

Utilizando Themes no ASP.NET 2.0 – Parte 1

por **Ricardo Oneda**
21 de novembro de 2005

Introdução

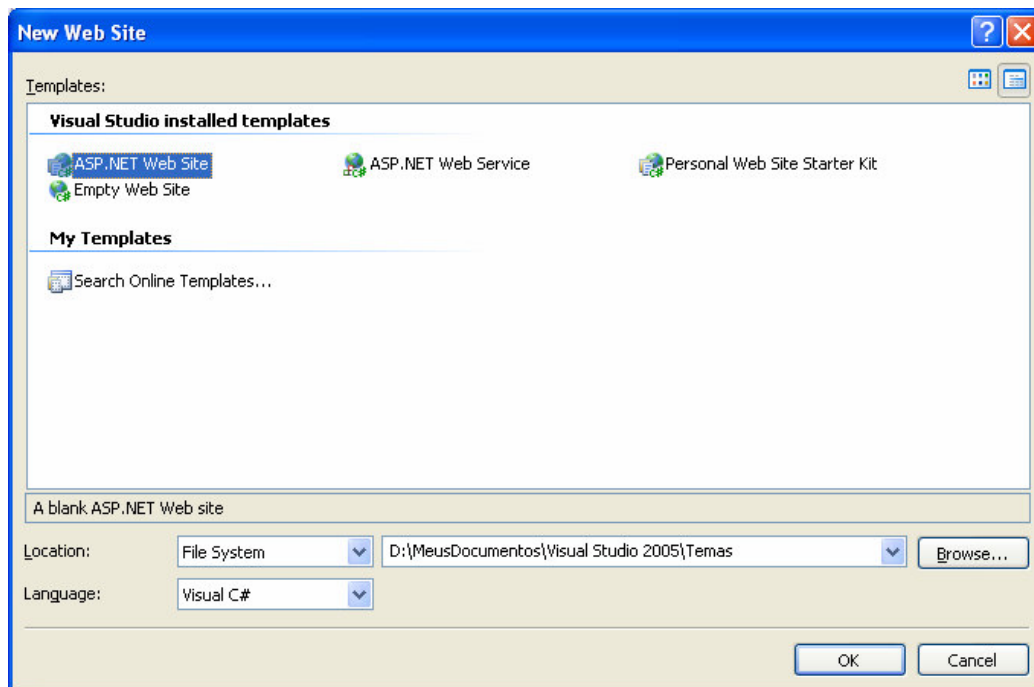
Themes, ou temas, são um novo recurso do ASP.NET 2.0 que permitem controlar a aparência de controles ASP.NET. Eles lembram as folhas de estilo, também conhecidas como **CSS – Cascade Style Sheet**, mas enquanto essas se aplicam a *tags* do HTML, os temas são aplicados em controles do lado servidor, inclusive em controles complexos, como o controle *GridView* ou o *Calendar*. Entre as vantagens de se utilizar temas está o fato de manter a aparência de sua aplicação consistente e ter controle centralizado sobre a mesma. Ao lado das **Master Pages**, os temas também auxiliam na manutenção da identidade visual de uma aplicação web.

Obs: este artigo é baseado na versão Beta 2 do .NET Framework 2.0/Visual Studio 2005. O conteúdo está sujeito a alterações até o lançamento da versão final dos produtos.

