# Laboratorio 05: Master Pages, Themes & Skins

#### Objetivos

Luego de finalizado este lab, usted podrá: Crear un Master Page. Aplicar un Master Page a páginas existentes. Crear una nueva página basada en un Master Page.

## **Master Pages**

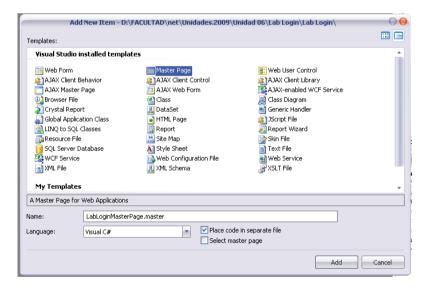
## 1. Ejercicio: Creando el Master Page

#### Pasos

- 1. Abra la solución del ejercicio Lab Login.
- 2. En la ventana Solution Explorer haga click derecho sobre el proyecto web Lab Login y luego elija Add New Item para agregar un Master Page que se llamará LabLoginMasterPage.master:
  - Template: Master Page
  - Name: LabLoginMasterPage.master

<%@ Master Language="C#" AutoEventWireup="true"</pre>

• Language: Visual C#



- 3. Presione Add.
- 4. Reemplace el código en LabLoginMasterPage.master por el siguiente:

```
<+r>
       Site Header
    <!-- site navigation -->
       Site Navigation
            <a href="Login.aspx">Login</a>
            <a href="Home.aspx">Home</a>
            <!-- this is the placeholder for the page content -->
    <!-- site footer -->
       Site Footer
    </form>
</body>
</html>
5. Ahora agregue el contenedor donde las páginas que utilicen este
  Master Page colocarán su contenido. Este contenedor es conocido como
  ContentPlaceHolder. Para agregar dicho contenedor al Master Page,
  reemplaze el código:
<!-- this is the placeholder for the page content -->
Por el siguiente:
<!-- this is the placeholder for the page content -->
  <asp:contentplaceholder id="PageContent" runat="server" />
```

## 2. Ejercicio: Aplicando el Master Page a WebForms Existentes

#### **Pasos**

Aplicando el MasterPage a la página Login.aspx

- 1. Para aplicar el Master Page a una página existente, haga doble click sobre el Web Form Login.aspx del proyecto Lab Login.
- 2. En la vista de código, con el objetivo de especificar cual será el Master Page a utilizar, reemplace el código:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Login.aspx.cs"
Inherits="Login" %>

Por el siguiente:
```

```
Page Language="C#" MasterPageFile="~/LabLoginMasterPage.master"
AutoEventWireup="true" CodeFile="Login.aspx.cs" Inherits="Login" %>
```

3. El contenido del WebForm deberá colocarse dentro de un tag </asp:Content> para ser incluido en el Master Page. Reemplaze el código:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"</pre>
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
   Por el siguiente:
<asp:Content ID="Content1" ContentPlaceHolderID="PageContent" Runat="Server">
Y el código:
   </form>
</body>
</html>
Por el siguiente:
</asp:Content>
```

- 4. Pase a la vista de diseño presionando el botón Design en la parte inferior izquierda del editor de código, o presione (SHIFT + F7)
- 5. En la vista de diseño, analice como el diseño del Master Page a afectado a la página Login.aspx.

Verifique que los sectores propios del Master Page no pueden ser editados (por ejemplo el menu de navegación).

Solo se tiene permitido agregar componentes al sector llamado Content, verifique que a ese sector puede agregar controles de ASPNet haciendo Drag & Drop desde el toolbox o bien escribir el texto que desee.

## 3. Ejercicio: Creando un WebForm a partir de un MasterPage

#### Pasos

1. Es posible asignarle un Master Page a una página nueva por agregar. Para probar esto, haga click derecho sobre el proyecto web Lab Login y seleccione Add New Item para agregar un nuevo formulario web que se llamará Home.aspx.

Asequrese que la opción Select Master Page esté seleccionada:

• Templates: Web Form

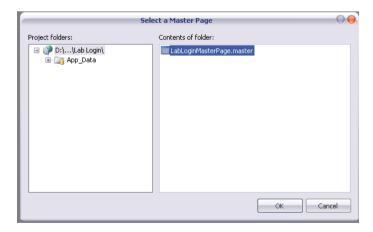
Name: Home.aspx

• Language: Visual C#

• Select Master Page: "Seleccionado"



