

Añade funciones extra a tu servicio de mensajería instantánea

Crea tu primer Addin para Messenger

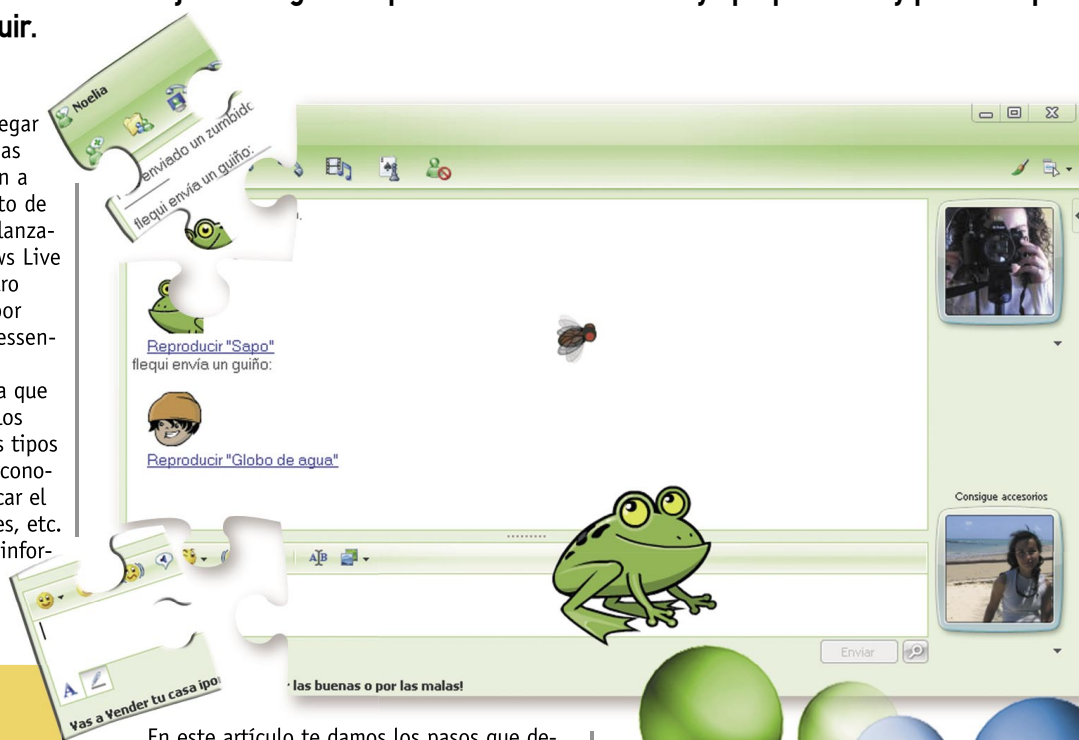
Incluido en el CD

EJERCICIO

Una de las maneras de añadir funciones extra a Live Messenger es mediante los addin. Se trata de pequeñas funciones que mejoran la funcionalidad de tu sistema de mensajería a tu gusto. Aprende a desarrollar el tuyo propio en muy poco tiempo. Verás lo que eres capaz de conseguir.

Windows Live Messenger permite agregar addins, es decir, pequeños programas creados por los usuarios que ayudan a mejorarlo y añaden nuevas funciones al gusto de cada uno. Cierto es que a pesar de haberse lanzado como novedad en la versión 8 de Windows Live Messenger no gozan de la popularidad de otro tipo de accesorios que pueden ser creados por particulares, como los famosos Scripts de Messenger Plus! Live.

Existe una amplia gama de posibilidades ya que con los addin se puede manejar y modificar los mensajes que envías y recibes, enviar nuevos tipos de mensaje, monitorizar cambios de estado, conocer las propiedades de tus contactos, modificar el nick, crear mensajes personalizados, imágenes, etc. Puedes crear desde robots o Bots (programa informático que realiza muy diversas funciones imitando a los humanos) hasta añadirle nuevas funcionalidades a tu cliente.

