

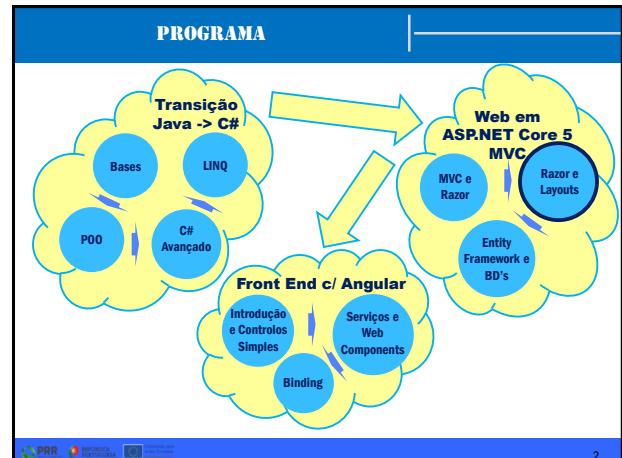
PROGRAMAÇÃO VISUAL TESP/TPSI

Introdução à Programação para a Web com .NET MVC ViewBag / ViewData / Layouts

Prof. José Brás
F262 - jose.brás.pereira@estsetubal.ips.pt
DEI / EST Setúbal / IPS
2025/26

1



2

5.2 WEB SITE C / LAYOUTS

5. .NET CORE WEB

- 5.2.0 – Criação do projeto MVC c/ Layouts
- 5.2.1 – Controladores (Controller), ações e resultados
 - 5.2.1.1 - Ações dos controladores
 - 5.2.1.2 - Resultados das ações dos controladores
- 5.2.2 – Passagem de Dados p/ Vistas (View)
 - 5.2.2.1 Model
 - 5.2.2.2 ViewBag
 - 5.2.2.3 ViewData
- 5.2.3 Criação e Utilização de novos Layouts
 - 5.2.3.1 Utilização de Layouts
 - 5.2.3.2 HTML Helpers
 - 5.2.3.3 Secção footer do _Layout preenchida na View
 - 5.2.3.4 Definição do Layout na _ViewStart
- 5.2.4 Sintaxe Razor (Vistas)

3

5.2.0 CRIAÇÃO DO PROJETO

5. .NET CORE WEB

5.2 WEB SITE C / LAYOUTS

Create a new project

Configure your new project

ASP.NET Core Web App (Model-View-Controller) > NEXT

Additional information

Framework: .NET 8.0 or 9.0
Authentication type: None
Configure for HTTPS: No
Do not use top-level statements: yes > Create

Blazor WebAssembly App
A project template for creating a Blazor app that runs on WebAssembly and is optionally hosted by an ASP.NET Core app. This template can be used for web apps with rich dynamic user interfaces (UIs).

ASP.NET Core Web App (Model-View-Controller)
A project template for creating an ASP.NET Core application with separate ASP.NET Core MVC Views and Controllers. This can also be used for RESTful HTTP services.

4

5.2.1 - CONTROLADORES

5.2 – WEB SITE C / LAYOUTS

Estão sempre na pasta com o nome Controllers

Por convenção têm o sufixo Controller como em FotoController

- O Controller é o elemento central do padrão MVC
- Os Routers instanciam um controlador (um objeto da classe) por cada pedido
 - Cada pedido é mapeado para uma ação específica
- Todos os controladores herdam da classe Controller
 - A classe Controller fornece o acesso ao Request (a informação do pedido)

5

5.2.1.1 – AÇÕES

5.2 – WEB SITE C / LAYOUTS

■ As ações são métodos dos controladores e representam o destino final dos pedidos

- São métodos públicos
- Não são de classe (não são static)
- Por omissão são HTTPGet (pelo que podemos omitir o "Decoration Atribute" [HttpGet])
- Não impõem restrições ao tipo de retorno mas ...


```
public class FotoController : Controller
{
    public IActionResult Index()
    {
        Foto f = new Foto()
        {
            Codigo = "ABC00001",
            Titulo = "Deserto"
        };
        return View(f);
    }
}
```

retornam normalmente um IActionResult (*)

(*) Uma variável do tipo Interface IActionResult

6

5.2.1.2 – RESULTADOS DAS AÇÕES

Os resultados das ações

- Respondem a um pedido do browser
- Normalmente retornam uma variável do tipo Interface `IActionResult`
 - Ou seja de um tipo que implemente os métodos requeridos pela Interface `IActionResult`

Existem vários tipos de retorno:

Nome	Comportamento	Produto
<code>ContentResult</code>	Por omissão retorna uma string mas pode retornar outros tipos	Conteúdo
<code>EmptyResult</code>	Sem resposta. Usado para comandos como create, update e delete	
<code>FileContentResult</code>	Retorna o conteúdo de um ficheiro.	File
<code>PhysicalFileResult</code>	Subclasse de <code>FileResult</code>	
<code>FileStreamResult</code>		
<code>VirtualFileResult</code>		
<code>ChallengeResult</code>	Authentication credential not valid or not present. Returns an HTTP 401 status code	

