

ASP.NET 2.0 - Visual Studio 2005 (Whidbey)

Ramon Durães

O ASP.NET que é a camada de desenvolvimento de aplicações web da plataforma .NET vem agradando em muito os desenvolvedores de todo o mundo desde o lançamento do em julho de 2000 o . Logo em sequência a Microsoft atualizou para versão 1.1 . Para o desenvolvedor realmente foi uma mudança muito radical na maneira de desenvolver pois não tinha nada e ganhou um ambiente completo.

Agora no início de julho de 2004 a Microsoft disponibilizou publicamente o Beta1 do ASP.NET 2.0 em conjunto com Visual Studio 2005 denominado de 'Whidbey'. Neste artigo vamos nos focar no ASP.NET, porém não podemos deixar passar em branco as funcionalidades do Visual Studio Team System, que além das novas facilidades do ASP.NET vai trazer maior suporte para toda equipe de desenvolvimento, integrando mais ainda as etapas de desenvolvimento, incluindo inclusive ferramentas de testes internas da Microsoft e a metodologia de trabalho MSF 4.x (Microsoft Solution Framework). Quem participou do PDC 2004 em São Paulo já viu uma preview do que teremos de novidade. Confira na **Figura 1**.

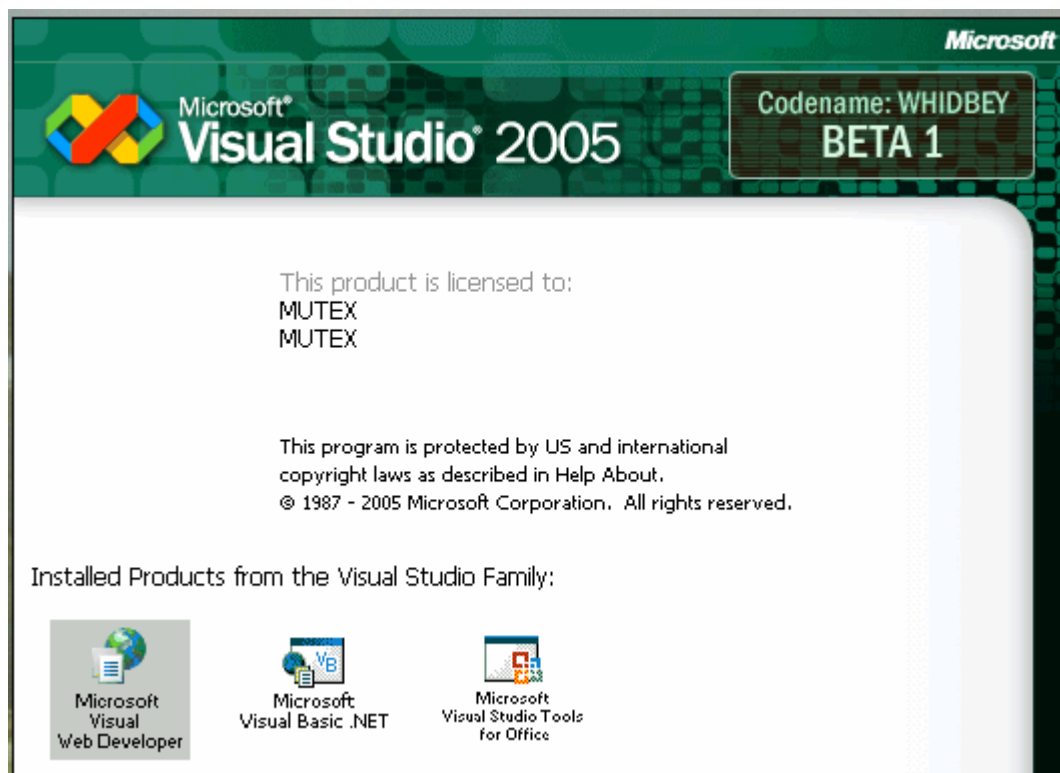


Figura 01 - Tela inicial do Visual Studio 2005 (Beta 1).

"A grande promessa desta nova versão está na otimização de muitas tarefas comuns, reduzindo em muito o trabalho com a diminuição da quantidade de linhas de código para executar as tarefas."