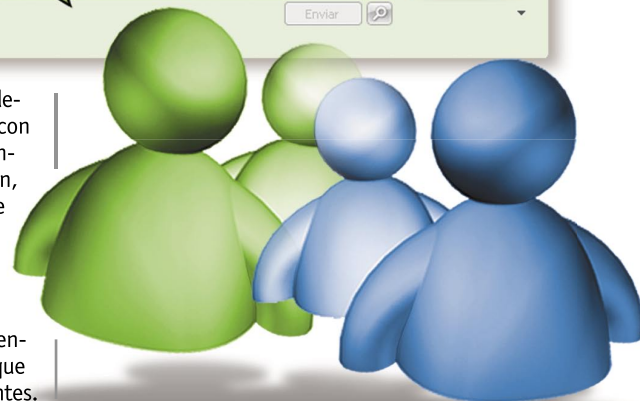


¿Qué es un addin?

Un addin o accesorio para Windows Live Messenger es un pequeño archivo .dll (librería dinámica de Windows) usado por Windows Live Messenger para añadir funciones extra al cliente de mensajería. Su funcionamiento se asimila al de cualquier plugin pudiendo ser activado o desactivado por el usuario. En cuanto a su funcionalidad puede llegar a ser muy potente aunque con algunas restricciones. Se desarrollan bajo la plataforma .NET Framework 2.0 de Microsoft pudiéndose elegir entre los lenguajes C# o Visual Basic .NET.

En este artículo te damos los pasos que debes seguir para conseguir desarrollar un addin con el que controles el cambio de estado de tus contactos, envíes un mensaje de texto si te saludan, controles el estado y cargues una imagen desde un fichero desde un solo addin.

Además, para que todo te resulte muy fácil puedes utilizar los archivos que encontrarás en el CD o bien descargártelos desde la página web www.messengeradictos.com. En el siguiente artículo te enseñamos a desarrollarlos para que conozcas cuáles son sus ventajas e inconvenientes.



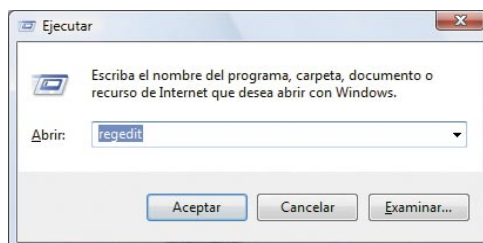
1 Antes de comenzar

1.1 Activa el soporte para addins

El soporte de addins para Windows Live Messenger está disponible en las versiones 8.1 y 8.5 Beta. En las versiones de prueba el soporte para addins viene desactivado por defecto por lo tanto tendrás que activarlo tal y como se detalla en los siguientes pasos.

Primero, haz clic en el menú **Inicio** de Windows y selecciona **Ejecutar**. Si prefieres puedes pulsar la combinación de teclas **Windows + R**.

A continuación, escribe **regedit** y pulsa sobre **Aceptar** para continuar. De esta manera entras



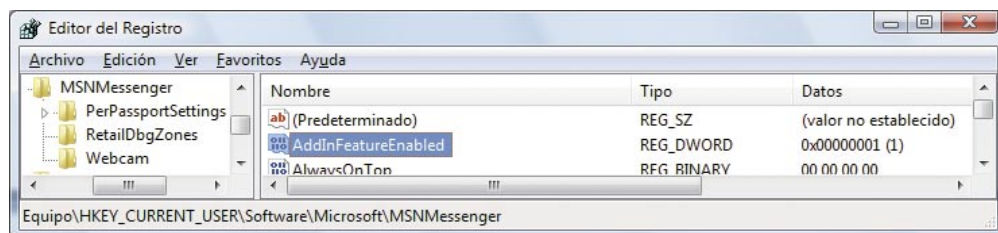
en el registro. Te aconsejamos que hagas una copia de seguridad del mismo antes de continuar. Modificar el registro por error puede hacer que tu equipo deje de funcionar.

A continuación, navega a través del árbol de directorios del **Registro** hasta la siguiente ubicación: **HKEY_CURRENT_USER\SOFTWARE\Microsoft\MSNMessenger**.

Una vez allí, añade un nuevo valor. Para ello, haz clic con el botón derecho del ratón sobre una zona en blanco de la parte derecha de la ventana y selecciona **Nuevo, Valor DWORD**.