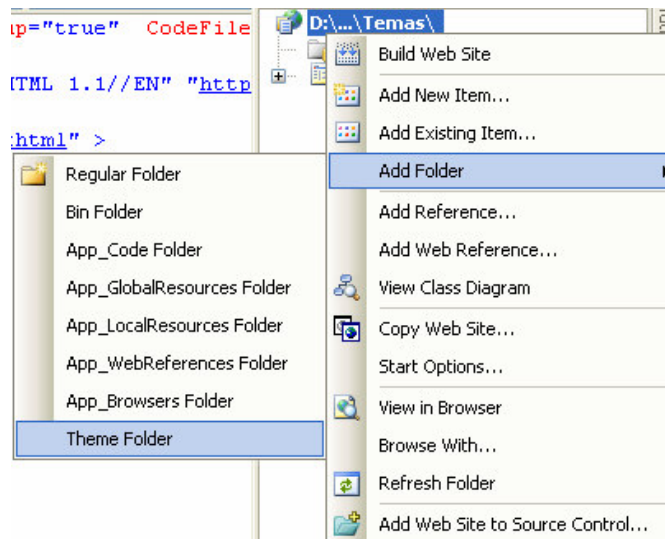
Themes e Skins

Um tema do ASP.NET 2.0 é um conjunto de *skins*, representados por arquivos com extensão **.skin** - nos quais estão as definições de aparência dos controles - localizados em uma pasta específica para os mesmos.

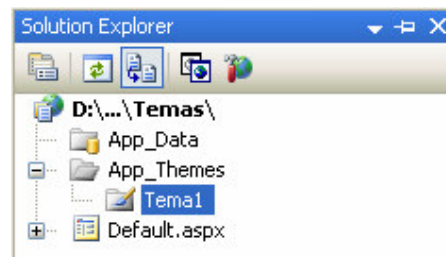
Veremos como definir um tema através de um exemplo. Abra o **Visual Studio 2005**, vá ao menu *File* → *New Web Site*. A janela *New Web Site* será apresentada. Escolha o *template ASP.NET Web Site* e em *Language* escolha *Visual C#*. O nome do *Web Site* será **Temas**. Escolha o diretório que lhe convier e clique em OK.



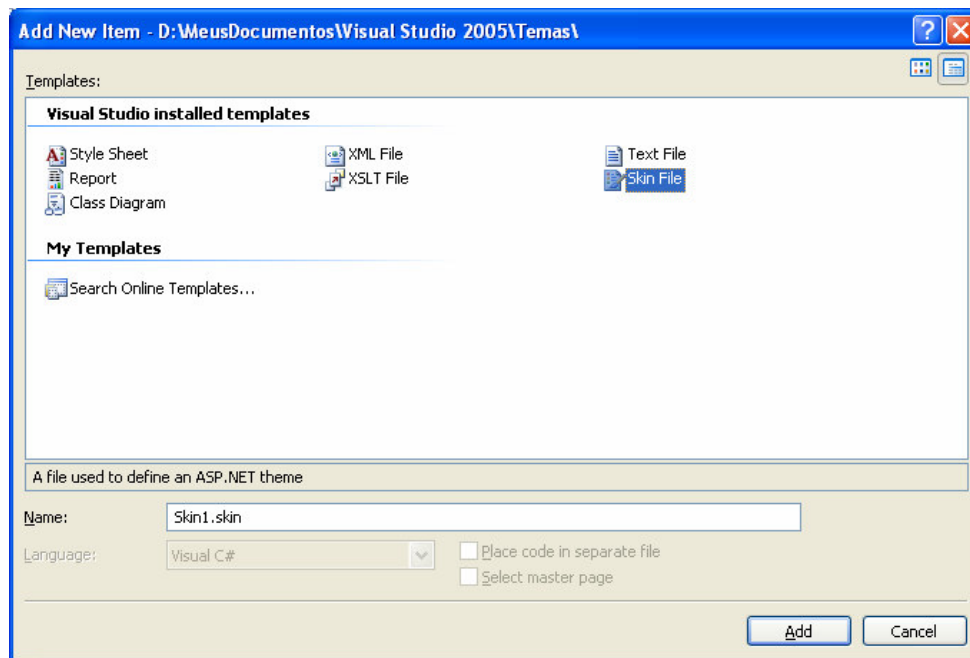
Exiba o *Solution Explorer*, indo ao menu *View* → *Solution Explorer*. Clique com o botão direito sobre o Web Site **Temas** e escolha a opção *Add Folder* → *Theme Folder*:



Perceba que foi criada uma pasta chamada **App_Themes**. É nela que o ASP.NET irá buscar os temas utilizados na aplicação. Note que dentro da pasta **App_Themes** foi criada outra pasta para o tema propriamente dito. Nomeie-a como **Tema1**. O *Solution Explorer* deverá se parecer com o da figura abaixo:



Dentro da pasta **Tema1**, iremos colocar todos os arquivos **.skin** do tema. Esta pasta pode conter vários arquivos **.skin**, além de arquivos **.css** e arquivos de imagem. Neste exemplo, só iremos criar um arquivo **.skin**. Ainda no *Solution Explorer*, clique com o botão direito sobre a pasta **Tema1** e escolha a opção *Add New Item*. Na janela que se abre, escolha *Skin File* e dê o nome **Skin1.skin**, conforme figura, e clique em *Add*:



No arquivo de *skin* que acabou de ser criado, adicione o seguinte código:

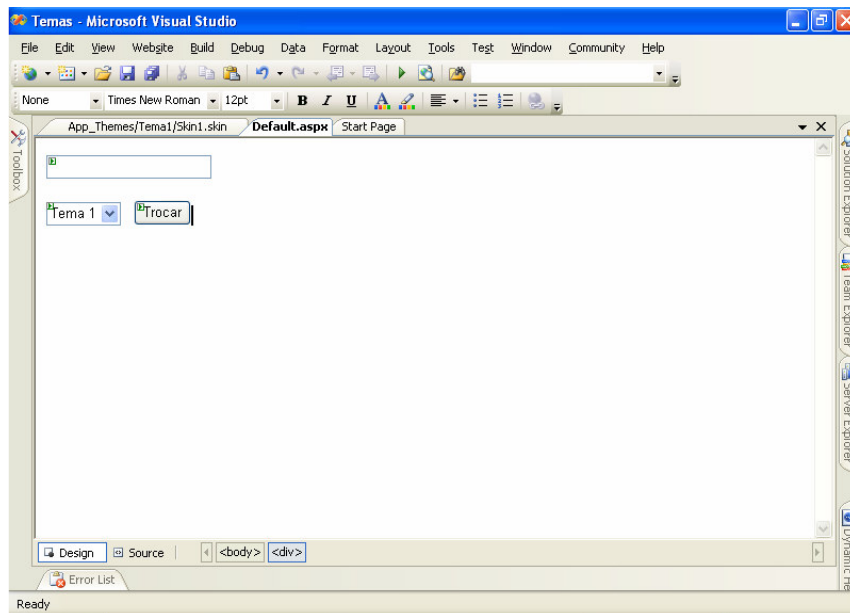
```
<asp:DropDownList BackColor="Green" ForeColor="white" Font-Bold="true" runat="server" />
<asp:TextBox BackColor="Yellow" Runat="Server" />
<asp:Button Font-Italic="True" BorderStyle="Outset" BorderWidth="5px" Runat="Server"/>
```

Acabamos de definir as aparências de controles do tipo *DropDownList*, *TextBox* e *Button*. Assim, qualquer um desses controles que estiver em nossa aplicação será formatado de acordo com o que foi definido: a propriedade *BackColor* define a cor de fundo, *ForeColor* define a cor da fonte, *Font-Bold* indica se o texto deve ou não estar em negrito, *Font-Italic* indica se o texto deve ou não estar em itálico, *BorderStyle* define o tipo da borda e *BorderWidth* define a largura dela.