SERVIDOR WEB PRÓPRIO

A primeira coisa que você vai observar é que não é mais necessário possuir um servidor web IIS instalado na máquina para realizar os testes da aplicação. O Próprio Visual Studio já traz embutido um servidor que será executado toda vez que você acionar uma aplicação web pelo Visual Studio. Confira na Figura 2.



Figura 02 - Servidor Web próprio.

Funcionalidades Gerais

- Agora não é mais necessário ter arquivos de projetos, basta apontar uma pasta e abrir os arquivos
- Não é mais necessário possuir um única dll por aplicação, pode-se agora modificar uma única pagina, compilar e enviar, similar ao que se fazia com asp 3.0
- O Código html agora é preservado, não sendo mais modificado. (HTML Source Preservation)
- Intellisense sempre disponível (Intellisense Everywhere)
- Você pode navegar pelas tags HTML similar ao #Region (HTML TAG NAVIGATOR)
- Já trás duas aplicações completas de exemplo (Samples)

MASTER PAGE

A herança visual já está disponível nessa versão e é conhecida como "Master Page", consiste na funcionalidade de se criar uma pagina padrão ou modelo, e gerar todas outras paginas apartir dessa, resultando em um reaproveitamento enorme da parte visual, pois você só precisa implementar uma única vez. Confira na Figura 3 e na Figura 4.



Figura 03 - Pagina padrao (Default.Master).

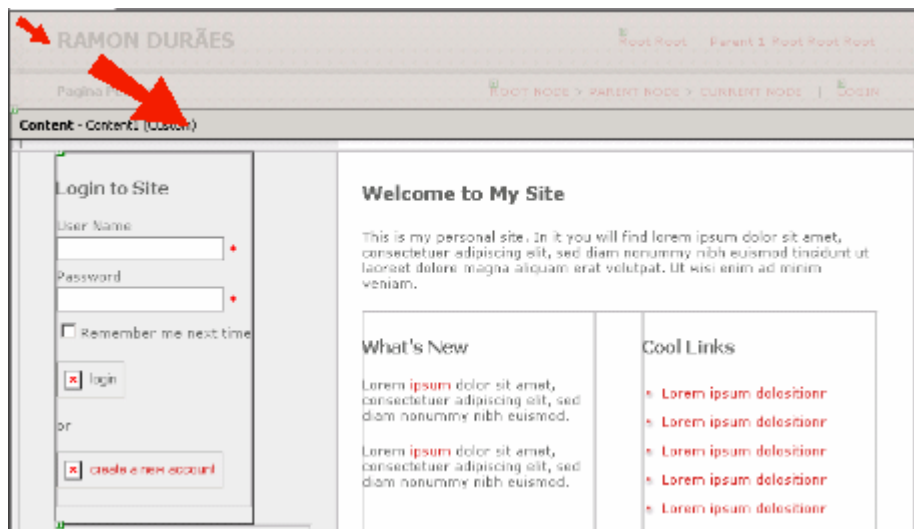


Figura 04 - Fazendo herança visual da pagina Master.

Passo a Passo (Wizard)

Para quem precisar criar telas de forma que os usuários coloquem os dados e siga avançando, agora já dispõem desse mecanismo. Confira nas Figuras 5 e 6.

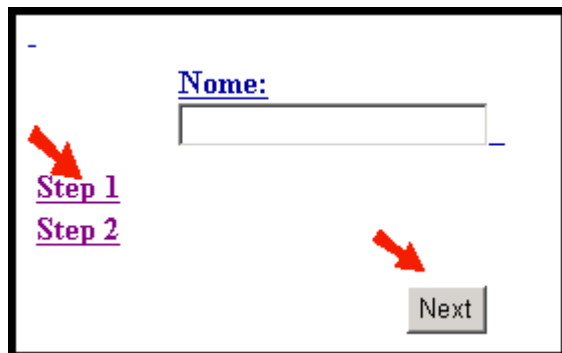


Figura 05 - Passo 1.