Como nombre ponle **addinFeatureEnabled**. A continuación, haz doble clic sobre el valor que acabas de crear para modificar su valor a

Con esto ya habrás habilitado el soporte para addins de Windows Live Messenger. No obstante, si has realizado estos pasos mientras Live

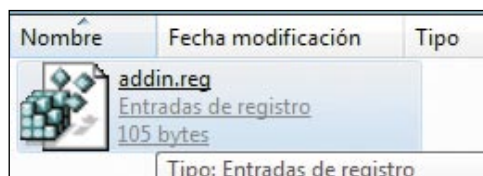


1 dentro del apartado llamado **Información del valor**. Una vez seguidos estos pasos, pulsa sobre **Aceptar** para guardar los cambios y sal del registro de **Windows**.

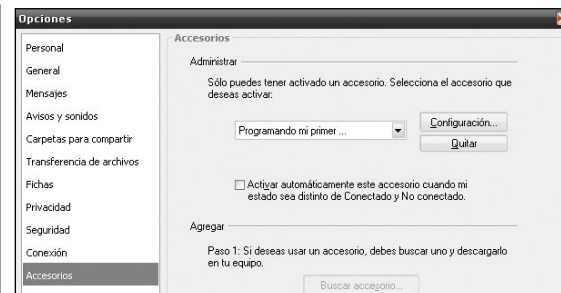
Otra forma de activar el soporte es crear un archivo de texto vacío. Guárdalo con el siguiente nombre **addin.reg**. A continuación, ábrelo para editarlo y copia el siguiente texto dentro:

```
REGEDIT4
[HKEY_CURRENT_USER\Software\Microsoft\
MSN Messenger]
"addinFeatureEnabled"=dword:00000001
```

Una vez que lo has copiado, guarda los cambios y a continuación, ejecútalo para activar el soporte.



Messenger estaba funcionando, debes cerrar la sesión y volverla a abrir para que los cambios surtan efecto. Para verificar que ahora Windows Live Messenger tiene habilitado el soporte para addins, ve a menú **Herramientas, Opciones**. Una vez allí, si todo ha ido bien, debes ver que se ha creado una nueva sección en el cuadro de la izquierda llamada **addins** o **Accesorios**.



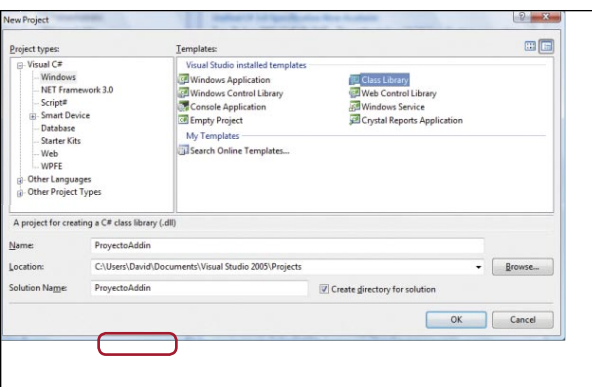
1.2 Instala Visual Studio

El software que necesitas para crear tu addin es el entorno de desarrollo Visual Studio 2005. Este entorno está creado para desarrollar bajo la tecnología .NET Framework 2.0 necesaria para la creación de los addins. Existen versiones Express que son gratuitas que te servirán para tus propósitos. Puedes obtener la versión de C# en el siguiente enlace www.microsoft.com/spanish/msdn/vstudio/express/VCS/default.aspx y la versión Visual Basic .NET en este otro: www.microsoft.com/spanish/msdn/vstudio/express/VB/default.aspx Una vez instalado el entorno ya sólo tienes que abrirlo y crear una biblioteca de clases. A continuación, se detallarán todos los pasos que debes seguir.

