

Imágenes de tema, URL, color de SVG y el selector de imágenes del diseñador

Guía de la lección · Guía del laboratorio / examen

● LearnWisej

Guía de la lección

Orígenes de imagen que no son archivos que despliegas junto a la página: imágenes de tema con nombre cargadas desde la caché de temas del cliente, URL relativas y absolutas, datos base64 incrustados en el tema, el Image Selector de Visual Studio que escribe por ti la cadena de origen, y el sufijo ?color= que recolorea un SVG monocromo adecuado con un nombre de color del tema. Qué significa esto para un desarrollador que conoce C# básico y ya ha trabajado algo con los controles y el diseñador de Wisej.NET — y cómo leer este módulo.



Enfoque de producción

Orígenes de imagen que no son archivos que despliegas junto a la página: imágenes de tema con nombre cargadas desde la caché de temas del cliente, URL relativas y absolutas, datos base64 incrustados en el tema, el Image Selector de Visual Studio que escribe por ti la cadena de origen, y el sufijo ?color= que recolorea un SVG monocromo adecuado con un nombre de color del tema.

1. El problema en producción

IconDesk ya tiene un panel de comandos y un laboratorio de imágenes, y todas sus imágenes son un archivo desplegado más una ruta que alguien escribió. Eso funciona hasta que la aplicación cambia de tema, se despliega bajo un directorio virtual distinto o tiene que mostrar el mismo símbolo en color de advertencia en una pantalla y en un color neutro en otra. Este módulo saca el origen de una imagen de la carpeta del proyecto y lo lleva al tema, a una URL o al diseñador, para que la cadena que lleva un control siga significando algo después de que cambien la ilustración, el tema o el servidor.

2. Conceptos clave

Concepto	Por qué importa
imagen de tema con nombre	Un nombre semántico como <code>icon-print</code> que el tema activo asocia a una ilustración, de modo que otro tema puede aportar otra imagen sin ningún cambio en C#.
caché de temas del cliente	Las imágenes de tema viajan con el tema y ya están en el navegador cuando un control pide una, lo que convierte al tema en el hogar adecuado para un conjunto pequeño de símbolos de comando y en el hogar equivocado para miles de ellos.
origen con URL relativa	Una cadena de origen de imagen resuelta con respecto a la aplicación: la forma más sencilla para un logotipo desplegado con la aplicación.
origen con URL absoluta	Una dirección externa completa que convierte la imagen en una dependencia de otro sistema, con consecuencias de disponibilidad, privacidad, CORS y Content Security Policy.
imagen base64 incrustada	Los bytes de la imagen escritos dentro de la propia definición del tema, de modo que la ilustración no necesita una segunda petición, a costa de un tema más pesado.
Image Selector	El selector de tiempo de diseño de Visual Studio que explora las imágenes del proyecto, las del tema y los recursos de los packs de iconos referenciados, y después serializa la cadena de origen elegida en <code>Designer.cs</code> .
sufijo <code>?color=</code>	Un sufijo en un origen de imagen que le pide a Wisej.NET que recoloree un SVG monocromo compatible con un color CSS o un nombre de color del tema.

3. Desarrollo de la lección

Cuatro formas que puede adoptar una imagen dentro de un tema

Un tema de Wisej.NET no son solo colores y bordes; también lleva definiciones de imagen que pueden usar tanto el framework como tu aplicación. La documentación del Theme Builder describe cuatro formas útiles para esas definiciones. Una **URL relativa** se resuelve con respecto a la aplicación. Una **URL absoluta** nombra una dirección externa completa. Los datos **base64 incrustados** ponen los bytes de la imagen dentro de la definición del tema, así que no hay que obtener nada más. Una **imagen con nombre** apunta a una definición de imagen, lo que añade una capa de indirección: muchas apariencias pueden referenciar el mismo nombre, y cambiar la definición una vez cambia todos los usos.

La indirección es la parte interesante para IconDesk. Cuando la galería pide un nombre en lugar de un archivo, la decisión sobre qué ilustración significa ese nombre pertenece al tema, y el tema se puede cambiar en el despliegue o por cliente. Cuando la página pide una URL, la decisión pertenece a quien escribió la URL, y cambiar de tema no cambia nada. El laboratorio pone ambos comportamientos en una misma pantalla para que la diferencia no sea una cuestión de fe.

Las URL absolutas merecen una decisión deliberada y no una costumbre. Una imagen obtenida de otro servidor es una dependencia en tiempo de ejecución: puede no estar disponible, puede exigir una autenticación que el navegador no tiene, puede estar bloqueada por una Content Security Policy y su

política de caché no es tuya. Un icono que siempre debe estar presente es mejor desplegarlo con la aplicación o empaquetarlo como recurso, que es a donde va el Módulo 5.

