

Tecnologia Em Analise e Desenv. de Sistemas

ASP.NET

Controles

Prof. Dr. Diego Rafael Moraes

diegorafaelmoraes@gmail.com

Notação Húngara

Controles	Prefixo	Exemplo
Button	btn	btnExit
ComboBox	cbo	cboEnglish
CheckBox	chk	chkReadOnly
CheckedListBox	clb	clbOptions
ColorDialog	cld	cldText
ContextMenu	cmn	cmnOpen
DateTimePicker	dtp	dtpPublished
ErrorProvider	err	errOpen
Form	frm	frmEntry

Controles	Prefixo	Exemplo
FontDialog	ftd	ftdText
DataGridView	grd	grdQueryResult
GroupBox	grp	grpActions
HelpProvider	hlp	hlpOptions
HScrollBar	hsb	hsbMove
ImageList	ils	ilsAllIcons
Label	lbl	lblHelpMessage
LinkLabel	lnk	lnkEmail
ListBox	lst	lstPolicyCodes

Notação Húngara

Controles	Prefixo	Exemplo
ListView	lvw	lvwHeadings
MonthCalendar	mcl	mclPeriod
Menu	mnu	mnuFileOpen
NotifyIcon	nti	ntiOpen
NumericUpDown	nud	nudPieces
OpenFileDialog	ofd	ofdImage
PictureBox	pic	picIcon
Panel	pnl	pnlGroup
PrintPreviewControl	ppc	ppcText

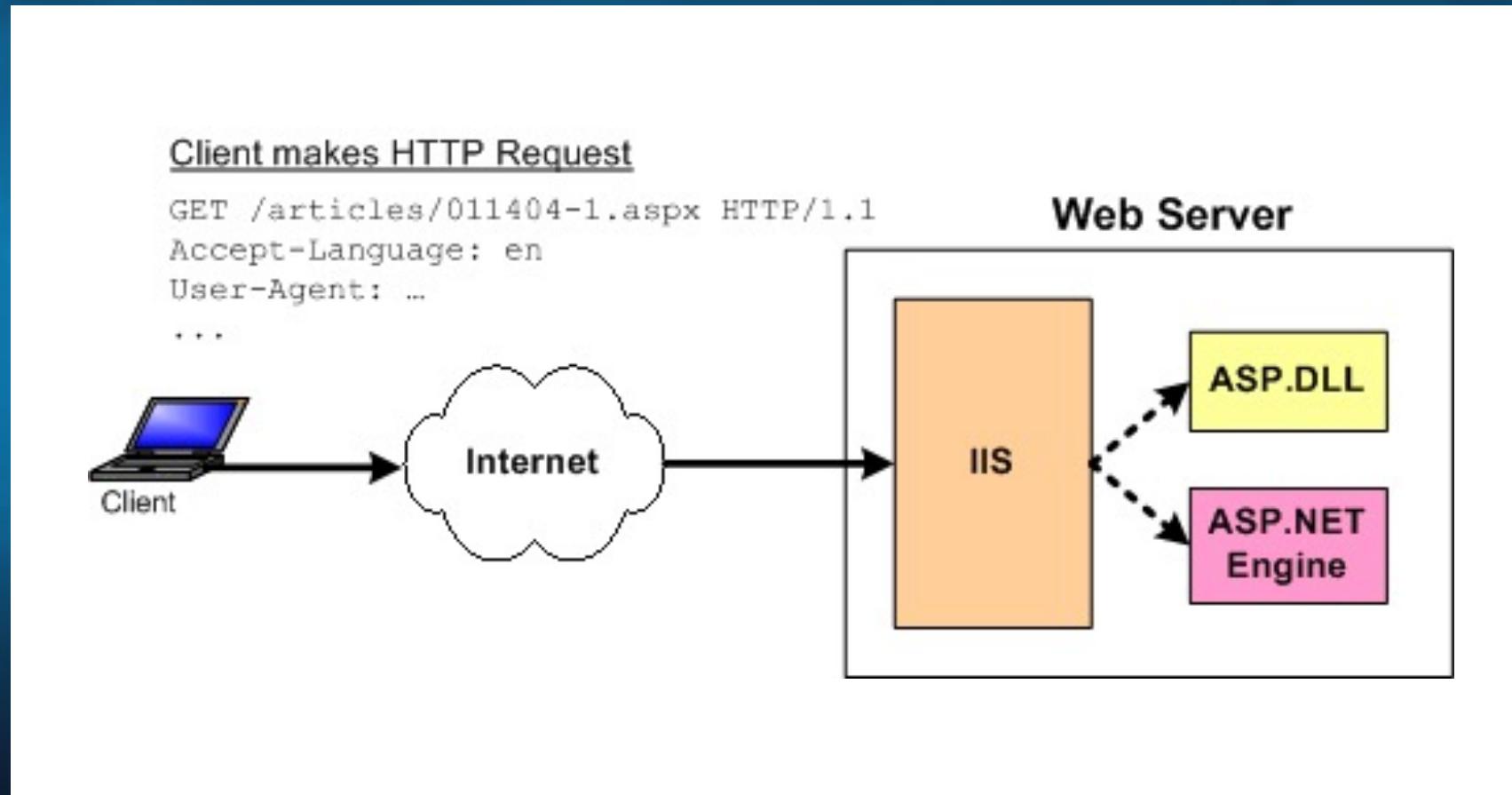
Controles	Prefixo	Exemplo
PrintPreviewDialog	ppd	ppdText
ProgressBar	prg	prgLoadFile
PrintDocument	prn	prnText
PageSetup Dialog	psd	psdReport
PrintDialog	ptd	ptdText
RadioButton	rad	radType
CrystalReportViewer	rpt	rptSales
RichTextBox	rtf	rtfReport
SaveFileDialog	sfd	sfdImage

Notação Húngara

Controles	Prefixo	Exemplo
Splitter	spt	sptDivision
StatusBar	sta	staDateTime
TabControl	tbc	tbcOptions
ToolTip	tip	tipIcons
ToolBar	tlb	tlbActions
Timer	tmr	tmrAlarm
TrackBar	trb	trbIndex
TreeView	tre	treOrganization
TextBox	txt	txtLastName
DomainUpDown	upd	updPages
VScrollBar	vsb	vsbMove

IIS

- Internet Information Services



Front-end vs Back-end

Front-end

- Parte visual (interface) responsável pela interação entre o usuário e a aplicação. Em aplicações web são utilizadas as tecnologias HTML, CSS e JavaScript para o desenvolvimento das interfaces.