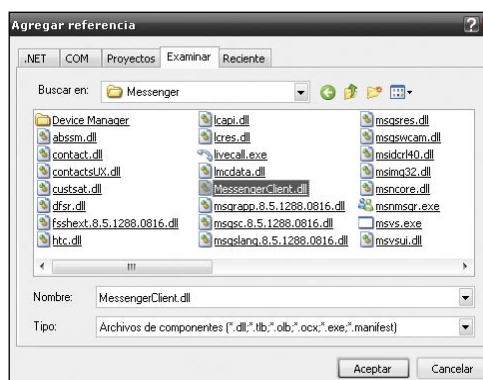
2 Crea los addins

2.1 Comienza el desarrollo

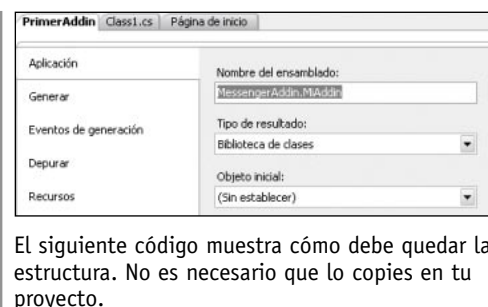
Para comenzar a desarrollar el addin abre Visual Studio. A continuación, crea un nuevo proyecto de la biblioteca de clases y selecciona un nuevo proyecto de biblioteca de clases. Para ello, ve a menú **Archivo, Nuevo, Proyecto**. Ahora guarda el proyecto con el nombre **Proyectoaddin** en un lugar que recuerdes porque luego lo vas a necesitar. A continuación, debes añadir la librería **MessengerClient.dll** que se encuentra ubicada en la carpeta de archivos de Messenger, siendo esta **C:\Archivos de Programa\MSN Messenger** en las



versiones antiguas de Messenger o bien **C:\Archivos de Programa\Windows Live\Messenger**. Esta librería contiene el espacio de nombres **Microsoft.Messenger**. Para añadirla haz clic en el menú **Proyecto** y luego en **Agregar Referencia**. Una vez abierta la ventana ve a la pestaña **Examinar** y busca la librería en la carpeta nombrada anteriormente.



Después, tienes que cambiar las propiedades del proyecto, de forma que el ensamblado (assembly) sea igual al nombre de la clase del addin. Esto es importante si quieres que tu addin funcione correctamente. En este caso defínelo como sigue. Haz clic en el menú del proyecto y luego sobre **Propiedades**. En la nueva ventana que se abre coloca en la caja de texto **Nombre de Ensamblado** el nombre correcto, en este caso será **Messengeraddin.Myaddin**.



El siguiente código muestra cómo debe quedar la estructura. No es necesario que lo copies en tu proyecto.

```
namespace Messengeraddin
{
    public class Myaddin
    {
```

Ahora debes importar el espacio de nombres **Microsoft.Messenger** e implementar la interfaz **IMessengeraddin**. Copia el siguiente código y dentro del archivo **Class1.cs** de tu proyecto:

```
using Microsoft.Messenger;
namespace Messengeraddin
{
    public class Myaddin : IMessengeraddin
    {
        private MessengerClient messenger;

        public void Initialize(MessengerClient messenger)
        {
            this.messenger = messenger;
        }
    }
}
```

En este paso ya tendrás creado tu primer addin aunque, de momento, su funcionalidad será nula.

2.2 Añade funcionalidad

Ahora comienza la parte creativa, y probablemente la más emocionante del trabajo que consiste en conseguir que tu addin haga algo útil. Es hora de poner a trabajar a tu cerebro ideando nuevas funciones con las que dotar a Windows Live Messenger. Pero antes de que eso ocurra, tienes que agregar propiedades a tu addin. Para ello, puedes copiar o modificar a tu gusto el siguiente código. Sólo tienes que cambiar los valores a la derecha por los que tú desees. Copia el siguiente código en el archivo **Class1.cs** de tu proyecto, concretamente dentro de la clase. **Cuadro1**

```
public void Initialize(MessengerClient messenger)
{
    messenger.addInProperties.Creator = "PCI";
    messenger.addInProperties.Description = "Ejemplo de addin para Messenger";
    messenger.addInProperties.FriendlyName = "Programando mi primer addin";
    messenger.addInProperties.PersonalStatusMessage = "Visita unapaginaweb.com";
    messenger.addInProperties.Status = UserStatus.Online;
    messenger.addInProperties.Url = new Uri("http://www.unapaginaweb.com");
    messenger.addInProperties.UserTile = PonImagen();

    messenger.IncomingTextMessage += new EventHandler<IncomingTextMessageEventArgs>(messenger_IncomingTextMessage);
    messenger.OutgoingTextMessage += new EventHandler<OutgoingTextMessageEventArgs>(messenger_OutgoingTextMessage);
}
```