No *Solution Explorer*, clique com o botão direito sobre a página **Default.aspx** e escolha a opção *View Designer*. Adicione um *TextBox*, *DropDownList* e um *Button* com as seguintes propriedades:

Controle	ID	Text	Items
TextBox	txtCaixa	-	
DropDownList	ddlTema	-	Tema 1, Tema 2
Button	btnTroca	Trocar	

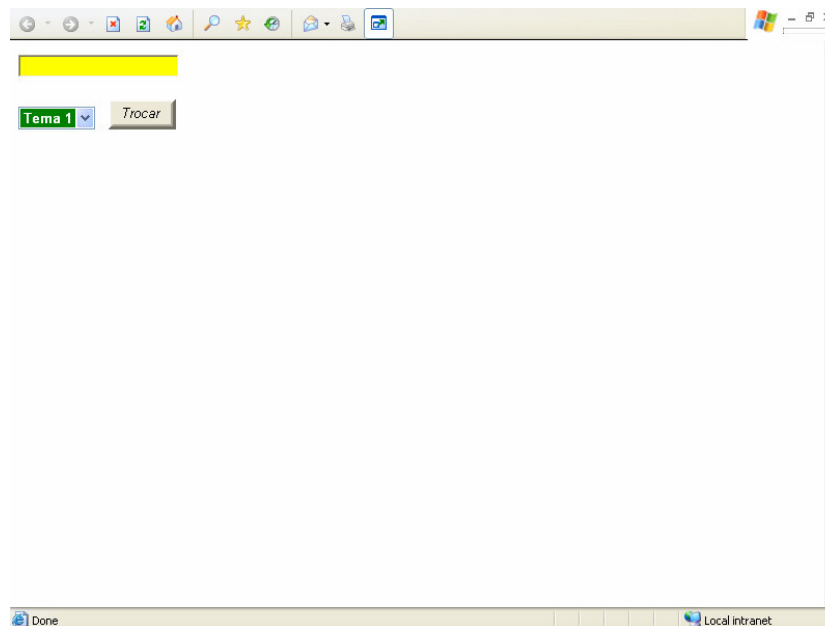
A página deverá se parecer com a da figura abaixo:



Exiba a janela de propriedades da página (tecla F4) e configure a propriedade *Theme* para **Tema1**. Visualize o código da página através do botão *Source* no rodapé da tela e note que foi incluído o atributo *Theme* na diretiva *@Page*:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" Theme="Tema1" %>
```

No *Solution Explorer*, clique com o botão direito sobre a página **Default.aspx** e escolha *View in Browser*. Você verá a página com o tema **Tema1** aplicado:



Definindo um Tema para toda a aplicação

Ao invés de definirmos um tema através do atributo *Theme* da diretiva *@Page*, podemos configurar qual tema a aplicação irá utilizar no arquivo de configuração **web.config**. O código abaixo, que deve ser colocado dentro da seção **<system.web>** do **web.config**, demonstra como isso pode ser feito:

```
<pages theme="Tema1" />
```

Neste caso, todas as páginas da aplicação irão se basear utilizar o tema **Tema1** definida no **web.config**.

Conclusão

Nesse artigo vimos o que são os **Themes** e como podemos utiliza-los para controlar a aparência dos controles do ASP.NET 2.0. Foi apresentada a maneira mais básica de se trabalhar com **Themes** e no próximo artigo veremos como podemos definir mais de um tema para um mesmo tipo de controle através dos **Named Skins**.

Referências

Código fonte do exemplo deste artigo

<http://www.msdnbrasil.com.br/sharepedia/visualizarartigo.aspx?id=62124>

Creating Web Application Themes in ASP.NET 2.0

<http://msdn.microsoft.com/asp.net/reference/ui/default.aspx?pull=/library/en-us/dnvs05/html/themes.asp>

ASP.NET Themes and Skins Overview

<http://msdn2.microsoft.com/en-us/library/ykzx33wh.aspx>

Sobre o Autor

Ricardo Oneda Pereira de Toledo é formado em Processamento de Dados pela Faculdade de Tecnologia de São Paulo, é *MCAD – Microsoft Certified Application Developer*, com ênfase em aplicações Web com C# e Banco de Dados SQL Server 2000 e é *MVP – Most Valuable Professional* em ASP/ASP.NET. Atualmente trabalha como Analista de Sistemas, desenvolvendo aplicações Web nas linguagens C e C++. Visite o meu blog em <http://thespoke.net/blogs/oned/default.aspx>