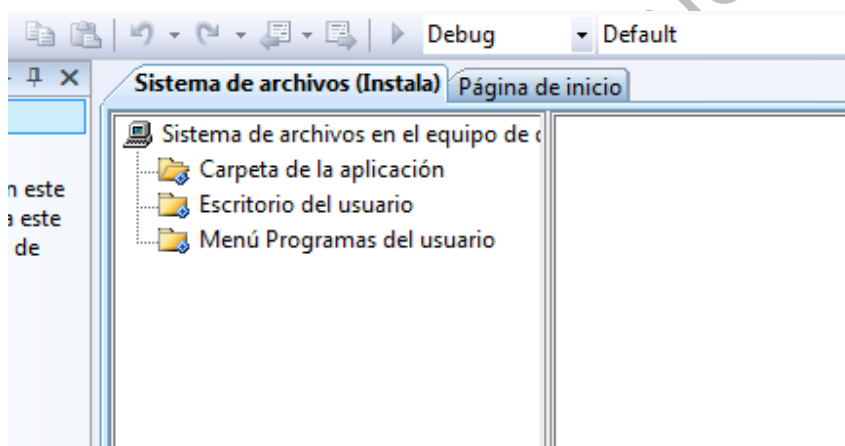


Figura: Sistema de Archivos

Para desarrollar este capítulo, vamos a desarrollar un instalador para la aplicación, previamente explicada en la actividad 1, que consiste en una calculadora. Para ello vamos a abrir la carpeta **“Carpeta de la aplicación”** y vamos a colocar en ella los archivos necesarios. Para ello hacemos clic con el botón secundario del ratón y del menú emergente elegimos la opción **Agregar → Archivo...**

Navegamos por nuestro sistema de archivos hasta localizar el ejecutable del proyecto de la calculadora y lo elegimos.

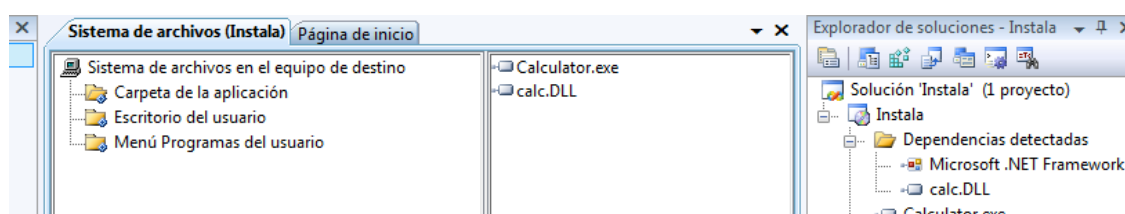


Figura:  
Añadiendo  
archivos



Es importante fijarse que el sistema ha añadido de forma automática, y sin pedir ningún tipo de confirmación, el archivo **calc.DLL** a la carpeta de la aplicación. Eso ocurre porque, como recordaremos del enunciado de la actividad, la calculadora se programó utilizando una librería programada por nosotros. Dicha librería es imprescindible para que nuestra calculadora funcione y debe estar junto al fichero ejecutable, por ese motivo, el entorno ha realizado la adición del fichero automáticamente. Podemos observar en la zona derecha del interfaz todas las dependencias necesarias en nuestra aplicación.

A continuación, vamos a crear iconos de acceso directo para nuestra calculadora. La imagen que queremos para nuestros iconos debe acompañar al proyecto, en ese caso, lo primero que vamos a hacer es elegir un archivo con extensión **.ico** y añadirlo a la carpeta del proyecto.