Cuadro2

```
using System;
using System.Collections.Generic;
using System.Text;
using Microsoft.Messenger;

namespace MessengerAddin
{
    public class MiAddin : Microsoft.Messenger.IMessengerAddin
    {
        public Microsoft.Messenger.MessengerClient messenger = new Microsoft.Messenger.MessengerClient();
        public void Initialize(MessengerClient e)
        {
            //Creamos el objeto
            messenger = e;

            //Propiedades del Add-in
            messenger.AddInProperties.Creator = "MessengerAdictos.com";
            messenger.AddInProperties.Description = "Ejemplo de addin para Messenger";
            messenger.AddInProperties.FriendlyName = "Programando mi primer Add-in";
            messenger.AddInProperties.PersonalStatusMessage = "Visita MessengerAdictos.com";
            messenger.AddInProperties.Status = UserStatus.Online;
        }
    }
}
```

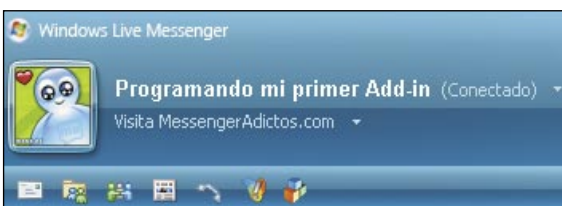
```
public void Initialize(MessengerClient messenger)
{
    messenger.addInProperties.Creator = "PCI";
    messenger.addInProperties.Description = "Ejemplo de addin para Messenger";
    messenger.addInProperties.FriendlyName = "Programando mi primer addin";
    messenger.addInProperties.PersonalStatusMessage = "Visita unapaginaweb.com";
    messenger.addInProperties.Status = UserStatus.Online;
    messenger.addInProperties.Url = new Uri("http://www.unapaginaweb.com");
    messenger.addInProperties.UserTile = PonImagen();
}
```

Cuadro1

La propiedad **UserTile** de los addins te permite cargar una imagen desde cualquier fuente ya sea creándola o desde un fichero. A continuación, copia el código mostrado a continuación y colócalo dentro de la clase como una nueva función. Debe estar entre **public class Myaddin** y **IMessengeraddin { }**. Te servirá de ejemplo cuya función es cargar una imagen desde un fichero:

```
private System.Drawing.Image PonImagen()
{
    return Image.FromFile("avatar.jpg");
}
```

Así verás las propiedades de tu addin una vez lo hayas cargado en tu Windows Live Messenger.



A continuación, añade el código del **Cuadro2** en la clase de tu proyecto para empezar a manejar los mensajes entrantes y salientes.

Esto te permite usar un manejador y llamar al procedimiento **messenger_IncomingTextMessage** y **messenger_OutgoingTextMessage** automáticamente cada vez que se envíe o reciba un mensaje de texto. El siguiente código (**Cuadro3**) es capaz de enviar un mensaje de texto si un contacto te dice hola y enviar un zumbido si tú escribes en una conversación la palabra zumbido.

```
private void messenger_IncomingTextMessage(object sender, IncomingTextMessageEventArgs e)
{
    string msg = e.TextMessage.ToLower();
    if (msg.Contains("hola"))
        messenger.SendTextMessage("hola!!!", e.UserFrom);
}

private void messenger_OutgoingTextMessage(object sender, OutgoingTextMessageEventArgs e)
{
    string msg = e.TextMessage.ToLower();
    if (msg.Contains("zumbido"))
        messenger.SendNudgeMessage(e.UserTo);
}
```

Cuadro3

En tu addin también puedes controlar cuándo se cambian de estado tus contactos, para ello debes añadir un nuevo manejador y una nueva función. Copia el siguiente código en la función **Initialize()**.

```
messenger.StatusChanged += new EventHandler<StatusChangedEventArgs>(messenger_StatusChanged);
```

y, a continuación, copia este otro código como una función nueva:

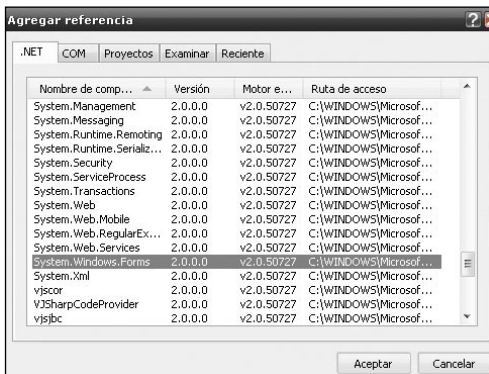
```
private void messenger_StatusChanged(object sender, StatusChangedEventArgs e)
{
    if (e.User.Status == UserStatus.Offline)
        messenger.addInProperties.Status = UserStatus.Offline;
}
```