Back-end

- Efetua o processamento das solicitações feitas pela interação entre o usuário e a interface (Front-end). Pode-se citar, como tecnologias utilizadas em Back-end, o C#, PHP, Java, etc.

Criação do projeto ASP.NET

O que você deseja fazer?

Abrir recente

Pesquisar recentes (Alt+S)

Introdução



Obter ou fazer check-out de código

Obter o código de um repositório online, como o GitHub ou o Azure DevOps



Abrir um projeto ou uma solução

Abrir um projeto local do Visual Studio ou arquivo .sln



Abrir uma pasta local

Navegar e editar o código dentro de qualquer pasta



Criar um projeto

Escolha um modelo de projeto com scaffolding de código para começar

Criação do projeto ASP.NET

Criar um novo projeto

Pesquisar modelos (Alt+S) Limpar tudo

C# Windows Web

Modelos de projeto recentes

- Aplicativo Web C#
- ASP.NET (.NET Framework)
- Aplicativo do Windows Forms (.NET Framework) C#
- Biblioteca de Classes (.NET Framework) C#
- Solução em branco
- Aplicativo Web C# Visual Basic
- ASP.NET (.NET Framework)
- Biblioteca de Classes (.NET Framework) C#

Aplicativo Web ASP.NET (.NET Framework)
Modelos de projeto para criar aplicativos ASP.NET. Você pode criar aplicativos de API Web, MVC ou ASP.NET Web Forms e adicionar muitos outros recursos no ASP.NET.

C# Windows Nuvem Web

Teste do WebDriver para Edge (.NET Core)
Um projeto que contém testes de unidade que podem automatizar os testes de interface do usuário de sites no navegador Edge (usando o Microsoft WebDriver).

C# Windows Web Teste

Teste do WebDriver para Edge (.NET Framework)
Um projeto que contém testes de unidade que podem automatizar os testes de interface do usuário de sites no navegador Edge (usando o Microsoft WebDriver).

C# Windows Web Teste

Serviço WCF
Um site da Web para criar serviços do WCF. Este modelo não produz um arquivo de projeto e tem suporte ao MSBuild limitado.

Voltar Próximo

Criação do projeto ASP.NET

Configurar seu novo projeto

Aplicativo Web ASP.NET (.NET Framework) C# Windows Nuvem Web

Nome do projeto

ProgramacaoWEB

Local

D:\Ricardo\Projeto\



Nome da solução ?

ProgramacaoWEB

Colocar a solução e o projeto no mesmo diretório

Framework

.NET Framework 4.7.2

Voltar

Criar

Criação do projeto ASP.NET

Criar um Aplicativo Web ASP.NET

 **Vazio**
Um modelo de projeto vazio para criar aplicativos ASP.NET. Este modelo não tem qualquer conteúdo nele.

 **Web Forms**
Um modelo de projeto para criação de aplicativos ASP.NET Web Forms. O ASP.NET Web Forms permite que você compile sites dinâmicos usando um modelo controlado por eventos do tipo "arrastar e soltar" familiar. Uma área de design e centenas de controles e componentes permitem que você compile sites controlados por IUs, poderosos e sofisticados com acesso de dados de forma rápida.

 **MVC**
Um modelo de projeto para a criação de aplicativos ASP.NET MVC. O ASP.NET MVC permite que você compile aplicativos usando a arquitetura Modelo-Exibição-Controlador. O ASP.NET MVC tem várias características que permitem o desenvolvimento rápido e baseado em teste para a criação de aplicativos que utilizam os padrões mais recentes.

 **API Web**
Um modelo de projeto para a criação de serviços de HTTP RESTful que podem alcançar a uma grande variedade de clientes, incluindo navegadores e dispositivos móveis.

 **Aplicativo de Página Única**
Um modelo de projeto para a criação de aplicativos HTML5 orientado por JavaScript no lado do cliente avançado utilizando a ASP.NET Web API. Os aplicativos de página única

Autenticação
Sem Autenticação
[Alterar](#)