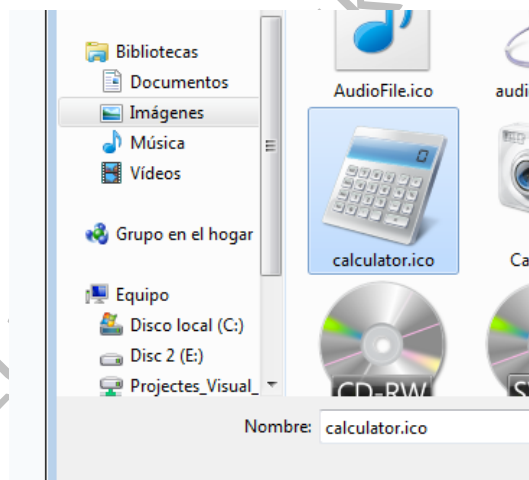


Figura: Añadiendo imágenes para iconos

Ahora ya podemos crear nuestros accesos directos. Para ello vamos a la carpeta “**Escritorio del usuario**” y del menú contextual elegimos **Crear nuevo acceso directo**.

En la ventana emergente que aparece debemos abrir la “Carpeta de la aplicación” y elegir el ejecutable de la calculadora como objetivo de nuestro icono de acceso directo.

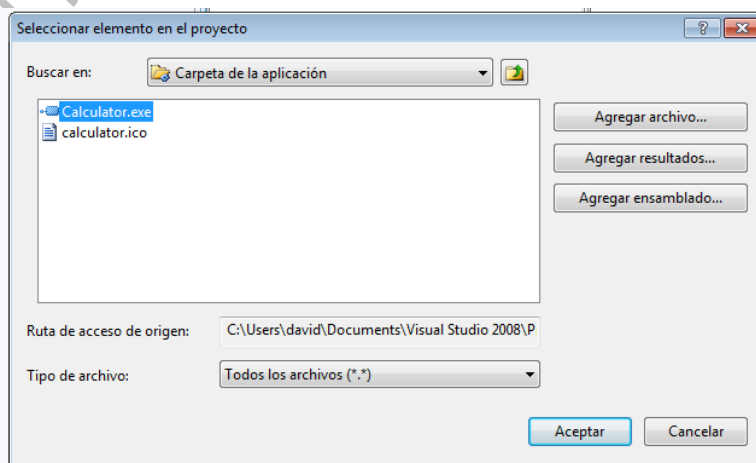


Figura: Icono de acceso directo (I)



Una vez creado, lo renombramos y le ponemos de nombre “Calculadora”. Si hacemos clic en el acceso directo recién creado y observamos la ventana de propiedades (abajo a la derecha), veremos una propiedad llamada **Icon** que actualmente pone (Ninguno). Elegimos examinar en dicha propiedad y aparece una ventana igual que la anterior, pero esta vez vamos a buscar el archivo de .ico que hemos añadido antes y lo seleccionamos.

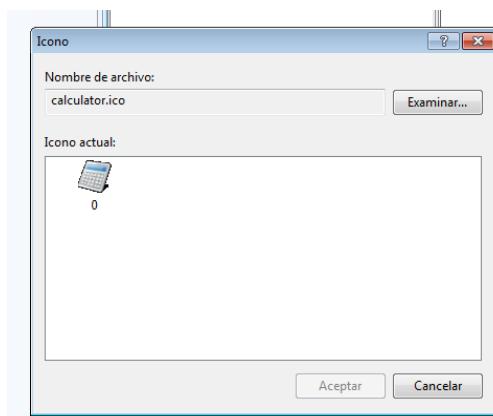


Figura: Icono de acceso directo (II)

Ahora hay que repetir todo el proceso para el icono de acceso directo en el menú inicio.

#### 4.1.5. Fase 3: Registro de Windows

Esta fase no siempre es necesaria. Aquí podemos especificar nuevas claves en el registro de Windows del equipo cliente o modificar los valores de claves ya existentes.

Esta es necesaria en el caso que haya que mapear unidades de disco, modificar opciones del sistema operativo o crear enlaces a bases de datos.