Esta función te desconectará del Windows Live Messenger cada vez que un usuario se desconecte, comprueba si el estado del contacto es desconectado y te muestra su estado. Por último, tienes la posibilidad de crear un formulario de configuración para tu addin. Estos aparecerán cuando hagas clic en el botón **Configuración** en el apartado **Accesorios** de Windows Live Messenger. Debes añadir un nuevo manejador en el apartado **Initialize()** y una nueva función que sea capaz de crear el entorno de configuración. Copia el siguiente código del **Cuadro4** para crearlo. La función crea un nuevo formulario con una caja de texto. Recuerda que para poder utilizar los formularios de Windows debes añadir la referencia **System.Windows.Forms.dll** e importar el espacio de nombres. Para añadir el espacio de nombre sólo tienes que copiar el siguiente código en la cabecera del fichero:

```
using System.Windows.Forms;
```


Y para añadir la referencia haz clic sobre el árbol de tu proyecto y luego sobre **Agregar referencia**. Después ve a la pestaña **.NET** y busca la referencia **System.Windows.Forms**.

Una vez que has llegado a este punto ya tendrás un addin completamente funcional.



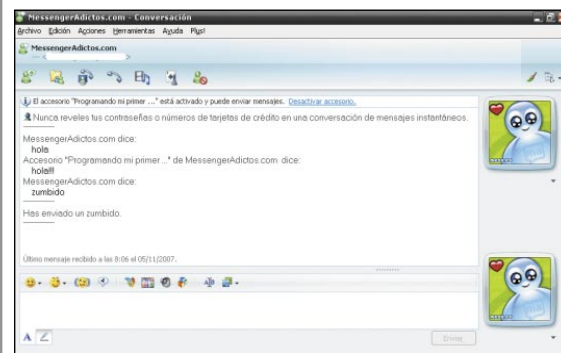
```
messenger.ShowDialog += new
EventHandler(messenger_ShowOptionsDialog);

private void messenger_ShowOptionsDialog(object sender, EventArgs e)
{
    TextBox texto;
    Form form1 = new Form();
    form1.Show();
    form1.Text = "Ventana de configuración";
    texto = new TextBox();
    texto.Dock = DockStyle.Top;
    texto.Location = new Point(20, 70);
    texto.Size = new Size(100, 50);
    form1.Controls.AddRange(new Control[] { texto });
}
```

Cuadro4

Ahora sólo tienes que compilarlo e instalarlo en tu Windows Live Messenger. En el CD de la revista puedes encontrar una plantilla creada en Visual Studio 2005 para el lenguaje C# que sólo tendrás que abrir y añadir el código que desees. En la siguiente imagen puedes ver

cómo será el funcionamiento de tu addin una vez lo hayas finalizado.



Crea un addin divertido: un Imitador

El siguiente ejemplo te servirá para imitar todo lo que dicen tus contactos, es decir todo lo que tus contactos te escriban, tú lo repetirás tal y como te lo han escrito. Sólo tienes que hacer uso del manejador **IncomingTextMessage** y para enviar el mismo mensaje. Debes utilizar el argumento **TextMessage**