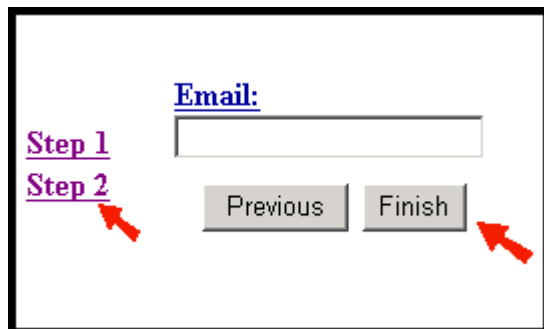


Figura 06 - Passo 2 e finalização.

DATA SOURCE

Para facilitar o acesso ao banco de dados foi criado um novo controle "Data Source", que veio para reduzir a zero a quantidade de linhas de código necessárias para acessar um banco de dados. Bastando informar a string de conexão de dados e a query e você já está com seu acesso a dados garantido. Confira na Figura 7. Um exemplo imediato você já pode observar no comando baixo:

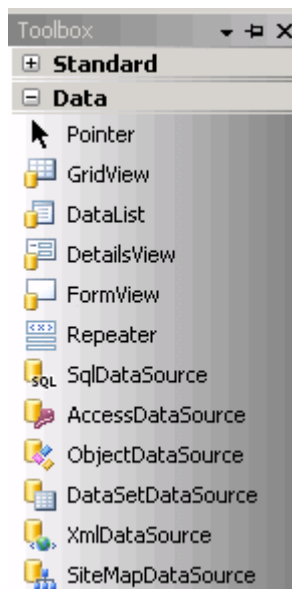


Figura 07 - DataSource Control.

Outro item importante é para quem gosta de implementar sua própria lógica de acesso a dados. Não se preocupe. Você dispõe do ObjectDataSource que vincula o DataSource a sua classe de dados.

GridView

Esse é o novo "DataGrid" e a principal mudança que você vai observar logo de cara é adicionar ele na página, configurar o DataSource, configurar para paginar, ordenar, editar sem adicionar uma única linha de código. E isso tudo quase que em 5 minutos!! Outra grande novidade é ordenar e paginar sem realizar "post" da página no servidor. Ele tem suporte a XMLHTTP de forma a realizar essa tarefa transparente para o programador. Outra nova funcionalidade é se ele encontra uma coluna do tipo Boolean, ele automaticamente já coloca um checkbox na tela. Ainda pode ser configurado em conjunto com o controle "DETAILSVIEW" para exibir detalhes de um registro do GridView. Confira a Figura 8 .

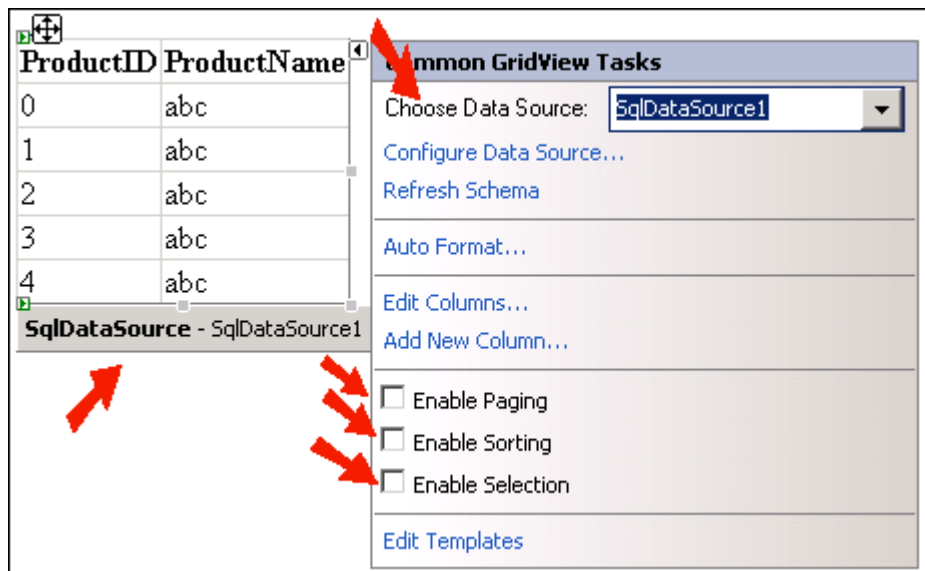


Figura 08 - GridView, ativando Paginação, Ordenação...