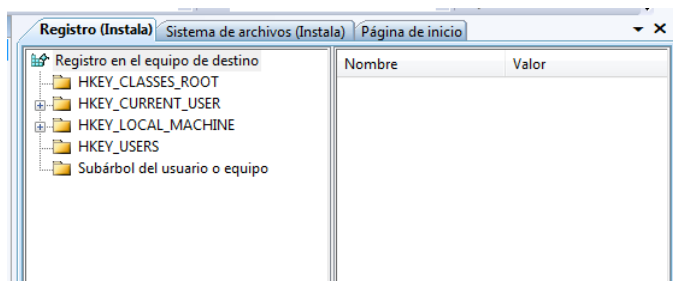


Figura: Editor del registro de Windows

En nuestro ejemplo, no necesitamos editar el registro. Por este motivo, nos saltamos esta fase y procedemos a la siguiente:



#### 4.1.6. Fase 4: Tipos de archivo

Esta fase no siempre es necesaria. Aquí podemos asociar determinadas extensiones de archivo con nuestra aplicación. Esta sería muy importante en el caso que nuestra aplicación generase archivos con extensiones inventadas por nosotros, ya que, en dicho caso, el sistema operativo Windows no podría saber con qué aplicación debe procesar dichos archivos.

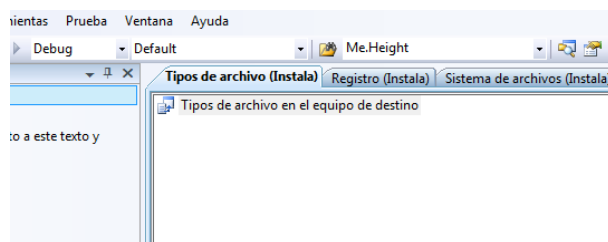
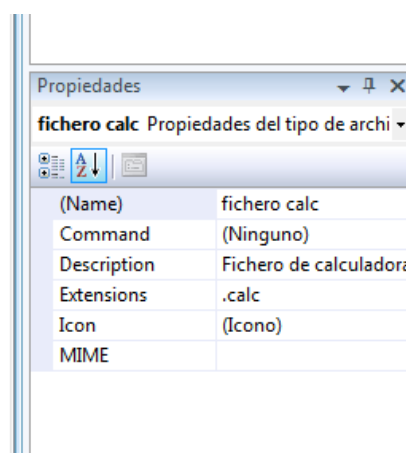


Figura: Tipos de archivo

Para nuestro caso, no necesitamos especificar ninguna extensión de archivo. Si quisiéramos hacerlo, solo tendríamos que hacer clic con el botón secundario del ratón y elegir **Agregar tipo de archivo** y luego editar sus propiedades para definir la extensión asociada, el nombre, el comentario, el icono, etc.

Figura: Propiedades del nuevo tipo de archivo



#### 4.1.7. Fase 5: Interfaz del asistente

Esta es otra de las fases clave de nuestro proyecto. Aquí debemos definir qué ventanas formarán nuestro asistente de instalación, la información que se mostrará en ellas, así como los datos que se van a pedir al usuario. El interfaz de la fase nos muestra un árbol con la secuencia de todas las ventanas que forma el asistente. Podemos eliminar las que no queramos utilizar. También podemos crear distintas secuencias en función del perfil de instalación elegido por el usuario (normal o como administrador).



