

**IIS Avanzado. FrontPage. ASP.NET 3.5**

# Indice

<b>2. IIS avanzado - FrontPage - Web Designer.....</b>	<b>1</b>
1. Propiedades de los sitios web.....	1
1.1 Sitio Web.....	1
1.2 Directorio particular.....	8
1.3 Origen del web.....	9
1.4 Permisos de ejecución.....	10
1.5 Configuración de la aplicación.....	10
1.6 Instalar filtros ISAPI.....	15
1.7 Mensajes de error personalizados.....	16
1.8 Documentos.....	19
1.9 Encabezados HTTP.....	20
1.10 Seguridad de directorios.....	21
1.11 Extensiones de servidor.....	27
1.12 Versión de ASP.NET.....	28
2. IIS y Frontpage.....	29
2.1 Abrir sitios web.....	30
2.2 Crear sitios web.....	30
2.3 Mantenimiento de permisos en el servidor.....	33
3. Introducción a ASP.NET 3.5.....	42
4. Primeros pasos con ASP.NET.....	45
4.1 Ejecución del lado del servidor.....	54
5. ASP.NET y el servidor web.....	55
<b>Ejercicios.....</b>	<b>61</b>
Ejercicio 1.....	61
Ejercicio 2.....	61
Ejercicio 3.....	61
Ejercicio 4.....	62

## 2. IIS avanzado - FrontPage - Web Designer

### 1. Propiedades de los sitios web

#### 1.1 Sitio Web

Esta hoja de propiedades se utiliza para configurar básicamente el sitio web:

The screenshot shows the 'Propiedades de Sitio Web predeterminado' dialog box. The 'Sitio Web' tab is active. The 'Identificación del sitio Web' section contains the following fields: 'Descripción' (Sitio Web predeterminado), 'Dirección IP' (Ninguna asignada), and 'Puerto TCP' (80). The 'Conexiones' section shows 'Tiempo de espera de la conexión' set to 120 segundos and 'Habilitar mantenimiento de conexiones HTTP abiertas' checked. The 'Habilitar registro' section shows 'Formato de registro activo' set to 'Formato de archivo de registro extendido W3C'. The 'Avanzadas...' button is visible next to the 'Dirección IP' field. The 'Aceptar', 'Cancelar', 'Aplicar', and 'Ayuda' buttons are at the bottom.

Nos falta una opción realmente importante: el registro, opción que se activa en la parte inferior de la ficha.

Es posible recopilar información acerca de la actividad de los usuarios si habilitamos este registro para los sitios Web. La información se almacena en archivos ASCII. Este registro tiene muchas posibilidades y supera por mucho el ámbito de las características del conocido registro de sucesos de Windows. Los registros pueden incluir información referente a quién ha visitado el sitio, qué ha visto el visitante y cuándo se vio la información por última vez. Podemos utilizar los registros para evaluar la popularidad del contenido o identificar los cuellos de botella de la información.

Podemos configurar los sitios Web o FTP para que graben estas entradas de registro generadas por la actividad de los usuarios y del servidor. Los datos de registro de IIS pueden ayudar a regular el acceso al contenido, evaluar la popularidad del contenido, planear los requisitos de seguridad y resolver problemas

potenciales en los sitios Web o FTP. El registro de actividad del sitio IIS no se debe confundir con el registro de sucesos efectuado por Windows XP ó 2000, que se muestra con el Visor de sucesos. El registro en IIS es más extenso y lo veremos a continuación.

### El proceso de registro

El registro de un sitio Web o FTP se realiza mediante unos módulos que funcionan independientemente de las demás actividades del servidor. Podemos elegir el formato de los registros para cada sitio Web o FTP individual. Si está habilitado el registro en un sitio, podemos habilitarlo o deshabilitarlo individualmente para cada uno de sus directorios.

Cada formato de registro utiliza una zona horaria diferente como base para las horas mostradas en los registros. El formato extendido W3C utiliza el Horario universal coordinado (UTC), lo que antes llamábamos hora del meridiano de Greenwich. Los otros formatos utilizan la hora local. Las horas mostradas en los archivos de registro reflejan la hora que el servidor utiliza para procesar las peticiones y las respuestas. Estas horas no reflejan el tiempo transcurrido en la red hasta llegar al cliente ni el tiempo de proceso del cliente.

### Formatos de archivo de registro

Podemos elegir el formato que el servidor Web utiliza para registrar la actividad de los usuarios. Disponemos de los siguientes formatos:

1. Formato de archivo de registro extendido W3C
2. Formato de registro de Microsoft IIS
3. Formato del archivo de registro común NCSA
4. Registro ODBC

El formato de archivo de registro extendido W3C, el formato de archivo de registro Microsoft IIS y el formato de archivo de registro NCSA son todos formatos de texto ASCII. El formato extendido W3C y el formato NCSA registran datos con formato de año de cuatro dígitos. El formato de Microsoft IIS utiliza un formato de dos dígitos para el año 1999 y anteriores y un formato de cuatro dígitos para los años posteriores. El formato de registro que se proporciona con Microsoft IIS asegura la compatibilidad con versiones anteriores de IIS. Únicamente se puede utilizar el formato de archivo de registro extendido W3C para crear formatos de registro personalizados con los campos precisos que se necesiten.

A título informativo comentaremos los tipos de registro que existen en IIS:

1. Formato de archivo de registro extendido W3C

El formato extendido W3C es un formato ASCII que puede personalizarse con diversos campos diferentes. Puede incluir campos que considere importantes y limitar al mismo tiempo el tamaño del registro si omite los campos que no desea. Los campos están separados por espacios. La hora se registra como UTC (Horario universal coordinado).

En el ejemplo siguiente se muestran líneas de un archivo que incluye los campos siguientes: Hora, Dirección IP del cliente, Método, Recurso (URL) visitado, Estado del protocolo y Versión del protocolo.

```
#Software: Servicios de Internet Information Server 5.1 de Microsoft
```

## IIS Avanzado. FrontPage. ASP.NET 3.5

#Versión: 1.0

#Fecha: 1998-05-02 17:42:15

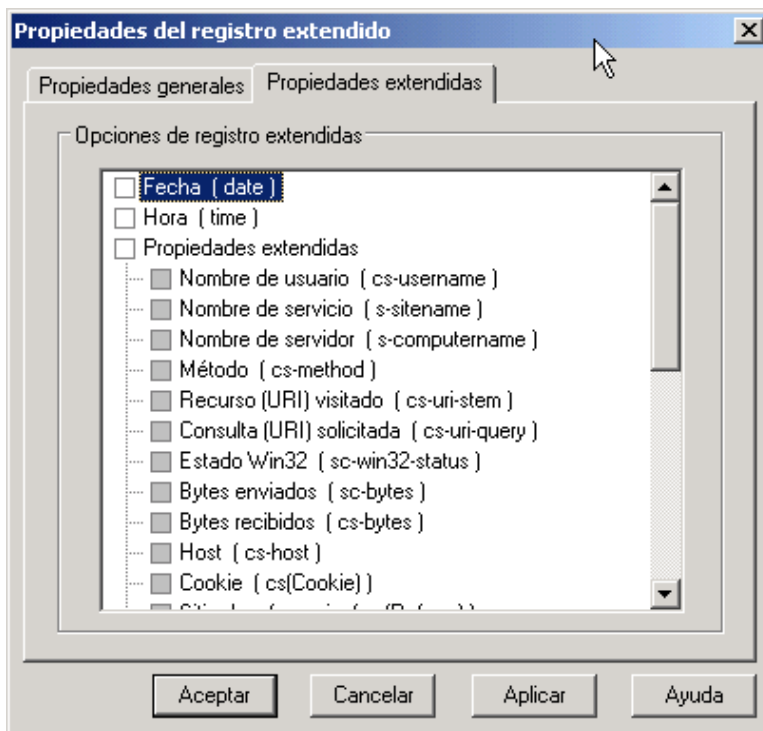
#Campos: time c-ip cs-method cs-uri-stem sc-status cs-version

17:42:15 172.16.255.255 GET /default.htm 200 HTTP/1.0

La entrada anterior indica que el 2 de mayo de 1998, a las 5:42 p.m., UTC, un usuario con HTTP versión 1.0 y dirección IP 172.16.255.255 emitió un comando GET de HTTP para el archivo \Default.htm. La petición se resolvió sin errores. El campo #Fecha: indica cuándo se hizo la primera entrada de registro, que es cuando se creó el registro. #Versión: indica que se utilizó el formato de registro W3C.

Podemos seleccionar cualquiera de los campos, pero puede que algunos no tengan información disponible para algunas peticiones. Para aquellos campos seleccionados que no tengan información aparecerá un guión ( ) en el campo como marcador de posición.

Al seleccionar este formato hemos dicho que podemos personalizar sus campos. Si pulsamos en el botón "Propiedades" nos aparecerá una pantalla cuya segunda ficha será como esta:



Esta pantalla permite indicar que campos queremos que figuren en el registro.

## 2. Formato de registro de Microsoft IIS

## IIS Avanzado. FrontPage. ASP.NET 3.5

El formato de Microsoft IIS es un formato ASCII fijo (no puede personalizarse) pero registra más datos que el formato común NCSA. El formato de Microsoft IIS incluye elementos básicos como la dirección IP del usuario, el nombre de usuario, la fecha y la hora de petición, el código de estado de servicio y el número de bytes recibidos. Además, incluye elementos detallados como el tiempo transcurrido, el número de bytes enviados, la acción (por ejemplo, una descarga realizada con un comando GET) y el archivo de destino. Los elementos se separan con comas, por lo que leer el formato resulta más sencillo que con los demás formatos ASCII, que utilizan espacios como separadores. La hora de registro es la local.

Al abrir un archivo con formato Microsoft IIS en un editor de textos, las entradas serán similares a las de los ejemplos siguientes:

```
192.168.114.201, -, 03/20/98, 7:55:20, W3SVC2, VHNTAS1, 192.168.114.201, 4502, 163, 3223, 200, 0, GHT  
172.16.255.255, anónimo, 03/20/98, 23:58:11, MSFTPSVC, VHNTAS1, 192.168.114.201, 60, 275, 0, 0, 0, PA
```

En las tablas siguientes se interpretan las entradas anteriores. La fila superior de cada tabla proviene de la segunda instancia del sitio Web (que aparece en "Servicio" como W3SVC2) y la fila inferior de la primera instancia del sitio FTP (que se indica en "Servicio" como MSFTPSVC1). El ejemplo se presenta en tres tablas por la limitación de ancho de página.

Dirección IP del usuario	Nombre del usuario	Fecha	Hora	Servicio instancia	Uso y resultado
192.168.114.201		03/20/98	7:55:20	W3SVC2	SALES1
172.16.255.255	anonymous	03/20/98	23:58:11	MSFTPSVC1	SALES1

Dirección IP del servidor	Tiempo empleado	Bytes enviados	Bytes recibidos	Código de estado de servicio	Código de estado de Windows
192.168.114.201	4502	163	3223	200	0
172.16.255.255	60	275	0	0	0

Tipo de solicitud	Destino de la operación	Parámetros
GET	/DeptLogo.gif	
[376] PASS	/intro.htm	

En el ejemplo, la primera entrada indica que un usuario anónimo, con la dirección IP 192.168.114.201, envió un comando GET de HTTP para el archivo de imagen /DeptLogo.gif a las 7:55 a.m. del 20 de marzo de 1998, desde un servidor llamado VENTAS1 que tiene la dirección IP 172.21.13.45. La petición HTTP de 163 bytes ha tenido un tiempo de proceso de 4502 milisegundos (4,5 segundos) y devolvió, sin errores, 3223 bytes de datos al usuario anónimo.

En el archivo de registro, todos los campos terminan en coma (.). Un guión ( ) actúa como marcador de posición si no hay un valor válido para un campo determinado.

### 3. Formato del archivo de registro común NCSA

## IIS Avanzado. FrontPage. ASP.NET 3.5

El formato común NCSA es un formato ASCII fijo (no puede personalizarse), disponible para sitios Web, pero no para sitios FTP. Registra información básica acerca de las peticiones de los usuarios, como nombre de host remoto, nombre de usuario, fecha, hora, tipo de petición, código de estado HTTP y número de bytes enviados por el servidor. Los elementos están separados con espacios en blanco y la hora de registro es la local.

Al abrir un archivo con formato común NCSA en un editor de textos, las entradas serán similares a las del ejemplo siguiente:

```
172.21.13.45 - REDMOND\fred [08/Apr/1997:17:39:04 -0800] "GET /scripts/iisadmin/ism.dll?http/serv HTTP/1.0"
```

Nota En la entrada anterior, el segundo campo (que mostraría el nombre de inicio de sesión remota del usuario) está vacío y está representado por el guión que sigue a la dirección IP 172.21.13.45.

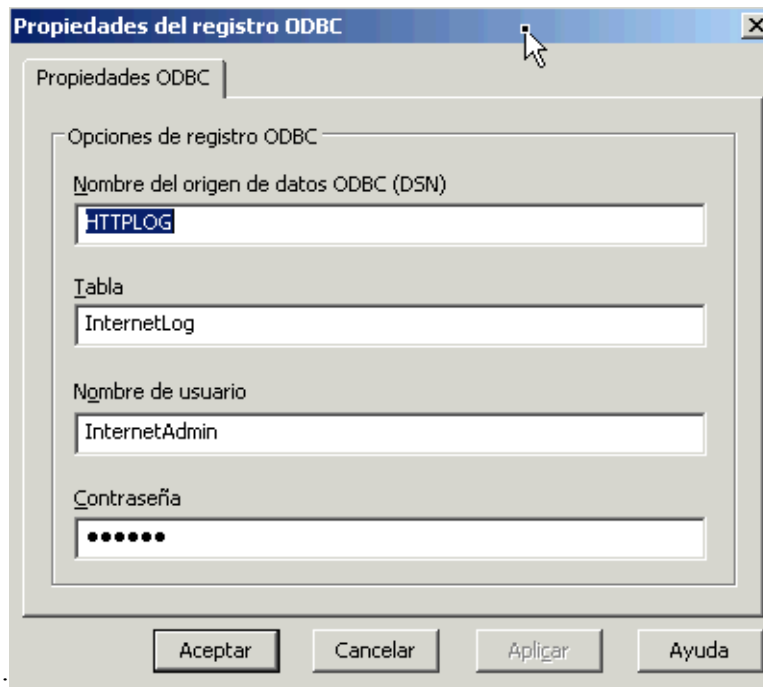
En las tablas siguientes se interpreta la entrada de ejemplo anterior. Se utilizan dos tablas por la limitación de ancho de página.

Nombre de host remoto	Nombre de inicio de sesión remota	Nombre de usuario	Fecha	Hora y diferencia con GMT
172.21.13.45		Rioja\jose	08 de abril de 1998	17:39:10 -0800

Petición/Versión	Código de estado de servicio	Bytes enviados
GET /scripts/iisadmin/ism.dll?http/serv HTTP/1.0	200	3401

La entrada indica que un usuario llamado Jose del dominio Rioja, con la dirección IP 172.21.13.45, envió un comando GET de HTTP (es decir, descargó un archivo) a las 5:39 p.m. del 8 de abril de 1998. La petición devolvió, sin errores, 3401 bytes de datos al usuario jose.

4. ODBC. Permite utilizar una base de datos para registrar los accesos. La configuración se realiza mediante el sistema estándar de Microsoft ODBC, indicando su



configuración:

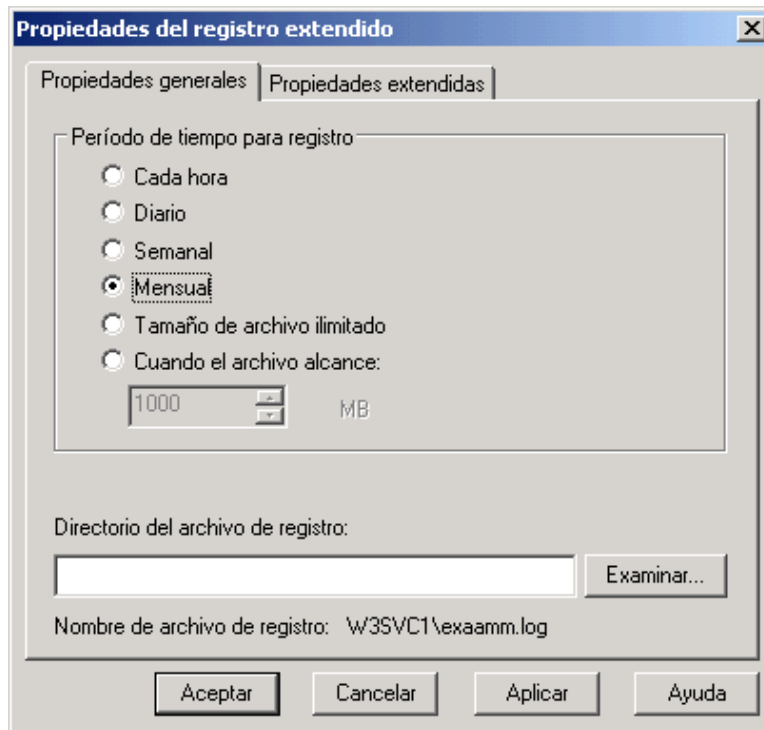
No te recomiendo esta configuración porque crea miles de entradas y crea unas bases de datos impresionantes. Como no hace falta una explotación de datos exhaustiva dejaremos cualquiera de los formatos anteriores, mas rápidos y sencillos de configurar.

### Tamaño de archivo de registro y creación de nuevos archivos de registro

Cuando está habilitado el registro de IIS, (lo está de manera predeterminada) se generan nuevas entradas de registro siempre que un usuario tiene acceso al servidor. Esto produce un incremento progresivo del tamaño del archivo de registro o del número de archivos de registro. Podemos necesitar equilibrar la recopilación de datos detallados con la necesidad de limitar los archivos a un número y tamaño fáciles de administrar. IIS ofrece dos opciones para administrar la generación de datos de registro y la creación de nuevos archivos de registro.

Una forma de administrar los datos de registro es personalizar el registro extendido W3C de modo que sólo se recopilen los datos que se necesitan. Otra opción para administrar archivos de registro es limitar el tamaño del registro mediante el cambio de la frecuencia de creación del archivo de registro. Esta opción aparece pulsando el botón Propiedades:





Los archivos de registro son simplemente archivos ASCII (de texto). Si ha creado muchos archivos pequeños y prefiere uno grande, puede combinarlos como haría con cualquier archivo ASCII.

Si el servidor se queda sin espacio en disco cuando IIS intenta agregar una entrada de registro a un archivo, el registro de IIS se cierra. Al mismo tiempo, en el registro de aplicación del Visor de sucesos de Windows, se registra un suceso. Cuando vuelve a haber espacio disponible en disco, se reanuda el registro de IIS. Esto hace que se registre un suceso adicional en el registro de aplicación del Visor de sucesos de Windows.

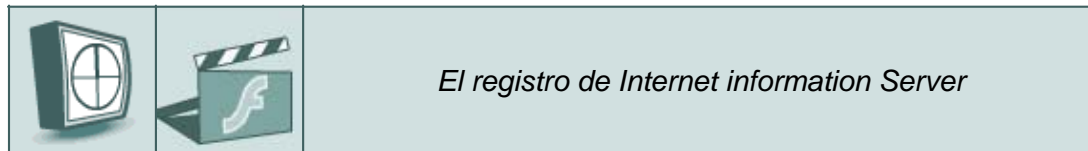
### Nombres de archivo de registro

En los nombres de los archivos de registro se utilizan las primeras letras para representar el formato y los números restantes para indicar el marco horario o la secuencia del registro. Consulta la tabla siguiente para obtener más información. Las letras en cursiva representan dígitos: nn para dígitos secuenciales, aa para el año, mm para el mes, ss para la semana del mes, dd para el día y hh para la hora en formato de 24 horas (es decir, 17 es 5:00 p.m.).

Formato	Criterio para nuevo registro	Patrón de nombre de archivo
Formato de registro de Microsoft IIS	Por el tamaño del archivo	inetsvnn.log
	Cada hora	inaammddhh.log
	Diario	inaammdd.log
	Semanal	inaammss.log
	Mensual	inaamm.log
Formato del archivo de registro común NCSA	Por el tamaño del archivo	ncsann.log
	Cada hora	ncaammddhh.log

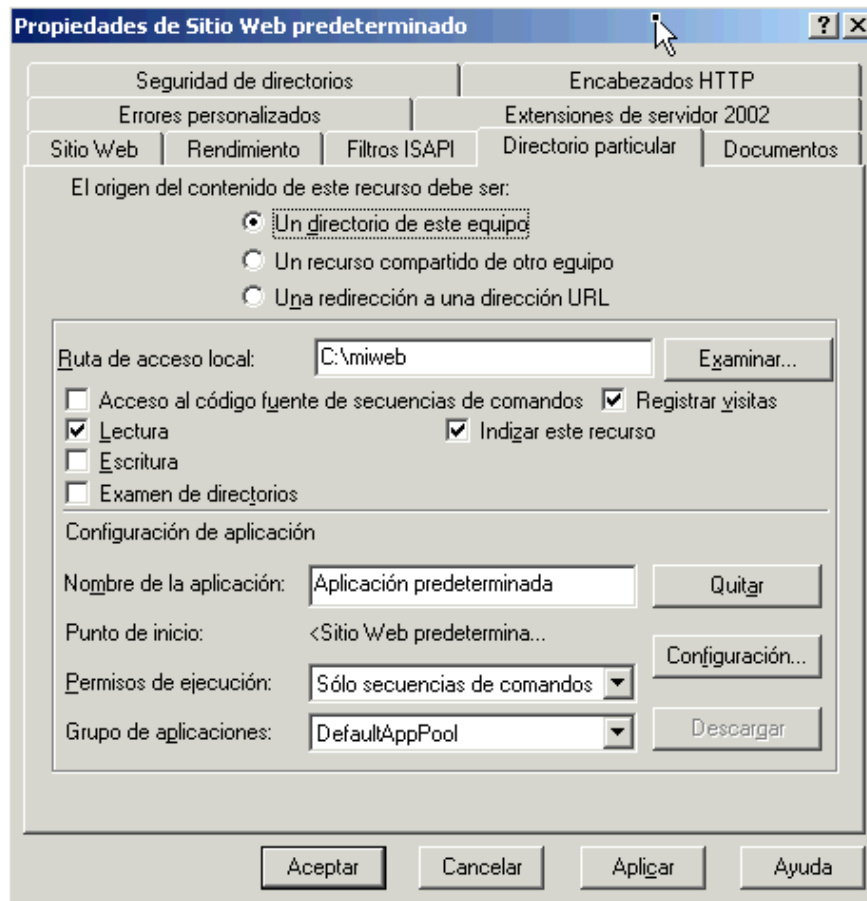
Formato de archivo de registro extendido W3C	Diario	ncaammdd.log
	Semanal	ncaammss.log
	Mensual	ncaamm.log
	Por el tamaño del archivo	extendnn.log
	Cada hora	exaammddhh.log
	Diario	exaammdd.log
	Semanal	exaammss.log
	Mensual	exaamm.log

**Nota** Existen muchos programas de otros fabricantes que recogen estos datos y los interpretan, creando potentes informes acerca del volumen de páginas descargadas, tamaños, usuarios, ...



## 1.2 Directorio particular

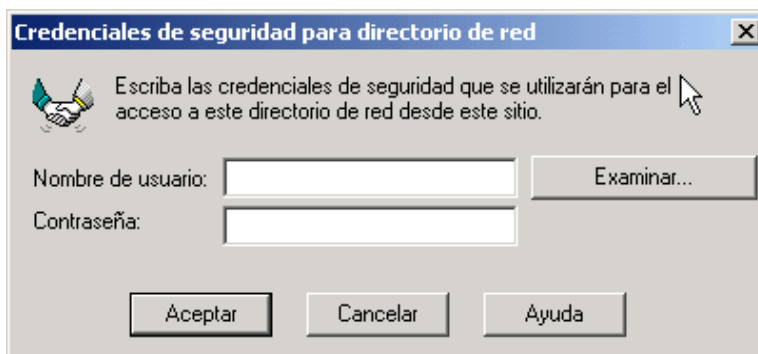
Esta hoja de propiedades se utiliza para configurar el directorio de publicación del sitio Web:



### 1.3 Origen del web

La primera parte hace referencia a la ubicación de las páginas web. Ya vimos en el capítulo anterior cómo asociábamos nuestro web con un directorio físico del disco duro, e incluso de directorios virtuales con directorios. Las tres opciones posibles son:

- Un directorio de este equipo: "c:\inetpub\wwwroot"
- Un recurso compartido de otro equipo. Si seleccionamos esta opción debemos indicar la ruta UNC \\servidor\recurso. Si además debemos identificarnos en ese directorio pulsaremos en el botón "conectar como" que se mostrará al elegir esta opción:



- Poco utilizada: redirección a una URL. Redirige un directorio virtual a la dirección URL de destino sin agregar ninguna parte de la dirección URL original. Puede utilizar esta opción para redirigir un directorio virtual completo a un archivo. Por ejemplo, para redirigir al archivo Predeter.htm en el directorio particular todas las peticiones realizadas al directorio virtual /scripts, escriba /Predeter.htm en el cuadro de texto Redirigir a y seleccione esta opción

### 1.4 Permisos de ejecución

Existen varias formas de dar permiso de acceso al sitio web:

- Acceso al código fuente de secuencias de comandos. Selecciona esta opción para permitir que los usuarios tengan acceso al código fuente si disponen de los permisos de escritura o de lectura. El código fuente incluye las secuencias de comandos en aplicaciones ASP.
- Lectura. Esta opción permite que los usuarios lean o descarguen archivos o directorios y sus propiedades asociadas.
- Escritura. Permitir que los usuarios carguen archivos y sus propiedades asociadas en un directorio con este permiso del servidor o para cambiar el contenido de un archivo con este permiso. La escritura sólo se puede realizar con un explorador que admita la característica PUT del estándar de protocolo HTTP 1.1.
- Examinar directorios. Esta opción permite que el usuario vea una lista con formato de hipertexto de los archivos y subdirectorios de este directorio virtual. Los directorios virtuales no aparecen en las listas de directorios. Los usuarios deben conocer su alias. Importante El servidor Web presentará el mensaje de error "Acceso prohibido" en el explorador Web del usuario si éste intenta tener acceso a un archivo o directorio y las dos condiciones siguientes se cumplen:
  - ◆ La opción Examinar directorios está deshabilitada.
  - ◆ El usuario no especifica un nombre de archivo, como NombreArchivo.htm.
- Registrar visitas. Es opción registra las visitas de este directorio en un archivo de registro. Las visitas sólo se registran si está habilitado el registro para este sitio Web.
- Indizar este recurso. Selecciona esta opción para que Servicios de Microsoft Index Server incluya este directorio en un índice de texto del sitio Web.

### 1.5 Configuración de la aplicación

Esta es la parte que enlaza el mundo ASP y ASP.NET con el IIS. Una aplicación es la capacidad del IIS para ejecutar secuencias de comandos como ASP y ASP.NET. Es decir si esta sección está desactivada no funcionarán nuestra páginas ASP ni ASP.NET

Una aplicación es un archivo que se ejecuta dentro de un conjunto definido de directorios de un sitio Web. Al crear una aplicación, el complemento IIS se usa para designar el directorio de punto de inicio de la aplicación, también denominado raíz de la aplicación, en el sitio Web. Cada archivo y subdirectorio incluido en este directorio del sitio Web se considera parte de la aplicación hasta llegar al directorio de punto de inicio de otra aplicación. Por tanto, los límites de los directorios permiten definir el alcance de las aplicaciones.