Adicionar as pastas e as principais referências

Formulários da Web
 MVC
 API Web

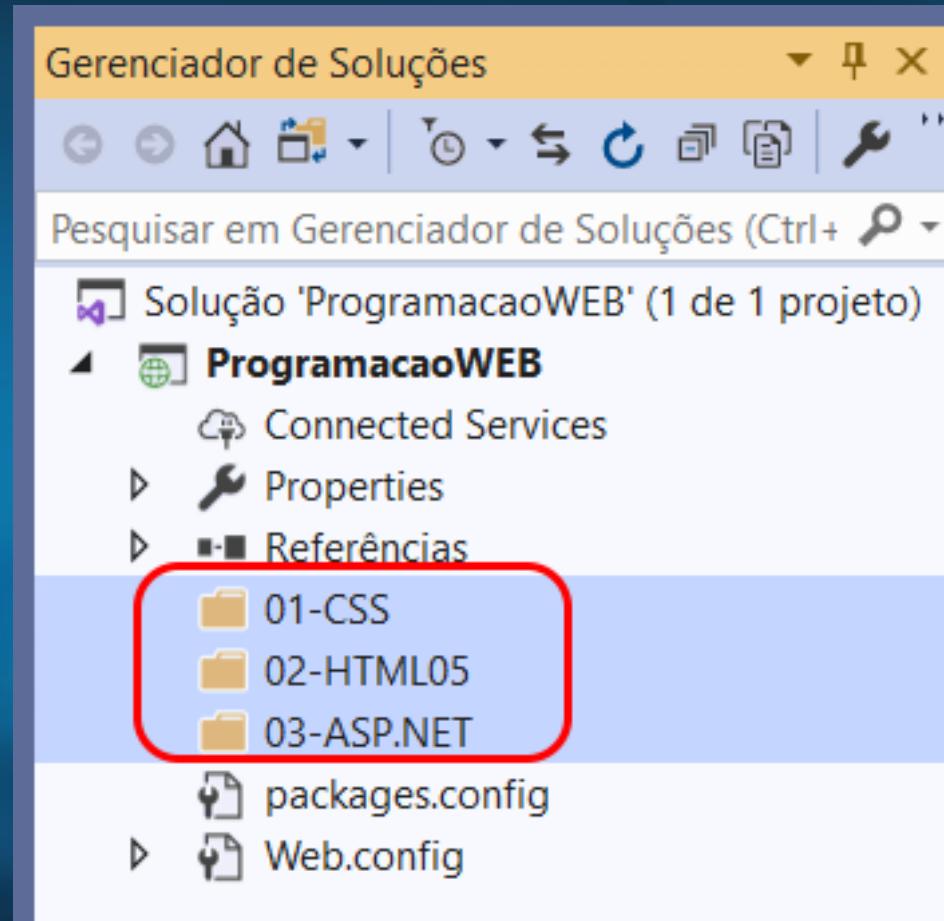
Avançado

Configurar para HTTPS
 Suporte do Docker
(Exige [Docker Desktop](#))
 Além disso, crie um projeto para testes de unidade

ProgramacaoWEB.Tests

[Voltar](#) [Criar](#)