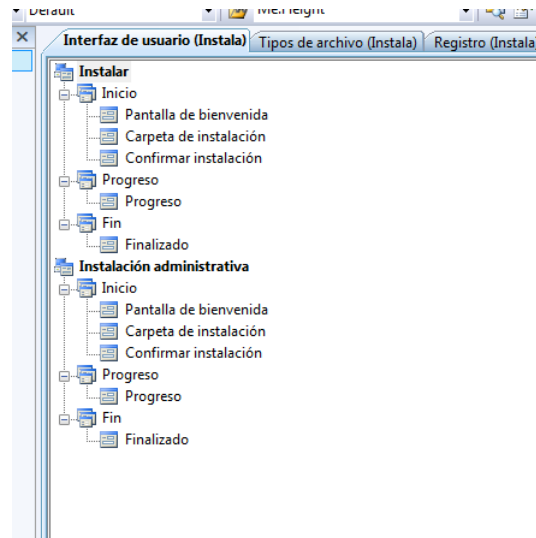
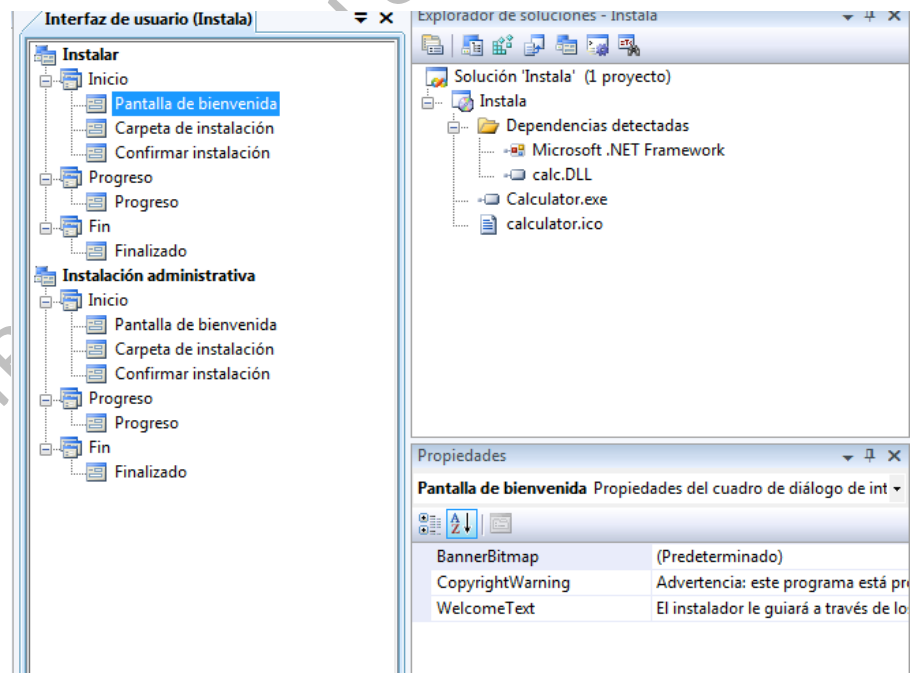


Figura: Interfaz del asistente

Una vez seleccionadas las ventanas que formarán nuestro asistente vamos a ir trabajándolas una por una. Para ello, solo hay que seleccionar una de las ventanas y editar sus propiedades para darles los valores deseados.

Figura: Propiedades de una ventana del asistente



Por ejemplo, en la imagen podemos observar las propiedades de la primera de las ventanas del asistente: la pantalla de bienvenida. En ella podemos definir el texto de bienvenida, el texto de advertencia de derechos de autor y la imagen de fondo.

Las propiedades que tenemos difieren en cada una de las ventanas.



#### 4.1.8. Fase 6: Acciones personalizadas

---

Esta fase no siempre es necesaria. Aquí podemos definir acciones personalizadas en momentos clave de la instalación.

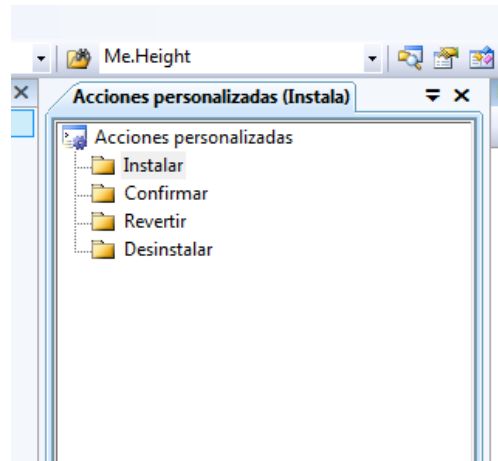


Figura: Acciones personalizadas

Para definir una acción personalizada es suficiente con hacer clic en el botón secundario del ratón en la fase que queremos editar y elegir **Agregar acción personalizada** del menú de contexto. A continuación, hemos de elegir un archivo ejecutable que contenga la acción que queremos llevar a cabo.

#### 4.1.9. Fase 7: Condiciones de inicio

---

Esta fase no siempre es necesaria. Aquí podemos definir las condiciones que debe cumplir el equipo destino para usar nuestra aplicación. No hay que confundir esto con los requisitos de instalación mencionados al principio del capítulo. En el caso de los requisitos, estos son librerías que debe contener el equipo destino para poder ejecutar nuestra aplicación. Las condiciones de inicio son librerías y programas que, aunque no sean estrictamente necesarios, nosotros queremos que el usuario cumpla.

