



Escuela de Sistemas y Tecnologías

Transparencias de ANALISTA DE SISTEMAS
Edición 2017- Materia: Diseño de Aplicaciones Web

TEMA: Mejoras en Diseño Web

Plantel y Contactos

➤ **Bedelía:**

- Mail: bedeliasistemas@bios.edu.uy

➤ **Encargado de Sucursal:**

- Pablo Castaño
- Mail: pablocasta@bios.edu.uy

Recursos

➤ Recursos Imprescindibles:

- Sitio Web de material
(comunicarse con Bedelía por usuario/contraseña).
- Transparencias del Curso.
- Contar con el software necesario

Agenda

- ☐ MasterPages:
- ☐ Themes y Skins

MasterPages

Definicion

- Logran herencia visual para las páginas Web.
- Permiten manejar áreas comunes de un sitio de manera consistente.
- Se basan en templates (MasterPage) y en páginas de contenido (ContentPage).
- Permiten incluir menús, encabezados, navegaciones, etc.

Páginas Maestras

- En lugar de la directiva **@Page**, utiliza la directiva **@Master**:

```
<%@Master Language="C#"
        CodeFile="MasterPage.master.cs"
        Inherits="MasterPage" %>
```

- Se trata como cualquier formulario web, con la extensión **.master**, y debe incluir el siguiente control:

```
<asp:contentplaceholder id="contenedor"
                        runat="server">
    contenido por defecto
</asp:contentplaceholder>
```

Paginas de Contenido

- Al crear un página de contenido, se elije la página maestra. La directiva **@Page** de la página sería:

```
<%@Page Language="C#"
    MasterPageFile="~/MasterPage.master"
    CodeFile="Default.aspx.cs"
    Inherits="_Default"
    Title="Home" %>
```

- No repiten los tags de la página maestra, pero se pueden acceder desde el código.
- Debe contener el control:

```
<asp:content ID="cntDefault"
    ContentPlaceHolderID="contenedor"
    Runat="Server">
    contenido de la página
</asp:content>
```


Web.Config

- Se puede configurar que todas las páginas hereden visualmente de una MasterPage específica.

```
<configuration>
  <system.web>
    <pages masterPageFile="~/MP.master"/>
  </system.web>
</configuration>
```

Acceso a objeto MasterPage

- Se define el objeto **MasterPage**:

```
<%@Page Language="C#"
    MasterPageFile="~/MP.master" %>
```

- Se usa en código el objeto **MasterPage**

```
void Page_Load(Object sender, EventArgs e)
{
    MasterPage.Title = "Título colocado
    desde la página de contenido
    codificando el objeto Master";
}
```

Anidar MasterPages

- Las páginas maestras se pueden anidar, ya que una página maestra puede hacer referencia a otra, que actúe como su página maestra principal.
- Las páginas maestras anidadas permiten crear páginas maestras divididas en componentes.

```
<%@Master Master="~/MPMain.master" %>
```

Navegador

- Permite que las páginas se programen de acuerdo a las características de cada navegador: Mozilla, Netscape , Nokia, Openwave, Opera, Palm, Panasonic, Pie, Webtv, Xiino:

```
<%@Page Language="C#"
```

```
    MasterPageFile="~/MG.master"
```

```
Mozilla:MasterPageFile="~/MGMozilla.master"
```

```
Opera:MasterPageFile="~/MGOpera.master" %>
```

Themes y Skins

Definicion

- **Skins:**
 - Son definiciones de formato y estilos que se aplican a los controles de servidor y se guardan en archivos de extensión **.skin**
- **Themes:**
 - Son “paquetes” de skins (alojados en carpetas). También pueden contener hojas de estilo en cascada (CSS) e imágenes asociados.

Ventajas de Uso

- La creación y utilización de Temas, es ventajoso en el diseño de sitios web:
 - se planifican colores y formas al comienzo del diseño
 - se los determina una sola vez
 - se aplican automáticamente a todos los elementos que se deseen dentro del sitio

Creación

- Agregar la carpeta *App_Themes* (en el sitio web).
- Crear una carpeta con el nombre del tema (dentro de la carpeta *App_Themes*).
- Incluir:
 - Archivos **.skin**
 - Archivos **.css**
 - Imágenes
 - Asignar las imágenes en los skins

Configuración (1)

- A partir de ASP.NET 2.0 se incluye una nueva carpeta virtual para la organización de los temas (*APP_Themes*).
- Se pueden configurar a nivel de:
 - Máquina (en el **machine.config**)
 - Aplicación (en el **web.config**)
 - Página (dentro de la directiva **@Page**)

Configuración (2)

- A nivel máximo (**machine.config**) :

```
<pages buffer="true"
        theme="UruguayNatural" >
    ...
</pages>
```

- A nivel de sitio (**web.config**):

```
<?xml version="1.0" encoding="UTF-8" ?>
<configuration>
    <system.web>
        <pages theme="UruguayNatural" />
    </system.web>
</configuration>
```

- A nivel de página (**@Page**):

```
<%@Page Language="C#" Theme="UruguayNatural" %>
```

Deshabilitar Themes

- La directiva **EnableTheming** permite manejar el bloqueo parcial de temas (si se aplica la definición al control) y puede ser utilizada como un atributo en un control:

```
<%@Page Language="C#" EnableTheming="False" %>
```

```
<asp:Textbox ID="TextBox1"  
             Runat="server"  
             BackColor="#000000"  
             ForeColor="#ffffff"  
             EnableTheming="false" />
```

Skins

- Los archivos **.skin** contienen declaraciones de estilo y formato de los controles de ASP.NET.
- En los siguientes ejemplos se muestra como se codifican dichas declaraciones:

```
<asp:LoginName runat="server"
    BorderWidth="1"
    BorderColor="#FF9900"
    ForeColor="navy"
    Font-Names="verdana" />
```

```
<asp:Label runat="server"
    ForeColor="#004000"
    Font-Names="Verdana"
    Font-Size="X-Small" />
```