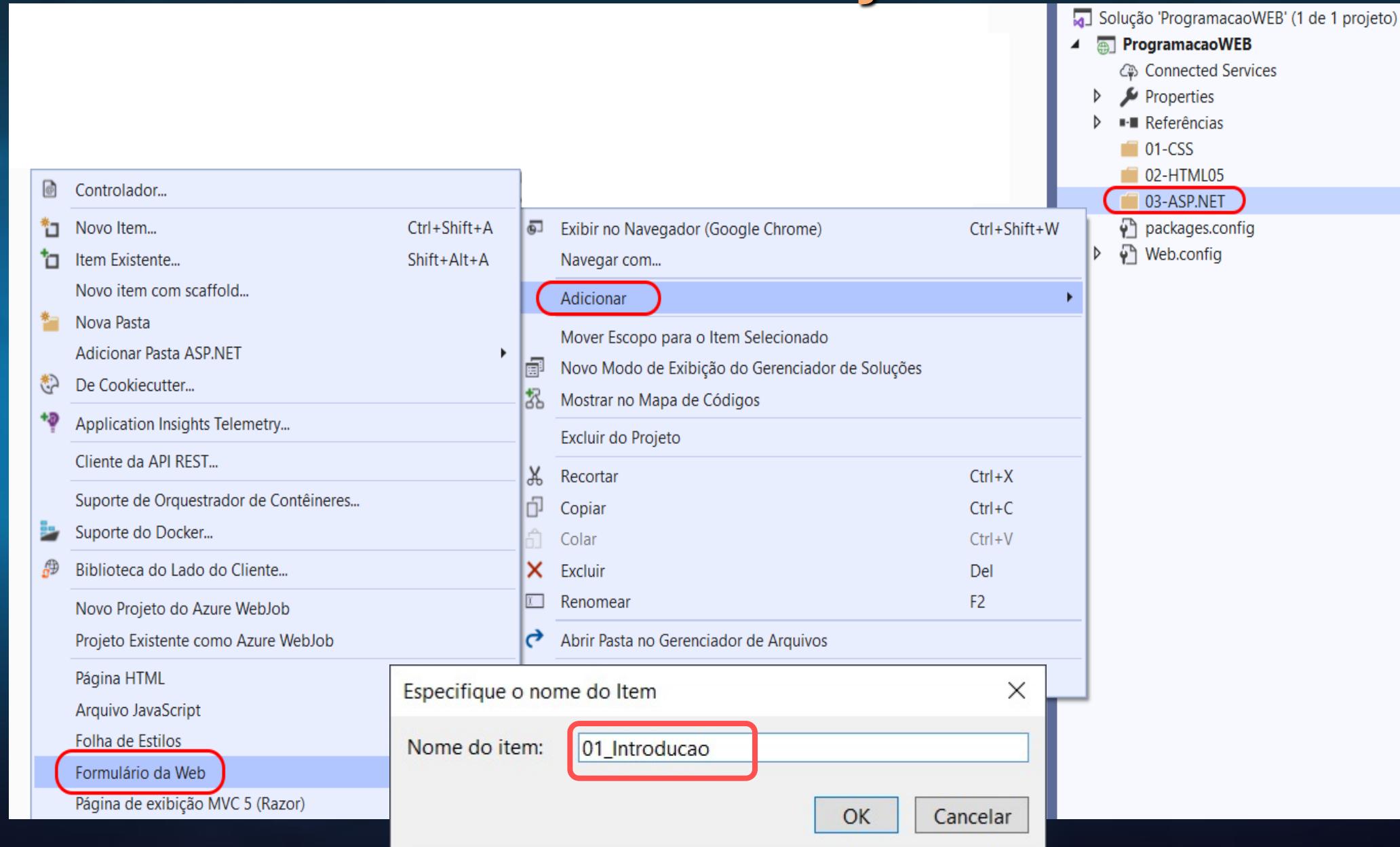
Criação do projeto ASP.NET



Estrutura de arquivo / pasta

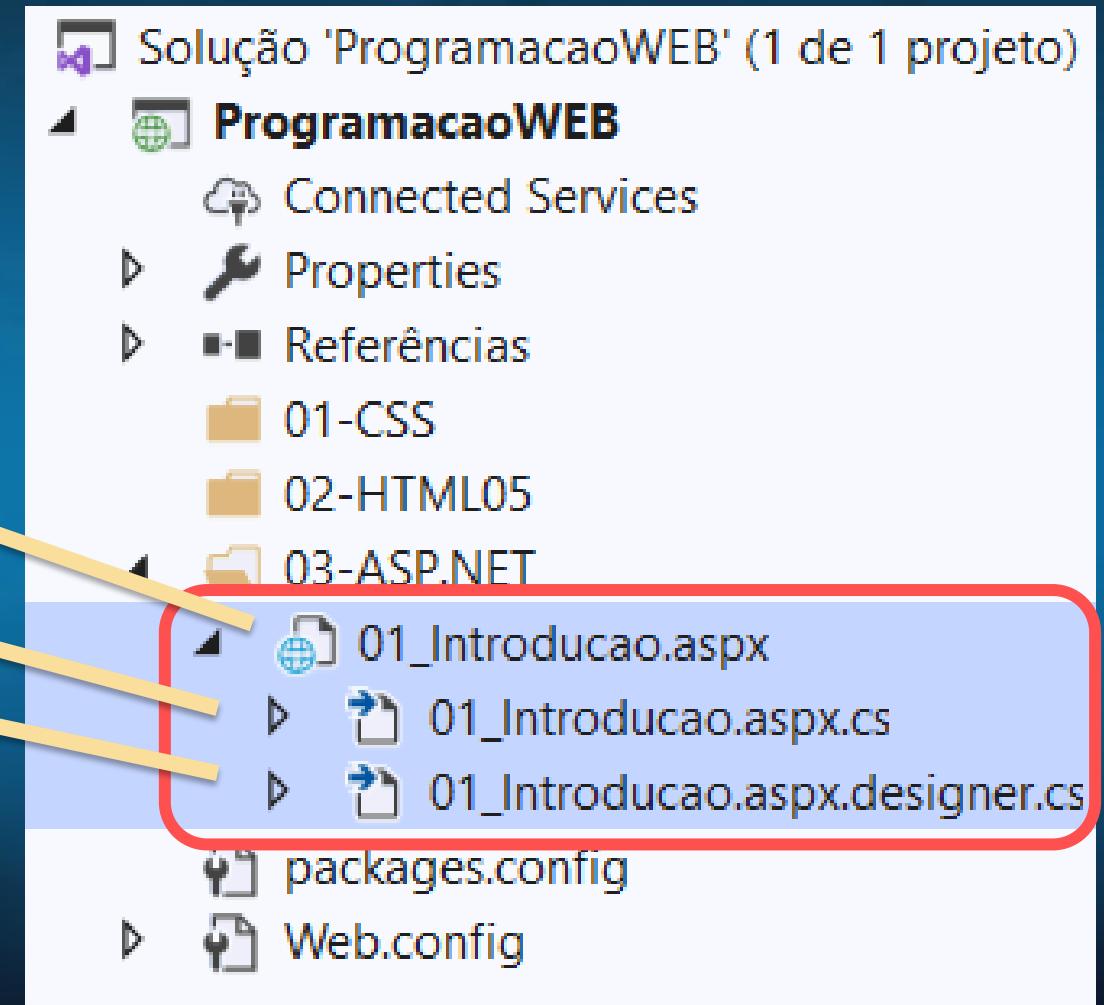
- **App_Code** → contém apenas arquivos de código C#.
- **App_Data** → contém pacotes e seu banco de dados.
- **Bin** → contém arquivos compilados.
- **Content** → contém imagens, e arquivos CSS, etc.
- **Account** → contém arquivos de usuários (login / registro etc). Se um usuário for para uma página restrita que exige um login, ele será redirecionado para Conta / Login, pois é o local padrão.
- **Scripts** → contém arquivos de script, como JavaScript.
- **Shared** → contém os arquivos de layout do seu site.