En el complemento IIS, el punto de inicio de una aplicación aparece indicado con un icono en forma de paquete. En el ejemplo siguiente se muestra una aplicación con un directorio de punto de inicio llamado /SiteAdmin:



Los archivos de los directorios \Login, \Order y \SetupDB se consideran parte de la aplicación /SiteAdmin.

Por lo tanto puede haber más de una aplicación por cada sitio Web. El sitio Web predeterminado creado al instalar Servicios de Internet Information Server es un punto de inicio de aplicaciones. Esto es importante porque cuando comencemos a construir páginas ASP.NET necesitaremos saber el ámbito de la aplicación para manejar nuestra variables. El concepto de aplicación es muy sencillo, simplemente es dividir nuestro web en "sitios independientes" donde cada uno puede tener sus variables y programas ejecutándose. Muchas veces nos encontraremos con que sólo necesitamos una "aplicación web" que coincide con el "Sitio Web predeterminado"

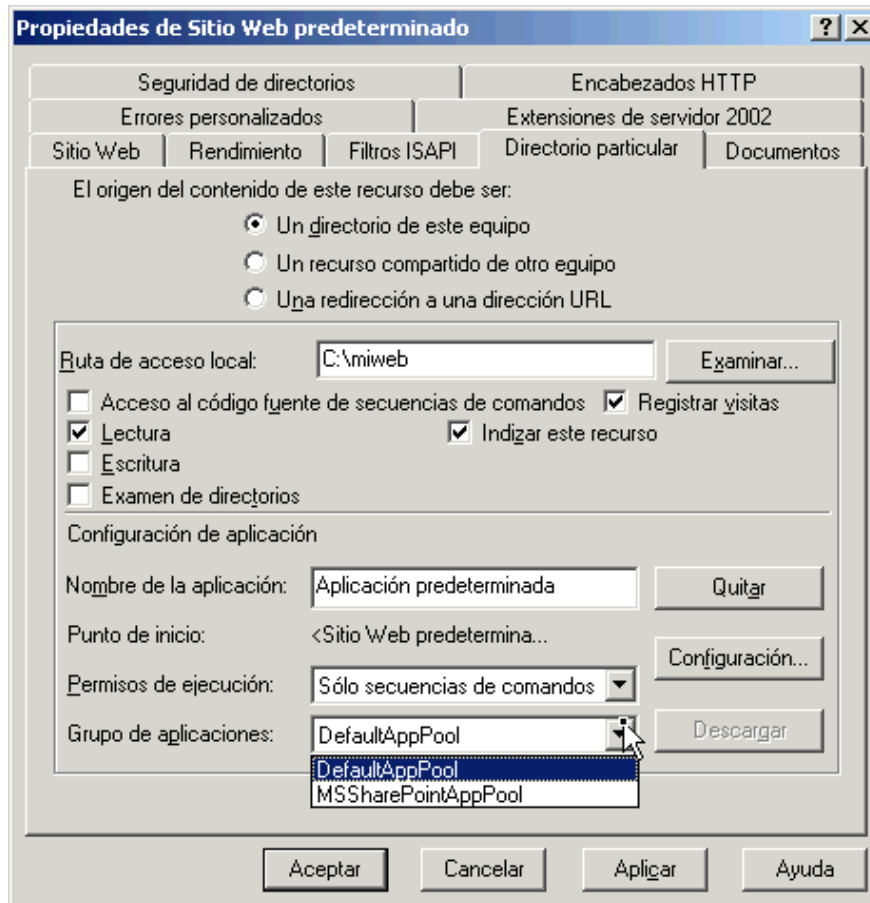
Resumiendo, sólo necesitamos indicar un nombre para la aplicación y con esto IIS creará internamente la estructura necesaria para que pueda ejecutar páginas ASP entre otras.

### Protección de aplicaciones

IIS 5.1 tenemos dos grados de protección de aplicaciones. La protección de aplicaciones se refiere al proceso en el que se ejecutan las aplicaciones. En IIS 4.0, se podían configurar las aplicaciones para ejecutarse en el mismo proceso como servicios Web (Inetinfo.exe) o en un proceso separado de los servicios Web (DLLHost.exe).

Estas opciones proporcionan dos grados de protección para situaciones en las que una aplicación que funciona de forma incorrecta puede tener un error de modo que el proceso en el que se ejecuta deje de responder. De forma predeterminada, los servicios Web (Inetinfo.exe) se ejecutarán en su propio proceso y otras aplicaciones lo harán en un único proceso agrupado (DLLHost.exe). La segunda opción "MSSahrePointApplPol" es mas apropiada cuando se emplea Microsoft Sharepoint como portal de páginas.

El gráfico siguiente ilustra la hoja de propiedades usadas para establecer las propiedades del grupo de aplicaciones.



La protección de aplicaciones sólo puede establecerse en un directorio de inicio de la aplicación.

## Establecer permisos para una aplicación

Los niveles de permisos para la aplicación son:

- Ninguno para impedir la ejecución de ningún programa o secuencia de comandos.
- Sólo secuencias de comandos para permitir la ejecución de las aplicaciones asignadas a un motor de secuencias de comandos en este directorio sin tener establecido el permiso Ejecución. Este es el que tenemos que tener activado para nuestra páginas ASP.NET. El permiso Secuencia de comandos es más seguro que el permiso Ejecución, ya que permite limitar las aplicaciones que se pueden ejecutar en el directorio.
- Establezca el permiso Secuencias de comandos y ejecutables para permitir la ejecución de cualquier aplicación en este directorio, incluso de las aplicaciones asignadas a motores de secuencias de comandos y archivos binarios de Windows (archivo .dll y .exe). (ojo con esto)

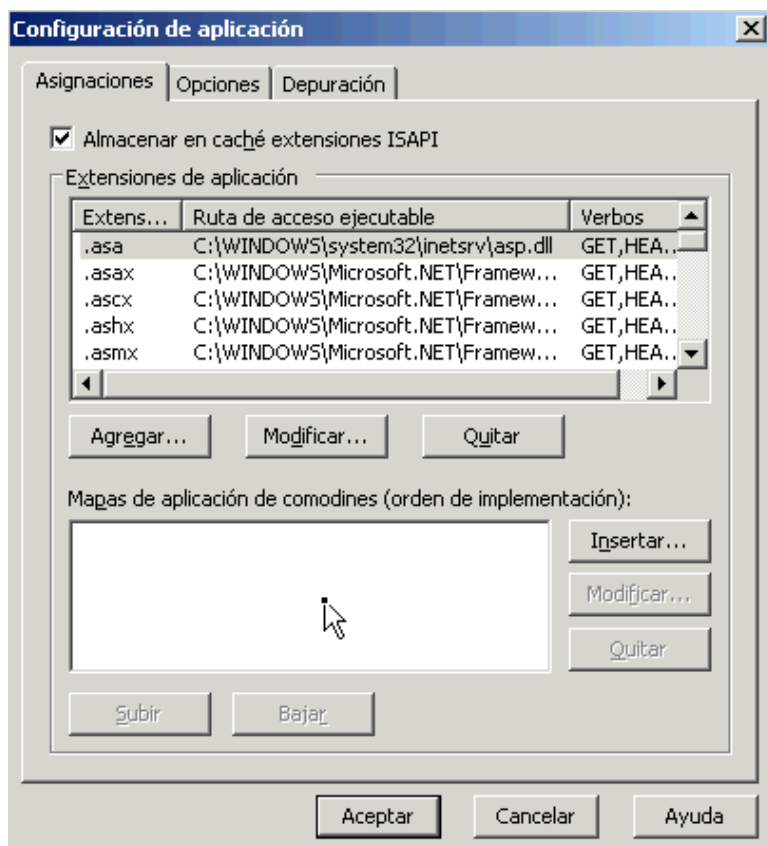
## Establecer asignaciones para la aplicación

Aunque esta sección entra en el nivel de avanzado la vamos a comentar para que tengamos una referencia completa de IIS. Se pueden desarrollar aplicaciones Web en diversos lenguajes de programación y de secuencias de comandos. Los Servicios de Internet Information Server (IIS) utilizan la extensión de archivo de un recurso solicitado en el sitio Web para determinar el programa ISAPI o CGI que se va a ejecutar para

procesar la petición. Por ejemplo, la petición de un archivo con la extensión .asp hace que el servidor Web inicie el programa ASP (Asp.dll) para procesar la petición. La asociación de una extensión de archivo a un programa ISAPI o CGI recibe el nombre de asignación de aplicación. IIS está configurado para permitir las asignaciones de aplicaciones comunes. Puede agregar o quitar asignaciones para todas las aplicaciones de un sitio Web o de aplicaciones individuales.

Para asignar una extensión a una aplicación:

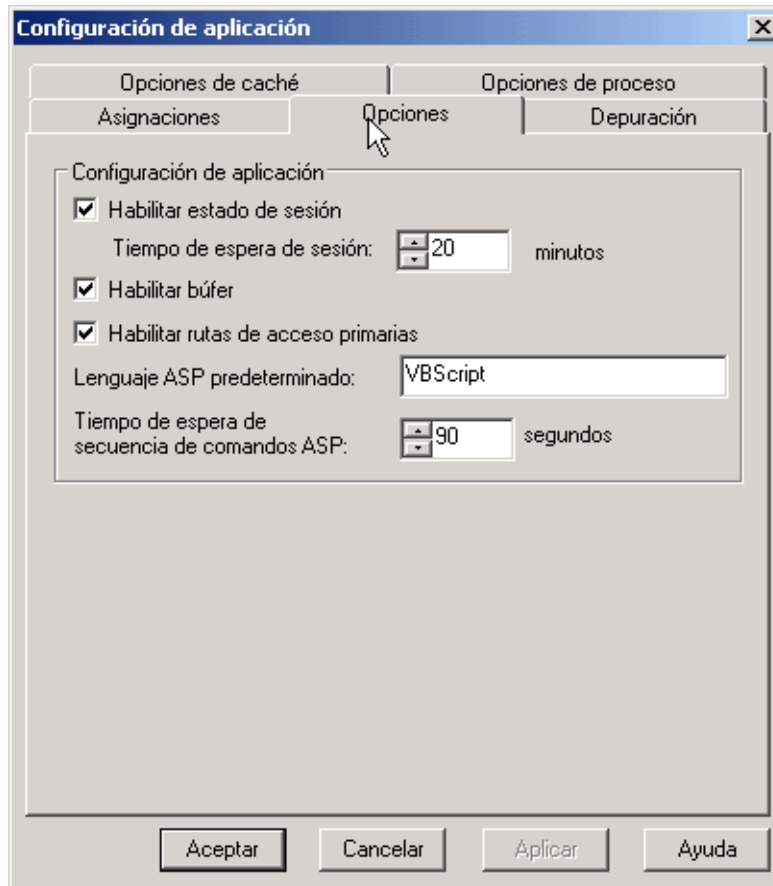
1. En la consola administrativa seleccionamos el sitio Web o el directorio que sirve como punto de inicio de una aplicación.
2. Abrimos las hojas de propiedades del directorio y haga clic en la ficha Directorio particular, Directorio virtual o Directorio.
3. Hacemos clic en el botón Configuración.
4. En la ficha Asignaciones seleccionamos Agregar.
5. En el cuadro Ejecutable, escribimos la ruta de acceso del programa ISAPI o CGI que procesará el archivo. Debemos especificar un programa en un directorio local del servidor Web.
6. En el cuadro Extensión, escribimos la extensión de archivo que desea asociar al programa ISAPI o CGI. Cuando el servidor Web reciba una dirección URL que identifique un archivo con esta extensión, iniciará el programa asociado para procesar la petición.
7. Para permitir el procesamiento de archivos de este tipo en un directorio con el permiso Secuencia de comandos, activamos la casilla de verificación Motor de secuencias de comandos. Si un directorio tiene establecido el permiso Secuencia de comandos (en lugar del permiso Ejecución), sólo se podrán procesar en el directorio los archivos asociados a aplicaciones que sean motores de secuencias de comandos designados.



Fíjate que las extensiones ASPX las va a interpretar el programa de la derecha algo así como "Framework". Esto es importante, cuando solicitamos una página, IIS mira en esta lista, si la extensión es una de ellas procesa primero esta página por ese programa. Así si tiene código ASP.NET lo ejecuta antes de devolver la página.

### Ficha de Opciones en propiedades de aplicación

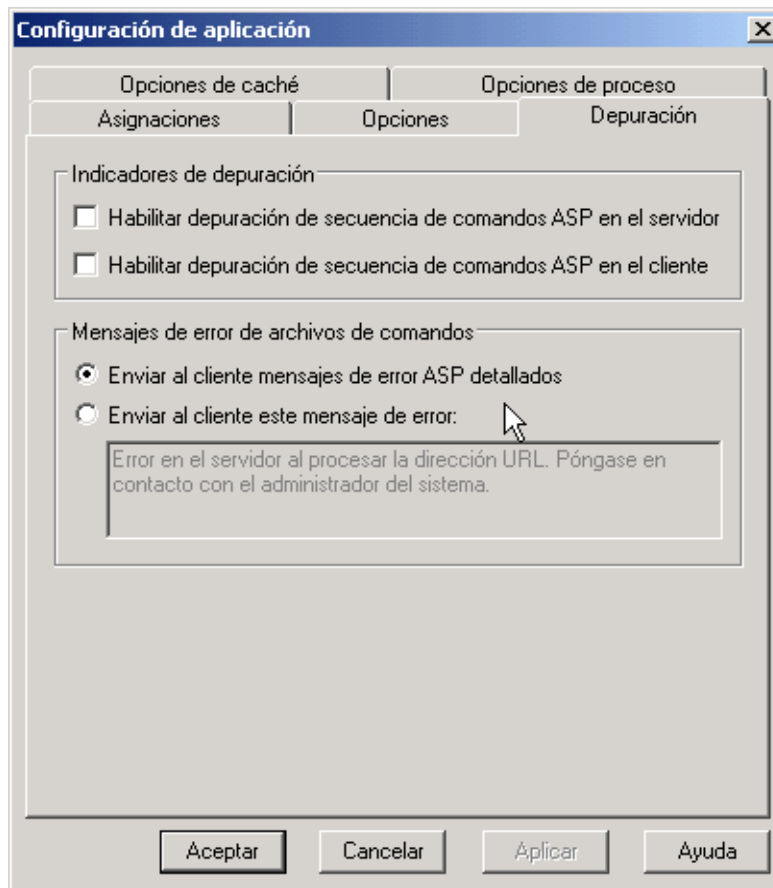
Aquí vamos a comentar las opciones que nos aparecen en la pestaña Opciones, algunas de ellas muy importantes:



- **Habilitar estado de sesión.** Esta opción es muy importante a la hora de trabajar con "variables globales" dentro de ASP/ASP.NET. Mas adelante veremos con profundidad todo el tema de las sesiones, tema vital en ASP.NET
- **Habilitar búfer.** Esta opción está deshabilitada por defecto en IIS 4.0. También afecta al funcionamiento de ASP/ASP.NET que veremos mas adelante.
- **Habilitar rutas de acceso primarias.** Permite que las páginas ASP utilicen rutas de acceso relativas para el directorio primario del directorio actual (rutas de acceso con la sintaxis ..)
- **Lenguaje ASP predeterminado.** Tenemos dos motores instalados: Vbscript y Javascript y podemos utilizar cualquiera de ellos. De forma predeterminada y el recomendado es VBScript. Esta opción es ya obsoleta porque ASP.NET trabaja de otra forma con los lenguajes, esto era solo para las viejas aplicaciones ASP.
- **Tiempo de espera:** Especifica el intervalo de tiempo que ASP/ASP.NET permitirá ejecutarse a una secuencia de comandos. Superado ese tiempo devolverá un error.

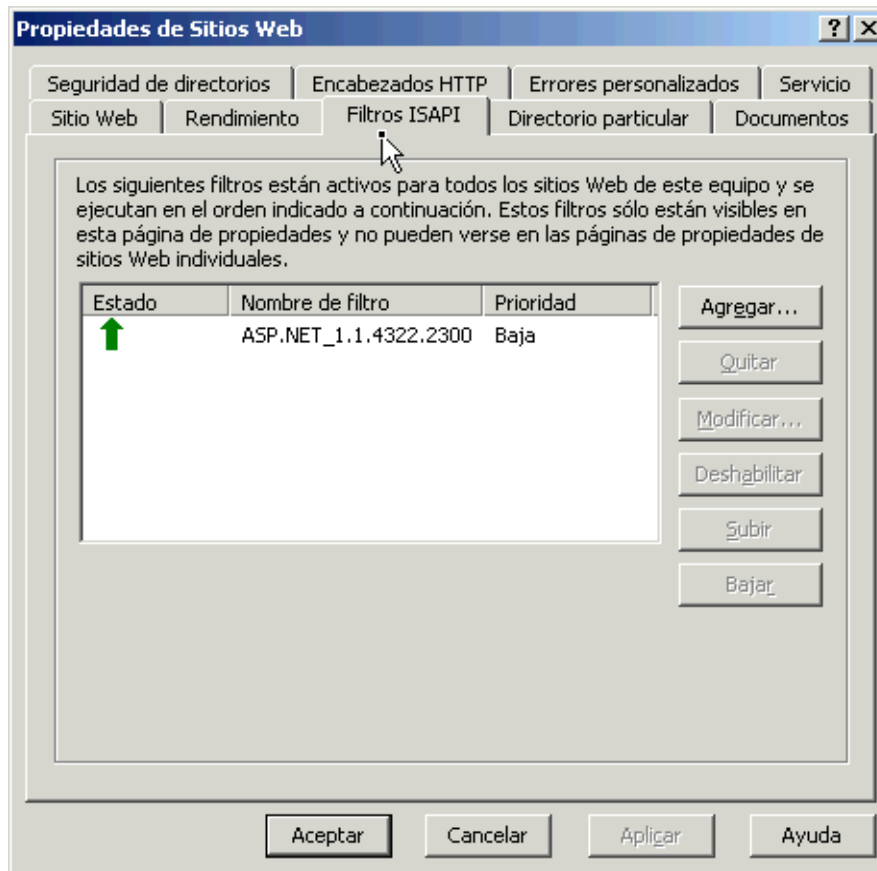


## Ficha de Opciones en propiedades de aplicación



Permite operaciones como la depuración de las secuencias ASP. Tenemos dos niveles o tipos de depuración: en el cliente o en el servidor.

## 1.6 Instalar filtros ISAPI



Es una opción que no utilizaremos seguramente nunca pero es bueno comentarla porque aprendemos más de las "tripas" de IIS. Al igual que las extensiones ISAPI (programación con IIS), los filtros ISAPI son programas que responden cuando el servidor Web recibe una petición HTTP. Sin embargo, a diferencia de las extensiones ISAPI, los filtros ISAPI se ejecutan siempre en los procesos del servidor. Los filtros ISAPI son diferentes de las aplicaciones porque se controlan por sucesos del servidor Web y no por peticiones de cliente. Puede asociar un filtro ISAPI a un suceso específico del servidor Web; el filtro recibirá una notificación cada vez que se produzca el suceso asociado. Por ejemplo, un filtro puede recibir una notificación cuando se produzca un suceso de lectura o escritura, y cifrará, a continuación, los datos que se van a devolver al cliente.

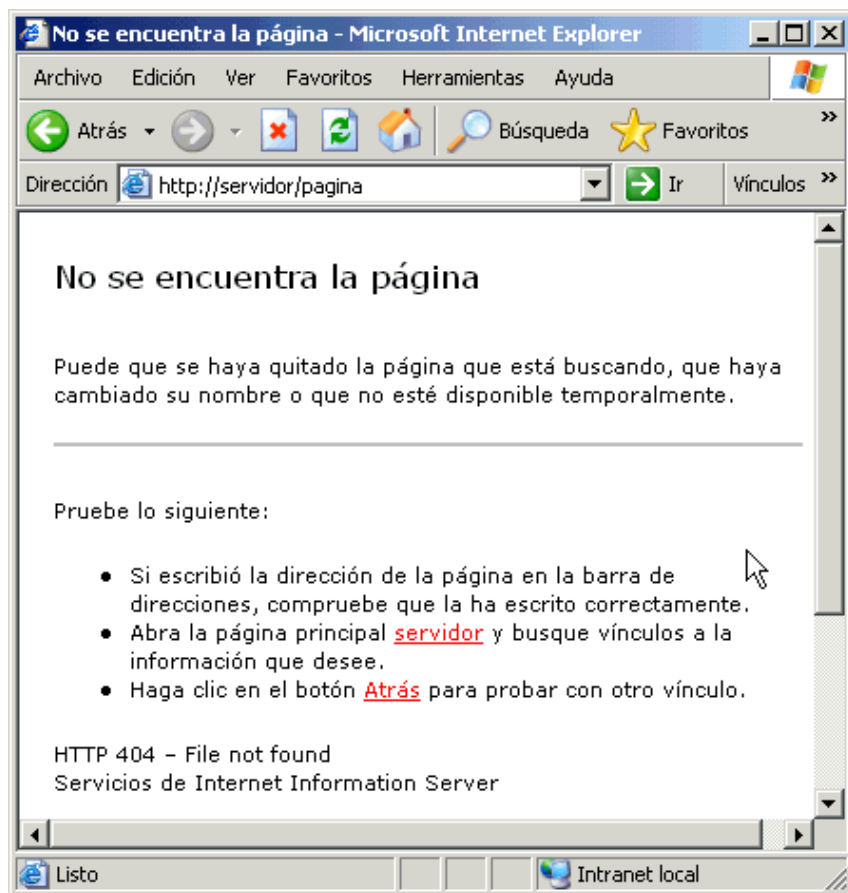
Puede instalar filtros para todos los sitios de un servidor (filtros globales) y también puede instalar filtros para sitios Web individuales. Si instala filtros globales y filtros de sitio, se combinarán las dos listas de filtros para el sitio.

Cuando se han registrado varios filtros para el mismo suceso, se les llama secuencialmente. Los filtros con mayor prioridad se ejecutan antes que los de menor prioridad. Si varios filtros tienen la misma prioridad, los filtros globales establecidos en las propiedades principales se ejecutan antes que los establecidos para el sitio. Los filtros que tienen la misma prioridad con el mismo grado de herencia se ejecutan según el orden en que se cargaron. Puede cambiar el orden de carga de los filtros en las hojas de propiedades del servidor Web o del sitio Web.

## 1.7 Mensajes de error personalizados

Cuando se produce un error en nuestra Intranet: no encuentra una página, error del servidor, excedido tiempo de espera... IIS muestra una página con el error. Estas páginas las encontramos en un directorio de nuestro

disco duro y podemos cambiarlas. Siempre es mejor mostrar un mensaje personalizado con el logo y aspecto de nuestra Intranet que uno genérico. Veamos un ejemplo del mismo mensaje de error visto por IIS de forma predeterminada y después he haber hecho alguna pequeña modificación en su aspecto:



Estos son algunos de los errores que se pueden personalizar y que son los mas comunes:

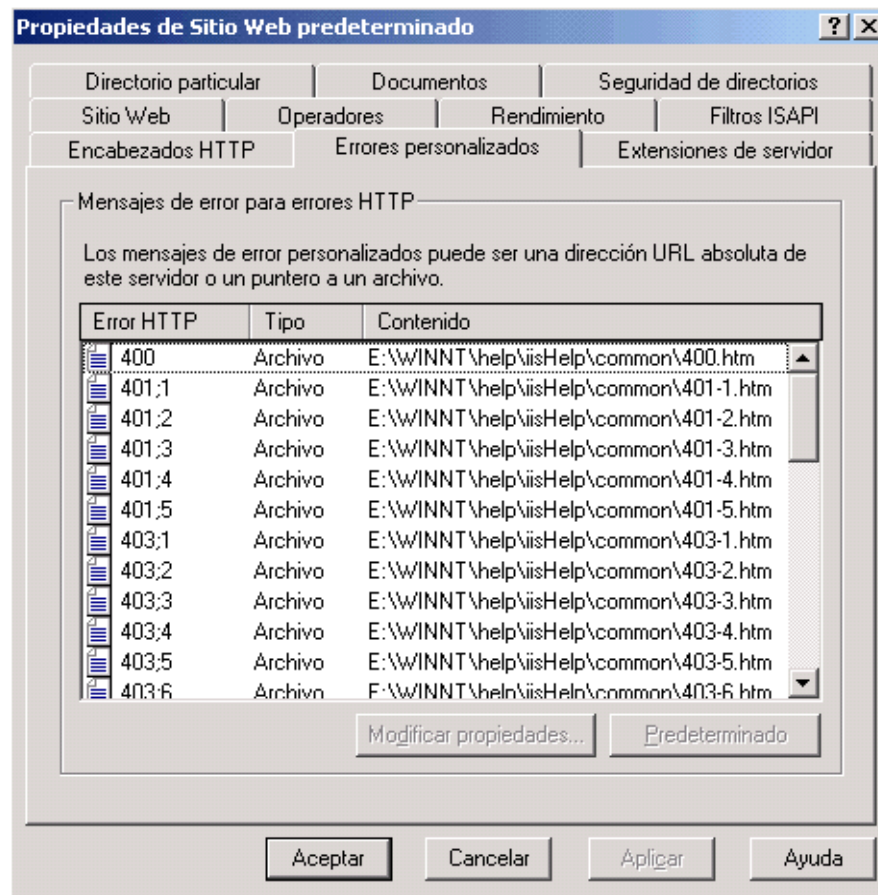
Código de error	Mensaje
400	Solicitud incorrecta
403.1	Acceso de ejecución prohibido
403.2	Acceso de lectura prohibido
403.3	Acceso de escritura prohibido
403.8	Acceso al sitio denegado
403.14	Lista de directorios denegada
404	No se encuentra
404.1	Sitio no encontrado
500	Error interno del servidor
500-100.asp	Error ASP

Los mensajes de error se muestran en una lista del complemento IIS que IIS trata como una sola propiedad.

Por ejemplo, cuando se configura un conjunto de mensajes de error personalizados para el sitio Web, todos los directorios de este servidor heredan la lista completa de mensajes personalizados. Es decir, no se combinan las dos listas de mensajes de error personalizados (para el servidor y para el directorio).

El error 404.1 sólo se produce en equipos con direcciones IP múltiples. Si se recibe una solicitud de cliente en una combinación de dirección y puerto IP determinada y la dirección IP no está configurada para la recepción en ese puerto específico, IIS devolverá el mensaje de error HTTP 404.1. Por ejemplo, si un equipo dispone de dos direcciones IP y solamente una de ellas está configurada para escuchar en el puerto 80, cualquier solicitud con puerto 80 que se reciba en la otra dirección IP hará que IIS devuelva el mensaje de error 404.1.

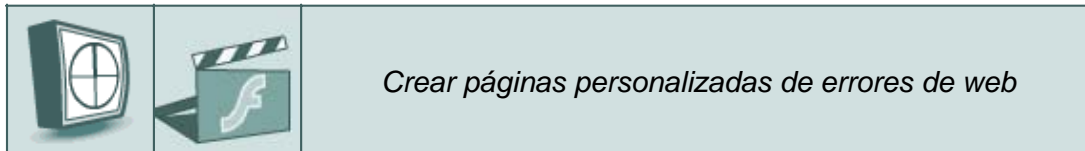
Podemos asignar mensajes de error personalizados a un archivo o a una dirección URL para esto utilizaremos la hoja de propiedades Errores personalizados del complemento IIS:



Para personalizar un mensaje de error mediante su asignación a un archivo:

1. Creamos un archivo que contenga su mensaje de error personalizado y colocamos el archivo en un directorio.
2. Seleccionamos el error HTTP que desea cambiar.
3. Hacemos clic en el botón Modificar propiedades.
4. Seleccionamos Archivo en el cuadro Tipo de mensaje.
5. Escribimos la ruta de acceso y el nombre del archivo que apunta al mensaje de error personalizado o utilizamos el botón Examinar para localizar el archivo en el disco duro del equipo.
6. Hacemos clic en Aceptar.