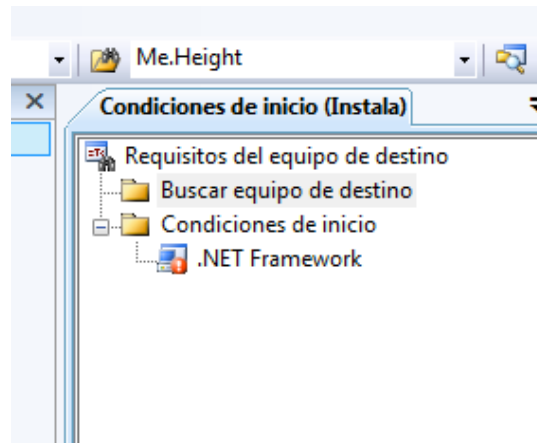


Figura: Condiciones de Inicio

Estas pueden referirse a determinados ficheros y programas, así como a elementos del registro de Windows. Por ejemplo, podríamos exigir que el usuario tuviera instalado Microsoft Word en su equipo.

#### 4.1.10. Toques finales

Lo único que nos queda por ejecutar son algunos retoques finales. Para ello, debemos hacer clic en el nombre del proyecto, en el explorador de soluciones y editar las propiedades.

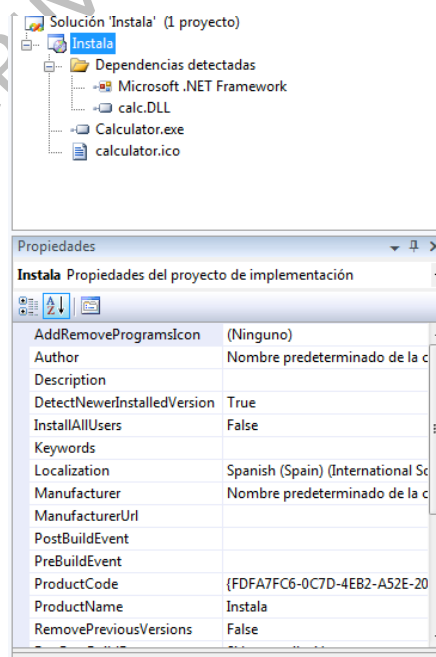
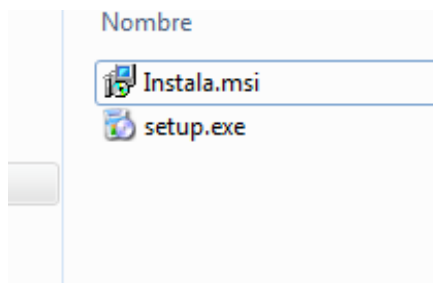


Figura: Toques finales



Aquí podemos definir elementos como el nombre del autor del programa, su compañía la versión, etc...Una vez terminado, salvamos el proyecto y lo generamos (Menú **Generar** y luego opción **Generar Solución** o F6). Esto genera los dos ficheros (uno .exe y otro .msi) que debemos distribuir al cliente.



Es recomendable probar el instalador en nuestro equipo para estar seguros que todo funciona como debería.



## TEMA 4 Creación de Instaladores

*Video: Creación de un Instalador*



## Conceptos clave

---

- **Distribución:** En este tema, hemos aprendido a crear nuevos paquetes instaladores para distribuir nuestras aplicaciones.
- **Personalización:** También hemos aprendido a personalizar la interfaz del instalador para adaptarlo lo mejor posible a nuestra aplicación.

VERSIÓN IMPRIMIBLE ALUMNOS LINKIAFP



## Test de autoevaluación

---

1. La extensión nativa de los instaladores para Windows es:
  - a) exe
  - b) msi
  - c) dll
  - d) lib
  
2. Para desarrollar un instalador:
  - a) Hay que cumplir TODAS las fases
  - b) Algunas fases son opcionales y no siempre hay que usarlas.
  - c) La selección de ficheros es la única fase obligatoria.
  - d) Hay 5 fases.
  