Interface de Administração

Um grande item que faltava era uma interface de administração, tínhamos que fazer tudo diretamente no arquivo web.config. Agora por meio dessa nova interface conforme **Figura 9**, Você vai poder realizar configuração de sua aplicação, e as informações serão salvas no web.config. Para chamar essa tela basta executar o arquivo **webadmin.axd** diretamente pelo browser na pasta de sua aplicação ou chamar pelo Visual Studio. Nesta tela você pode configurar o tipo de autenticação da sua aplicação, definir aonde serão salvos os usuários, e ainda conta com um sistema de estatística, isso mesmo. Vai ter um log de utilização da aplicação, inclusive com suporte a contar automaticamente quantidade de cliques em um determinado botão ou link por exemplo.

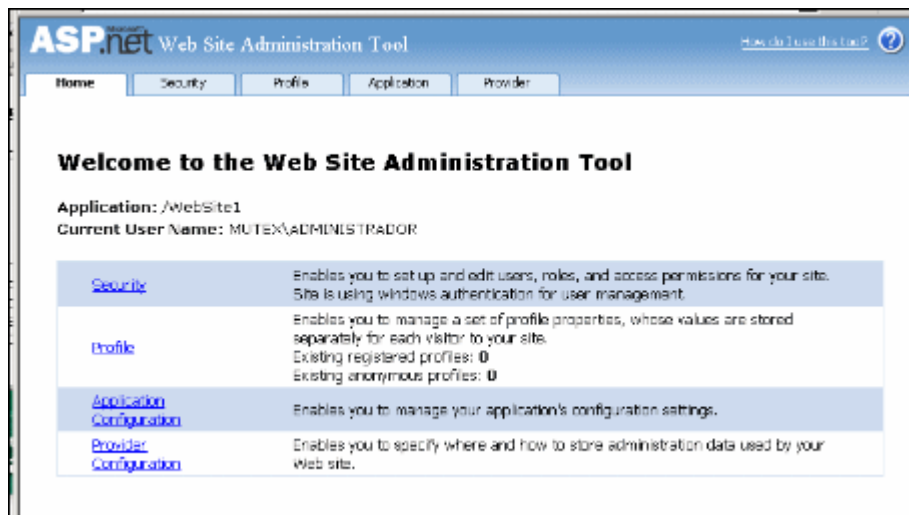


Figura 09 - Tela de Administração da aplicação.

Controles de Autenticação

A administração de usuários agora está automatizada desde a primeira tela de entrada no sistema, ou seja até a tela de login já vem pronta para você utilizar na Autenticação Forms, confira na Figura 10 e 11 os novos controles. Até para o usuário solicitar reenvio de de senha por e-mail já possui uma tela. Disponibilizando ainda um controle para exibir o status se está logado ou não.

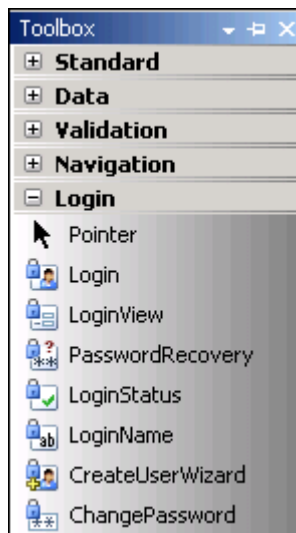


Figura 10 - Novos controles de segurança.