que almacenará el mensaje recibido de tu contacto. Copia el siguiente código en el archivo de clases de tu proyecto (**Class1.cs**) y te servirá como ejemplo en el desarrollo de tu primer addin. También puedes usarlo desde el CD de la revista o descargarlo desde www.messengeraddictos.com

```
using System;
using Microsoft.Messenger;

namespace MiPrimeraddin
{
    /// <summary>
    /// Este ensamblado deberá llamarse 'MiPrimeraddin.addin.dll'
    /// para que el windows live messenger pueda cargarlo.
    /// </summary>
    public class Miaddin : IMessengeraddin
    {
        MessengerClient messenger;

        /// <summary>
        /// Implementación de la inicialización del addin es usada
        /// por el windows live messenger cuando se carga el addin
        /// </summary>
        void IMessengeraddin.Initialize(MessengerClient client)
        {
            messenger = client;

            // Propiedades del addin
            messenger.addinProperties.FriendlyName = "Imitador";
            messenger.addinProperties.PersonalStatusMessage =
                "unpaginaweb.com";

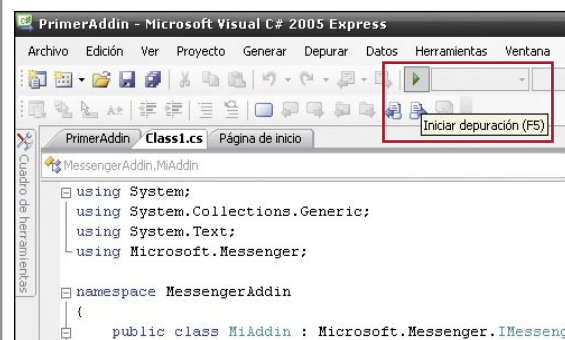
            // Manejador de los mensajes entrantes
            messenger.IncomingTextMessage +=
                new EventHandler<IncomingTextMessageEventArgs>(
                    this.OnIncomingMessage);

        }

        /// <summary>
        /// Cuando el addin está activado y el usuario recibe un
        /// mensaje se envía el mismo mensaje al contacto.
        /// </summary>
        public void OnIncomingMessage(object sender,
            IncomingTextMessageEventArgs args)
        {
            messenger.SendTextMessage(args.TextMessage,
                args.UserFrom);
        }
    }
}
```

23 Compila e instala el addin

Para compilar un addin sólo debes hacer clic en el botón **Iniciar depuración** que se encuentra en la barra de herramientas. También puedes pulsar la tecla **F5**. Una vez que ha terminado la depuración y no aparecen errores ya puedes instalarlo en Windows Live Messenger.



El siguiente paso consiste en instalar en Live Messenger el addin que acabas de crear. Para ello, ve a menú **Herramientas, Opciones, Accesorios**. A continuación, haz clic en el botón **Agregar a Messenger**.

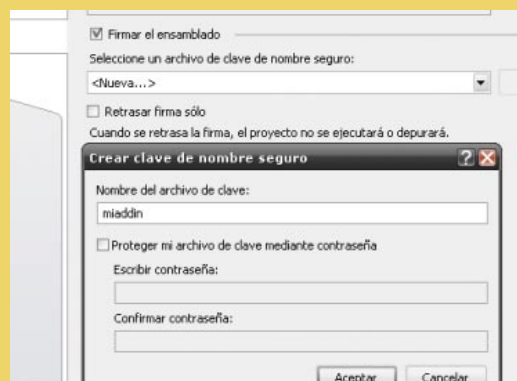
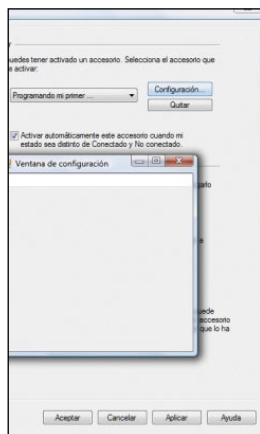
Una vez compilado tu addin éste se encuentra ubicado en la carpeta donde creaste el proyecto. Por defecto, lo encontrarás en **C:\Documents and**



¿Restricciones de la compilación?

Los addins para Windows Live Messenger se ejecutan en una especie de caja de arena de seguridad, es decir, que por defecto no te va a dejar acceder al sistema de ficheros, registro de Windows, Internet, etc. Esta restricción la puedes evitar introduciendo tu addin en la caché de ensamblados global (GAC). Para ello, primero debes dar un nombre seguro a tu proyecto. Ve a las propiedades del proyecto y marca la casilla **Firmar el ensamblado**.