El nombre de una imagen de tema es una clave, no una ruta

Asignar una imagen de tema es la línea más corta del curso: establece una propiedad `ImageSource` con el nombre y Wisej.NET la carga desde la caché de temas del cliente. Nada en el servidor abre un archivo y no se crea ningún `System.Drawing.Image`, así que esta es la segunda canalización del Módulo 1, no la primera. La cadena es una clave de tema. Si otro tema asocia esa clave a otra ilustración, la aplicación cambia de aspecto sin una sola edición en C#, y por eso los nombres semánticos como `icon-print`, `icon-save` o `icon-warning` ganan a una ruta pegada.

Ese comportamiento de precarga es también una regla de dimensionamiento. Las imágenes de tema llegan con el tema, así que un tema encaja con un conjunto pequeño y estable de símbolos de comando del nivel del framework. Los recursos de un pack de iconos se comportan al revés: cada icono se solicita de forma individual y después lo cachea el navegador, lo que escala a catálogos enormes pero hace que el primer uso de cada icono cueste una petición. Saber cuál de los dos estás usando te dice qué aspecto debería tener el panel de red, y esa es la prueba que el Módulo 4 te pide recoger.

```
// IconGallery: three source strings, three different resolutions.
buttonPrint.ImageSource = "icon-print";           // theme key
logo.ImageSource        = "Images/company-logo.svg"; // relative URL
avatar.ImageSource      = "https://cdn.example.com/users/42.jpg"; // absolute

// Runtime choices stay semantic, never assembled from user text.
buttonStatus.ImageSource = isDanger
    ? "icon-warning?color=error"
    : "icon-info?color=activeText";
```

Deja que el Image Selector escriba la cadena de origen

El diseñador de Wisej.NET en Visual Studio ofrece una experiencia de selección de imágenes para las propiedades de origen. Explora las imágenes del proyecto, las que define el tema actual y, cuando hay ensamblados de packs de iconos referenciados, los packs que descubre automáticamente. El explorador de packs de iconos muestra una lista o cuadrícula con vista previa, y un icono monocromo se puede previsualizar en el color elegido antes de confirmarlo. Cuando eliges uno, el diseñador serializa un origen concreto, como una URL `resource.wx/...`, en `Designer.cs`.

Esto es más que una ayuda para escribir. Leer la línea generada enseña la representación en tiempo de ejecución: elegir un icono de un pack **no** copia el SVG dentro del formulario. El formulario guarda una cadena de origen que apunta de vuelta al ensamblado del pack, y por eso cinco formularios que usen un mismo icono siguen distribuyendo una única copia de la ilustración. El laboratorio te pide abrir `Designer.cs` y leer esas cadenas exactamente por ese motivo.

- Imágenes del proyecto: elígelas con el selector para no escribir una ruta dos veces.
- Imágenes de tema: usa la lista del tema y la vista previa para ver la ilustración del tema actual.
- Recursos de packs de iconos: busca y previsualiza en lugar de memorizar nombres de archivo.

Recolorear un SVG monocromo con ?color=

Wisej.NET puede recolorear un SVG monocromo compatible añadiendo `?color=` al origen de imagen. El valor puede ser un color al estilo CSS como `red` o `#3078d4`, o el nombre de un color del tema como `error` o `activeText`. Un color del tema es la opción más sólida porque sigue teniendo sentido cuando cambia el tema: un gris oscuro fijado en el código puede desaparecer sobre un tema oscuro, mientras que un color semántico del tema está pensado para adaptarse con él. El mismo sufijo funciona con el nombre de una imagen de tema y con un origen de recurso `resource.wx`, porque se aplica a la cadena de origen y no a un mecanismo de almacenamiento concreto.

El recoloreado no es universal, y el fallo es silencioso. La documentación advierte de que puede fallar cuando el propio SVG fija valores de `fill` o `stroke`, porque esos valores ganan. Para un icono pensado para que lo coloree Wisej.NET, quita los colores fijados que entren en conflicto cuando sea apropiado; para un logotipo multicolor, déjalo tal cual. La galería de IconDesk debería tener uno de cada para que la nota de laboratorio pueda decir qué ilustración aceptó el sufijo y cuál lo ignoró. Mantén semánticas las cadenas de origen en tiempo de ejecución y no construyas nunca una ruta de recurso a partir de texto de usuario no confiable.



Nombra primero el mecanismo

Antes de asignar cualquier imagen, di qué mecanismo estás usando: un objeto `System.Drawing.Image` que el servidor renderiza como PNG, una cadena de origen de imagen que resuelve el navegador, una imagen de tema con nombre, un recurso incrustado servido por `resource.wx` o una fuente de iconos dibujada como HTML. Asigna una sola propiedad de imagen por control para que las demás no puedan restablecerla en silencio, mantén la cadena de origen con significado en lugar de una ruta mágica pegada, prefiere SVG para los iconos de interfaz para que se mantengan nítidos en cualquier tamaño y escala, y dale una alternativa textual a todo icono que transmita significado.