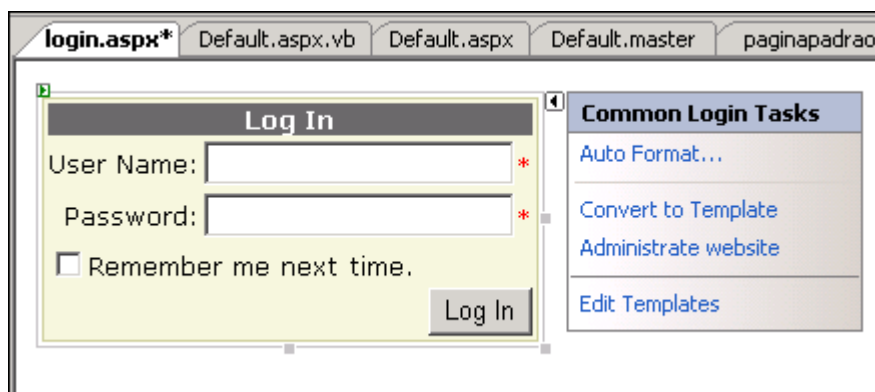


Figura 11 - Novos controles de segurança.

Grupos de Usuários (Roles)

Para utilizar o recurso de Roles durante a autenticação Forms era necessário escrever boa quantidade de código, agora com o novo "Role Manager" do ASP.NET 2.0 você já tem disponível métodos como : CreateRole, DeleteRole, AddUserToRole podendo configurar pela interface de administração já citada. Garantindo assim por exemplo que usuários do grupo "Alunos" não acessem recursos de usuários do grupo

"Professores". Caso tenha alguma tentativa de acesso indevido o usuário é redirecionado automaticamente para tela de Login.

Personalização

Este é um novo recurso oferecido para salvar informações do usuários, dados que salvamos anteriormente em cookies ou no banco de dados ou em variáveis de sessão. Basta agora criar o campo desejado no web.config e pronto o ASP.NET vai persistir essas informações para você, que vai poder recuperar ou alterar a qualquer momento. Essa informações serão salvas no Access (data\aspnetdb.mdb) ou sqlServer, necessitando de um outro banco de dados você precisar criar um provider. Por padrão o profile somente é armazenado para usuários autenticados, podendo você configurar para também trabalhar com usuários anônimos. Confira Abaixo:

Web.config:

```
<profile>
  <properties>
    <add name="Nome" />
    <add name="Quantidade_Visitas" Type="System.Int32" defaultvalue="0" />
  </properties>
</profile>
```

Agora no Código:

```
Profile.Nome="Ramon Durães"
Profile.Quantidade_Visitas=Profile.Quantidade_Visitas+1
```

Dependência de Cache SQL (SQL Cache Dependencies)

Um ótimo recurso do asp.net 1.x é o cache e objetos, podemos realizar uma consulta a um banco de dados e salvar o retorno no cache para não ter que ficar consultado o banco de dados toda hora. Agora evoluiu de forma que podemos criar uma dependência de cache a uma tabela do banco de dados. Ou seja, o cache somente vai ser invalidado se a tabela do banco de dados for alterada. A mesma dependência agora pode ser configurada também na diretiva de pagina. Para conhecer mais sobre Cache consulte o artigo do **Miguel Ferreira** na revista MSDN MAGAZINE (Dezembro/2003).

Controle de Validação (Validation Groups)

Um grande trabalho do desenvolvedor é escrever código javascript para gerenciar a entrada de dados do usuário, desde a implementação dos controle de validação eles foram muito bem recebidos pela facilidade e pelo bom resultado oferecido. Posteriormente foi observando um pequeno problema. Quando se tinha mais de um botão na tela os mesmos ocasionavam validação de dados. Fato esse resolvido, que você agora pode separar os controles de validação por grupos, resolvendo de vez o problema, sendo que cada botão ficaria associado a um grupo. Outra novidade é a possibilidade de definir um focus para o curso em caso de erro. Podendo colocar o focus numa caixa de texto.

Além do Page.IsValid foi criado o Page.Validate("Grupo") para verificar se especificamente o grupo foi validado.