Introdução

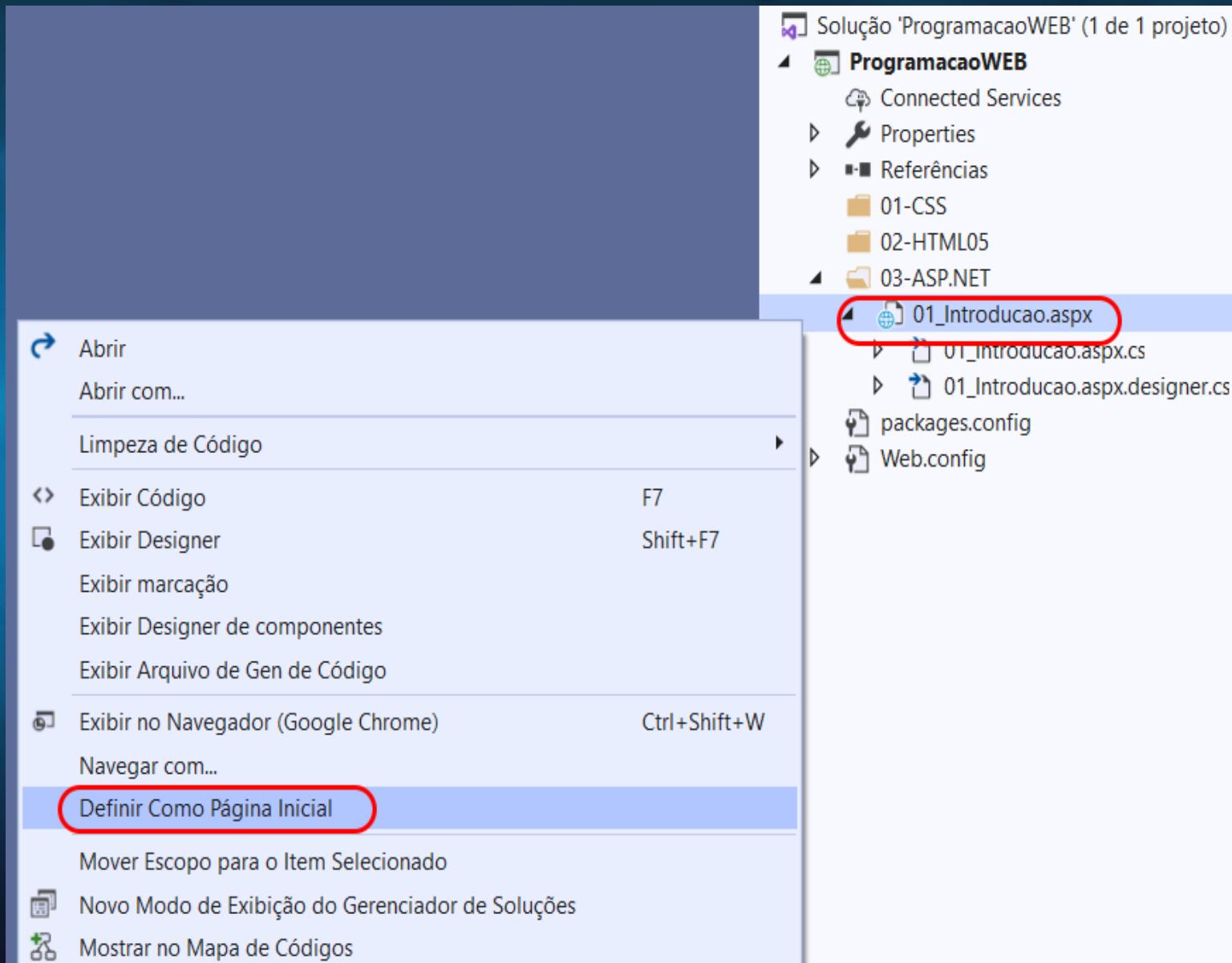


Introdução

- HTML
- Código C#
- Definição dos controles



Introdução



Introdução

```
4
5 <html xmlns="http://www.w3.org/1999/xhtml">
6 <head runat="server">
7   <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
8   <title></title>
9 </head>
10 <body>
11   <h1> Introdução ao ASP.NET</h1>
12
13   <form id="form1" runat="server">
14     <div>
15     </div>
16   </form>
17 </body>
18 </html>
19
```

90 % ▾ Não foi encontrado nenhum problema ◀ ▶ Ln: 17 Car: 1

body

Introdução ao ASP.NET

Introdução

- **Response.Write** → exibe conteúdo na tela.

```
namespace ProgramacaoWEB._03_ASP.NET
{
    public partial class _01_Introducao : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            Response.Write("Olá Mundo!");
        }
    }
}
```

TextBox – SingleLine

Principais propriedades:

- **ID** → identificador
- **Text** → define o texto a ser mostrado para o controle.
- **MaxLength** → define o número máximo de caracteres que podem ser inseridos.
- **ToolTip** → exibe o texto quando o mouse está sobre o controle.
- **Readonly** → usado para tornar o controle somente leitura.
- **TextMode** →

TextBox - SingleLine

```
0 referências
protected void btnExecutar_Click(object sender, EventArgs e)
{
    string nome;

    // Text Box
    nome = txtNome.Text;
    lblSaida.Text = nome.ToUpper();
}
```

TextBox - Number

- Conversão de números com casas decimais.
- Namespace → **System.Globalization**
- A **InvariantCulture** propriedade pode ser usada para persistir dados em um formato independente de cultura.

```
float altura;  
altura = float.Parse(txtNumero.Text, CultureInfo.InvariantCulture);  
lblSaida.Text = altura.ToString("N2");
```

Saiba mais:

<https://docs.microsoft.com/pt-br/dotnet/api/system.globalization.cultureinfo.invariantculture?view=netcore-3.1>

Radio Button

Principais propriedades:

- **ID** → identificador
- **Text** → define o texto a ser mostrado para o controle.
- **GroupName** → define o nome do grupo de botões de opção.
- **Checked** → define a opção selecionada.

Radio Button List

Principais propriedades:

- **ID** → identificador
- **Text** → define o texto a ser mostrado para o controle.
- **RepeatDirection** → define a direção na qual os botões de opção dentro do grupo são exibidos.
- **RepeatColumns** → Obtém ou define o número de colunas a serem exibidas no controle RadioButtonList.
- **RepeatLayout** → Obtém ou define um valor que especifica se a lista será renderizada usando um elemento table, um elemento ul, um elemento ol ou um elemento span.

CheckBox

Principais propriedades:

- **ID** → identificador
- **Text** → define o texto a ser mostrado para o controle.
- **Checked** → define a opção selecionada.

Page.IsPostBack

- Obtém um valor que indica se a página está sendo renderizada pela primeira vez ou está sendo carregada em resposta a um postback.

DropDownList

The screenshot illustrates the configuration of a DropDownList control in a Windows application.

Properties Window: Shows the properties of the DropDownList1 control.

ClientIDMode	Inherit
Enabled	True
EnableTheming	True
EnableViewState	True
SkinID	
ToolTip	
ValidateRequestMode	Inherit
ValidationGroup	
ViewStateMode	Inherit
Visible	True

Dados (Data) section:

- (Expressions)
- DataMember
- DataSourceID
- DataTextField
- DataTextFormatString
- DataValueField
- ItemType

Control on Form: The DropDownList control is displayed on the form with the value "Fiat Strada". A context menu is open over the control, showing options: "Escolher Fonte de Dados...", "Editar Itens...", and "Habilitar AutoPostBack". The "Editar Itens..." option is highlighted with a red box.