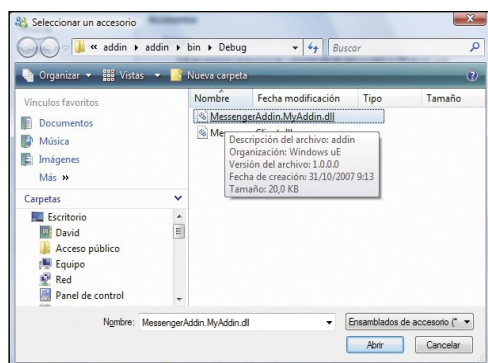
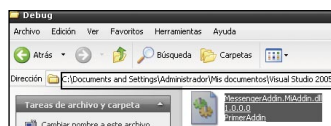
A continuación, pulsa sobre **Nuevo** y ponle el nombre **miaddin**. Para finalizar debes ir a **Inicio, Programas, Visual Studio 2005, Herramientas, Visual Studio 2005 Command Prompt**. Se abre la ventana **Terminal**. Una vez abierta dirígete con el comando **cd** a la carpeta de tu proyecto, exactamente en donde se encuentra la **dll** y ejecuta el siguiente comando: **gacutil /i NombreMiaddin.dll**. Esto debes hacerlo una vez que hayas compilado tu addin y siempre antes de cargarlo en tu Windows Live Messenger.



Settings\NOMBRE USUARIO WINDOWS\Mis documentos\Visual Studio 2005\Projects\Nombre-Proyecto. Una vez que te encuentres en la carpeta de tu proyecto, verás una nueva carpeta llamada como el nombre de tu proyecto. Dentro de está se encuentra otra carpeta llamada **bin** y a su vez

dentro de esta otra carpeta llamada **Debug**. Aquí encontrarás tu addin para cargarlo en tu Windows Live Messenger. Aquí

aparecen dos librerías (dll), una será **Messenger-Client.dll** y la otra será la que has creado y la que tienes que añadir a tu Windows Live Messenger.



Si tu addin tiene formulario de configuración, es decir, un diálogo que permite introducir datos o almacenar sobre tu addin, haz clic en

el botón **Configuración** para abrirlo. Una vez cargada debes activarla, para ello dirígete a la ventana de contactos y haz clic sobre tu nick. En el menú que aparece haz clic en **Activar "Mi Primer addin"**. De esta manera ya has finalizado



la instalación por lo que ahora sólo te queda disfrutar del addin que has creado.

Detalles de la API

A continuación, te detallamos todas las funcionalidades de la API (Interfaz de programación de aplicaciones) para que te sea más fácil trabajar con ellas.

Clases Messenger addin: clases soportadas por la API

Clase	Descripción
MessengerClient.addinProperties Class	Atributos del addin.
MessengerClient.IncomingTextMessageEventArgs Class	Guarda información sobre el evento IncomingTextMessage.
MessengerClient.OutgoingTextMessageEventArgs Class	Guarda información sobre el evento OutgoingTextMessage.
MessengerClient Class	Información del addin y su instancia.
MessengerClient.StatusChangedEventArgs Class	Guarda información sobre el evento StatusChanged.
MessengerClient.User Class	Contiene los datos sobre el usuario local y remoto.

Eventos Messenger addin: eventos soportados por la API

Evento	Descripción
MessengerClient.IncomingTextMessage Event	Evento que sucede cuando un addin recibe un mensaje de texto.
MessengerClient.OutgoingTextMessage Event	Evento que sucede cuando el usuario local envía un mensaje a un contacto.
MessengerClient.ShowOptionsDialog Event	Evento que sucede cuando se accede al dialogo de opciones.
MessengerClient.Shutdown Event	Evento que sucede cuando se cierra Messenger.
MessengerClient.StatusChanged Event	Evento cuando el usuario o un contacto cambia de estado.

Métodos Messenger addin: métodos soportados por la API

Método	Descripción
MessengerClient.SendActionMessage Method	Envía un mensaje de acción a un contacto. Sólo se permite enviar un mensaje de texto, acción o zumbido por evento recibido.
MessengerClient.SendNudgeMessage Method	Envía un zumbido a un contacto. Sólo se permite enviar un mensaje de texto, acción o zumbido por evento recibido.
MessengerClient.SendMessage Method	Envía un mensaje de texto a un contacto. Sólo se permite enviar un mensaje de texto, acción o zumbido por evento recibido.

Clase MessengerClient.addinProperties

Parámetro	Descripción
Creator	Creador del addin.
Description	Descripción.
FriendlyName	Nick.
PersonalStatusMessage	Mensaje Personal.
Status	Estado.
URL	URL.
UserTile	Avatar.