**Nota** si es una URL seleccionaremos "Dirección URL" en el cuadro Tipo de mensaje.

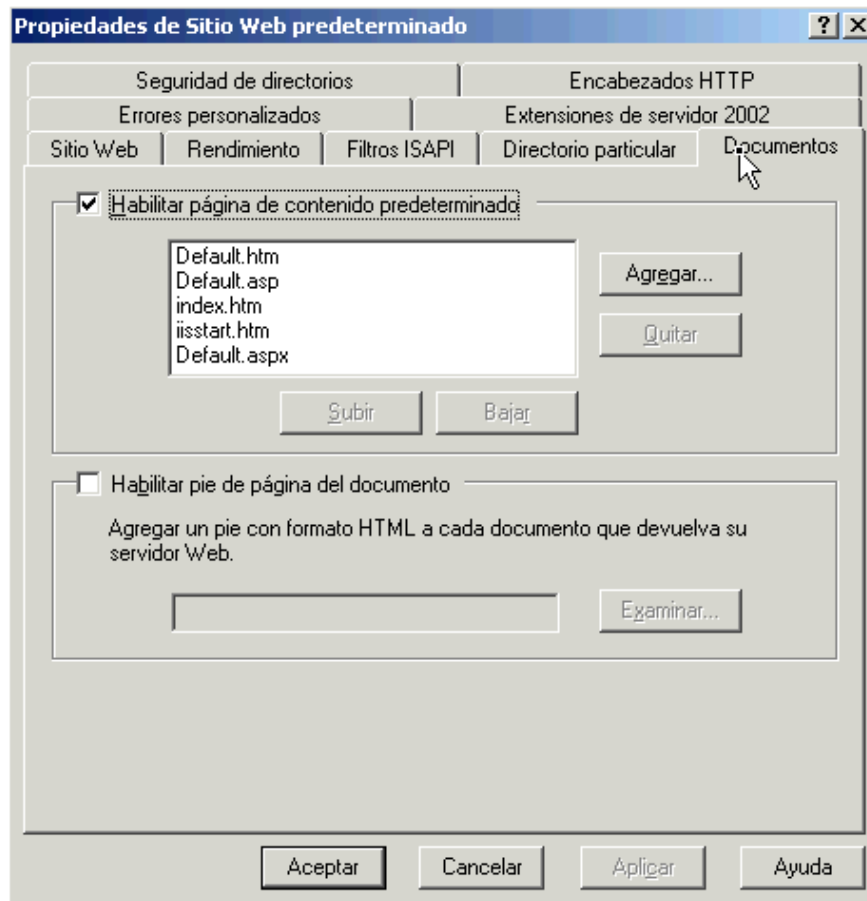


Podemos modificar los ficheros con FrontPage pero ojo, si incluimos un gráfico debe ser con la forma `http://servidor/images/logo.gif`. De lo contrario incluirá un enlace a una ruta local `c:\imagenes\logo.gif` que no funcionará bien. Por ejemplo, modificando el fichero "404b.htm" que es el que se muestra cuando no encuentra la página quedaría así:



## 1.8 Documentos

En esta hoja podremos indicar la página por defecto del sitio Web. Cuando un usuario no especifique en el navegador ningún fichero en concreto de nuestro sitio Web se le mostrará esta página por defecto. Podemos indicar diferentes documentos predeterminados dentro de la lista que vemos a continuación:

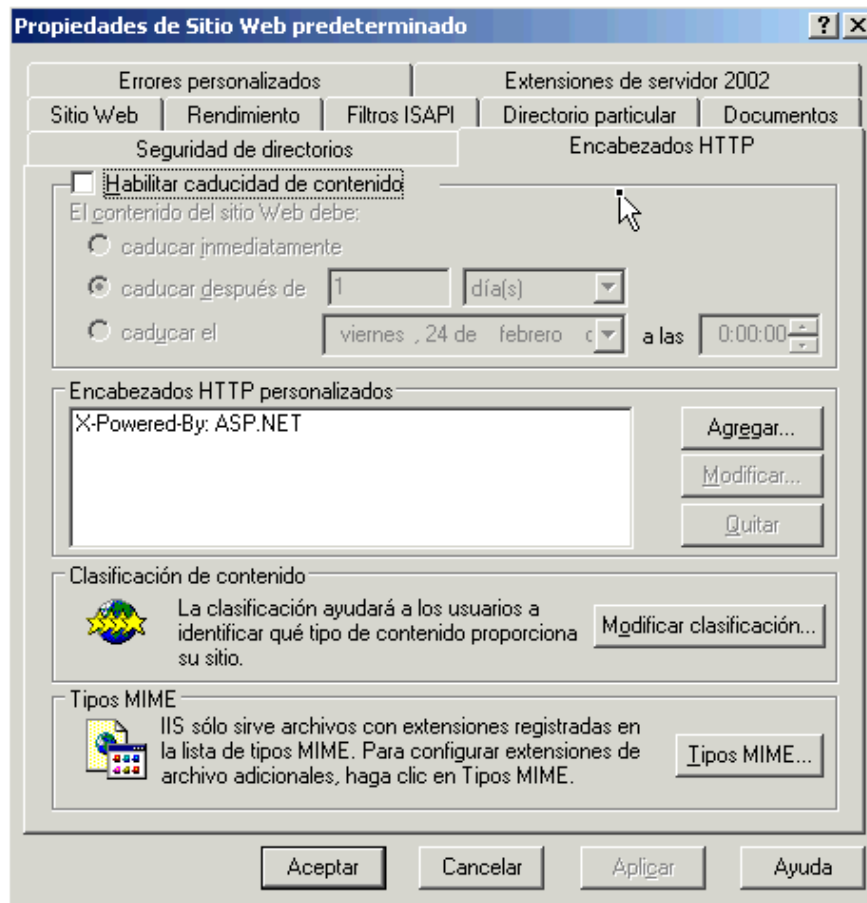


El orden en el que se indican los documentos es significativo, el servidor Web devolverá el primero que encuentre. El orden de prioridad de los documentos lo podemos modificar utilizando las flechas. El documento predeterminado podrá ser tanto una página HTML como una página ASP.

Con la opción inferior "Habilitar pie de página del documento" podremos configurar el servidor Web para que inserte de manera automática un fichero en formato HTML en la parte inferior de todos los documentos del sitio Web.

## 1.9 Encabezados HTTP

En esta hoja de propiedades vamos a poder establecer los valores de las cabeceras HTTP que se envíen a nuestros clientes (navegadores Web). Puesto que no nos van a afectar mucho para nuestra intranet las enumeraremos para que el alumno las identifique...



Si activamos la casilla de verificación **Habilitar caducidad de contenido** nos permitirá establecer el momento en el que la información de la página se considera no válida u obsoleta (ha caducado), el navegador Web comparará la fecha del sistema con la de la página que le llega y determinará si se visualiza la página almacenada en la caché o bien pedirá una actualización de la misma.

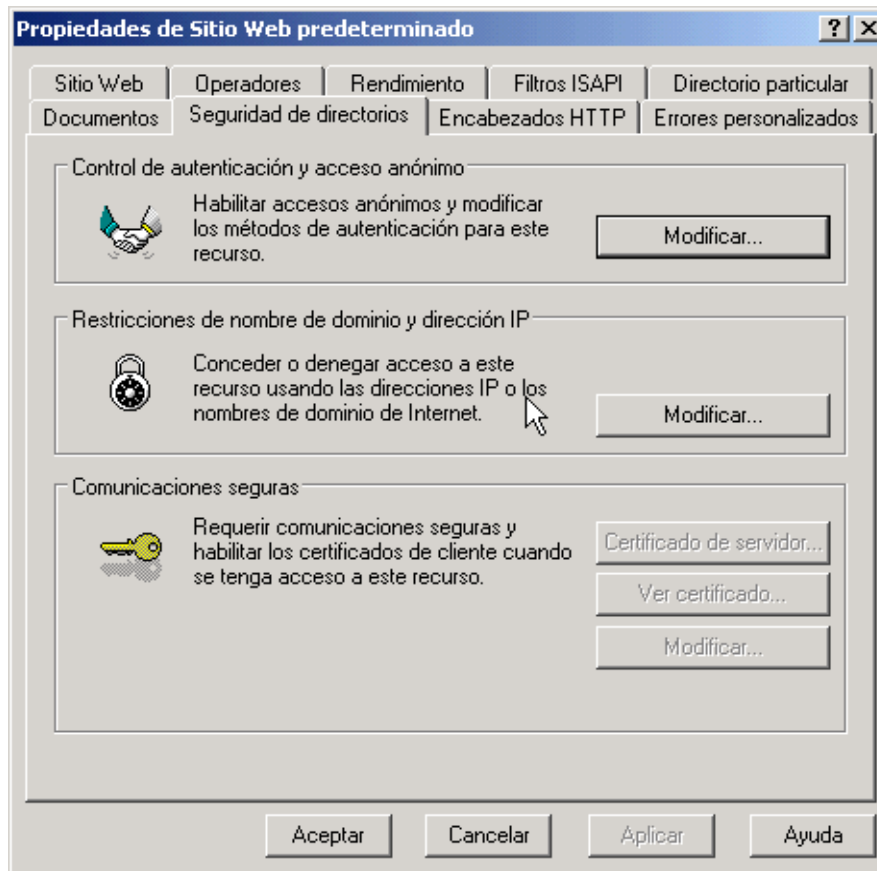
En la propiedad **Encabezados HTTP personalizados** podremos enviar un encabezado HTTP personalizado del servidor Web al navegador del equipo cliente.

Las páginas Web pueden incorporar cabeceras con información que identifique sus contenidos desde un punto de vista moral, la configuración de estas cabeceras la realizaremos a través del apartado "Restricción de contenido". De esta manera los usuarios pueden filtrar las páginas en función de ese contenido. IIS utiliza un método desarrollado por el organismo RSAC (Recreational Software Advisory Council) que clasifica los contenidos atendiendo a diversos grados de violencia, pornografía y lenguaje ofensivo. Para establecer estas restricciones de contenido se debe pulsar el botón "Modificar clasificación".

En la opción **Tipos MIME** (Multipurpose Internet Mail Extensions) podemos determinar los tipos de archivos que se enviarán al cliente Web.

## 1.10 Seguridad de directorios

Esta es una de las partes mas complejas de la administración del sitio web con IIS. No es que revista dificultad sino que son los parámetros que van a determinar cómo va a funcionar nuestra Intranet. Además concederá y negará el acceso a usuarios y otros aspectos relativos a la seguridad.



Estas son las tres secciones referentes a la seguridad y los permisos. La tercera parte la dejaremos porque escapa al objeto de nuestra intranet al entrar en el campo de los certificados y la seguridad SSL, nos centraremos en las dos primeras. De ellas la primera parte es la fundamental para establecer el comportamiento del servidor.

## Qué es autenticación

En este capítulo explicaremos con profundidad el tema de la autenticación y en el siguiente lo resumiremos y nos quedaremos ya con la configuración apropiada para nuestra intranet. Podemos configurar IIS para autenticar o determinar la identidad de la cuenta de un usuario de Windows, antes de permitir que éste establezca una conexión de red con su servidor. Sin embargo, la autenticación de los usuarios sólo tendrá lugar cuando esté desactivado el acceso anónimo o cuando los permisos NTFS requieran que los usuarios se identifiquen con el nombre y la contraseña de una cuenta de usuario válida de Windows.

Con las opciones de autenticación que ofrece IIS, podemos elegir un método de autenticación que se ajuste a los requisitos de seguridad y a las capacidades del explorador Web del usuario.

Puede especificar que los usuarios proporcionen el nombre de usuario y la contraseña de una cuenta de usuario válida de Microsoft Windows para poder tener acceso a cualquier información del servidor. Este proceso de identificación recibe el nombre de autenticación. La autenticación, como muchas de las características de IIS, se puede establecer para sitios Web, directorios o archivos. IIS proporciona los siguientes métodos de autenticación para controlar el acceso al contenido del servidor. Veamos los métodos mas importantes



### Autenticación anónima

La autenticación anónima proporciona a los usuarios acceso a las áreas públicas del sitio Web o FTP sin preguntar el nombre de usuario o la contraseña. Cuando un usuario intenta conectarse al sitio Web o FTP público, el servidor Web le asigna la cuenta de usuario de Windows llamada IUSR\_nombre\_equipo, donde nombre\_equipo es el nombre del servidor en el que se ejecuta IIS. De manera predeterminada, la cuenta IUSR\_nombre\_equipo está incluida en el grupo de usuarios Invitados de Windows. Este grupo tiene restricciones de seguridad impuestas por los permisos NTFS, que designan el nivel de acceso y el tipo de contenido que hay a disposición de los usuarios públicos.

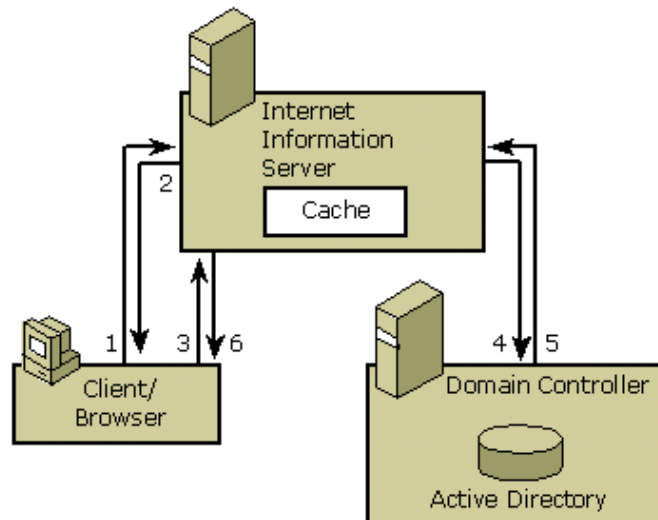
Si tiene varios sitios en el servidor o si tiene áreas de su sitio que requieren diferentes privilegios de acceso, puede crear varias cuentas anónimas, una para cada sitio Web o FTP, directorio o archivo. Al dar a estas cuentas diferentes permisos de acceso o al asignar estas cuentas a grupos de usuarios de Windows diferentes, puede otorgar a los usuarios acceso anónimo a distintas áreas de contenido público Web y FTP.

1. La cuenta IUSR\_nombre\_equipo se agrega al grupo Invitados del equipo IIS durante la instalación.
2. Cuando se recibe una solicitud, IIS suplanta la cuenta IUSR\_nombre\_equipo antes de ejecutar cualquier código o de tener acceso a cualquier archivo. IIS puede suplantar la cuenta IUSR\_nombre\_equipo porque conoce el nombre de usuario y la contraseña de esta cuenta.
3. Antes de devolver una página al cliente, IIS comprueba los permisos de archivos y directorios NTFS para determinar si la cuenta IUSR\_nombre\_equipo tiene acceso al archivo.
4. Si está permitido el acceso, se completará la autenticación y los recursos estarán disponibles para el usuario.
5. Si no está permitido el acceso, IIS intentará utilizar otro método de autenticación. Si no hay ninguno seleccionado, IIS devolverá al explorador un mensaje de error "HTTP 403 Acceso denegado".

La cuenta anónima debe tener el derecho de usuario para el inicio de sesión local. Si la cuenta no tiene el permiso Inicio de sesión local, IIS no podrá atender ninguna solicitud anónima. La instalación de IIS concede específicamente el permiso Inicio de sesión local a la cuenta IUSR\_nombre\_equipo. De forma predeterminada, las cuentas IUSR\_nombre\_equipo de los controladores de dominio no se asignan a las cuentas de invitados. Para permitir inicios de sesión anónimos, debemos cambiar las cuentas IUSR\_nombre\_equipo a Inicio de sesión local

### Autenticación básica

La autenticación básica es un método estándar muy extendido para recopilar información de nombre de usuario y contraseña. El proceso para autenticar se realiza de la siguiente forma:



1. El explorador Web Internet Explorer muestra un cuadro de diálogo en el que el usuario debe escribir su nombre de usuario y contraseña de Windows previamente asignados, que también se conocen como **credenciales**.
2. El explorador Web intentará establecer la conexión con un servidor, mediante las credenciales del usuario. La contraseña de texto simple se codifica en Base64 antes de enviarla a través de la red.
  - ◆ **Importante** La codificación en Base64 no es un cifrado. Si una contraseña codificada en Base64 es interceptada en la red por un husmeador de redes, la contraseña puede ser decodificada y utilizada por personas no autorizadas.
3. Si las credenciales de un usuario son rechazadas, Internet Explorer muestra una ventana de diálogo de autenticación para que el usuario vuelva a escribir sus credenciales. En Internet Explorer se permite al usuario tres intentos de conexión antes de informarle de que no se puede establecer la conexión.
4. Cuando el servidor Web compruebe que el nombre de usuario y la contraseña corresponden a una cuenta de usuario válida de Microsoft Windows, se establecerá una conexión.

La autenticación básica tiene la ventaja de que forma parte de la especificación HTTP y es compatible con la mayoría de los exploradores. La desventaja de los exploradores Web que utilizan la autenticación básica es que transmiten las contraseñas sin cifrar. Mediante la supervisión de las comunicaciones en la red, alguien podría fácilmente interceptar y decodificar estas contraseñas mediante herramientas de dominio público. Por tanto, **la Autenticación básica no es recomendable** a menos que tenga la seguridad de que la conexión entre el usuario y el servidor Web sea segura, como una línea dedicada o una conexión Capa de sockets seguros (SSL)

#### Autenticación de texto implícita

La autenticación de texto implícita ofrece la misma funcionalidad que la autenticación básica. Sin embargo, la autenticación de texto implícita supone una mejora en la seguridad debido a la forma en que se envían las credenciales del usuario a través de la red. La autenticación de texto implícita transmite las credenciales a través de la red como un hash MD5, también conocido como mensaje implícito, en el que el nombre de usuario y la contraseña originales no pueden descifrarse del hash. La autenticación de texto implícita está disponible para los directorios del Sistema distribuido de creación y control de versiones Web (WebDAV).

Antes de habilitar la autenticación de texto implícita en el servidor IIS, asegúrese de que se cumplen los requisitos mínimos siguientes. Sólo los administradores de dominio pueden comprobar que se cumplen los

## IIS Avanzado. FrontPage. ASP.NET 3.5

requisitos del controlador de dominio (DC). Si tiene dudas, consulte al administrador del dominio si el controlador del dominio cumple con los requisitos siguientes:

Todos los clientes que tienen acceso a un recurso protegido con autenticación de texto implícita van a utilizar Internet Explorer 5.0 o posterior.

- El usuario y el servidor IIS deben ser miembros o tener la confianza del mismo dominio.
- Los usuarios deben tener una cuenta válida de usuario de Windows almacenada en Active Directory en el controlador de dominio.
- El controlador del dominio debe ser un equipo con Windows 2000 o posterior.
- El servidor IIS debe ser un equipo con Windows 2000 o posterior.

### **Autenticación de Windows integrada**

La autenticación de Windows integrada (anteriormente llamada NTLM, también denominada autenticación de desafío y respuesta de Windows NT) es un método seguro de autenticación, ya que el nombre de usuario y la contraseña se procesan con el método de hash antes de enviarlos a través de la red. Al habilitar la autenticación de Windows integrada, el explorador del usuario demuestra que conoce la contraseña mediante un intercambio criptográfico con el servidor Web, en el que interviene el método de hash.

La autenticación de Windows integrada utiliza los métodos de autenticación Kerberos v5 y NTLM. Si los servicios Active Directory están instalados en un controlador de dominio con Windows 2000 o posterior y el explorador del usuario es compatible con el protocolo de autenticación Kerberos v5, se utilizará la autenticación Kerberos v5; de lo contrario, se utilizará la autenticación NTLM.

**Nota** El protocolo de autenticación Kerberos v5 es una característica de la arquitectura de Servicios distribuidos de Windows 2000. Para que la autenticación Kerberos v5 se realice correctamente, tanto el cliente como el servidor deben tener una conexión de confianza a un Centro de distribución de claves (KDC) y ser compatibles con los Servicios de directorio.

Traduciendo esta definición digamos que el navegador no se valida cada vez, sino que envía las credenciales del dominio/usuario y establece una comunicación donde no vuelve a enviar la contraseña. Es el mejor sistema para una Intranet pero debemos tener un sistema de dominios instalado en nuestra red.

### **Qué autenticación elegir**

Revisamos ya con la consola administrativa los niveles de seguridad que son:

**Métodos de autenticación**

☒ **Habilitar el acceso anónimo**

Usar la siguiente cuenta de usuario de Windows para acceso anónimo:

Nombre de usuario:

Contraseña:

---

**Acceso autenticado**

Para los siguientes métodos de autenticación, se requiere nombre de usuario y contraseña cuando:

- el acceso anónimo está deshabilitado, o
- el acceso está restringido mediante listas de control de acceso NTFS

☒ **Autenticación de Windows integrada:**

☐ Autenticación de texto implícita para servidores de dominio Windows

☐ Autenticación básica (la contraseña se envía como texto no cifrado)

☐ Autenticación de .NET Passport

Dominio predeterminado:

Territorio:

Y corresponden a:

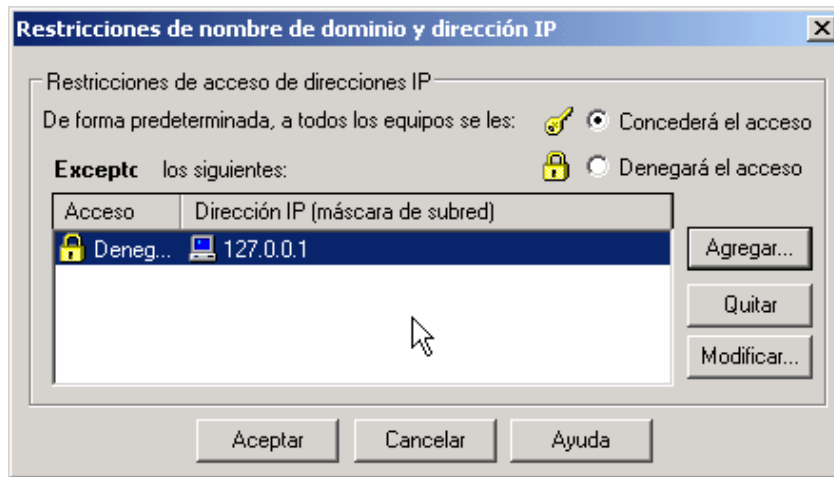
- Autenticación anónima: permite que cualquier usuario tenga acceso sin que se le pida su nombre de usuario y contraseña.
- Autenticación básica: solicita al usuario su nombre de usuario y contraseña, que se envían sin cifrar a través de la red.
- Autenticación de texto implícita: funciona de manera similar a la autenticación básica, pero difiere de ella en que las contraseñas se envían como un valor de hash. La autenticación de texto implícita sólo está disponible para los dominios con un controlador de dominio de Windows 2000.
- Autenticación de texto implícita avanzada La autenticación de texto implícita avanzada es idéntica a la autenticación de texto implícita, excepto en que la avanzada almacena las credenciales de cliente como un hash MD5 de Active Directory en el controlador de dominio de Windows XP para mejorar la seguridad.
- La autenticación de Windows integrada utiliza la tecnología de hash para identificar al usuario sin enviar realmente la contraseña a través de la red.

Si utilizamos la primera opción el acceso será anónimo y no tendremos ningún tipo de identificación a lo largo de la sesión del usuario. Es una seguridad propia para un web de Internet, ya que nadie debe, en principio, identificarse. La segunda opción la rechazamos inmediatamente por no tener ningún tipo de seguridad a la hora de transmitir la contraseña.

Y de las demás nos quedamos con la "Seguridad Integrada" Esta envía los credenciales del usuario en cada petición. Esto funcionalmente hace que desde ASP podamos saber siempre que usuario ha solicitado la página. Como veremos en el capítulo 10 es un aspecto fundamental a la hora de manejar una Intranet.

## Restricciones de nombres de dominio y direcciones IP

Otro de los métodos para controlar el acceso es a través de las direcciones IP. Si observamos la siguiente pantalla:

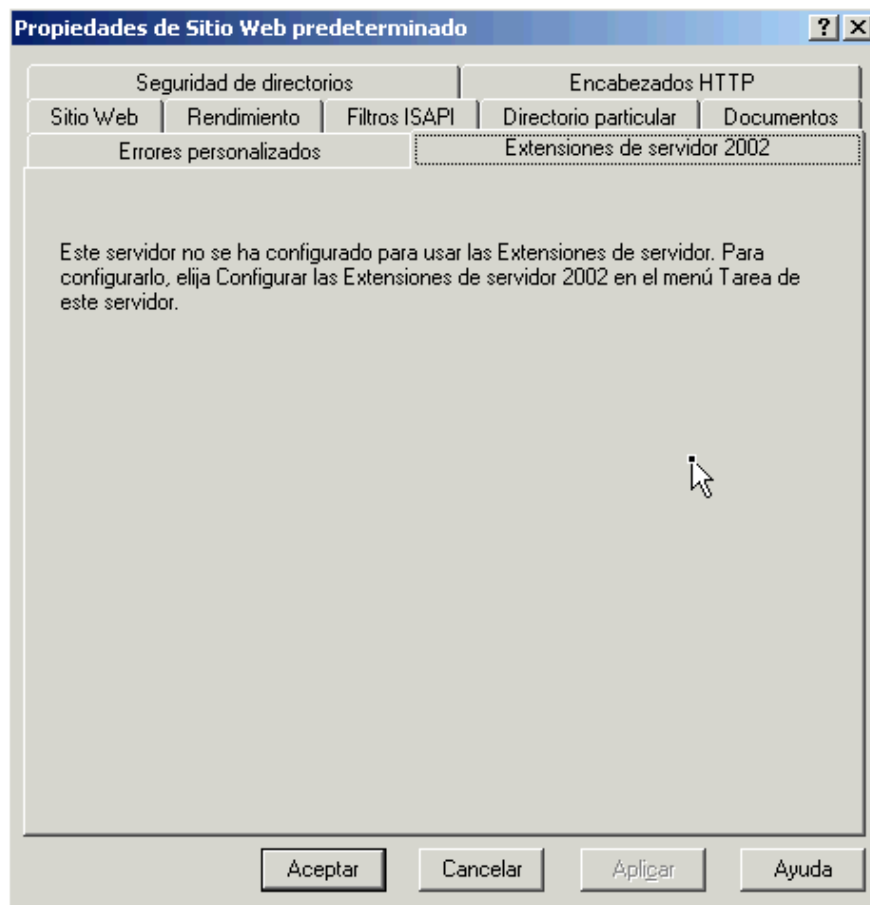


Observamos que podemos indicar una dirección IP que será la o las excepciones de lo que indiquemos. Es decir, en el ejemplo por defecto se les deniega a todos el acceso excepto para la dirección 127.0.0.1, que es una dirección especial que indica al propio equipo.

Esta restricción la tiene el web de administración de nuestro IIS. Si recuerda el alumno no nos funcionaban las páginas web del sitio de administración mas que en el propio servidor. Esta es la razón. Si queremos conceder acceso a alguna dirección IP en particular, la podemos realizar desde aquí.