3. Cual es el objetivo principal de un instalador:
  - a) Agrupar todos los ficheros que forman nuestra aplicación en uno solo..
  - b) Restringir los usuarios que pueden usar nuestras aplicaciones
  - c) Permitir la venta y distribución de nuestras aplicaciones.
  - d) Permitir la instalación de nuestra aplicación en el equipo cliente.
  
4. La interfaz del instalador.
  - a) Se organiza en tres fases: Inicio, Progreso, Fin.
  - b) Tiene tres fases, pero se pueden agregar nuevas fases.
  - c) El nombre de las fases puede cambiar.
  - d) Tiene 3 fases, pero podemos eliminar cualquiera de ellas.



5. Sobre las ventanas que forman la interfaz del instalador.
  - a) Se pueden personalizar a nuestra elección.
  - b) Se pueden añadir nuevas ventanas creadas por nosotros.
  - c) No se pueden personalizar.
  - d) Solo podemos personalizar lo que nos permite el cuadro de propiedades.
  
6. Las carpetas del sistema de archivos:
  - a) No se pueden eliminar, pero se pueden añadir
  - b) La “Carpeta de la Aplicación” es obligatoria y no la podemos eliminar.
  - c) Les podemos cambiar el nombre si es necesario.
  - d) No podemos crear subcarpetas nuevas en su interior.

## Ponlo en práctica

---

### Actividad 1

Vamos a crear un instalador para nuestra aplicación de seguimiento de partidos de futbol. Para ello hay que instalar todos los archivos necesarios y personalizar todo lo posible la interfaz del instalador. Agregaremos la extensión personalizada (.fut) como archivos que deben abrirse con nuestra aplicación de futbol.



## Solucionario

### Test de autoevaluación

---

1. La extensión nativa de los instaladores para Windows es:
  - a) exe
  - b) msi**
  - c) dll
  - d) lib
  
2. Para desarrollar un instalador:
  - a) Hay que cumplir TODAS las fases
  - b) Algunas fases son opcionales y no siempre hay que usarlas.**
  - c) La selección de ficheros es la única fase obligatoria.
  - d) Hay 5 fases.
  
3. Cual es el objetivo principal de un instalador:
  - a) Agrupar todos los ficheros que forman nuestra aplicación en uno solo.
  - b) Restringir los usuarios que pueden usar nuestras aplicaciones
  - c) Permitir la venta y distribución de nuestras aplicaciones.
  - d) Permitir la instalación de nuestra aplicación en el equipo cliente.**
  
4. La interfaz del instalador.
  - a) Se organiza en tres fases: Inicio, Progreso, Fin.**
  - b) Tiene tres fases, pero se pueden agregar nuevas fases.
  - c) El nombre de las fases puede cambiar.
  - d) Tiene 3 fases, pero podemos eliminar cualquiera de ellas.





5. Sobre las ventanas que forman la interfaz del instalador.
  - a) Se pueden personalizar a nuestra elección.
  - b) Se pueden añadir nuevas ventanas creadas por nosotros.
  - c) No se pueden personalizar.
  - d) Solo podemos personalizar lo que nos permite el cuadro de propiedades.**
  
6. Las carpetas del sistema de archivos:
  - a) No se pueden eliminar, pero se pueden añadir
  - b) La “Carpeta de la Aplicación” es obligatoria y no la podemos eliminar.**
  - c) Les podemos cambiar el nombre si es necesario.
  - d) No podemos crear subcarpetas nuevas en su interior.

## Ponlo en práctica

---

### Actividad 1

Vamos a crear un instalador para nuestra aplicación de seguimiento de partidos de futbol. Para ello hay que instalar todos los archivos necesarios y personalizar todo lo posible la interfaz del instalador. Agregaremos la extensión personalizada (.fut) como archivos que deben abrirse con nuestra aplicación de futbol.

**Solución:** La encontrarás en la versión interactiva del aula.