Objetivos de aprendizaje

Al terminar este módulo serás capaz de:

- ✓ Usar URL relativas, URL absolutas, datos base64 incrustados e imágenes con nombre en un tema de Wisej.NET.
- ✓ Asignar nombres de imágenes de tema directamente a `ImageSource`.
- ✓ Usar el selector de imágenes de tiempo de diseño de Visual Studio para seleccionar recursos locales, de tema y de packs de iconos.
- ✓ Recolorear los iconos SVG adecuados con la sintaxis `?color=` de Wisej.NET y los nombres de color del tema.

Guía del laboratorio / examen

Lo que construirás en el laboratorio práctico, el enfoque sugerido y los entregables requeridos.



Laboratorio práctico

Rellena una página IconGallery en IconDesk usando el Image Selector del diseñador en lugar de escribir las cadenas de origen a mano. Asigna imágenes a cinco controles: una imagen de tema con nombre, un SVG del proyecto referenciado por URL relativa, una URL absoluta y al menos dos recursos de un pack de iconos si hay un pack instalado. Recolorea un SVG monocromo con el sufijo `?color=` y un nombre de color del tema, abre el archivo `Designer.cs` generado y lee las cadenas de origen de imagen que produjo el selector, después cambia la aplicación entre dos temas de Wisej y anota qué recursos cambiaron y cuáles se quedaron exactamente igual.

Punto de partida

Abre la solución IconDesk tal como la dejó el Módulo 2: el panel de comandos del Módulo 1 y la página ImageLab con su cuadrícula de modos de tamaño, su PictureBox con `LoadAsync` y su SVG nítido. Añade una nueva página llamada `IconGallery` con cinco controles vacíos y un selector de tema, para que todas las imágenes que haya en ella se asignen durante este laboratorio y no se herede nada de un módulo anterior.

Enfoque sugerido

- ✓ Coloca primero los cinco controles de la galería y dale a cada uno una etiqueta que nombre el mecanismo que piensas usar, antes de asignar ninguna imagen.
- ✓ Asigna todas las imágenes con el Image Selector del diseñador: la lista del tema para la imagen de tema con nombre, el árbol del proyecto para el SVG relativo, el campo de dirección para la URL absoluta y el explorador de packs de iconos para los recursos del pack.
- ✓ Añade el sufijo `?color=` con un nombre de color del tema a un SVG monocromo, y pruébalo también en un recurso multicolor para poder ver cómo se ignora el sufijo.
- ✓ Abre `Designer.cs`, copia en la nota de laboratorio las líneas de origen de imagen generadas y marca cada una como clave de tema, URL relativa, URL absoluta u origen de recurso.
- ✓ Cambia la aplicación entre dos temas de Wisej, captura la galería con los dos y anota qué controles siguieron al tema y cuáles no.

Entregables requeridos

- ✓ Página IconGallery con cinco controles cuyas imágenes se asignaron con el Image Selector del diseñador
- ✓ Entre ellas, una imagen de tema con nombre, un SVG relativo del proyecto, una URL absoluta y al menos dos recursos de un pack de iconos
- ✓ SVG monocromo recoloreado con el sufijo `?color=` y un nombre de color del tema
- ✓ Extracto de `Designer.cs` que muestre las cadenas de origen de imagen generadas por el selector

- ✓ Nota del cambio de tema que recoja qué recursos siguieron al tema y cuáles se mantuvieron fijos

Preguntas de repaso

- ✓ La galería se renderiza correctamente en tu máquina y no muestra nada para la URL absoluta en la de un compañero. ¿Cuál de los cinco mecanismos podría producir esa diferencia y qué comprobarías primero?
- ✓ Dos de tus iconos se niegan a cambiar de color cuando añades el sufijo. ¿Qué hay dentro de esos archivos y cuándo editarlos es la respuesta equivocada?
- ✓ Un diseñador pide el mismo símbolo en la galería, en una barra de herramientas y en una celda de cuadrícula. ¿Qué cuesta menos cambiar después, la clave de tema, la URL relativa o el recurso del pack, y por qué?



Nombra primero el mecanismo

Antes de asignar cualquier imagen, di qué mecanismo estás usando: un objeto `System.Drawing.Image` que el servidor renderiza como PNG, una cadena de origen de imagen que resuelve el navegador, una imagen de tema con nombre, un recurso incrustado servido por `resource.wx` o una fuente de iconos dibujada como HTML. Asigna una sola propiedad de imagen por control para que las demás no puedan restablecerla en silencio, mantén la cadena de origen con significado en lugar de una ruta mágica pegada, prefiere SVG para los iconos de interfaz para que se mantengan nítidos en cualquier tamaño y escala, y dale una alternativa textual a todo icono que transmita significado.