## 1.11 Extensiones de servidor

Uno de los elementos mas importantes en nuestro servidor IIS son las "Extensiones de servidor". Esta es una aplicación incluida con IIS (hasta la versión 4.0 era externa) que permite conectar programas de edición, como FrontPage y SharePoint Designer (Office 2007) con el IIS. Por ejemplo desde FrontPage podremos crear y administrar distintos sitios web trabajando con un IIS. Es decir le podemos decir "Crear un sitio Web" en el servidor "intranet". Frontpage buscará un servidor de páginas web (IIS) en esa ubicación y luego que esté el módulo de "extensiones de servidor". Si está correctamente podrá recuperar los datos de este servidor: sitios web, permisos, ... y otros detalles que harán que no necesitemos manejar mas veces la consola administrativa de IIS. Veamos la siguiente pantalla:

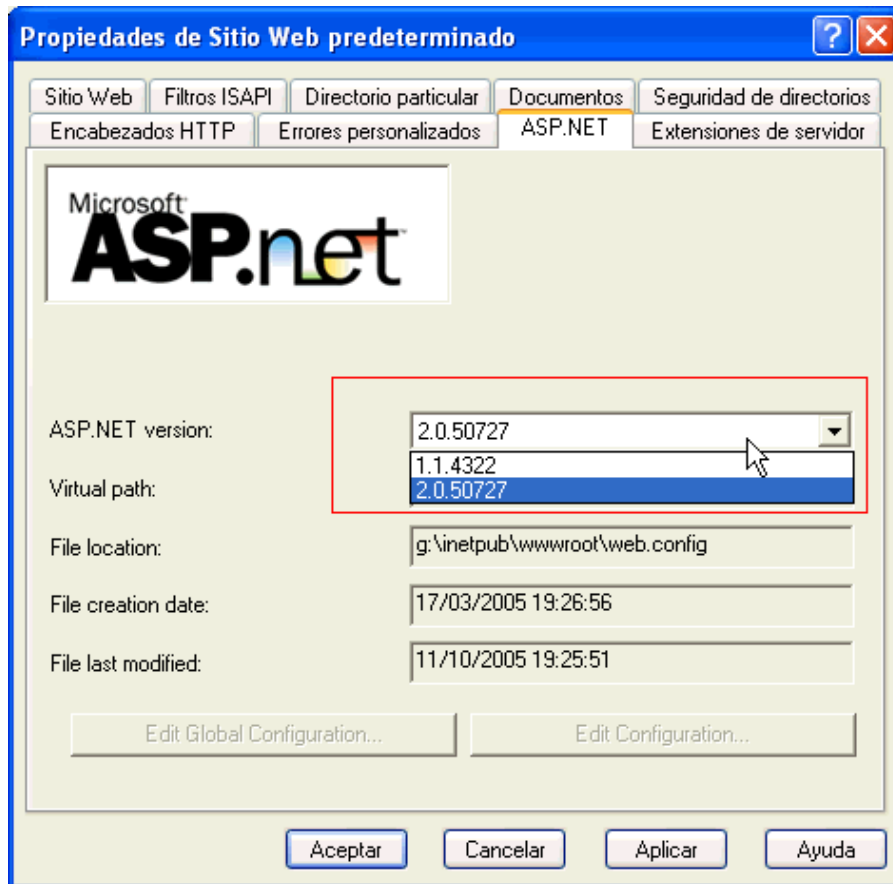


## 1.12 Versión de ASP.NET

Atento a este punto ya que es vital para que nos funcione bien. Sabemos que hay varias versiones de ASP.NET: la 1.1, 2.0 y la nueva 3.5. Resulta que hay comandos nuevos que funcionan en una versión y en la otra no. Obviamente si la versión 3.5.0 es mas moderna ésta debe tener mas comandos.

Ahora bien, las dos versiones tienen las mismas páginas "ejemplo.aspx". ¿Cómo sabe nuestro IIS qué motor (1.1, 2.0 ó 3.5) debe utilizar? Vale, pues tenemos una opción con la que tenemos que tener mucho cuidado porque al activarla cambiaremos el comportamiento de las páginas web ya que van a ser ejecutadas ahora por el "framework" versión 2.0

Bien, el cambio es muy sencillo. Cuando están los motores de ASP.NET instalados tendremos una nueva solapa:



Donde nos indica la versión de ASP.NET que ejecuta este servidor. Debemos asegurarnos de tener marcada la 2.0. Date cuenta que si ponemos comandos de la última versión (2.0) y esto no está habilitado, IIS intentará ejecutar la página con la versión "1.1" que nos devolverá seguramente un error al no encontrar ese comando.

## 2. IIS y Frontpage

Ya hemos terminado de ver todas las opciones de IIS y pasaremos a la acción con FrontPage. Como hemos comentado este programa es el idóneo para trabajar con nuestra web ya que IIS incluye un módulo especial que conecta IIS con FrontPage: las "extensiones de servidor"

Toda la información de este punto está centrada en el programa FrontPage porque está mas extendido que el nuevo SharePoint Designer de Office 2007 pero la idea funciona igual en los dos programas y es la conexión de IIS con un editor de sitios web como son los dos que acabo de mencionar. Estas extensiones van a permitir entre otras cosas crear webs directamente desde Frontpage en nuestro IIS o incluir componentes web avanzados dentro de las páginas. Veamos las opciones mas importantes:

**Nota** Hemos hablado que FrontPage era el editor de páginas web hasta Office 2003 ya que luego ha sido sustituido. Oficialmente se ha sustituido por Microsoft Web Expression en lugar del que estoy comentando del SharePoint Designer. Aquí tienes información de los dos:

<http://office.microsoft.com/es-es/frontpage/FX100743233082.aspx>

Como ves los dos productos son de la línea Office y trata al SharePoint Designer como de mas nivel que el Web Expression. Yo me quedo con el primero porque le he visto mas utilidades que al segundo pero eso ya es a gustos del consumidor. Lo importante es que no nos va a afectar a la conexión con un IIS que es lo que estamos tratando ahora con FrontPage

## 2.1 Abrir sitios web

Para abrir un web simplemente seleccionaremos la opción de abrir web. Si escribimos `http://servidor` abriremos el web raíz del servidor IIS. Para abrir otros webs haremos la misma operación pero escribiendo la ruta correcta: `http://servidor/miweb`

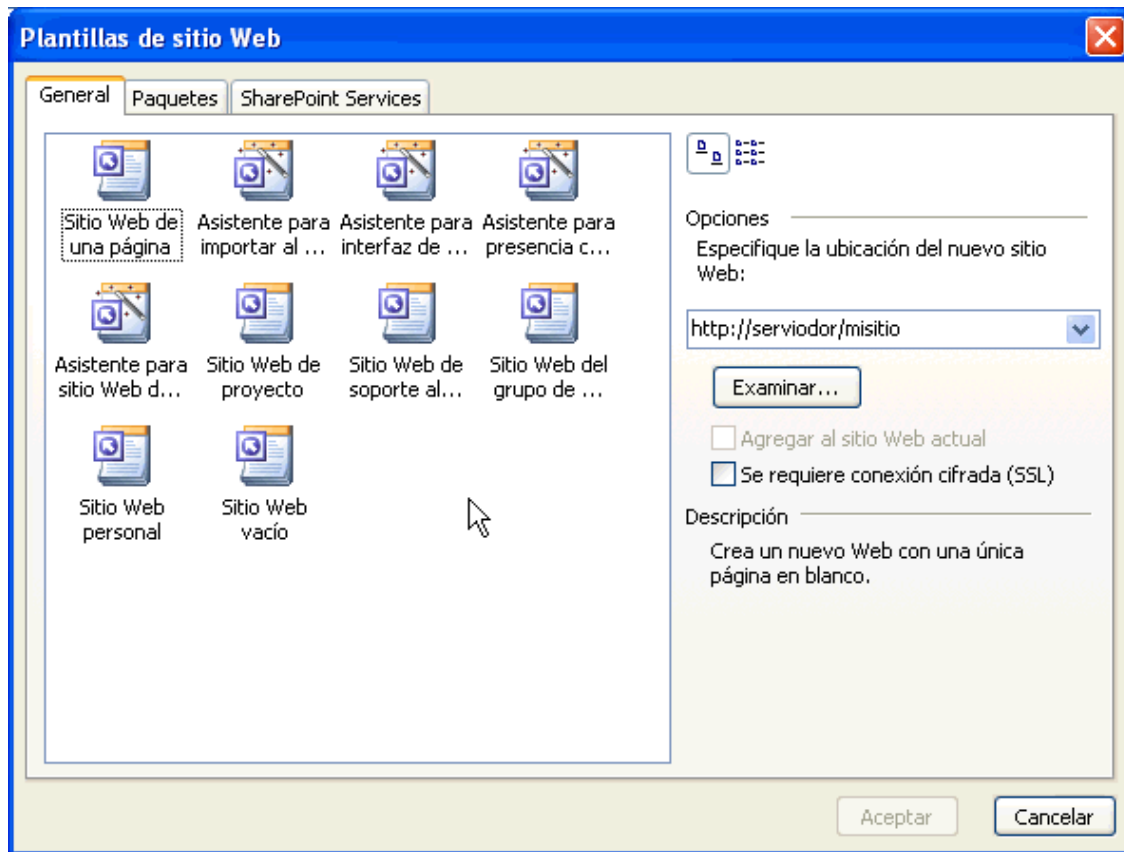
## 2.2 Crear sitios web

1. En Microsoft FrontPage, en el menú Archivo, seleccionamos Nuevo y, a continuación Web.
2. En el panel de tareas Nueva página o Web, en Nuevo a partir de una plantilla, haga clic en Plantillas de sitio Web.
3. Hacemos clic en Web de una página y, a continuación, en el cuadro Especificar la ubicación del nuevo sitio Web, escriba la dirección URL del sitio Web. La dirección URL la escribiremos con el formato `http://servidor/miweb`

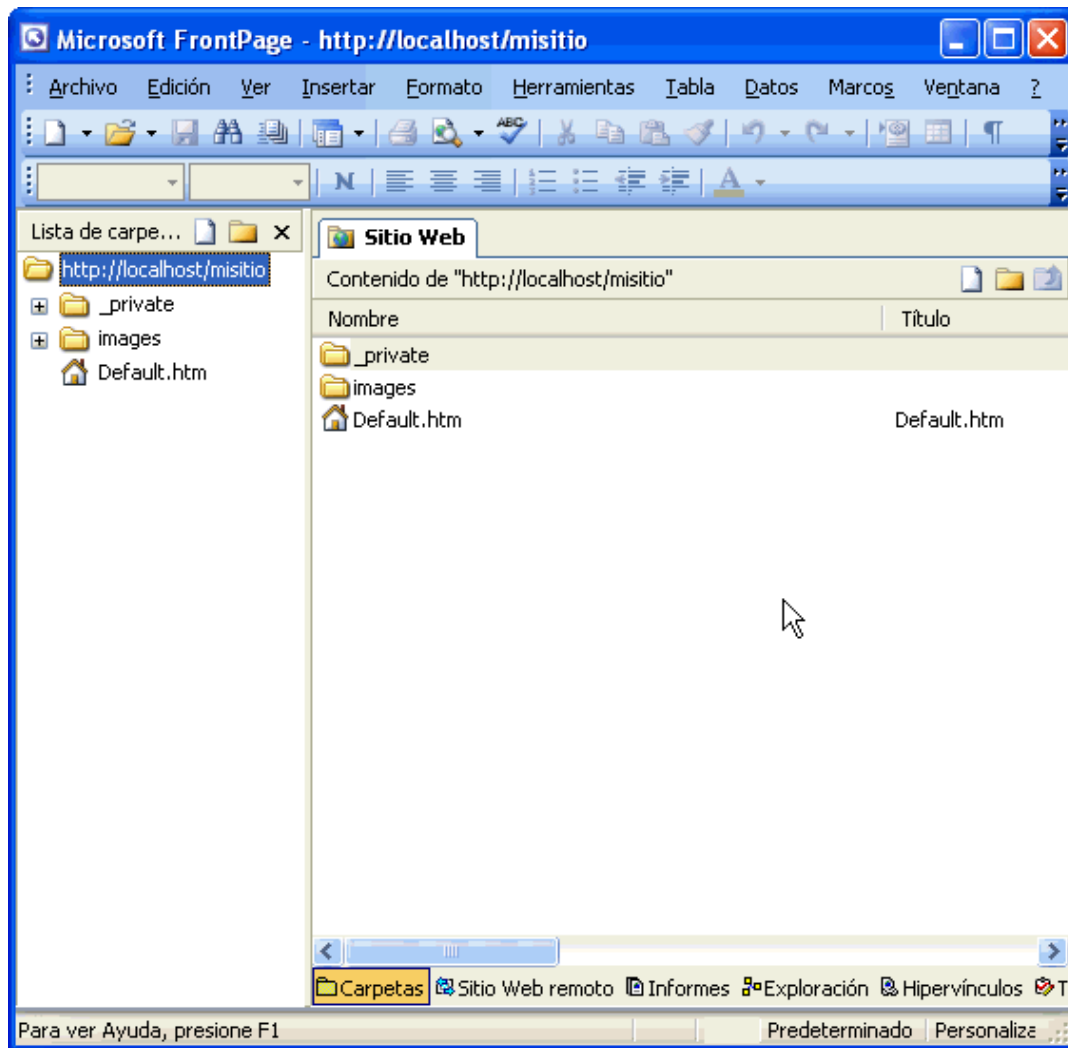


Si es FrontPage 2003 nos aparecerá a la derecha una lista de opciones para realizar, seleccionaremos "crear Web de una página" y nos aparecerá una pantalla:

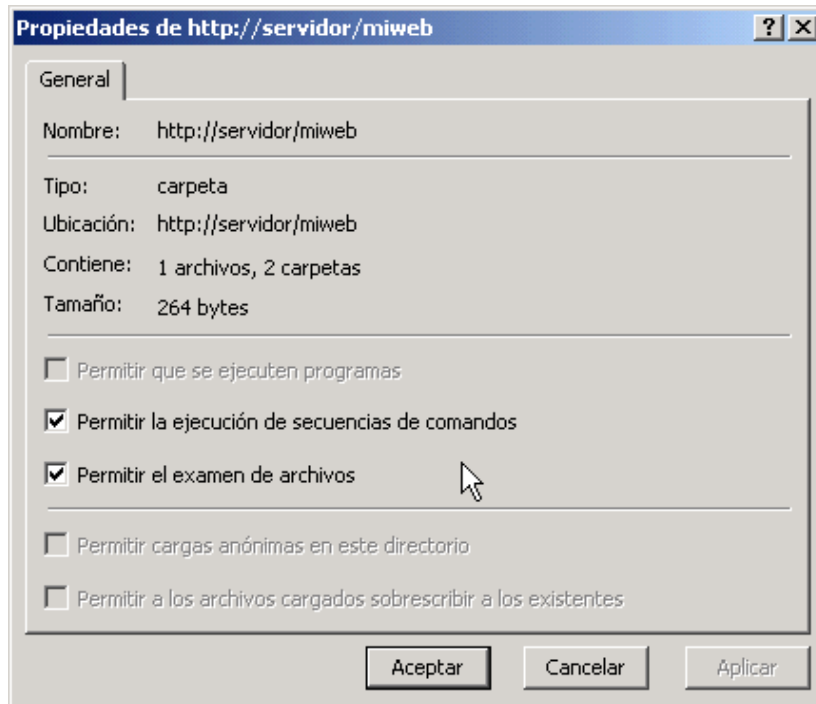




Hacemos clic en Aceptar y FrontPage creará el sitio Web en el servidor:



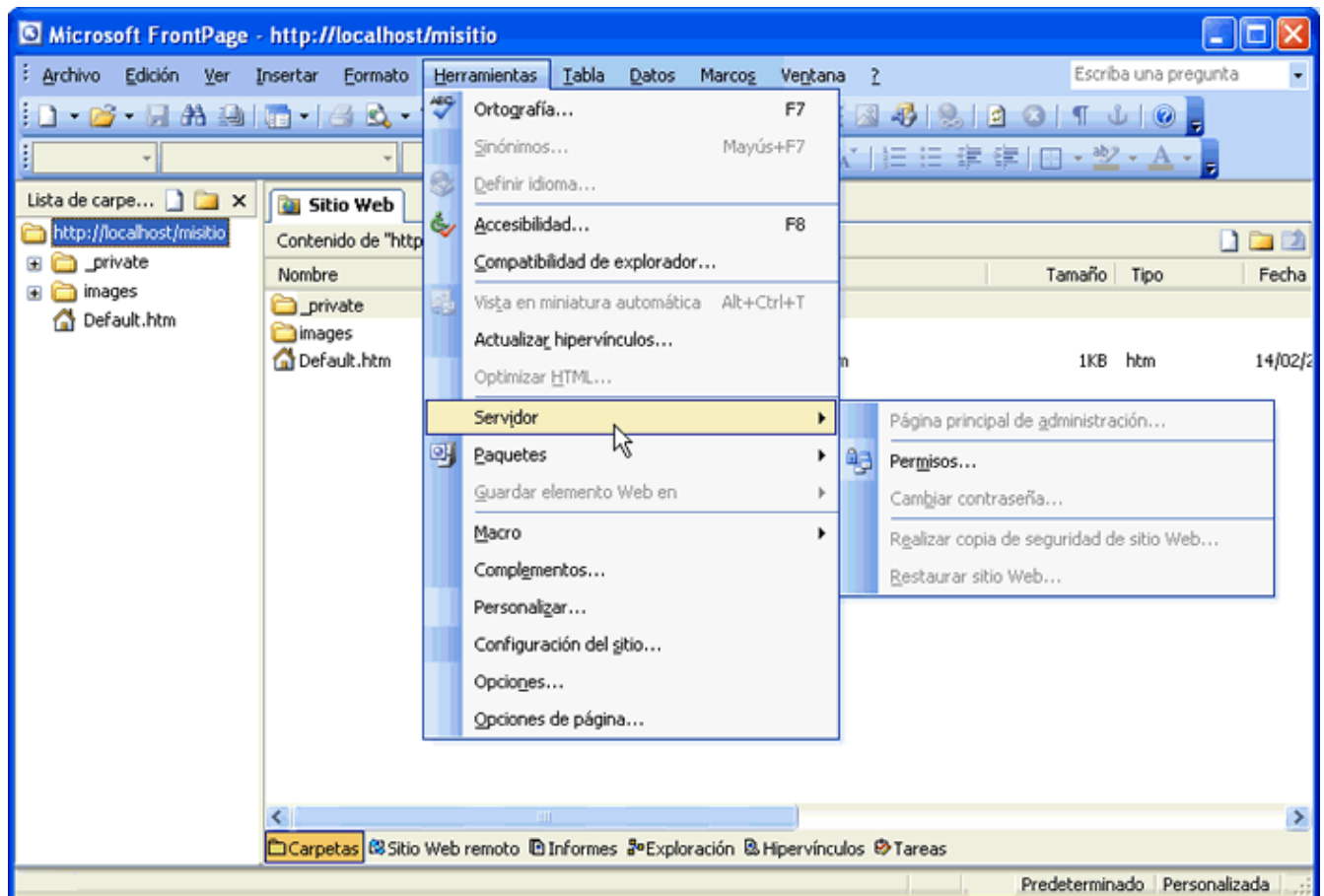
Para comprobar que los parámetros de " ejecución de comandos " están activos pulsamos con el botón derecho en la raíz del Web:



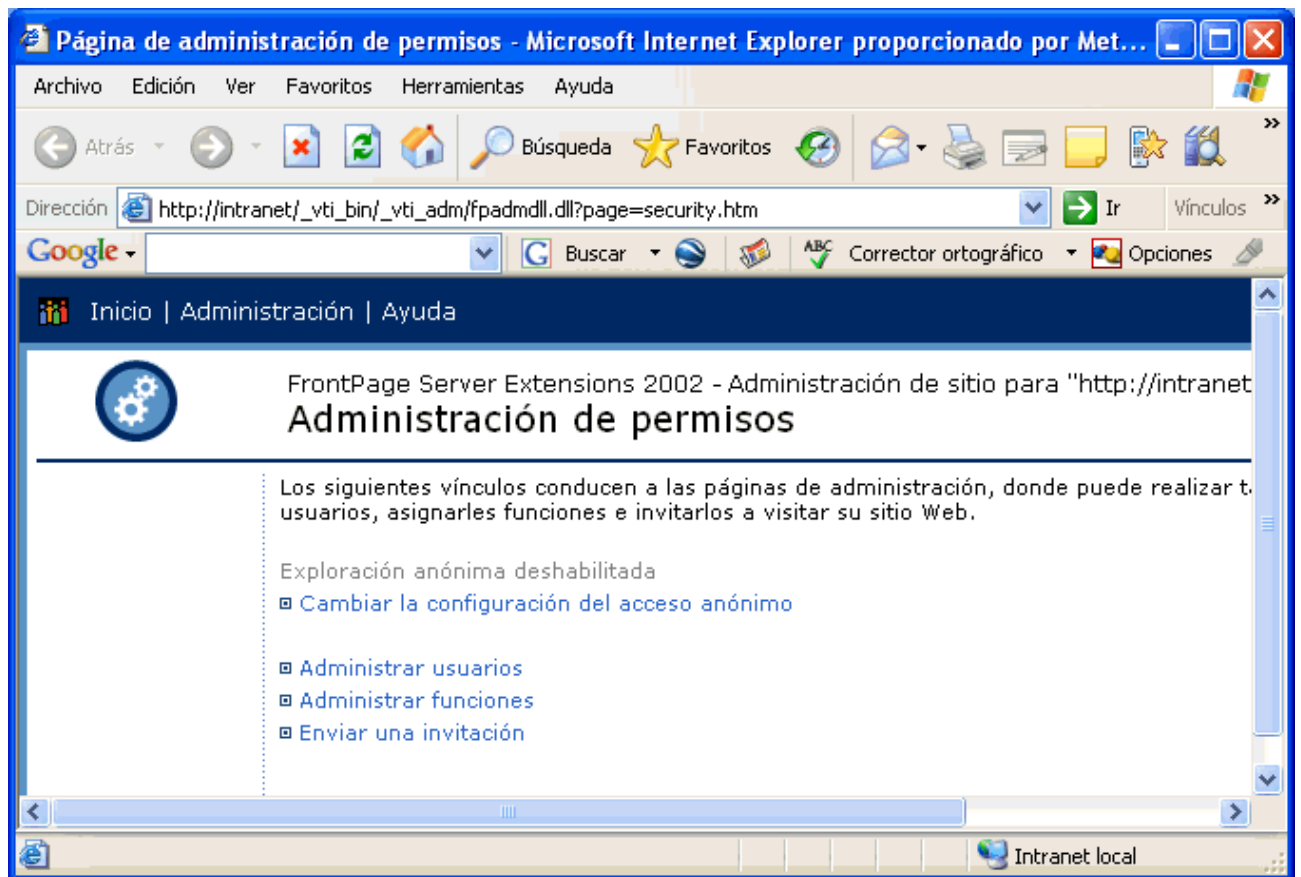
En este caso también está activa la opción de “ Permitir el examen de directorios ” opción no recomendable pero que en desarrollo es útil mientras se construyen las páginas con los hipervínculos

## 2.3 Mantenimiento de permisos en el servidor:

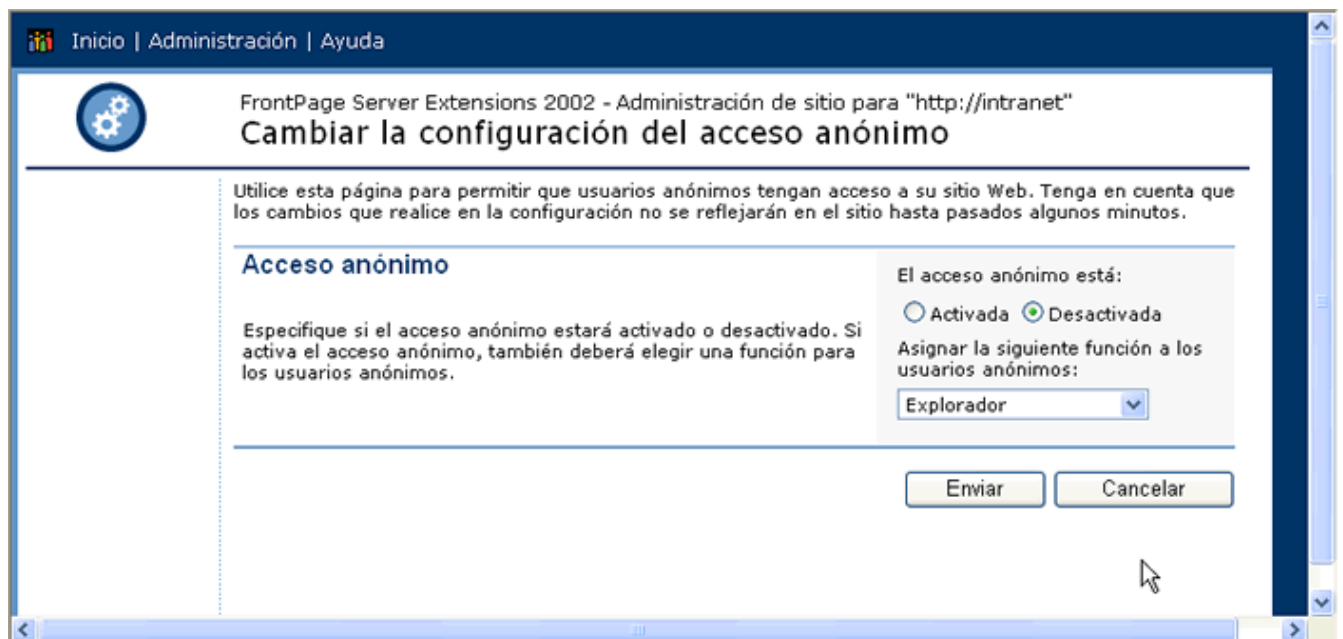
Los permisos para que los usuarios puedan administrar y modificar se encuentran en la opción “ Permisos ” de la sección “ Servidor ” del menú “ Herramientas



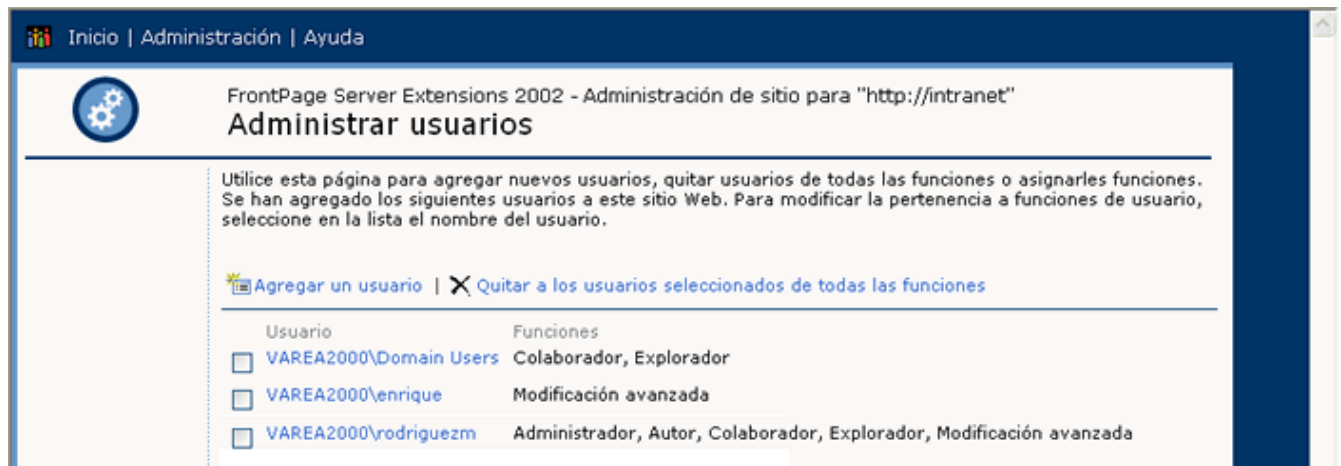
En estas páginas de permisos podremos acceder ya a todas las opciones para definir los usuarios que pueden modificar el Web, con varios tipos de permisos. Si seleccionamos esa opción veremos el sitio Web administrativo para estos permisos. Antes esto se hacía con pantallas normales, ahora se hace todo con el navegador. Veamos estas sencillas pantallas. De momento podemos definir si nuestro Web va a permitir la exploración anónima:



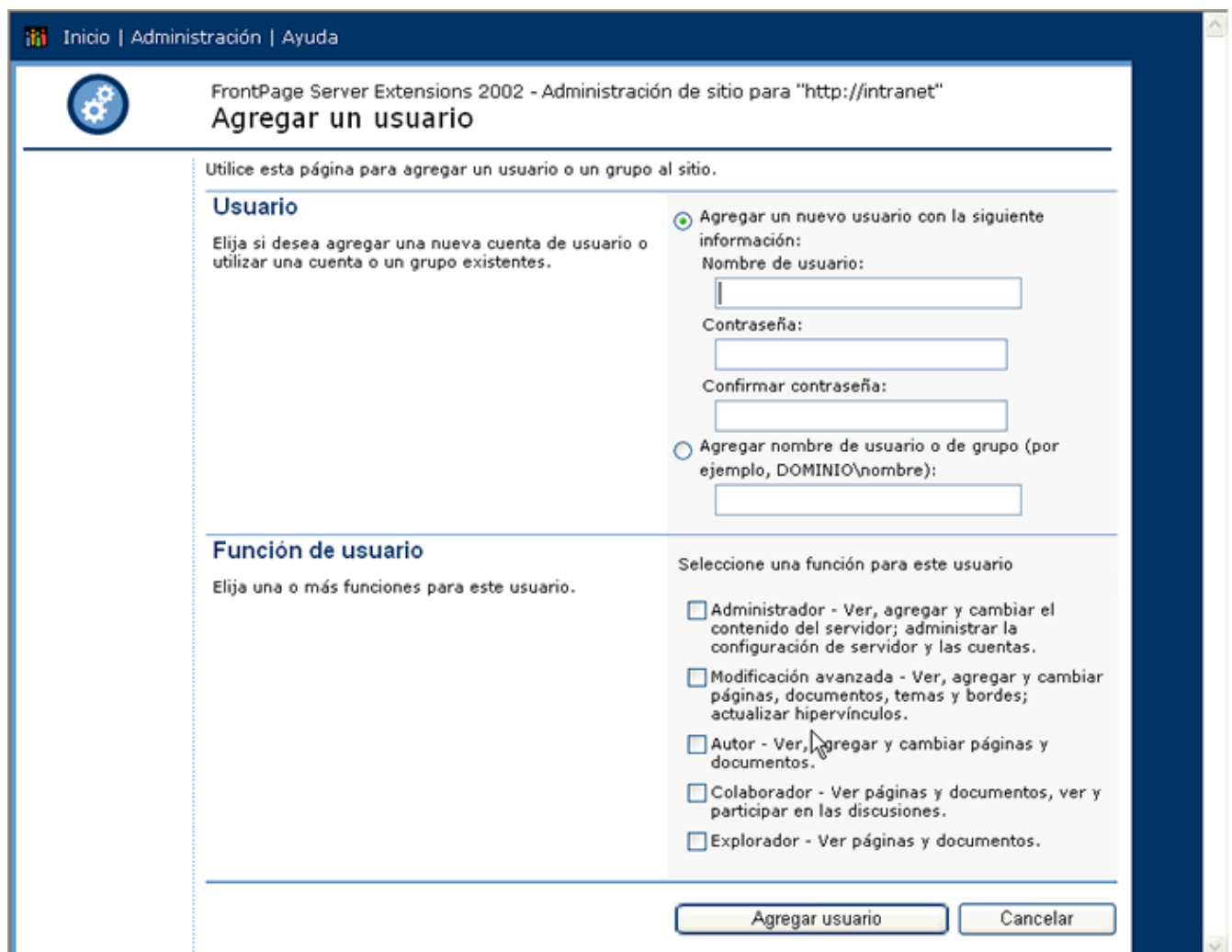
Para sitios Web internos lo mejor es desactivar esta opción pero para sitios Web externos como los usuarios no van a ser los del dominio de usuarios de Windows lo normal es que sea anónimo. Luego cuando veamos la identificación de usuarios de ASP.NET avanzaremos sobre esta identificación anónima.



Volviendo a la primera pantalla vemos la administración de los usuarios:



Como ves tenemos una lista de usuarios que tienen un perfil o "funciones" que puede realizar. Si queremos añadir un usuario pulsamos y:



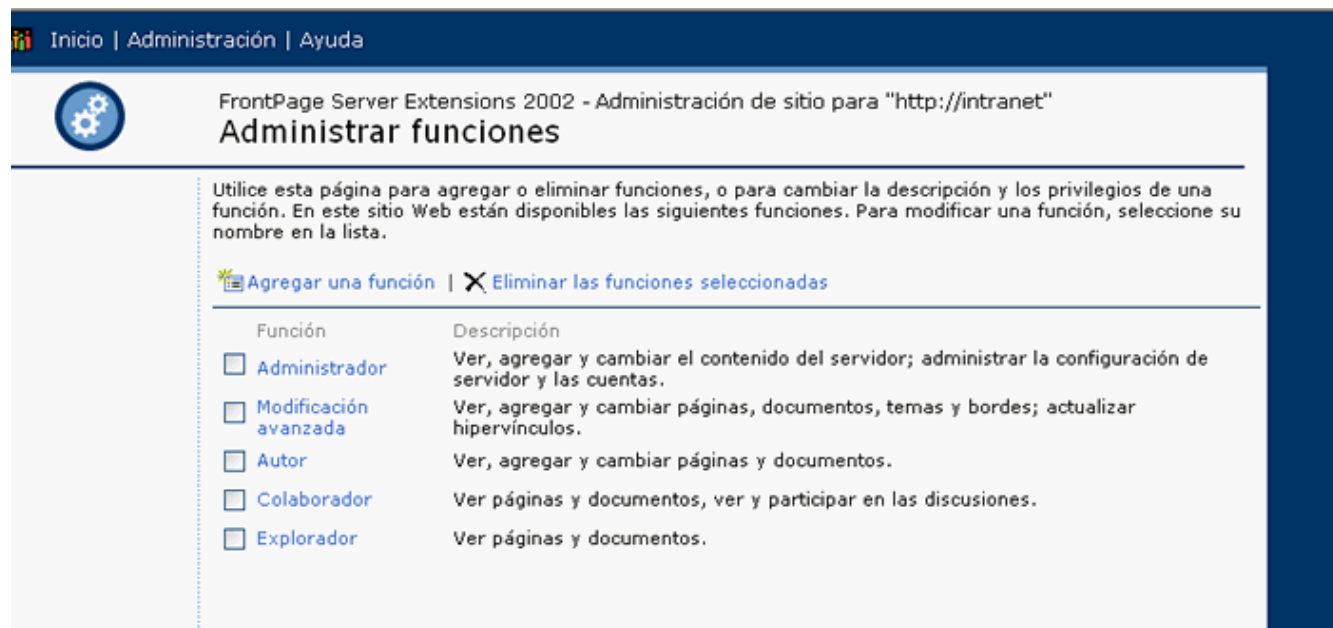
Por un lado si va a ser un usuario independiente, es decir, que no pertenece al dominio de usuarios de la red ponemos todos sus credenciales: usuario y contraseña. Si es un usuario del dominio no hace falta mas que indicar el nombre de usuario de la forma DOMINIO\usuario. La contraseña no hace falta porque coge la

## IIS Avanzado. FrontPage. ASP.NET 3.5

almacenada en el Directorio Activo de nuestra red. Luego debajo indicaremos las funciones que va a realizar este usuarios: si va explorar o va a editar páginas, ...

Ojo, fíjate que en la pantalla anterior aparece en la primera línea "Dominio\Domain Users" con permiso de explorar, si no lo tuvieran no podrán acceder a ver las páginas Web ya que antes quitamos el acceso anónimo. Así que si lo quitamos debemos dar permiso a los usuarios del dominio que es el grupo anterior para que pueda "Explorar" el Web.

Otras funciones que nos permite este sitio administrativo es la creación de perfiles como los que hemos asignado antes:



Función	Descripción
<input type="checkbox"/> Administrador	Ver, agregar y cambiar el contenido del servidor; administrar la configuración de servidor y las cuentas.
<input type="checkbox"/> Modificación avanzada	Ver, agregar y cambiar páginas, documentos, temas y bordes; actualizar hipervínculos.
<input type="checkbox"/> Autor	Ver, agregar y cambiar páginas y documentos.
<input type="checkbox"/> Colaborador	Ver páginas y documentos, ver y participar en las discusiones.
<input type="checkbox"/> Explorador	Ver páginas y documentos.

Podemos añadir, modificar, ... de esta forma:

Inicio | Administración | Ayuda

## FrontPage Server Extensions 2002 - Administración de sitio para "http://intranet"

### Agregar una función

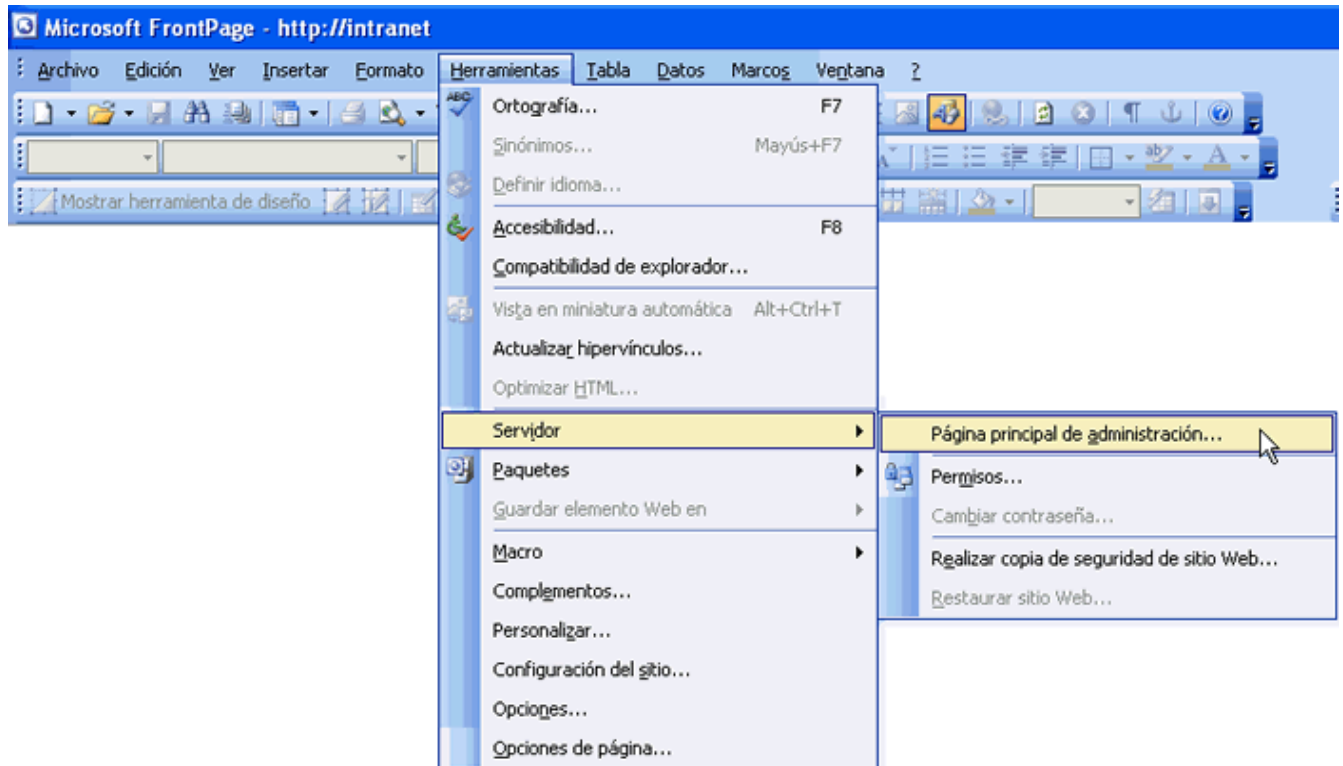
Utilice esta página para crear una función.

<h4>Nombre y descripción de la función</h4> <p>Escriba un nombre y una descripción para esta función.</p>	<p>Nombre de la función:</p> <input type="text"/> <p>Descripción:</p> <input type="text"/>
<h4>Privilegios</h4> <p>Elija los privilegios que se incluyen en esta función. Utilice la casilla de verificación <b>Seleccionar todo</b> para seleccionar o borrar todos los privilegios.</p>	<p>Seleccione los privilegios a incluir en esta función.</p> <p><input type="checkbox"/> <b>Seleccionar todo</b></p> <p><b>Derechos de diseño de Web</b></p> <ul style="list-style-type: none"><li><input type="checkbox"/> Modificar páginas - El usuario puede crear, editar o eliminar páginas HTML y directorios.</li><li><input type="checkbox"/> Examinar - El usuario puede explorar páginas en este sitio Web, incluidas las que contienen componentes Web.</li><li><input type="checkbox"/> Configurar el control de origen - El usuario puede configurar la base de datos de control de código fuente.</li><li><input type="checkbox"/> Aplicar temas a páginas Web - El usuario puede aplicar un tema a un sitio Web.</li><li><input type="checkbox"/> Configurar bordes de páginas Web - El usuario puede aplicar un borde a un sitio Web.</li><li><input type="checkbox"/> Vincular hojas de estilo - El usuario puede aplicar una hoja de estilo a un sitio Web.</li></ul> <p><b>Derechos de administración de Web</b></p> <ul style="list-style-type: none"><li><input type="checkbox"/> Configurar acceso - El usuario pueden crear, eliminar o modificar funciones, agregar usuarios a dichas funciones y especificar qué derechos están asignados a una función.</li><li><input type="checkbox"/> Crear cuentas - El usuario puede crear cuentas para máquinas locales.</li><li><input type="checkbox"/> Administrar el mantenimiento del servidor - El usuario puede ejecutar las funciones de estado de servidor en un servidor virtual.</li><li><input type="checkbox"/> Administrar el análisis de uso - El usuario puede modificar la configuración de análisis de uso en un servidor virtual.</li><li><input type="checkbox"/> Administrar subwebs - El usuario puede crear, cambiar de nombre o quitar subwebs.</li><li><input type="checkbox"/> Actualizar Web - El usuario puede actualizar un sitio Web.</li></ul>

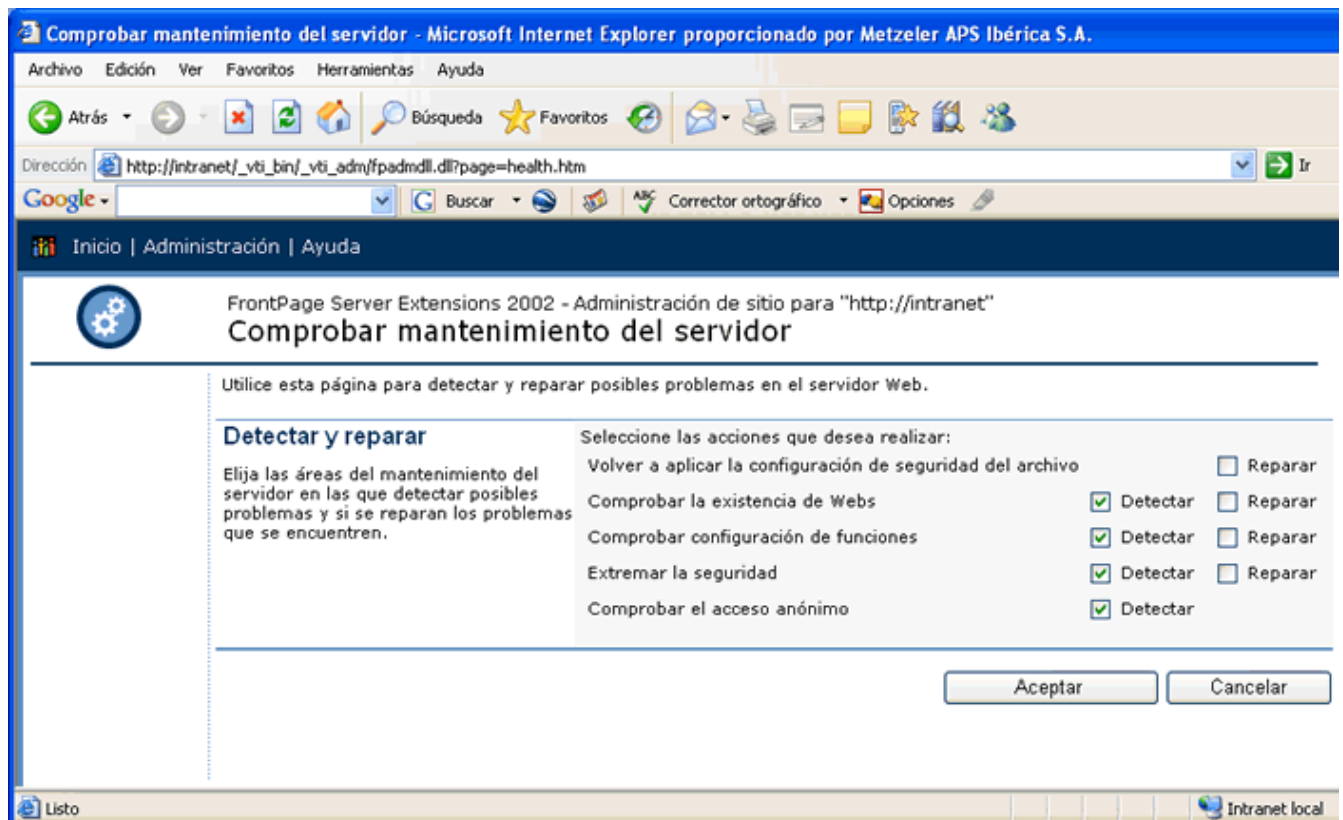
Crear función Cancelar

## Administración del sitio Web



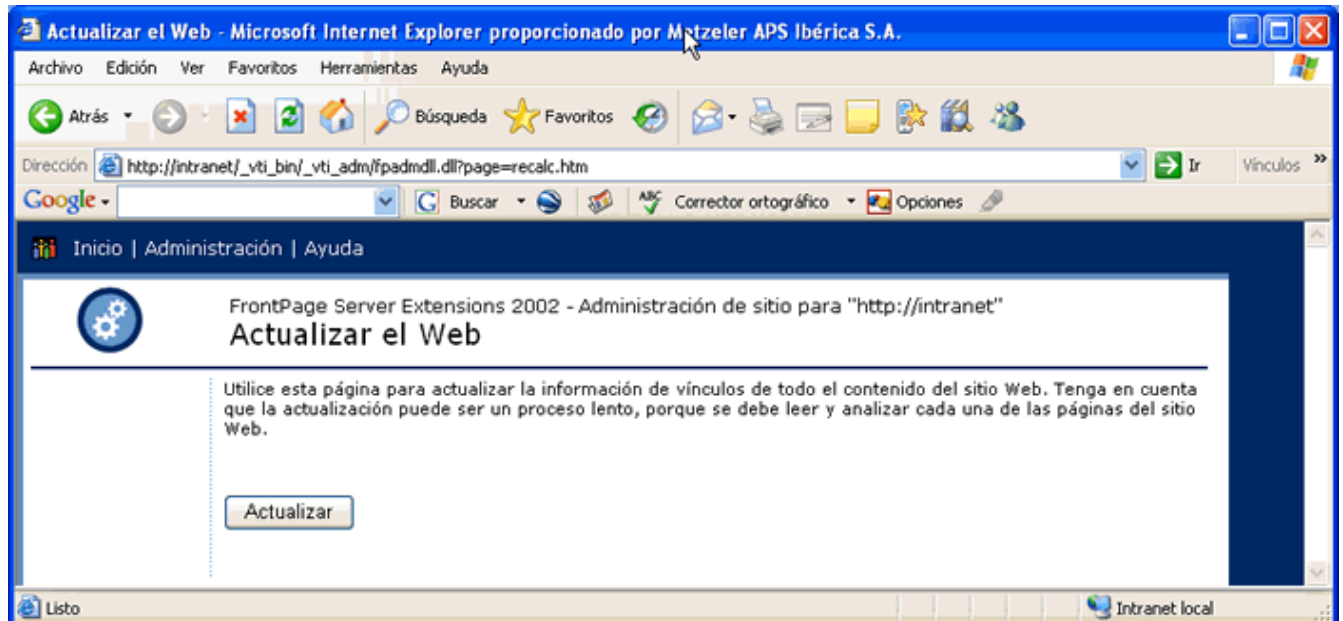


Con esto llegamos a la administración completa del sitio Web con varias opciones, las primeras eran para administrar los usuarios que ya hemos visto antes (están repetidas las mismas opciones de los usuarios y perfiles en esta parte). Además:

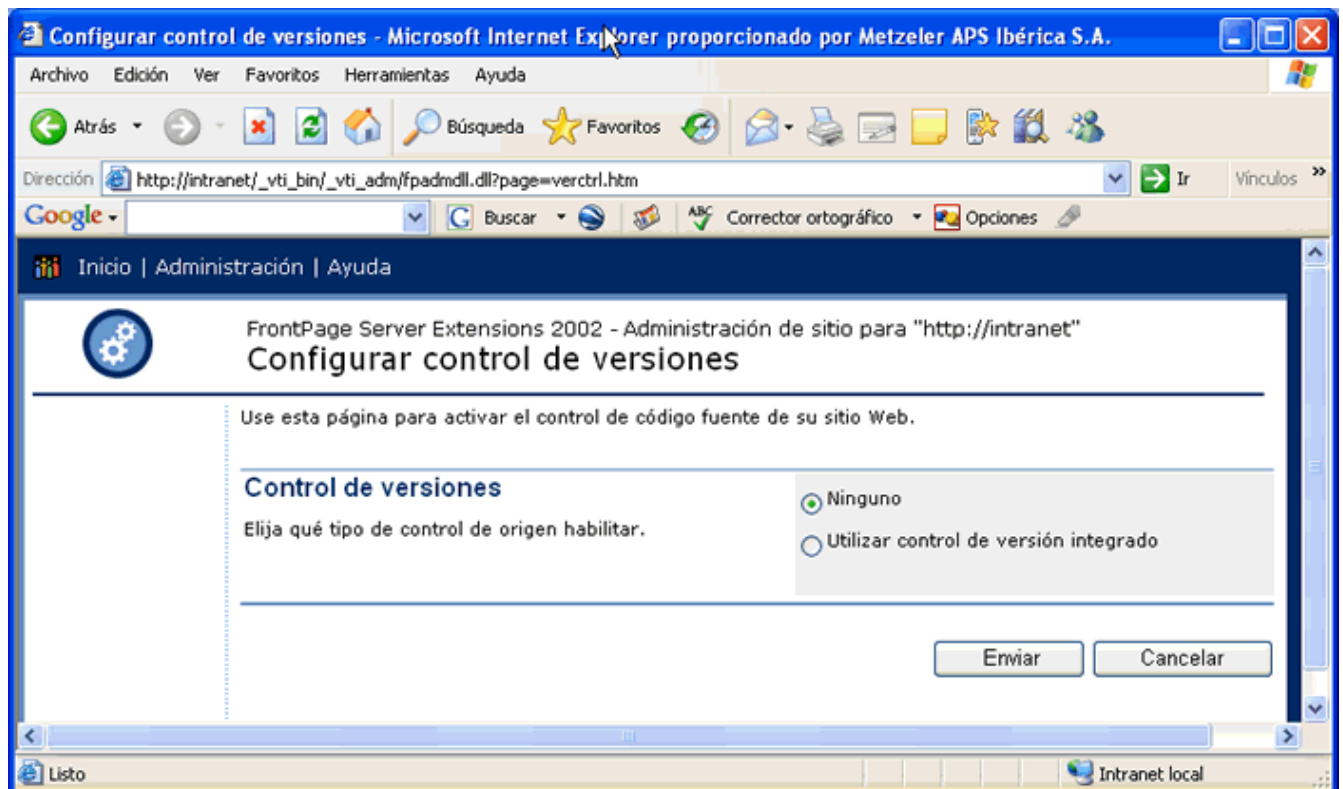


## IIS Avanzado. FrontPage. ASP.NET 3.5

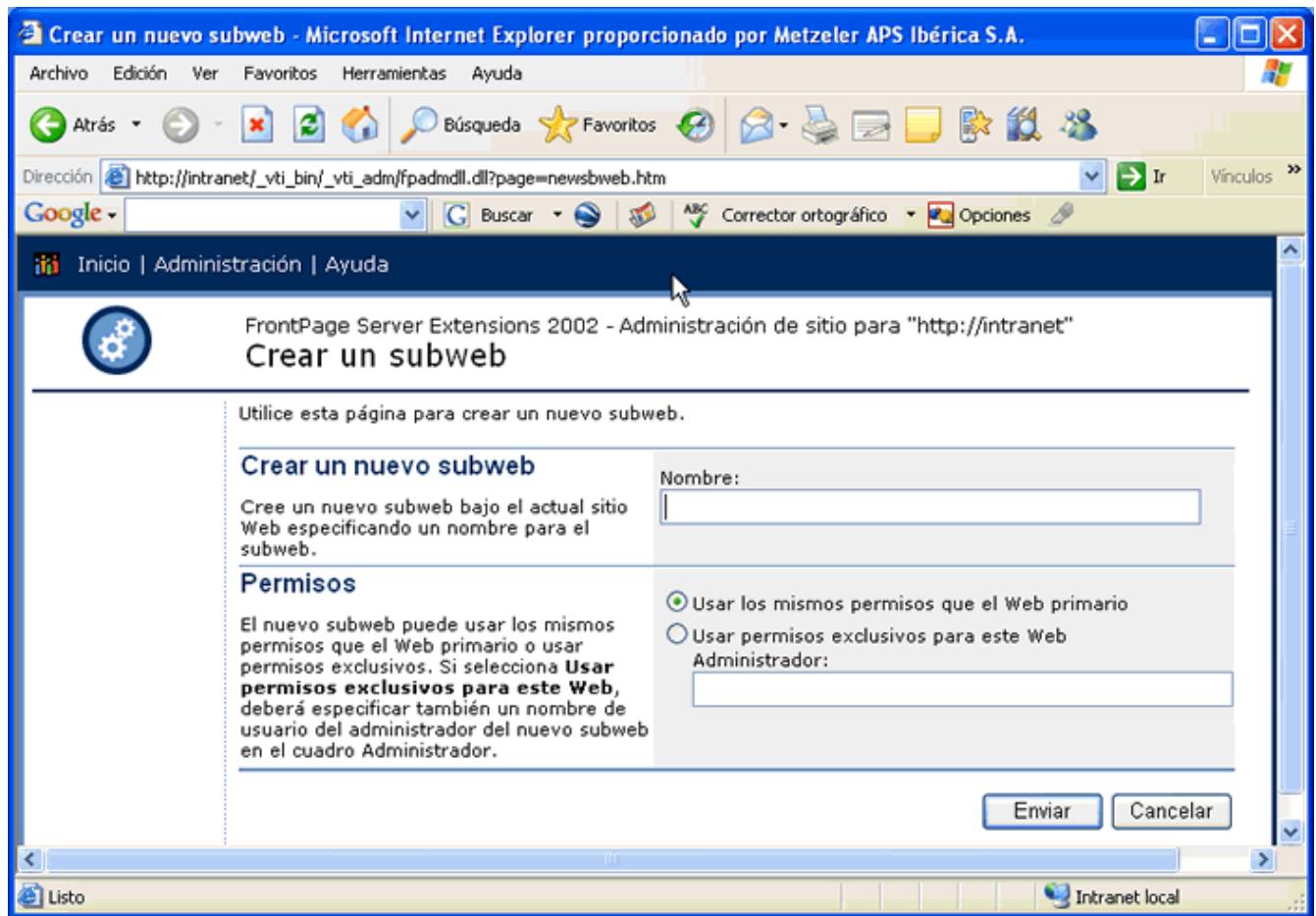
Si alguna vez nos dejan de funcionar correctamente el sitio Web desde FrontPage, por ejemplo se corrompen los permisos, podemos detectar y reparar desde esta página. Además podemos:



Que es para refrescar los enlaces y la base de datos de FrontPage por si quedase desactualizada. Otra opción es controlar quienes modifican las páginas web:



Que gestiona un sencillo control de versiones de las páginas, anotando quien modifica cada página y protegiéndolas para no modificarlas o borrarlas por accidente. Por defecto está desactivada. Finalmente podemos crear nuevos sitios web internos al raíz:



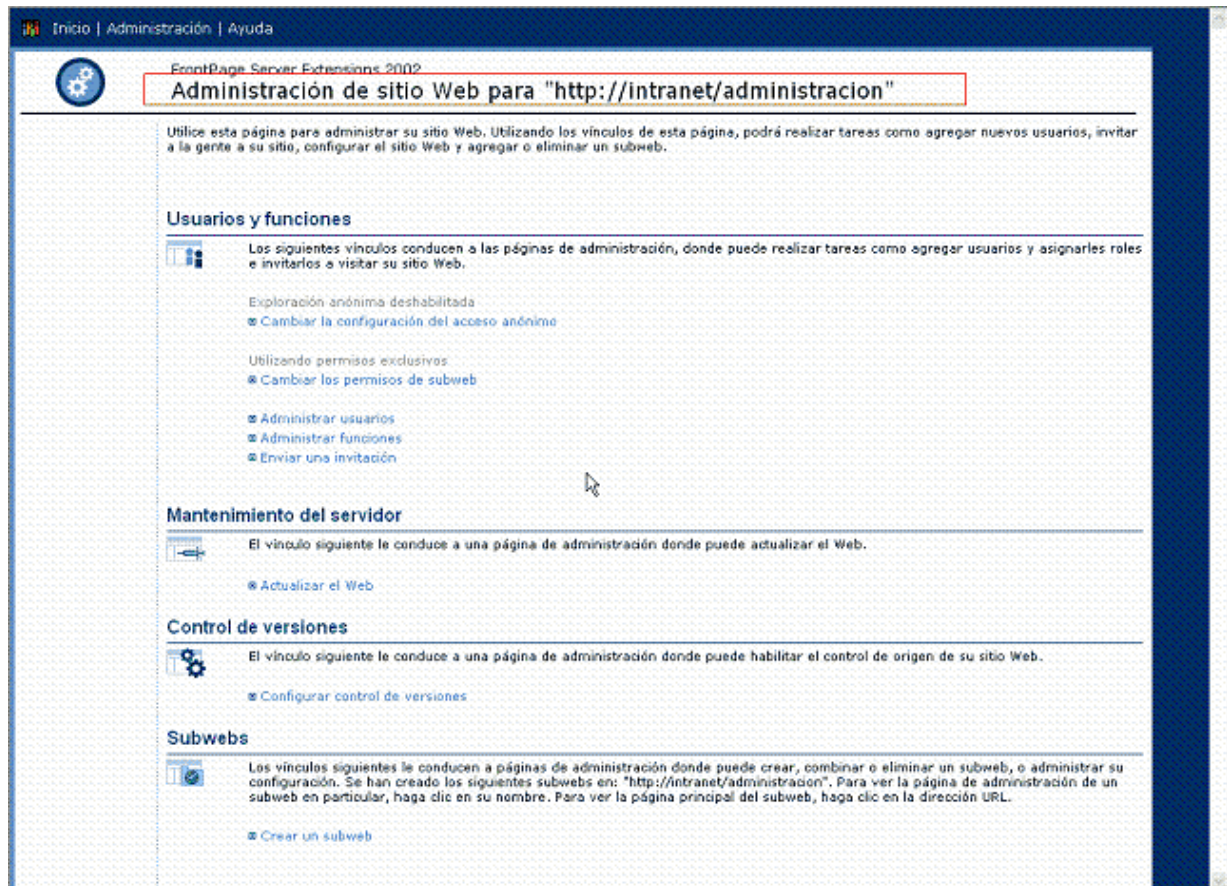
¿Por qué? Muy sencillo si disponemos de varios departamentos queremos que solo determinados usuarios lo administren y modifiquen sus páginas Web. Y por supuesto si hay varios subwebs esos usuarios serán distintos en cada uno de ellos. Como ves en la pantalla podemos indicarle además del nombre si queremos utilizar los mismos permisos y usuarios utilizados en el Web de nivel superior o si queremos crear unos propios indicando en ese caso quien es el administrado. Una vez creados:

### Subwebs

Los vínculos siguientes le conducen a páginas de administración donde puede crear, combinar o eliminar un subweb, o administrar su configuración. Se han creado los siguientes subwebs en: "http://intranet". Para ver la página de administración de un subweb en particular, haga clic en su nombre. Para ver la página principal del subweb, haga clic en la dirección URL. Para ver una lista de todos los subwebs y administrar su configuración, haga clic en **Más subwebs...**

<ul style="list-style-type: none"> <li>■ <a href="#">Crear un subweb</a></li> <li>■ <a href="#">Combinar un subweb</a></li> <li>■ <a href="#">Eliminar un subweb</a></li> </ul>	
<p>Nombre del subweb</p> <ul style="list-style-type: none"> <li>• <a href="#">administracion</a></li> <li>• <a href="#">aspnet/net2006</a></li> <li>• <a href="#">calidad</a></li> <li>• <a href="#">compras</a></li> <li>• <a href="#">costesproducti</a></li> <li>• <a href="#">direccion</a></li> <li>• <a href="#">guias</a></li> <li>• <a href="#">ingenieria</a></li> <li>• <a href="#">logistica</a></li> <li>• <a href="#">nuevosmat</a></li> </ul>	<p>Dirección URL</p> <ul style="list-style-type: none"> <li><a href="http://intranet/administracion">http://intranet/administracion</a></li> <li><a href="http://intranet/aspnet/net2006">http://intranet/aspnet/net2006</a></li> <li><a href="http://intranet/calidad">http://intranet/calidad</a></li> <li><a href="http://intranet/compras">http://intranet/compras</a></li> <li><a href="http://intranet/costesproducti">http://intranet/costesproducti</a></li> <li><a href="http://intranet/direccion">http://intranet/direccion</a></li> <li><a href="http://intranet/guias">http://intranet/guias</a></li> <li><a href="http://intranet/ingenieria">http://intranet/ingenieria</a></li> <li><a href="http://intranet/logistica">http://intranet/logistica</a></li> <li><a href="http://intranet/nuevosmat">http://intranet/nuevosmat</a></li> </ul>

Podemos crear nuevamente mas niveles internos o administrarlos. Fíjate que tenemos si pulsamos en uno creado de nivel inferior:



Tenemos otra vez todas las opciones completas como si fuese un Web completo con sus usuarios, funciones y demás opciones.

Aunque hemos visto Frontpage, esto es obviamente como trabajábamos hasta ahora con ASP NET 2.0. Hemos comentado en el capítulo 1 que no utilizaremos este editor aunque ha sido muy útil saber como se conecta con el servidor de páginas WEB IIS. Ahora comenzaremos con el Visual Web Developer para hacer sitios web en IIS con este entorno...

### 3. Introducción a ASP.NET 3.5

Por fin ha llegado el momento de comenzar con en serio con ASP.NET. Recordemos un poco algunos aspectos de ASP.NET que vimos en el primer capítulo:

¿Qué es ASP.NET?

ASP.NET es un marco de trabajo de programación generado en Common Language Runtime (recuerdas lo del capítulo 1) que puede utilizarse en un servidor para generar aplicaciones Web. ASP.NET ofrece varias ventajas importantes acerca de los modelos de programación Web anteriores:

- **Mejor rendimiento.** ASP.NET es un código de Common Language Runtime compilado que se ejecuta en el servidor. A diferencia de sus predecesores, ASP.NET puede aprovechar las ventajas del enlace anticipado, la compilación just-in-time, la optimización nativa y los servicios de caché desde el primer momento. Esto supone un incremento espectacular del rendimiento antes de siquiera escribir



una línea de código.

- **Compatibilidad con herramientas de primer nivel.** El marco de trabajo de ASP.NET se complementa con un diseñador y una caja de herramientas muy completos en el entorno integrado de programación (Integrated Development Environment, IDE) de **Visual Studio**. La edición WYSIWYG, los controles de servidor de arrastrar y colocar y la implementación automática son sólo algunas de las características que proporciona esta eficaz herramienta. Al final del curso veremos un entorno similar a este Visual Studio.
- **Eficacia y flexibilidad.** Debido a que ASP.NET se basa en Common Language Runtime, la eficacia y la flexibilidad de toda esa plataforma se encuentra disponible para los programadores de aplicaciones Web. La biblioteca de clases de .NET Framework, la Mensajería y las soluciones de Acceso a datos se encuentran accesibles desde el Web de manera uniforme. ASP.NET es también independiente del lenguaje, por lo que puede elegir el lenguaje que mejor se adapte a la aplicación o dividir la aplicación en varios lenguajes. Además, la interoperabilidad de Common Language Runtime garantiza que la inversión existente en programación basada en COM se conserva al migrar a ASP.NET.
- **Simplicidad.** ASP.NET facilita la realización de tareas comunes, desde el sencillo envío de formularios y la autenticación del cliente hasta la implementación y la configuración de sitios. Por ejemplo, el marco de trabajo de página de ASP.NET permite generar interfaces de usuario, que separan claramente la lógica de aplicación del código de presentación, y controlar eventos en un sencillo modelo de procesamiento de formularios de tipo Visual Basic. Además, Common Language Runtime simplifica la programación, con servicios de código administrado como el recuento de referencia automático y el recolector de elementos no utilizados.
- **Facilidad de uso.** ASP.NET emplea un sistema de configuración jerárquico, basado en texto, que simplifica la aplicación de la configuración al entorno de servidor y las aplicaciones Web. Debido a que la información de configuración se almacena como texto sin formato, se puede aplicar la nueva configuración sin la ayuda de herramientas de administración local. Esta filosofía de "administración local cero" se extiende asimismo a la implementación de las aplicaciones ASP.NET Framework. Una aplicación ASP.NET Framework se implementa en un servidor sencillamente mediante la copia de los archivos necesarios al servidor. No se requiere el reinicio del servidor, ni siquiera para implementar o reemplazar el código compilado en ejecución.
- **Escalabilidad y disponibilidad.** ASP.NET se ha diseñado teniendo en cuenta la escalabilidad, con características diseñadas específicamente a medida, con el fin de mejorar el rendimiento en entornos agrupados y de múltiples procesadores. Además, el motor de tiempo de ejecución de ASP.NET controla y administra los procesos de cerca, por lo que si uno no se comporta adecuadamente (filtraciones, bloqueos), se puede crear un proceso nuevo en su lugar, lo que ayuda a mantener la aplicación disponible constantemente para controlar solicitudes.
- **Posibilidad de personalización y extensibilidad.** ASP.NET presenta una arquitectura bien diseñada que permite a los programadores insertar su código en el nivel adecuado. De hecho, es posible extender o reemplazar cualquier subcomponente del motor de tiempo de ejecución de ASP.NET con su propio componente escrito personalizado. La implementación de la autenticación personalizada o

de los servicios de estado nunca ha sido más fácil.

- **Seguridad.** Con la autenticación de Windows integrada y la configuración por aplicación, se puede tener la completa seguridad de que las aplicaciones están a salvo.

Bueno esto es un poco técnico así que vamos a pasar a la acción... pero antes de continuar veamos varias formas hacer algunas páginas web para seguir situando nuestras páginas ASP.NET. Digamos que históricamente estas son las fases por las que ha pasado la historia de las páginas web.

1. Primero fueron las **páginas estáticas**. Muestran información estática sin ningún elemento variable (no incluyo los gráficos).
2. **Páginas con programación de cliente**. En este caso las páginas incorporan pequeños programas que se ejecutan en el cliente (Internet Explorer, Firefox, Opera...) y no en el servidor (ASP, ASP.NET). Esta programación suele ser pequeñas validaciones de entrada de datos y algún efecto visual. Normalmente están realizadas con el **lenguaje JavaScript**.
3. **Páginas con programación de servidor**. Este es el caso que nos ocupa, se envían unos datos al servidor (consulta de un nº de teléfono) y éste nos devuelve una tabla con el resultado. Esto se puede hacer con muchos programas, por ejemplo en el sistema operativo Linux se utiliza el PHP cuyos resultados son superiores a las viejas páginas ASP pero muy inferiores a nuestras nuevas páginas ASP.NET. **Lenguajes del "lado del servidor"** (así se llaman cuando se ejecutan en el servidor) son:

- ◆ ASP (antigua tecnología de Microsoft) ó Active Server Pages
- ◆ ASP.NET, la nueva tecnología de páginas de servidor basada en .NET
- ◆ PHP, lenguaje utilizado en los servidores Linux
- ◆ JSP, tecnología que permite utilizar Java para crear páginas dinámicas
- ◆ CGI, antigua tecnología utilizada sobre todo en servidores UNIX

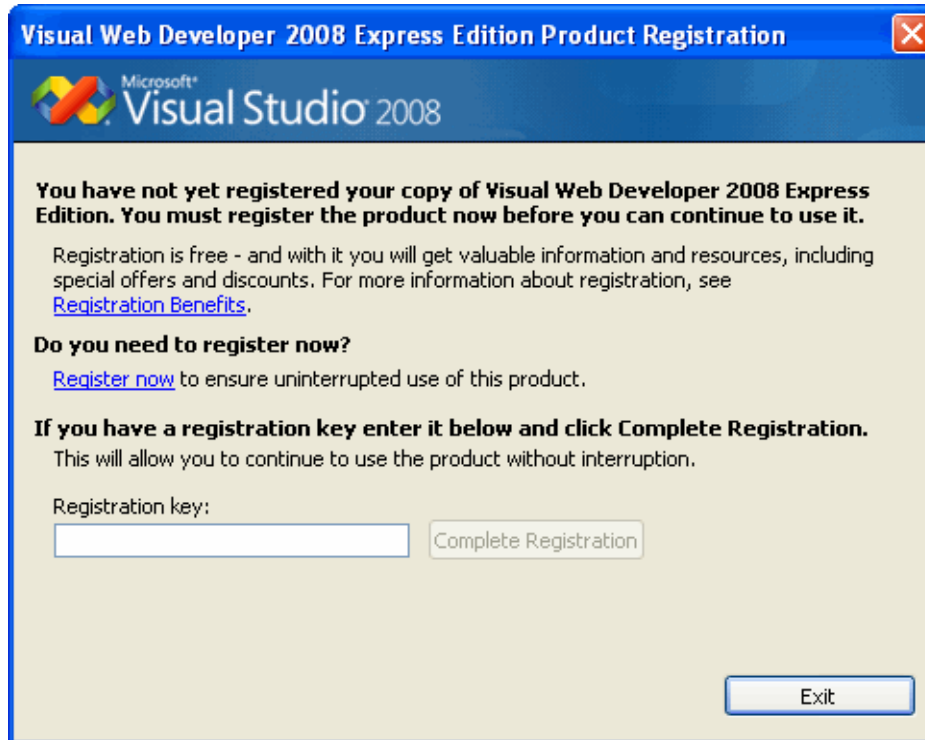
4. **Páginas con programas**. Hay dos formas de incrustar programas en las páginas web: con **Activex de Microsoft** y con el **lenguaje Java de Sun**. Los dos casos los conocemos de sobra porque están presentes en muchas de las páginas que vemos por internet, el primero instala en nuestro ordenador el programa o ActiveX necesario y a partir de ahí la página puede ejecutar esos programas dentro de ella, por ejemplo las animaciones con Flash utilizan un ActiveX. En el caso de Java se realizan programas de cualquier tipo ya que es un lenguaje multipropósito y se ejecutan dentro de las páginas. En contra tiene la velocidad y la dificultad de programar en este lenguaje.

Nuestro objetivo es el mas interesante y es el punto 3: realizar páginas que se ejecutan en un servidor web (Internet Information Server) y que devuelven un resultado (consulta a bases de datos) utilizando el lenguaje ASP.NET

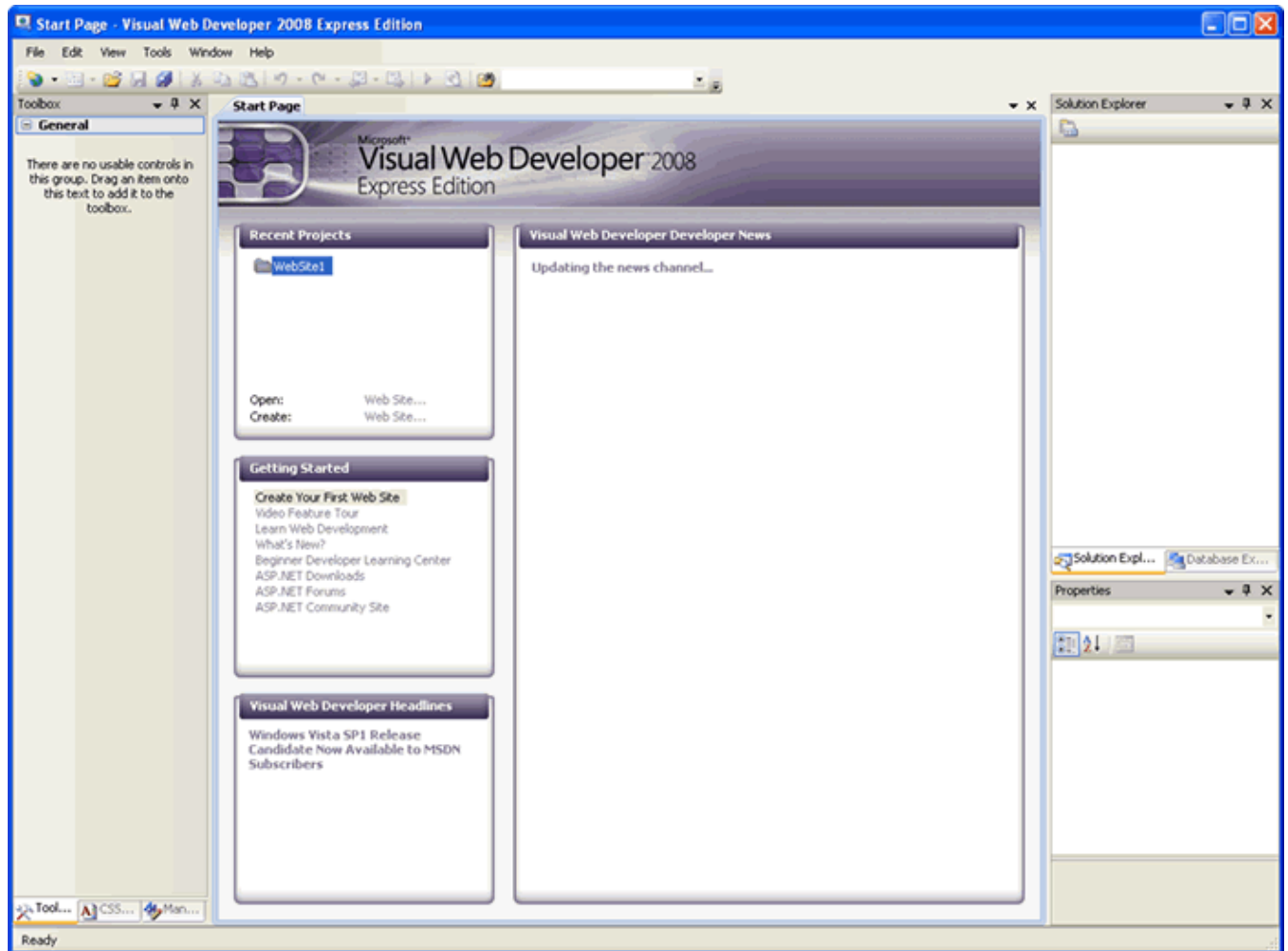
**Nota** Para los programadores de ASP (en los que me incluyo) decir que "tenemos que cambiar el chip" y reconstruir nuestro esquema de servidor. El modelo de objetos que nos proporciona .NET Framework es impresionante. Seguramente haga falta un manual de 1000 páginas para ver con detalles todos los objetos que nos ofrece este nuevo entorno. Comenzaremos con hacer las mismas cosas que hacíamos con ASP... ¡paciencia!

## 4. Primeros pasos con ASP.NET

Vamos a poner en marcha nuestro entorno de desarrollo integrado para crear nuestra primera página ASP.NET y así probar que tenemos todo en marcha. Ejecutamos el Visual Web Developer 2.008 donde nos encontramos con el ultimo punto de la instalación y es el registro del programa:

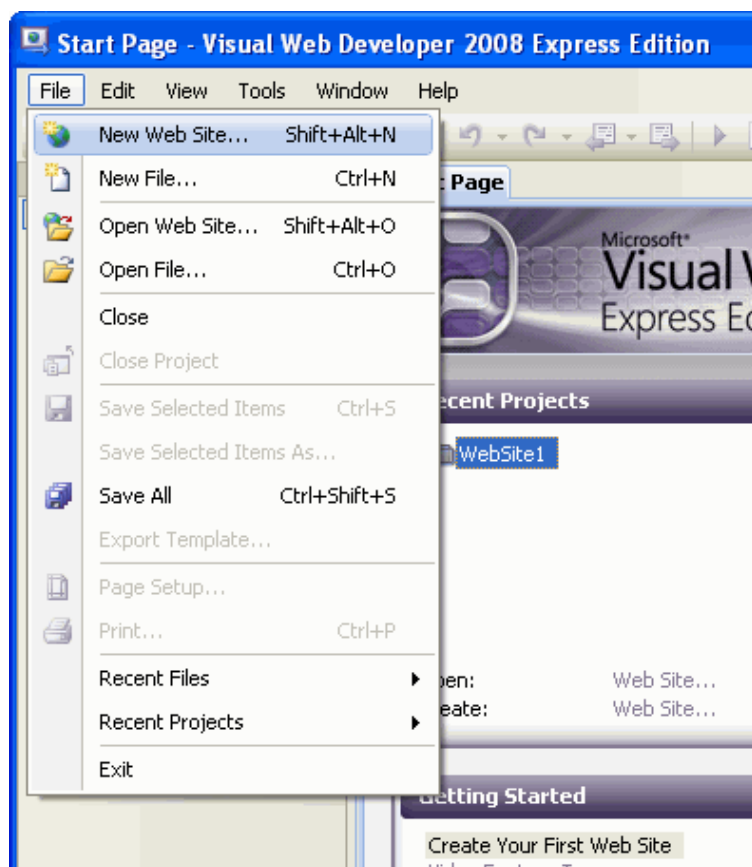


Nos registramos para obtener la clave y nos encontraremos después con el entorno:

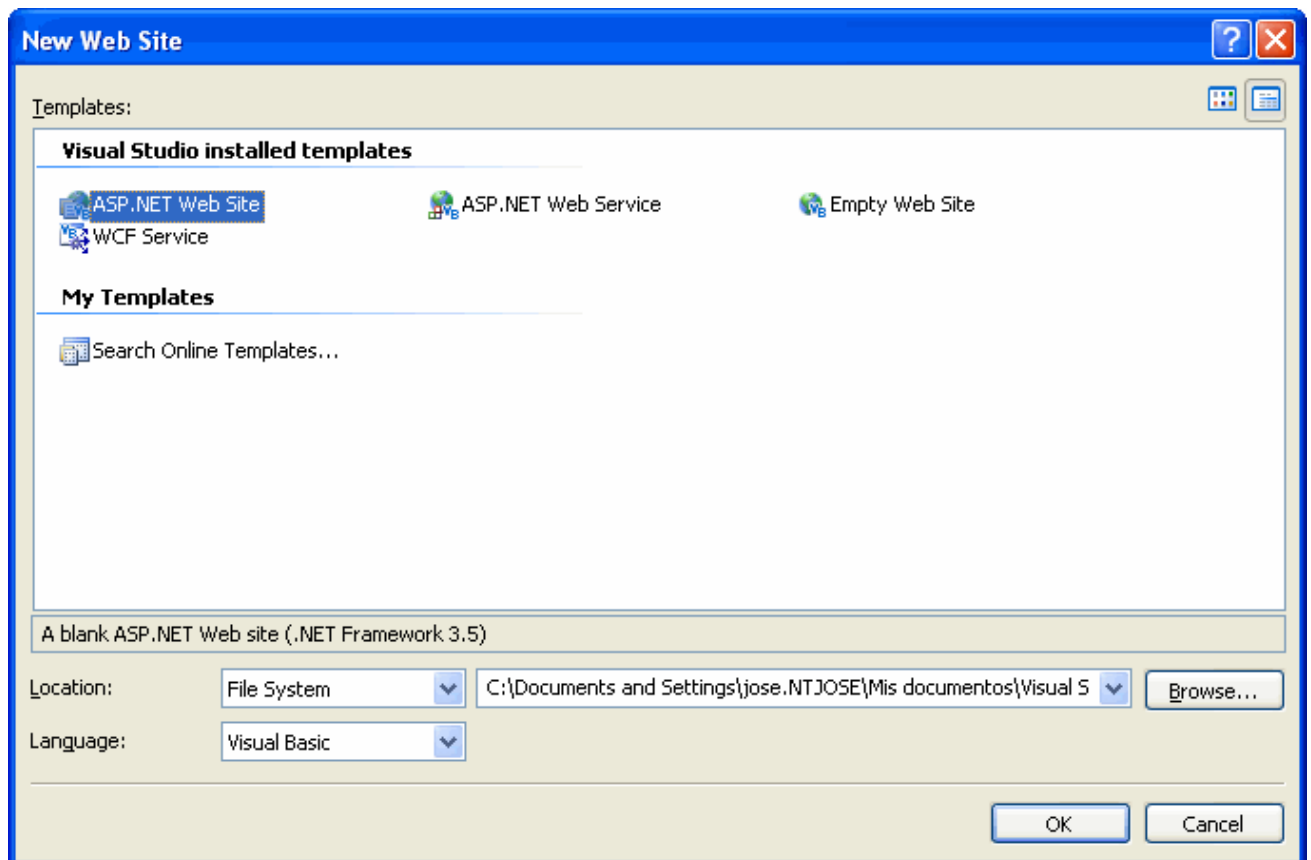


Tenemos multitud de opciones que iremos viendo poco a poco, de momento vamos a crearnos un web para poder empezar a crear páginas. Pulsamos en la opción de menu "File" y luego "New Web Site" para crear un sitio web. Es lógico, no vamos a crear páginas sueltas sino un sitio web completo que compartirá ciertas partes como diseños, variables, ... Por tanto lo que queremos crear es un sitio web.