Mapeamento de URL (URL Mapping)

Um grande problema dos desenvolvedores web é como lembrar de grandes urls, tais como as da ajuda do msdn, você agora poderá resolver esse problema internamente dentro da sua própria aplicação. Simplesmente adicionando o mapeamento no web.config

```
<urlMappings enabled="true">
  <add url="/sair.aspx" mappedUrl="/login.aspx?Status=Sair" />
</urlMappings>
```

Tornar objeto ativo (FOCUS)

Quanto vezes precisamos colocar o focus em um determinada caixa de texto, ou definir qual botão da pagina é o padrão.

```
<Form default tfocus="txtNome" Default tButton="button1" runat="Server" >
```

Acesso ao Help online

Uma modificação bem observada no Visual Studio é o fato de você poder fazer a consulta diretamente na base do MSDN Online e em segundo plano na sua máquina local. Com isso você poderá sempre consultar a versão da Library mais atualizada. É notório o numero de pessoas conectadas em tempo real e usando banda larga, portanto esse será um bom recurso.

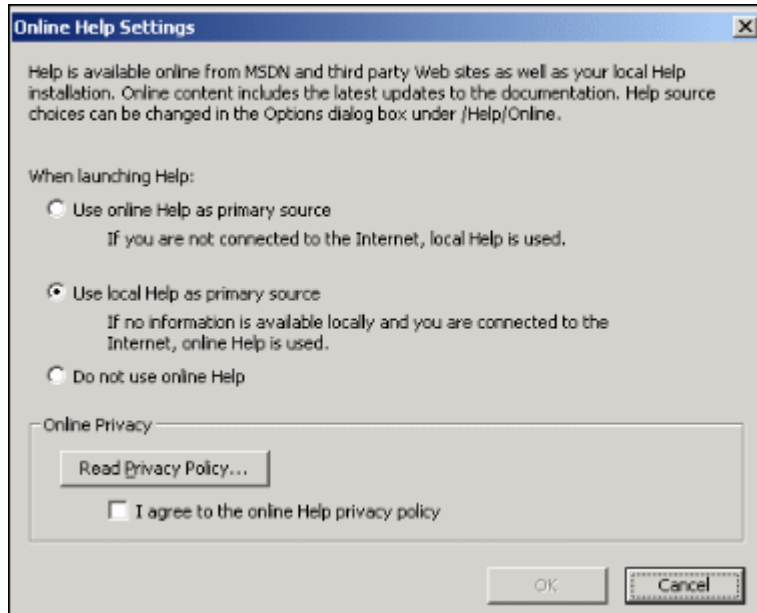


Figura 12 - Configurando ajuda para consulta on-line.

WEB PARTS

Esse recurso é bem interessante, e já vem sendo utilizado com sucesso na solução Share Point. Tem o objetivo de dar ao usuário o poder de modificar a aparência e customizar a visualização de partes do site. No asp.net essa customização já fica salva automaticamente no profile do usuário. Pense numa situação que você coloca um calendário em um local da tela. E em determinado momento um certo usuário não quer o mesmo naquele local. Com esse recurso o usuário pode move para outra webparts ou mesmo ocultar. Confira os controles na Figura 13.

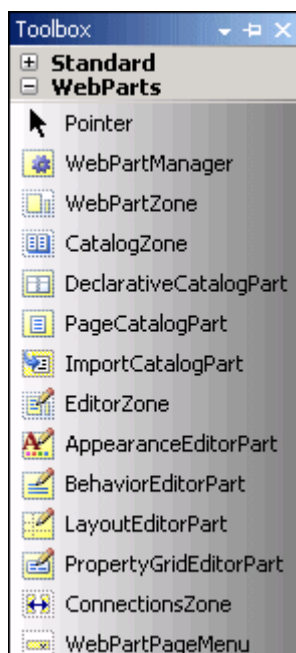


Figura 13 - Web Parts.