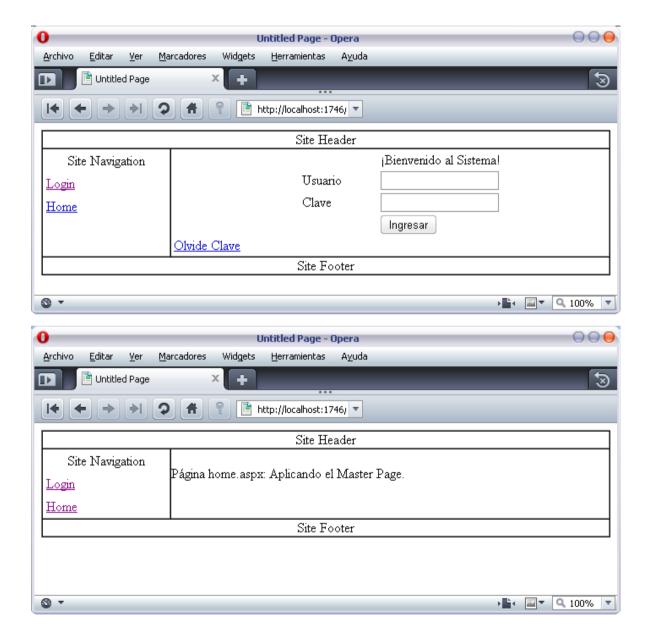
- 2. Presione Add.
- En el diálogo Select a Master Page, en la sección Contents of Folder, seleccione el Master Page LabLoginMasterPage.master creado anteriormente.



- 4. Haga click en OK.
- 5. Agregue el siguiente código al contenido del Web Form Home.aspx recién creado:
- <br /> Página home.aspx: Aplicando el Master Page.

#### Ejecutándo el Sitio Web

- 1. Haga click derecho sobre el proyecto Lab Login y seleccione Set as Startup Proyect.
- 2. Haga click derecho sobre el web form Login.aspx y seleccione Set as Start Page.
- 3. Del menu Build seleccione Build Solution...
- 4. Presione Ctrl+F5 para ejecutar el Web Site.
- 5. Verifique que la navegación a través del menu en el master page lo direcciona a los demás Web Forms creados y que sobre cada web Form se aplica el Master Page:



## **Temas & Skins**

Un tema es un conjunto de valores de propiedad que permiten definir la apariencia de las páginas y de los controles y, a continuación, aplicar esa apariencia de manera coherente a las páginas de una aplicación Web, en toda una aplicación Web o en todas las aplicaciones Web de un servidor.

Los temas están formados por un conjunto de elementos: máscaras (skins), hojas de estilos en cascada (CSS), imágenes y otros recursos. Como mínimo, un tema contendrá máscaras. Los temas se definen en directorios especiales en un sitio Web o un servidor Web.

## 4. Ejercicio: Crear un tema de página (theme)

#### Pasos

- 1. Cree una nueva carpeta denominada App\_Themes en su sitio Web. (La carpeta debe denominarse App Themes)
- 2. Cree una nueva subcarpeta de la carpeta App\_Themes para que contenga los archivos de tema.

#### **☑**Not.a

El nombre de la subcarpeta es el nombre del tema que decida crear. Por ejemplo, para crear un tema denominado Azul, cree una carpeta denominada \App Themes\Azul.

## 5. Ejercicio: Crear una máscara (skin)

#### Pasos

1. Agregue un nuevo item de tipo Skin en la subcarpeta de tema Azul que ha creado. Llamelo Button.skin.

#### **☑**Nota

La convención que se suele seguir es crear un archivo .skin por control, como Button.skin o Calendar.skin. No obstante, los archivos .skin pueden contener varias definiciones de máscaras.

2. En el archivo Button.skin, agregue una definición normal de control (con sintaxis declarativa), pero incluyendo únicamente las propiedades que vaya a establecer para el tema y sin incluir el atributo ID. La definición del control debe incluir el atributo runat="server". Agregue el siguiente código en Button.skin:

```
<asp:Button runat="server"
    BackColor="lightblue"
    ForeColor="Black"
    Font-Name="Arial"
    Font-Size="9px" />
<asp:Label Runat="server"
    Font-Bold="True"
    Font-Names="Garamond"
    Font-Overline="False"
    BackColor="LightGoldenrodYellow" />
```

## ☑Nota

Una forma cómoda de crear una máscara consiste en agregar el control a una página y configurarlo hasta conseguir el aspecto deseado. Por ejemplo, podría agregar un control Calendar a una página y establecer su encabezado de días, de fecha seleccionada y otras propiedades. A continuación, puede copiar la definición del control de la página a un archivo de máscara y quitar luego el atributo ID.

Repita los pasos 1 y 2 para cada máscara de controles que desee crear.

## ☑Nota

Sólo podrá definir una máscara predeterminada por control. Utilice el atributo SkinID en la declaración del control de la máscara para crear una máscara con nombre que pueda aplicar a las instancias concretas de un control.

## 6. Ejercicio: Aplicando un tema de página (theme)

#### Pasos

1. En el archivo web.config, dentro de las etiquetas <sistem.web>,
 escribir:

<pages Theme="Azul"/>

Esto aplicara el tema Azul a todas las páginas de la aplicación.