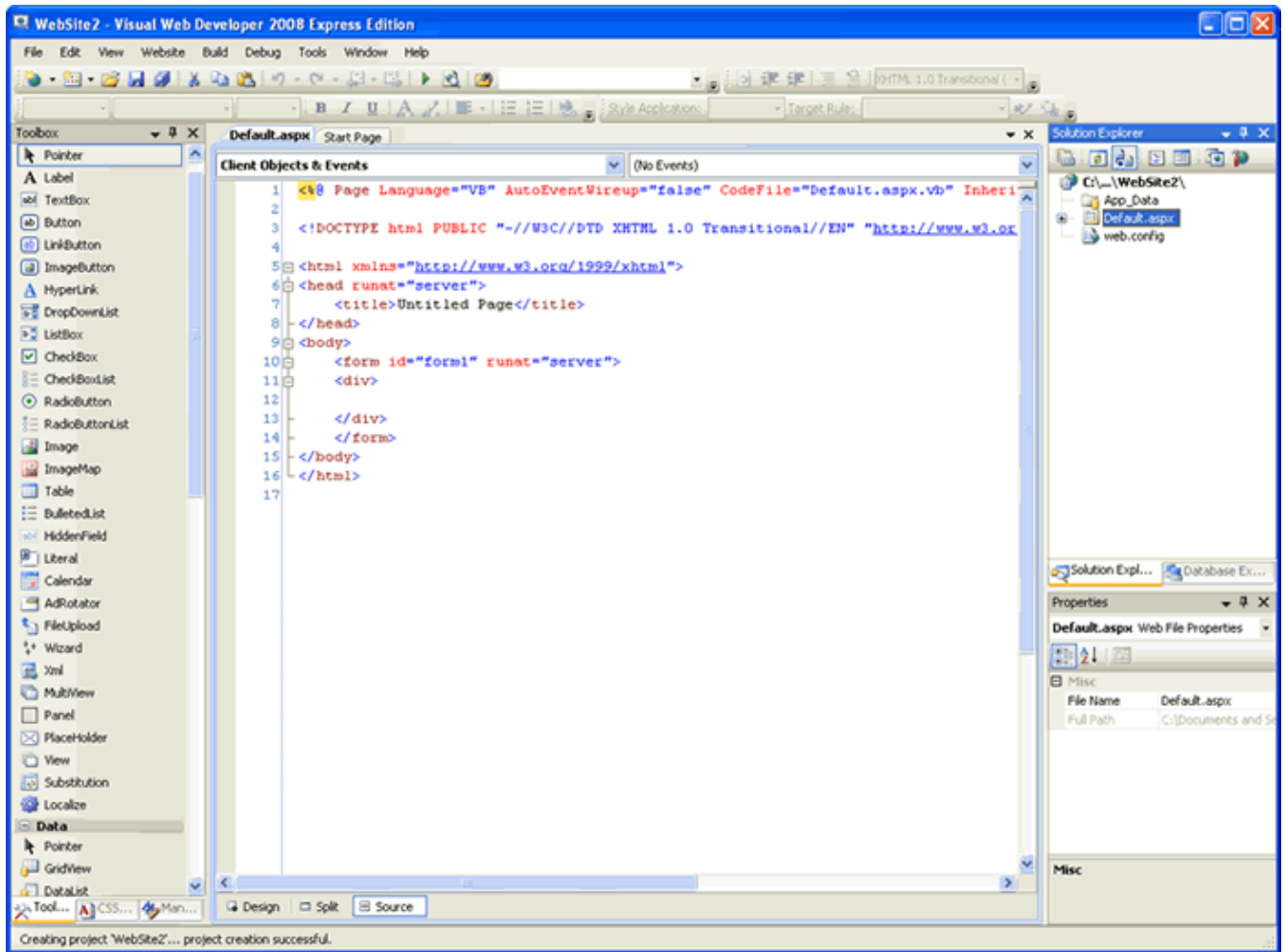




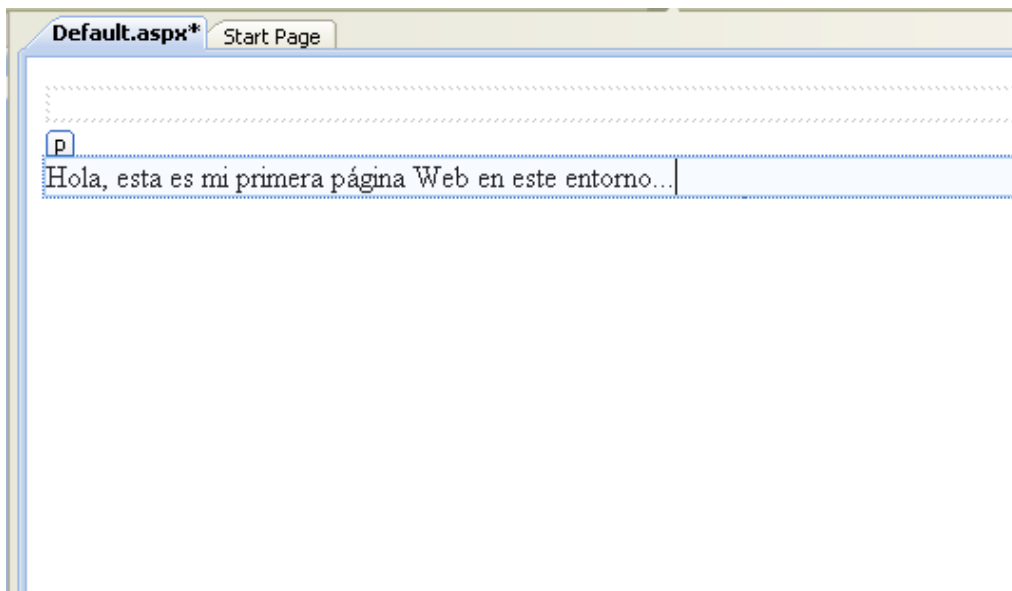
Ahora nos muestra las opciones disponibles:



En nuestro caso queremos crear un sitio web ASP.NET, luego esa es la opción que seleccionamos. Debajo fíjate que el lenguaje que vamos a utilizar el "Visual Basic" y "Location" indica que es "File System" es decir, una carpeta normal de nuestro disco duro. Mas adelante ya veremos todas las opciones ya que tendremos que crear y publicar nuestros sitios web en nuestro flamante IIS instalado en los capítulos 1 y 2. Pulsamos "OK" para continuar:

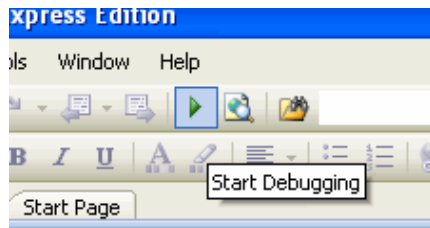


No está mal ¿eh? Vaya cantidad de componentes y pantallas! No te preocupes veremos todas las opciones de este entorno. De momento no lo vamos a describir, lo que queremos es ejecutar nuestra primera página en el navegador. Así que hacemos clic en la parte central y abajo para poner la pantalla en la vista de diseño en lugar de la del código, así que haz clic en "Design" y escribe un texto de prueba:

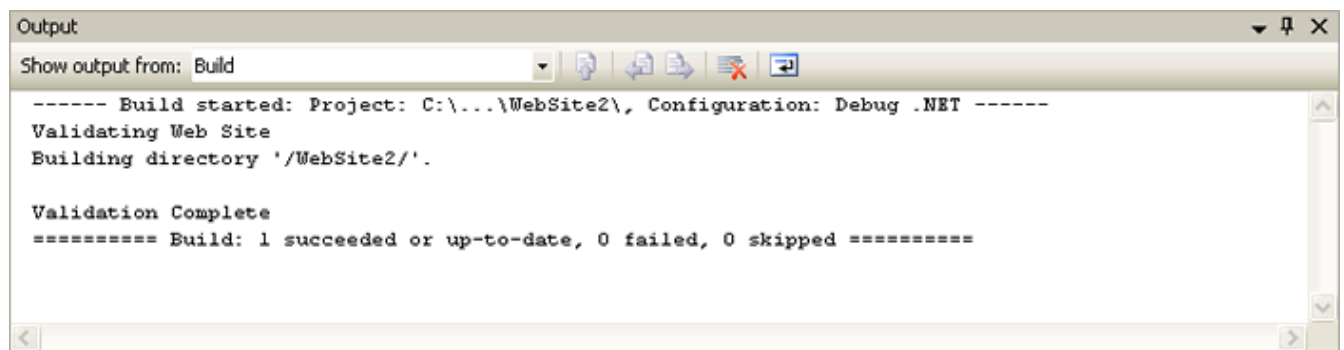


## IIS Avanzado. FrontPage. ASP.NET 3.5

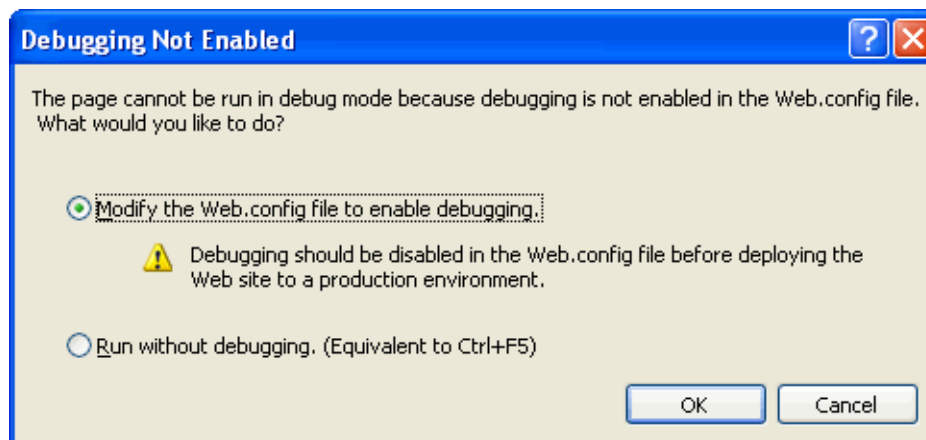
Ahora vamos a ejecutarla, aunque como no hay nada de código ASP.NET simplemente nos la deberá mostrar en el navegador. Así que vamos a pulsar en el "depurador":



Luego veremos todos estos conceptos pero es mejor que pongamos todo en marcha para comprobarlo y así aprender alguna cosilla por encima. Al darle a este botón pasarán varias cosas, nuestro entorno ejecutará la página. Para esto primero debe "compilarla" es decir, preparar todo el código ASP.NET que tiene dentro, comprobar que su sintaxis está bien y finalmente mostrarla en pantalla. Así que verás en la parte de abajo la siguiente pantalla:



Son mensajes de ayuda y de alerta que nos vendrá muy bien mas adelante. Sigamos... al poco te aparecerá esta pantalla:



Que nos avisa que no puede ejecutar la página porque no tiene configurado el modo de depuración nuestro sitio web. Pero nos ofrece la posibilidad de activar esto para poder continuar, así que modificará esta opción pulsando en "OK" y podremos continuar... Ahora vemos algo mas raro todavía, si veis la parte de abajo a la derecha nos ha aparecido este mensaje:



¿y esto que es? Esto merece un rato de explicación. Sabemos que tenemos que ejecutar las páginas en un servidor Web, como el IIS que hemos instalado antes. Ya sabes que es el servidor Web de Microsoft y que viene con 2000, 2003, 2008, XP y Vista

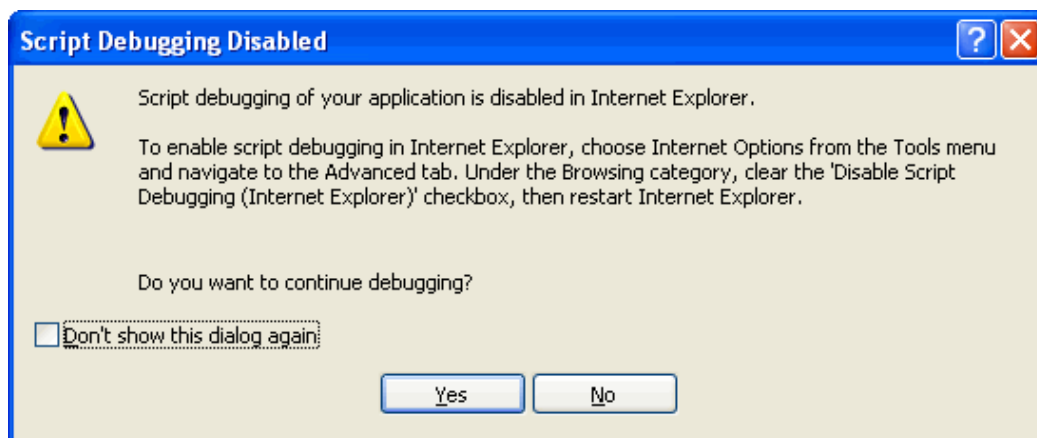
**Nota** En las versiones Home no viene el servidor de páginas web Internet Information Server (IIS), hay que tener la versión profesional.

Pues bien... si todavía no hemos conectado nuestro sitio web de ejemplo con el IIS... ¿cómo podemos ejecutar las páginas? Recuerda que no se pueden hacer doble clic en ellas en el explorador de archivos sino que deben ejecutarse en un servidor web. Bueno, pues la solución es muy buena y muy sencilla: nuestro entorno de desarrollo incorpora un pequeño servidor web para la depuración y la ejecución de las páginas. No se puede utilizar como servidor web, es sólo para la prueba de las páginas, así no tenemos que moverlas a un servidor IIS para probarlas. Luego muy bien, las crearemos aquí y podremos probarlas completamente antes de pasarlas al IIS. Si te fijas pone:

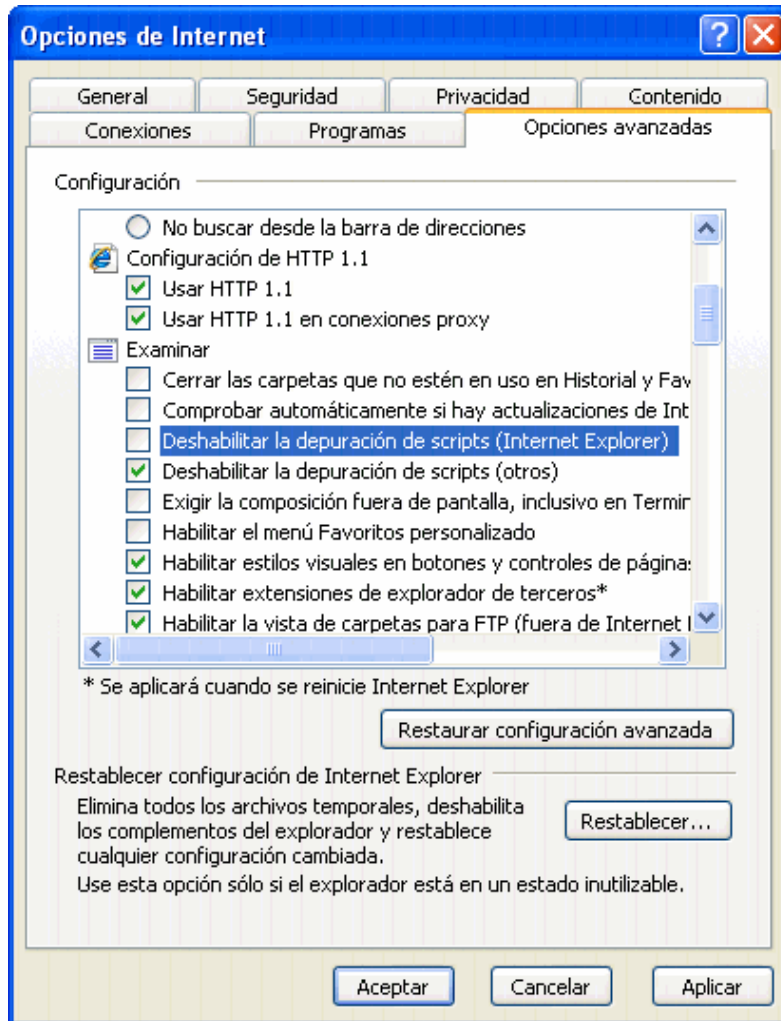
http://localhost:4450/Website2

El servidor es "localhost" que es una palabra clave que apunta a nuestro propio equipo y ejecuta el servidor web en el puerto 4450 de ahí localhost:4450. Ya sabemos que los servidores web se ejecutan en el puerto 80 pero como este es uno de pruebas lo ejecuta en ese puerto, no importa ya que es para probar las páginas. Y por fin pone en marcha la página predeterminada del sitio web llamado "Website2". Así que muy bien, tenemos en nuestro equipo un pequeño servidor web para poder ejecutar las páginas, no hay que configurar nada de este servidor web porque lo gestiona todo nuestro entorno.

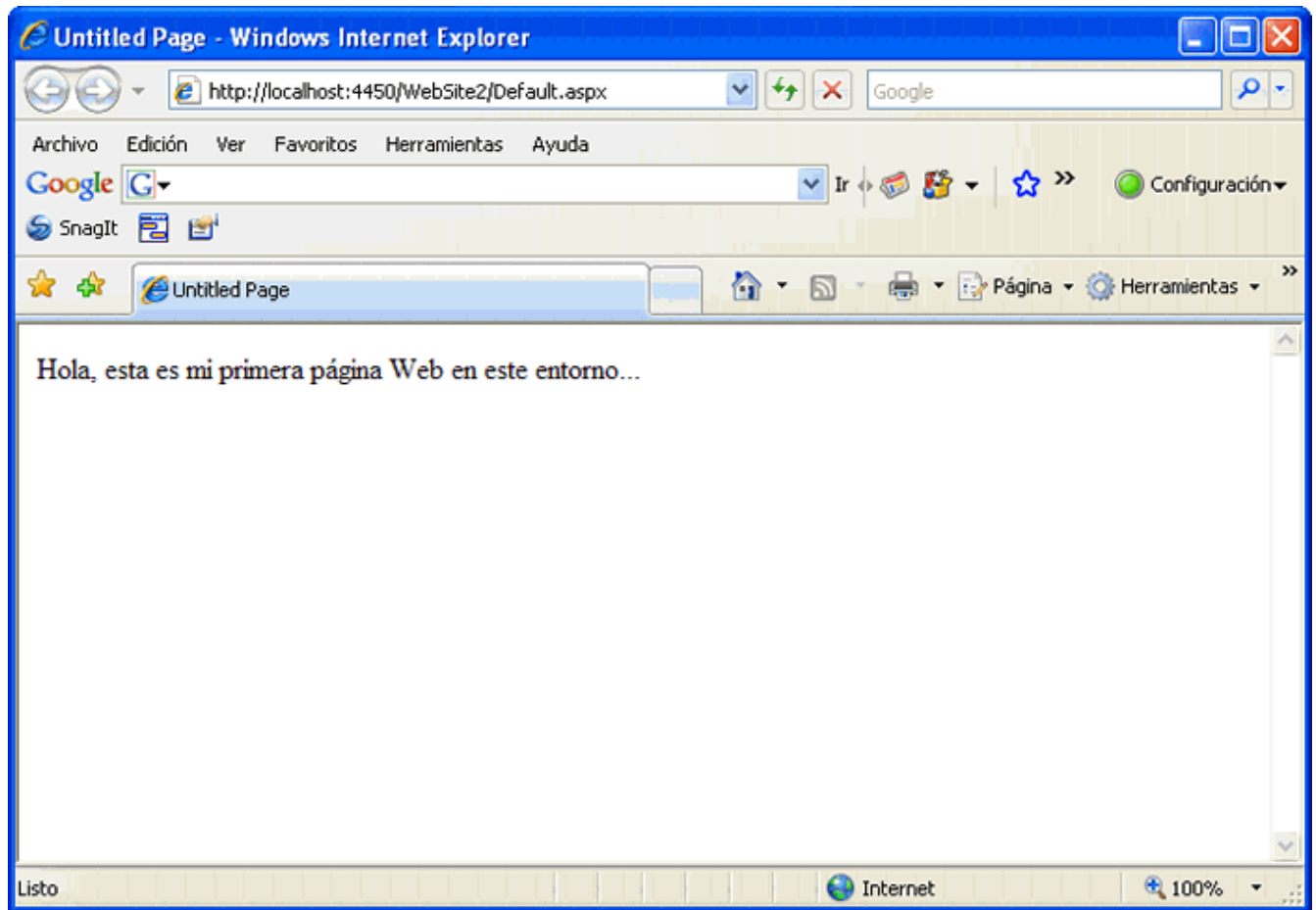
En mi caso, al intentar ejecutar la página me ha dado otro mensaje:



Que me avisa que no tengo activada la depuración en el Internet Explorer, es así por defecto, así que me iré al Internet Explorer a cambiar esta opción. Nos vamos a las Opciones del Internet Explorer del menu herramientas y seleccionamos la solapa "Opciones Avanzadas":



Desmarcamos la opción de "Deshabilitar la depuración de scripts (Internet Explorer)" y reiniciamos el navegador. Cancelamos la ejecución de la página del mensaje anterior para que coja estos cambios y le volvemos a pulsar al botón de depuración para ejecutar la página:

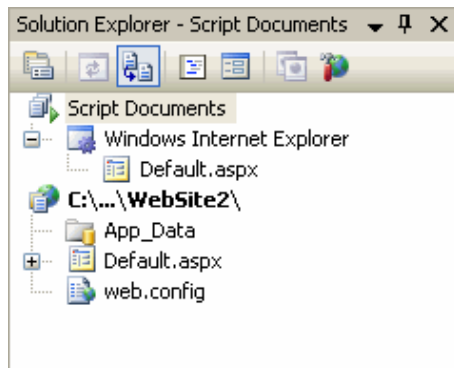


Por fin! ya hemos conseguido ejecutar nuestra primera pagina web, aunque sin código ASP.NET dentro. De todas formas hemos aprendido muchas cosas con la ejecución ya que sabemos que podemos probar en nuestros equipos las páginas con el miniservidor web que incorpora el entorno de desarrollo del Visual Web 2.008. En otras condiciones se editaban las paginas con FrontPage, por ejemplo, y se ejecutaban en el servidor Web IIS. Ahora podemos probarlas en local antes de moverlas al servidor de explotación, o de ejecución.

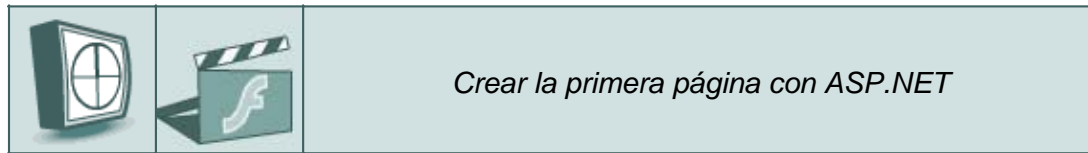
Si te fijas en la barra de direcciones del navegador web, la página que ha ejecutado es:

`http://localhost:4450/WebSite2/Default.aspx`

SI te fijas ahora en la parte derecha de nuestro entorno de desarrollo verás:



Esta pantalla es el "explorador de soluciones" que contendrá todos nuestros componentes del sitio web. Puedes ver en la lista de abajo que está la página "default.aspx" que es la que nos creó automáticamente al crear el sitio web y es donde escribimos antes nuestro primer texto.



## 4.1 Ejecución del lado del servidor

Vamos a crear ahora nuestra primera página con código ASP.NET. Será un sencillo ejemplo para saber como se ejecutan estas páginas y así repasamos la ejecución otra vez. Vamos al editor de páginas y volvemos a la edición de nuestra página de ejemplo "default.aspx"

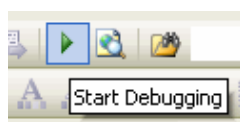
Activamos la vista en código fuente seleccionando de las solapas de debajo la que pone "Source" y escribimos lo siguiente:

```

1  <%@ Page Language="VB" AutoEventWireup="false" CodeFile="Default.aspx.vb" Inherits="Default" %>
2
3  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
4
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7      <title>Untitled Page</title>
8  </head>
9  <body>
10     <form id="form1" runat="server">
11     <div>
12
13     </div>
14     </form>
15     <p>
16         Hola, esta es mi primera página Web en este entorno...</p>
17         Hoy es <%= Now %>
18     </body>
19 </html>
20

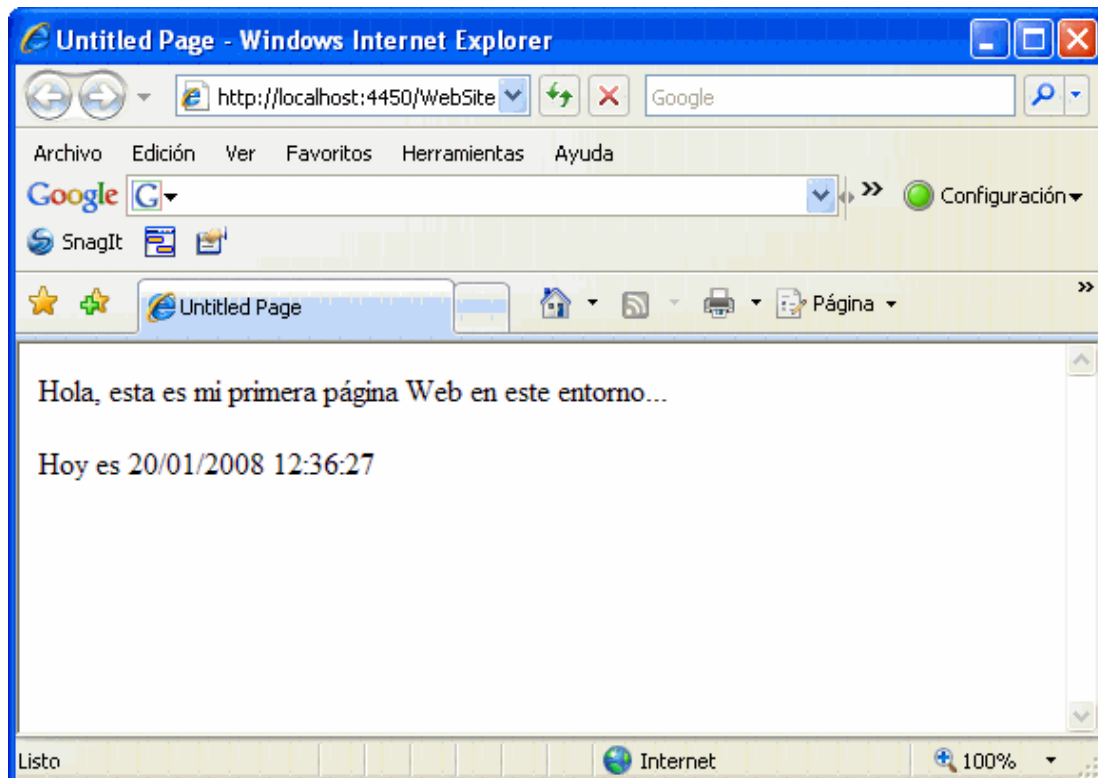
```

Ten cuidado y escribe ese código tal y como lo estoy poniendo, teniendo cuidado en poner los símbolos: <% = now %>. Una vez escrito pulsa otra vez en ejecutar la página:



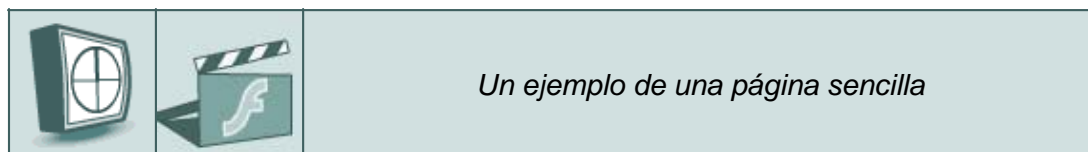
Y observa el resultado:





"Now" es una función que devuelve la fecha y hora del sistema. Veamos lo que ha pasado... Nuestro servidor web a ejecutado la página, esto es, ha localizado el código ASP.NET que hemos insertado y lo ha ejecutado antes de devolver la página. Como en este caso era una simple llamada a la función Now, le hemos dicho con ese código que devuelva el resultado de ejecutar esa función, y por tanto, la fecha y hora que ves en pantalla. Esto es la ejecución del lado del servidor, o lo que es lo mismo, la ejecución de una página por el servidor de páginas web. Si hubiésemos ido por el explorador de archivos y le hubiésemos dicho que abriera esa página, no habría funcionado porque debe pasar por el servidor web para poder hacer esa ejecución. Recuerda que en el capítulo anterior vimos como en nuestro IIS sabe que las páginas .aspx debe procesarlas antes de devolverlas.