Editor de Coleção ListItem: This dialog shows the list item collection for the DropDownList.

0	Fiat Strada
---	-------------

Properties Dialog (propriedades Fiat Strada): Displays the properties of the selected list item.

Diversos	
Enabled	True
Selected	False
Text	Fiat Strada
Value	1

Buttons: OK and Cancelar.

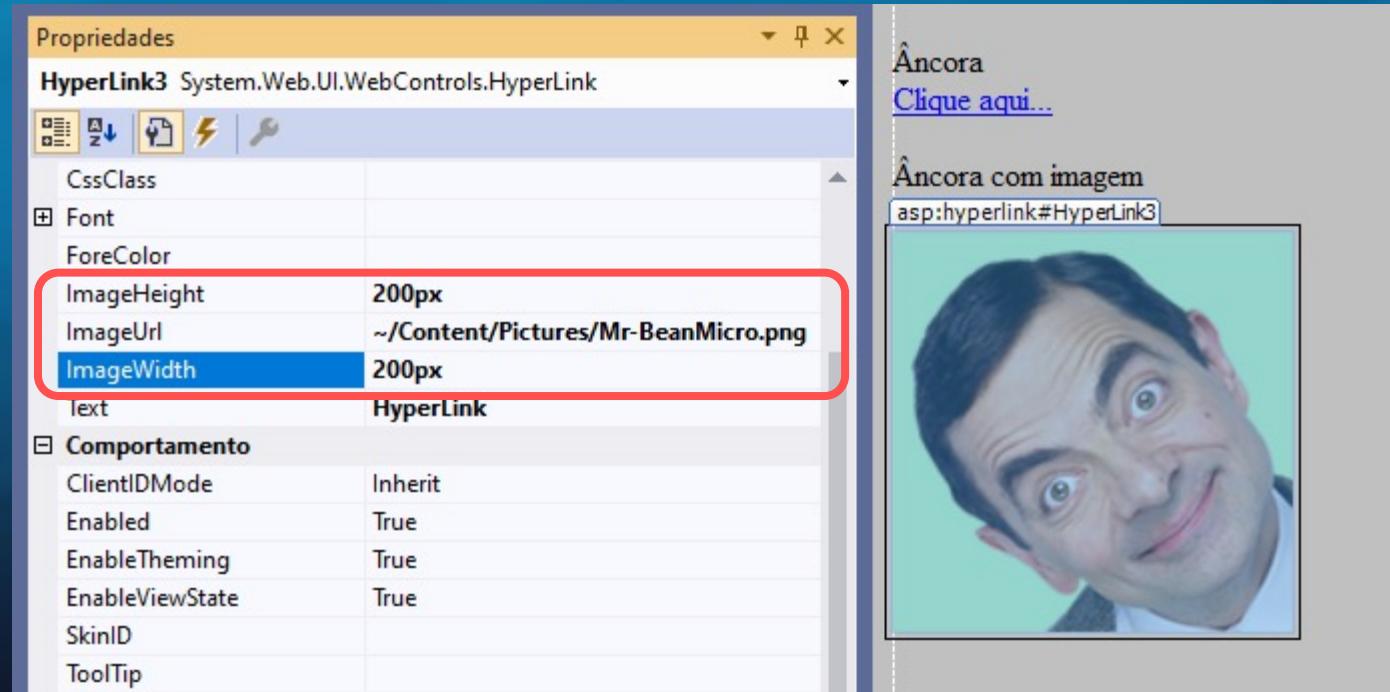
Âncora

Principais propriedades:

- **ID** → identificador
- **Text** → define o texto a ser mostrado para o controle.
- **NavigateUrl** → define a URL que será usada para link quando um usuário clicar no controle HyperLink.
- **Target** → define a janela ou o quadro de destino no qual deverá ser exibido o conteúdo da página da Web vinculada quando um usuário clicar no controle HyperLink.
- **ImageUrl** → define o caminho para uma imagem a ser exibida para o controle de HyperLink.

Âncora

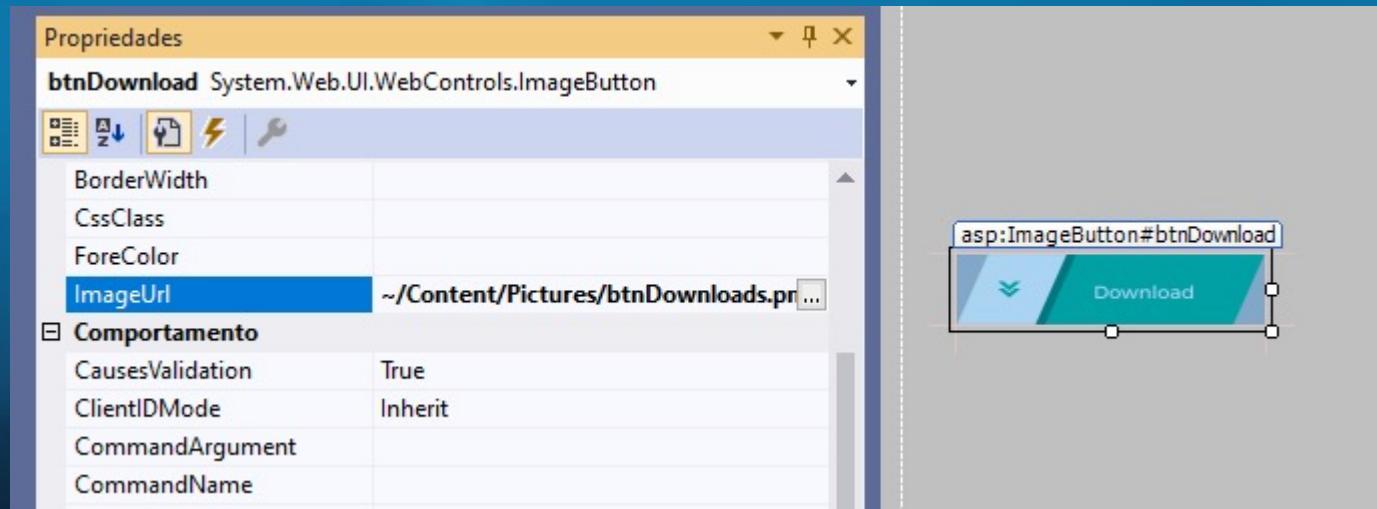
- **ImageUrl** → Obtém ou define o caminho para uma imagem a ser exibida para o controle de HyperLink.
- **ImageWidth** → Obtém ou define a largura do hiperlink quando o hiperlink é uma imagem.