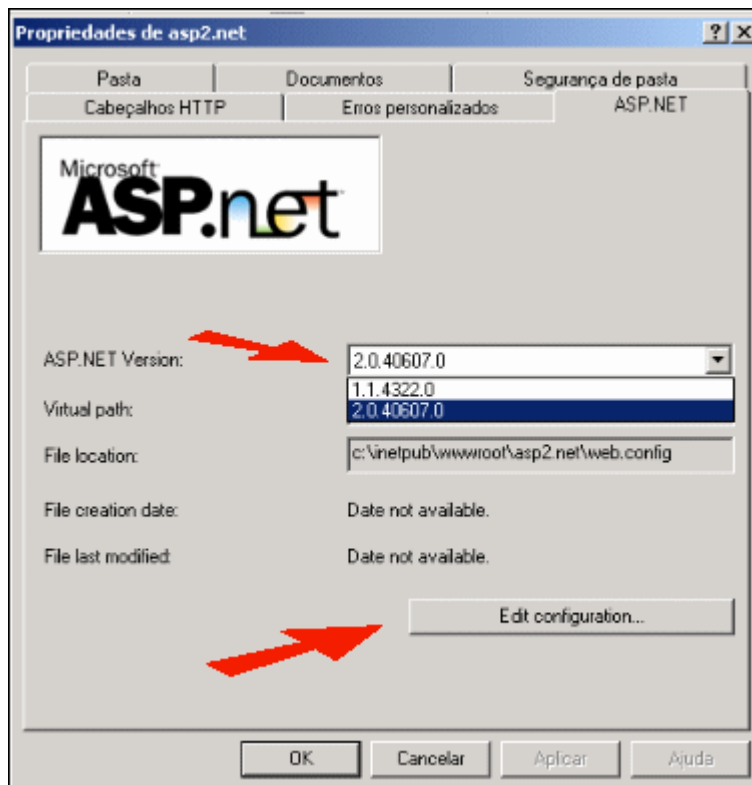


Figura 14 - Nova aba no IIS com novas opções de configuração.

Temas (Skins)

Outra grande novidade é você preparar sua aplicação para ter suporte a esse tipo de recurso, de forma que apenas trocando o skin você mude completamente a aparência do seu site. Um recurso parecido já foi implementado no ASP.NET Start Kit Community.

Novos controles

Existes outros controles novos que não citei, como Menu, Upload, Treeview, DynamicImage, ImageMap, Sitemap, DetailsView e certamente mais alguns outros que nem vi ainda ou que ainda não estão presentes nessa versão.

Projetos modelos (Start Kits)

Um novo conceito adotado dentro do Visual Studio é a questão de soluções iniciais, para que com pequenos ajustes você já comece a montar sua aplicação e uma delas já disponível em fase inicial é o Personal Web Site Starter Kit, confira na **Figura 15**.

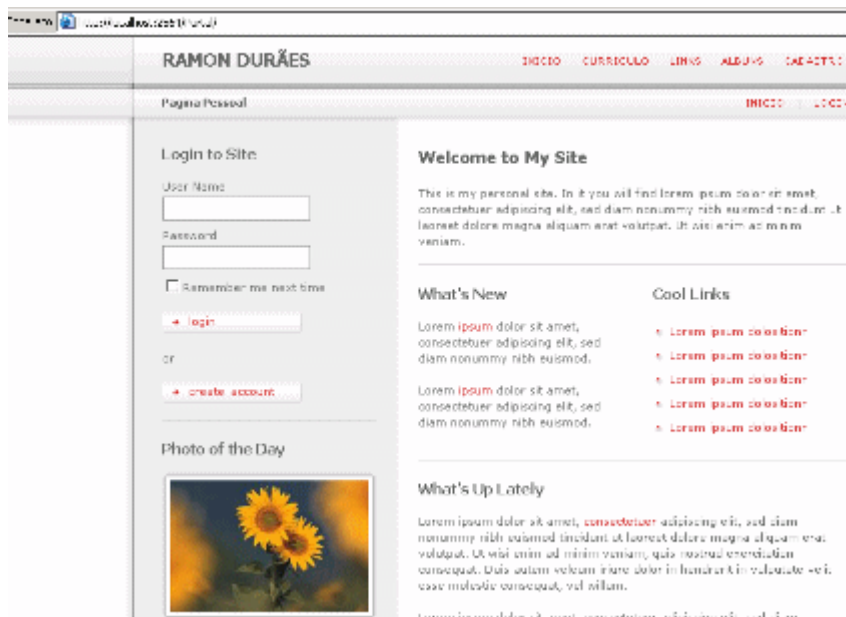


Figura 15 - Modelo de site pronto.