El otro tipo de ejecución es la del "lado del cliente", es decir código que está dentro de la página pero que se ejecuta en el Internet Explorer. Este código es muy distinto ya que solo hace pequeñas operaciones sobre el código de la página. Este código se suele escribir en Java o en Javascript y no tiene comparación en cuanto a potencia porque no tenemos a nuestra disposición todas las funciones de ASP.NET ejecutándose en el servidor. De momento nos quedamos con esta idea en cuanto a que las páginas asp.net se ejecutan en el servidor y nos devolverá la página resultante de interpretar el código ASP que hemos incluido dentro. De momento solo la escritura de la hora pero pronto la consulta a bases de datos, realización de gráficos segun consultas a bases de datos, envío de correo y mil funciones mas...



## 5. ASP.NET y el servidor web

## IIS Avanzado. FrontPage. ASP.NET 3.5

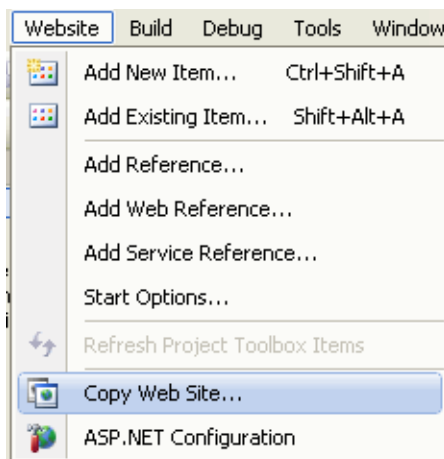
ASP.NET está pues conectado con un servidor que puede ser un IIS, como seguramente sea cuando esté en funcionamiento nuestra aplicación web o ejecutándose con el miniservidor web que nos ofrece el entorno de desarrollo integrado.

En el ejemplo del punto anterior, donde escribíamos la fecha veíamos que la página debe ser ejecutada por un servidor. Si hemos obtenido esa página con la fecha correcta significa que todo está correcto y que tenemos en marcha nuestro ASP.NET porque ha sido él quien ha analizado la página ha visto que había código ASP.NET dentro (la función now) lo ha ejecutado y nos ha devuelto una página web a nuestro navegador.

Una vez hayamos terminado nuestro sitio web tendremos que colocarlo desde nuestro equipo, donde lo hemos creado y puesto a prueba al servidor principal con IIS. Es decir copiar el web de nuestro disco duro al del servidor web. Tendremos que realizar dos sencillos pasos:

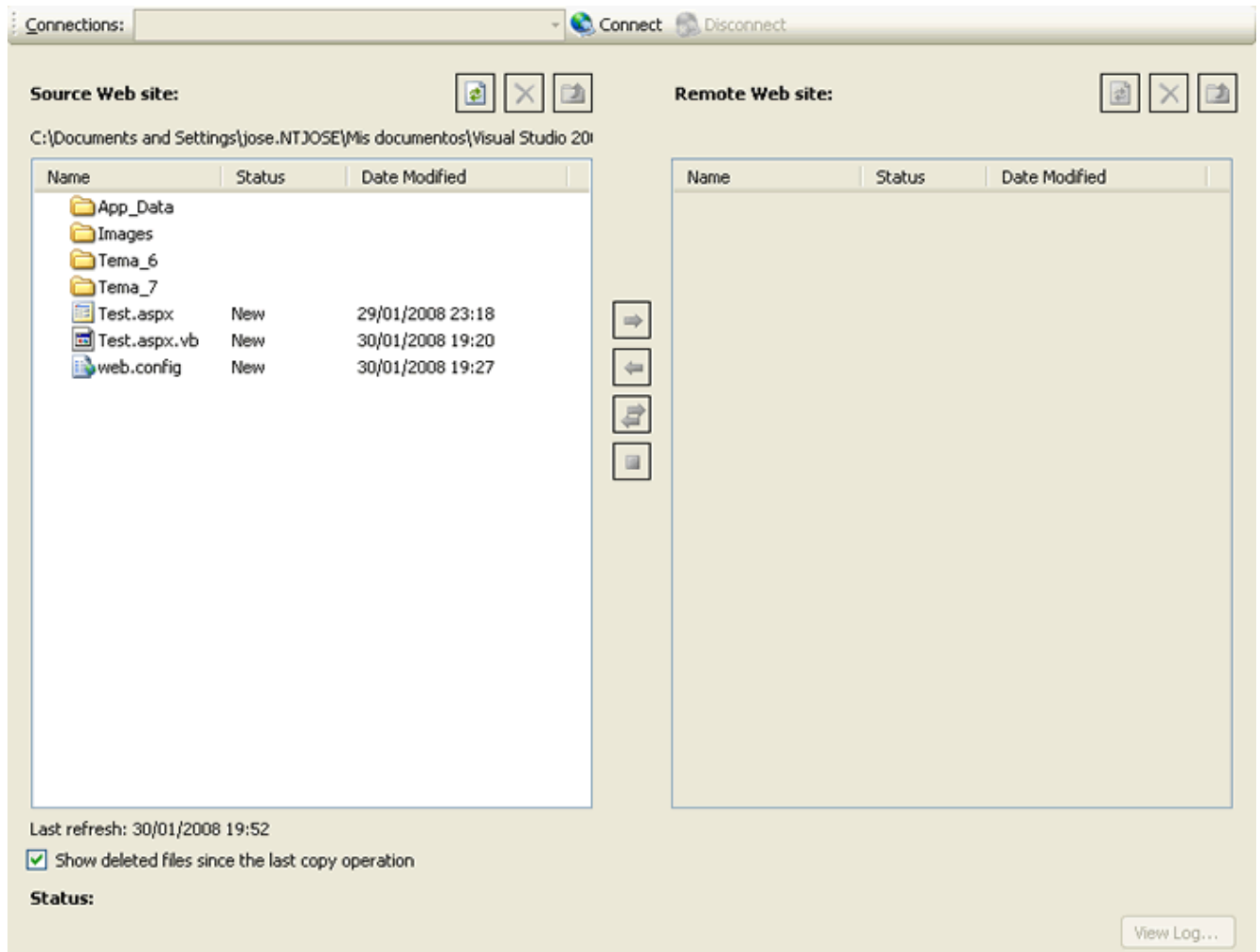
1. Crear un directorio virtual en el servidor
2. Copiar el sitio web completo

Antes de copiarlo debemos desactivar todo el código que hayamos puesto para depuración y que veremos mas adelante. Sigamos con nuestra labor de publicar el web en el servidor. Te voy a mostrar un ejemplo, pero luego puedes utilizar otro de los distintos métodos que hay. Supongamos que tenemos el web de pruebas creado en una carpeta en nuestro disco y queremos moverlo al servidor final. Para realizar esta labor nos iremos a la opción de publicar web en el menu "website":

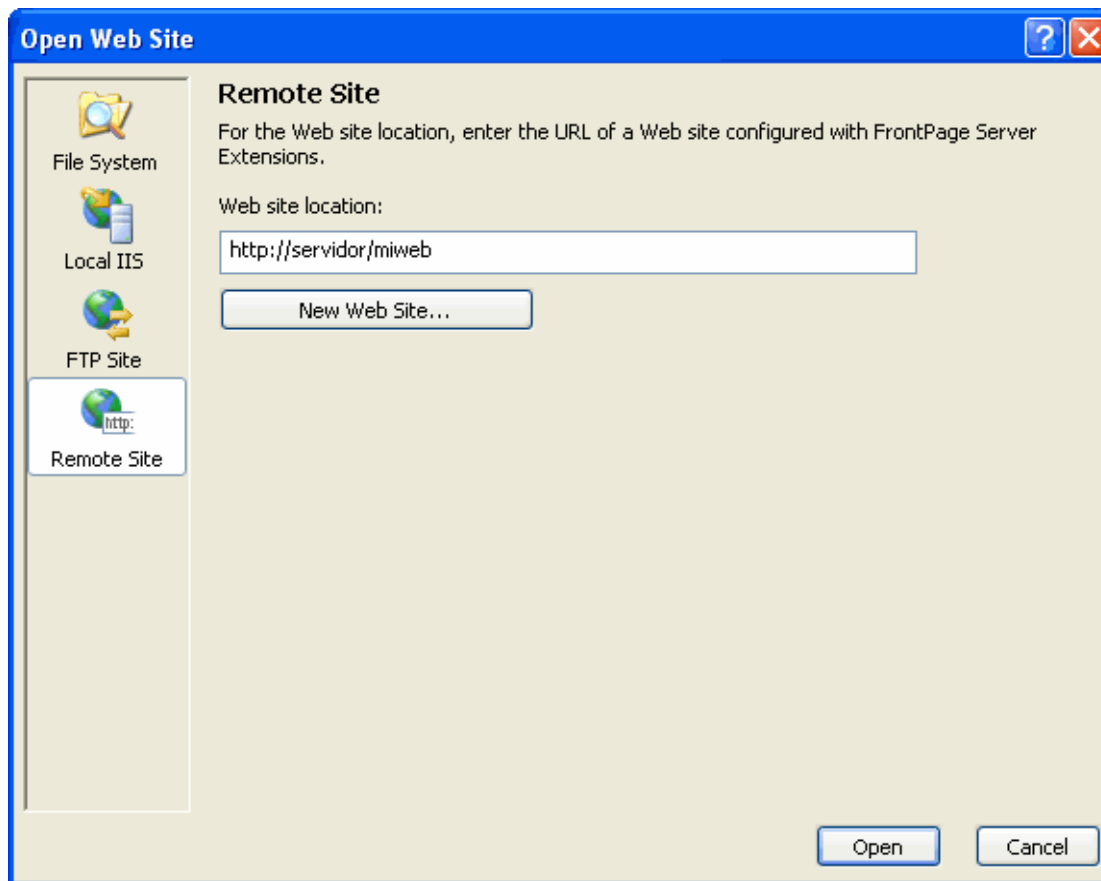


Se nos mostrará esta pantalla:

## IIS Avanzado. FrontPage. ASP.NET 3.5

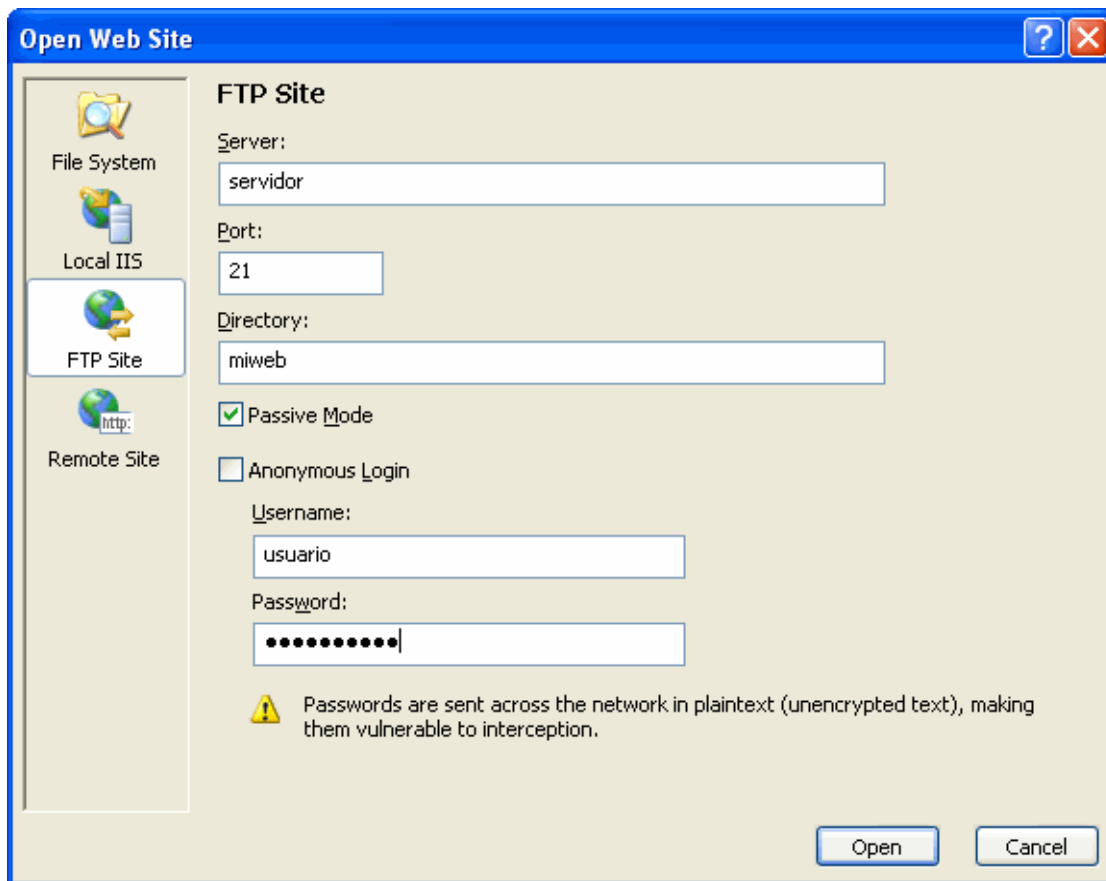


A la izquierda tenemos nuestro web, que en este caso se creó en una carpeta del disco duro y a la derecha el sitio donde debemos copiarlo. Primero debemos conectarnos al destino pulsando el botón de arriba de "Connect":



Elegiremos la forma de conexión adecuada:

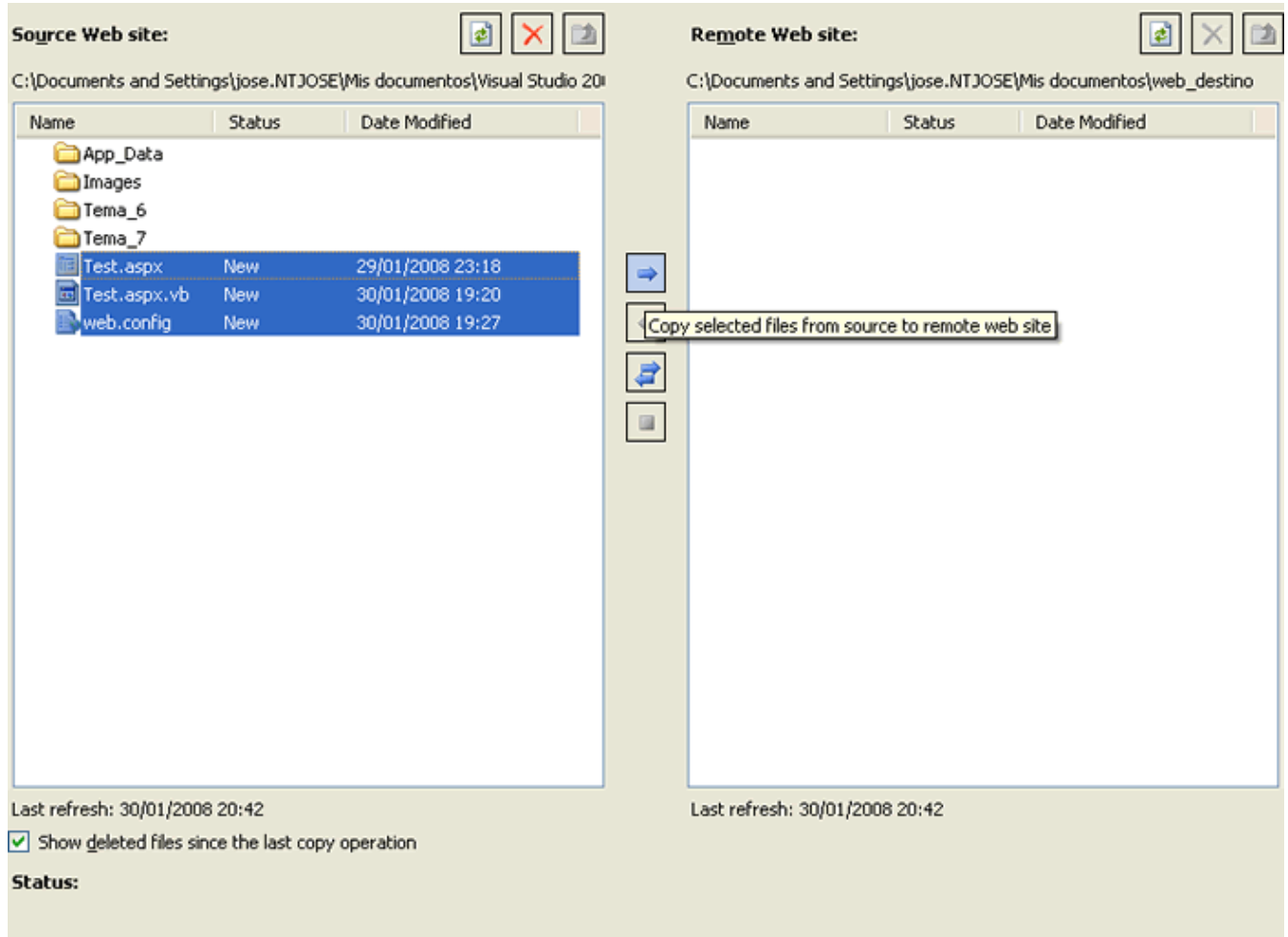
- Si elegimos File System necesitaremos indicarle simplemente la carpeta destino. Si tenemos mapeada en una unidad la ruta del servidor será una forma sencilla
- "Local IIS" lo instalará en el servidor web IIS de nuestro equipo. Esto es útil si hemos estado desarrollando en el propio Windows Server, que no es habitual. Esta opción la utilizaremos poco ya que es muy poco común.
- FTP Site. Copia los ficheros mediante el protocolo de transferencia de ficheros FTP:



Los datos que necesita para conectarnos son los de un servidor con el servicio FTP habilitado, una carpeta como destino y los "credenciales" para poder copiar. Normalmente los sitios FTP no son de escritura por eso tendremos que disponer de una cuenta con acceso de escritura. Este sistema es muy comun cuando tenemos nuestro web alojado fuera de nuestra empresa. Por ejemplo, realizamos los desarrollos en local o en nuestra empresa pero el servidor de explotación está en un proveedor de Internet. Éste nos habrá comunicado nuestra cuenta de FTP para que podamos transferir allí el web y los sucesivos cambios que realicemos en él.

- Remote Site. Es la opción mas habitual si tenemos un IIS funcionando en nuestra web, nos podremos conectar a él para transferir con en esta utilidad las páginas.

Vamos a realizar una conexión de prueba, lo haré con un destino como directorio para probar si funciona.



El funcionamiento es muy sencillo ya que simplemente seleccionaremos las partes que queremos mover y pulsaremos en la flecha de la dirección que queremos mover. Ni que decir tiene que es muy conveniente que hagas copia de seguridad antes de este proceso. Nunca se sabe, igual sin querer sobrescribes precisamente la página que quieres actualizar, recuerda que las leyes de Murphy son implacables.

# Ejercicios

## Ejercicio 1

Localizar los archivos de registro de actividad y comprobar los accesos realizados. Cambiar a registro semanal

Nota: ver el punto (1.1)

## Ejercicio 2

Localizar las páginas web de los errores y personalizar la página 404b.htm. Escribir una ruta incorrecta en el servidor y comprobar el resultado



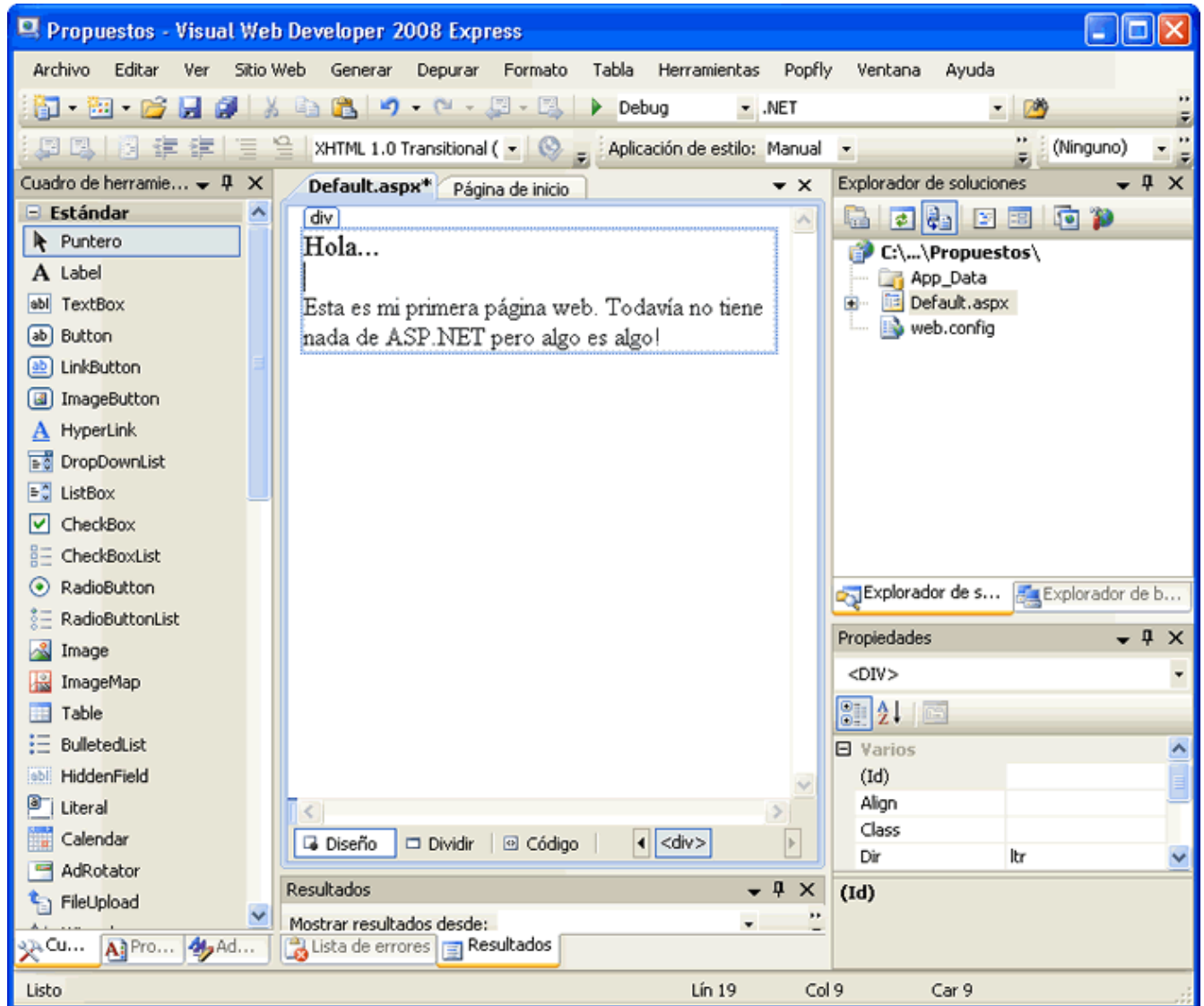
## Ejercicio 3

Si tienes FrontPage abre la página de inicio del servidor de Internet Information Server y modifica el contenido de la página de inicio predeterminada. Para esto asegurate de haber instalado las extensiones de servidor en el servidor de IIS

## Ejercicio 4

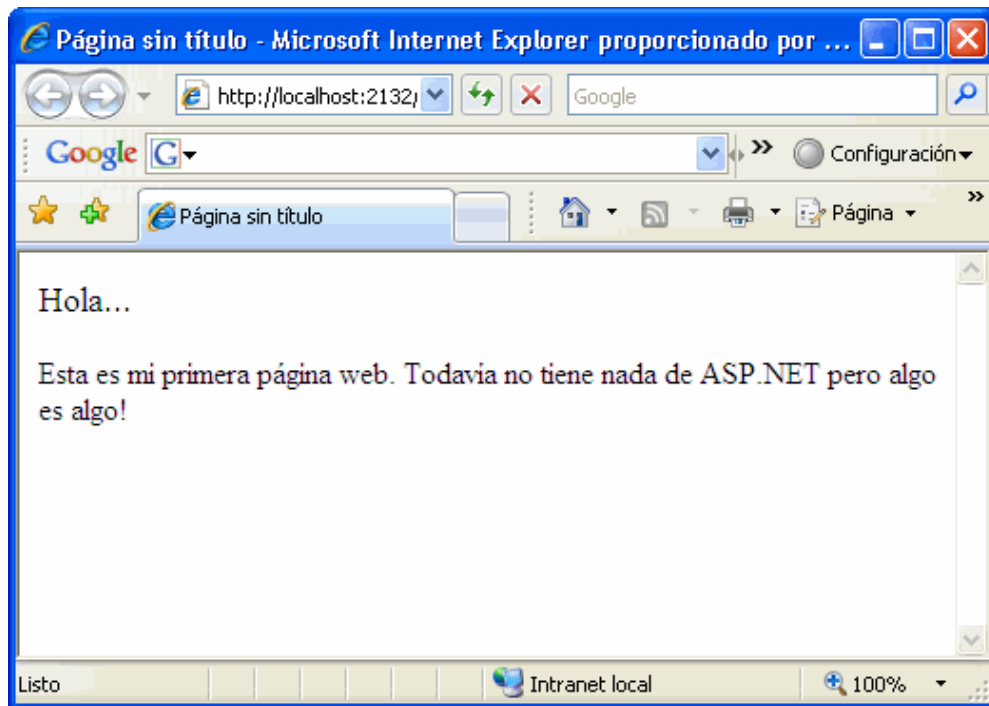
Crea un nuevo sitio web con el Visual Web Developer. Este será nuestro web para el resto de las lecciones así que organizaremos las lecciones en carpetas.

Pon la vista de diseño y escribe algun texto en pantalla:



Ejecuta la página habilitando todo lo necesario para que la depuración esté activa y comprueba que se ve en el navegador:





Nota, no se adjuntan solución a estos ejercicios porque no es mas que seguir los pasos del tema. Lo mas importante es que hagáis este ultimo ejercicio siguiendo al detalle el punto 4 del tema. Debemos hacer funcionar esta página porque así comprobamos que está todo preparado para empezar a hacer páginas ASP.NET