ImageButton

Principais propriedades:

- **ImageUrl** → Obtém ou define a URL que fornece o caminho a uma imagem para exibir o controle Image.



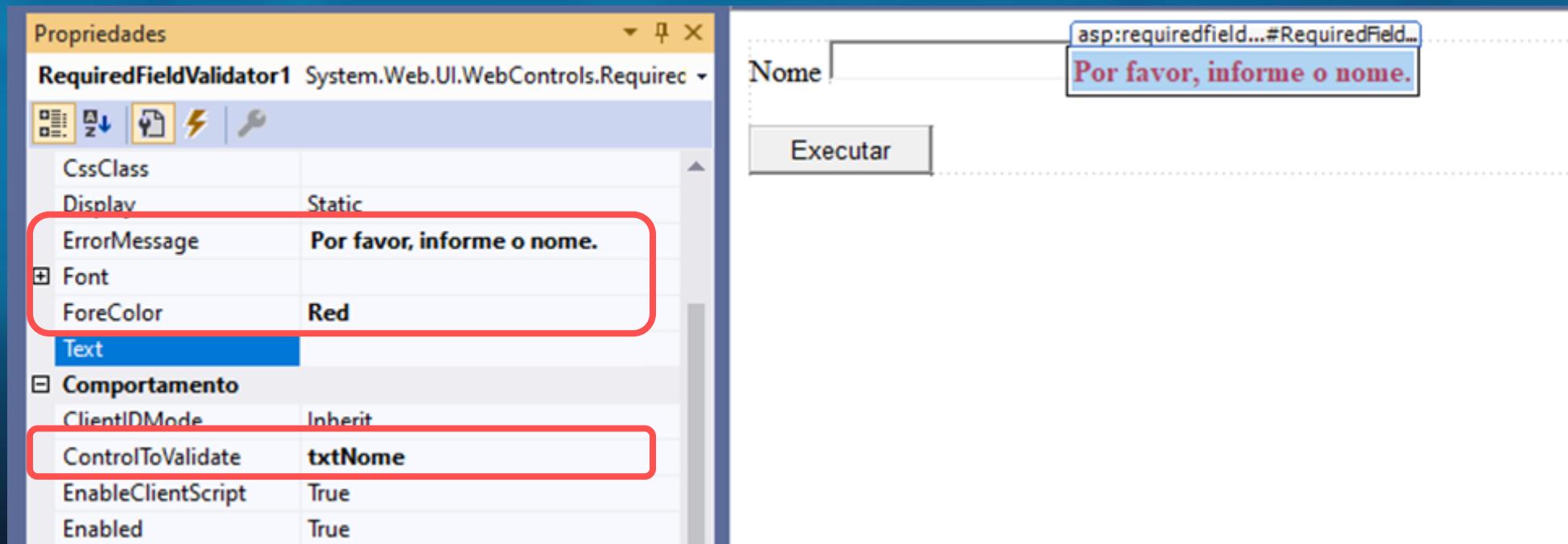
Validação

- Ajuste do WEB.CONFIG para solução de erro do RequiredFieldValidator.

```
7 <configuration>
8
9   <appSettings>
10    <add key="ValidationSettings:UnobtrusiveValidationMode" value="None"/>
11  </appSettings>
12
13 <system.web>
14
15  <!--cookieless="true" envia o IDSession na URL da página -->
16  <sessionState timeout="1" cookieless="false"></sessionState>
17
18  <compilation debug="true" targetFramework="4.7.2"/>
19  <httpRuntime targetFramework="4.7.2"/>
20 </system.web>
```