Posicionador de janelas

Uma das coisas que eu mais reclamava era quando uma janela saia da ordenação para coloca-la em um local tínhamos a maior dificuldade. Agora todos problemas estão solucionados. Ao arrastar aparece na tela o botão de posicionamento. Confira um exemplo na Figura 16 em que estou movendo a janela Solution Explorer de um local para outro.

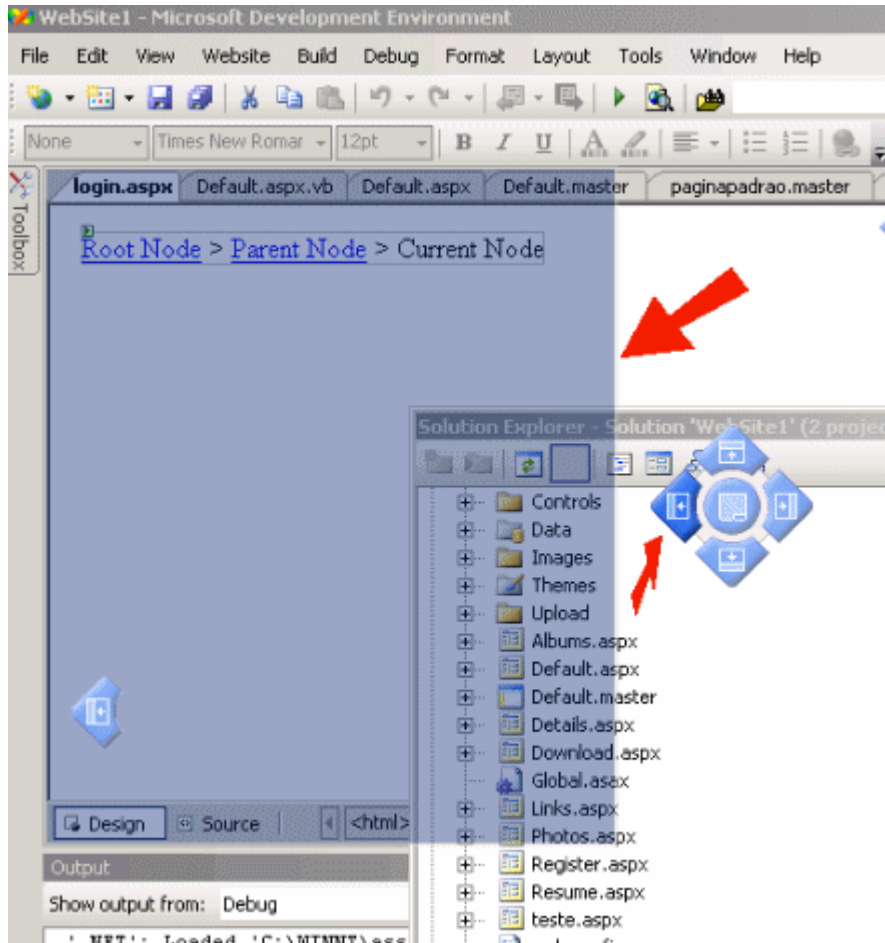


Figura 16 - Nova material de organizar as janelas.

[↶ Início da página](#)

Referências

- <http://msdn.microsoft.com>
- <http://www.msdnbrasil.com.br/msdnmag> (Brasil) - Ano 01 - Ed. 08
- Msdn Magazine - Junho/2004
- <http://www.aspallianse.com>

[↶ Início da página](#)

Finalizando

Recentemente me perguntaram se não era novo para já ter uma nova versão. Eu respondi que isso indica uma preocupação constante da Microsoft em manter a evolução do produto de forma a oferecer ao desenvolvedor sempre uma ferramenta atualizada e com mais recursos para facilitar o trabalho de implementação. Isso foi isso realmente que tenho observado desde essa versão (beta1). E já estou confiante que a Microsoft continua acertando e muito nesse seguimento. Pela qualidade que está esse produto no primeira beta, já poderemos pensar em adotar logo depois do beta 2.

"Use .NET não é mais necessário reinventar a roda..."

O que você gostaria saber?

Aguardo seu comentário! Ele é muito importante. Participe!!! Até próximo artigo!