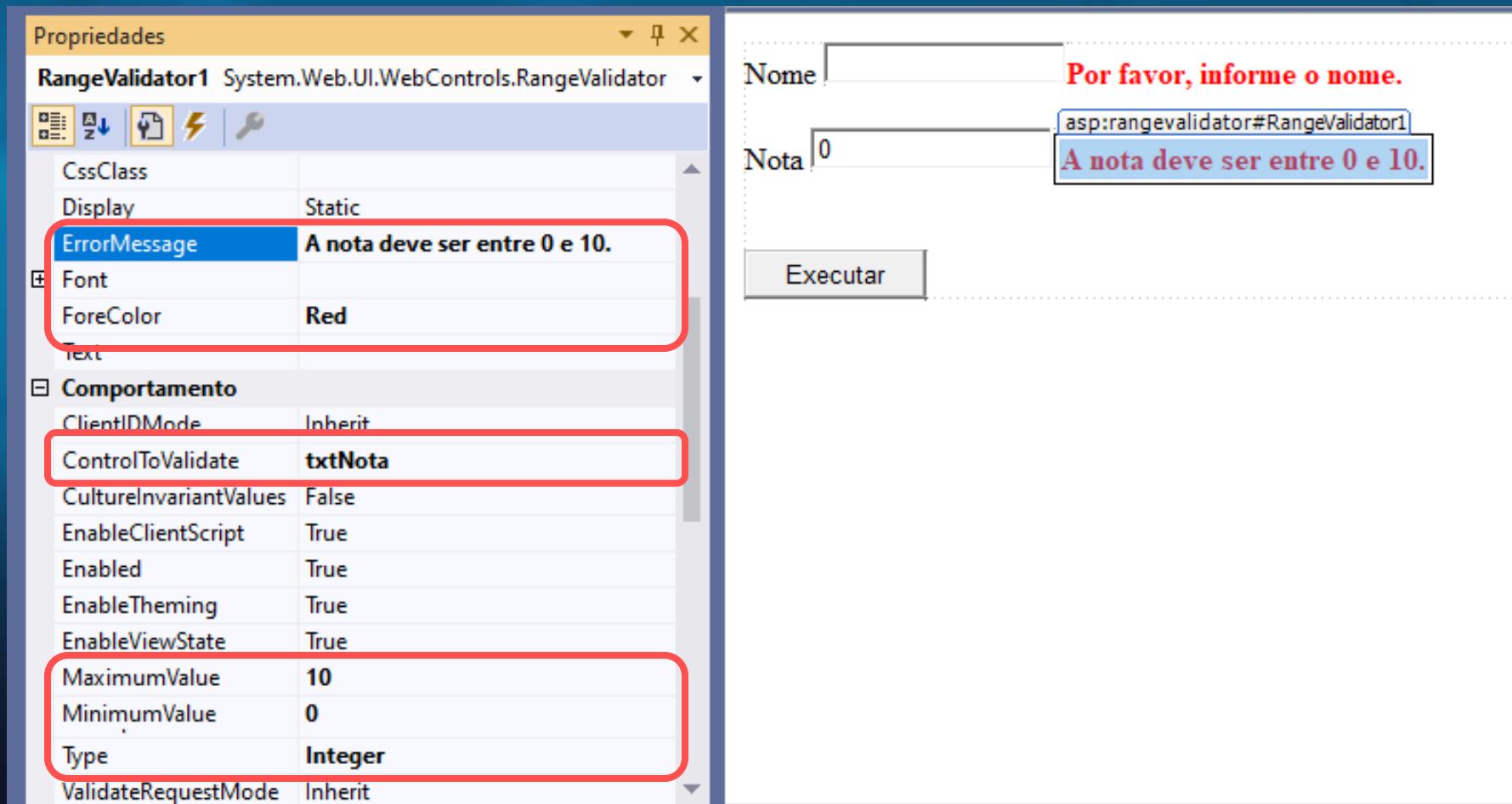
RequiredFieldValidator

- RequiredFieldValidator → verifica se o usuário inseriu algum valor na caixa de texto.



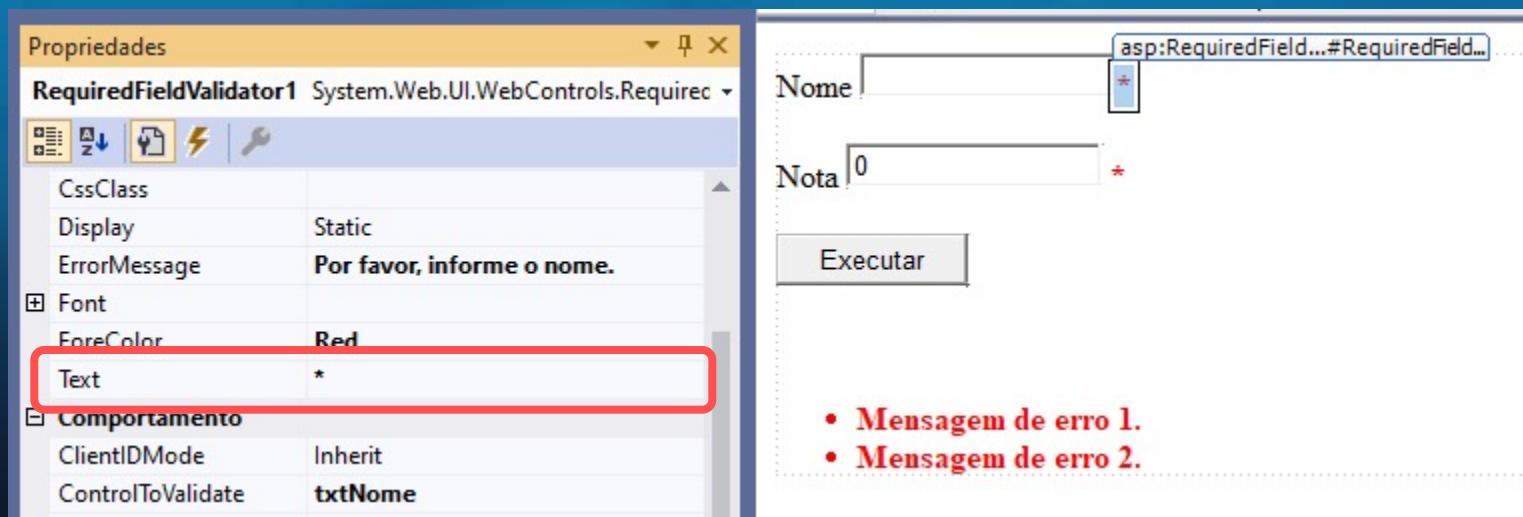
RangeValidator

- **RangeValidator** → Verifica se o valor de um controle de entrada está dentro de um intervalo de valores especificado.



ValidationSummary

- Exibe um resumo de todos os erros de validação embutidos em uma página da Web, em uma caixa de mensagem, ou em ambas.
- A propriedade **Text** das validações devem ser configuradas com *****.



ValidationSummary