PRR Portugal

7

5.2.1.2 – RESULTADOS DAS AÇÕES

Nome	Comportamento	Produto
<code>UnauthorizedResult</code>	Returns an HTTP 401 status code and nothing else	
<code>RedirectResult</code>	Redirects the client to a new URL. Derived from <code>RedirectToActionResult</code>	Redirect / RedirectPermanent
<code>RedirectToActionResult</code>		
<code>RedirectToRouteResult</code>		
<code>LocalRedirectResult</code>		
<code>ForbidResult</code>	To refuse a request to a particular resource. Returns an HTTP 403 status	
<code>JsonResult</code>	Returns data in JSON format	Jason
<code>ViewResult</code>	Response is the responsibility of a view engine-To render a view or part of it.	View / PartialView
<code>PartialViewResult</code>		
<code>ViewComponentResult</code>	Returns html content for a view	
<code>SignInResult</code>	Sign in or sign out the user based on provided mechanism	
<code>SignOutResult</code>		

... Existem mais tipos de retorno

PRR Portugal

8

7

8

5.2.2 - PASSAGEM DE DADOS P/ VISTAS

É possível passar informação para as vistas através de:

- ViewBag**
 - Action: `ViewBag.Message = "Text of message";`
 - View: `@ViewBag.Message`
- ViewData**
 - Action: `ViewData["messageIdentifier"] = "message text";`
 - View: `@ViewData["messageIdentifier"]`
- Model (strongly-typed)**
 - Action: `return view(model);`
 - View: `@model ModelDataType`

PRR Portugal

9

5.2.2.1 – PASSAGEM DE DADOS C/MODEL

5. .NET CORE WEB

5.1 – WEB SITE SIMPLES

1. BDR sobre a pasta Models -> Add -> New Item -> Class -> Foto

```
namespace TP11.Models
{
    public class Foto
    {
        public String Código { get; set; }
        public String Título { get; set; }
    }
}
```

2. Adicionar 2 propriedades Código, Título

3. Adicionar ao HomeController uma ação PorTítulo que aceite um título e retorne uma view com um objeto da classe foto

```
public IActionResult PorTítulo(string título)
{
    Foto f = new Foto
    {
        Código = "ABC000001",
        Título = título
    };
    return View(f);
}
```

PRR Portugal

10

9

10

5.2.2.1 – PASSAGEM DE DADOS C/MODEL

5. .NET CORE WEB

5.2 – WEB SITE C/ LAYOUTS

1. BDR sobre a pasta Home -> Add -> View

2. No diálogo Add New Scaffolded Item -> RazorView

3. No diálogo Add Razor View:
View name: **PorTítulo**
Template: **Details**
Model Class: **Foto**
Layout page: **_Layout**
>> Add

4. Run & Try

Copy/paste url to browser navigation box ->
https://[domain]/Home/PorTítulo?Título=Deserto

PRR Portugal

11

5.2.2.1 – PASSAGEM DE DADOS C/MODEL

5. .NET CORE WEB

5.1 SIMPLE WEB SITE

Generated Output

Data + Template

View PorTítulo.cshtml

```
1 <model>TP12.Models.Foto
2 
3 <!
4     ViewData["Title"] = "PorTítulo";
5     Layout = "~/Views/Shared/_Layout.cshtml";
6 </!
7 
8 <h1>@ViewData["Title"]</h1>
9 <p>@Model.Código<br/>@Model.Título<br/>@Model.Código</p>
10 
11 </>
```

Model (Foto.cs)

```
namespace TP12.Models
{
    public class Foto
    {
        public String Código { get; set; }
        public String Título { get; set; }
    }
}
```

PRR Portugal

12

11

12

5.2.2.2-PASSAGEM DE DADOS C/ VIEWBAG

5..NET CORE WEB

5.2 WEB SITE C/ LAYOUTS

```
public IActionResult Index()
{
    ViewBag.Welcome = "Bem Vindo";
    Foto f = new Foto() { Titulo = "Deserto" };
    ViewData["ObjetoLista"] = lista;
    ViewData["Texto1"] = "Bem Vindo ao FotoShopping";
    return View();
}
```

1. Reescreva a action Index no ficheiro HomeController.cs

2. Reescreva o código do ficheiro Index.cshtml da view Home

3. Run & Try

ViewBag – é uma propriedade de tipo dinâmico da classe Controller

Serve para passar dados do controlador para a view

TP16_Slides

Bem Vindo
Deserto

© 2022 - TP16_Slides - Privacy

13

5.2.2.3-PASSAGEM DE DADOS C/ VIEWDATA

5..NET CORE WEB

5.2 WEB SITE C/ LAYOUTS

```
public IActionResult Index()
{
    List<String> lista = new List<String>() {
        "aqui vai encontrar", "as melhores fotos"
    };
    ViewData["ObjetoLista"] = lista;
    ViewData["Texto1"] = "Bem Vindo ao FotoShopping";
    return View();
}
```

1. Reescreva a action Index no ficheiro HomeController.cs

2. Reescreva o código do ficheiro Index.cshtml da view Home

3. Run & Try

ViewData – é um Dicionário de pares chave-valor em que cada chave é uma String

Tal como a ViewBag serve para passar dados do controlador para a view

12_StandAuto

Bem Vindo
ao
FotoShopping

aqui vai encontrar
as melhores fotos

© 2022 - 12_StandAuto - Privacy

14

5.2.3 CRIAÇÃO E USO DE LAYOUTS

5..NET CORE WEB

5.2 WEB SITE C/ LAYOUTS

1. BDR na pasta Shared -> New Item

2. Selecionar C# -> Razor Layout

3. Nome: _MyLayout (com _ no inicio) -> Add

4. Substituir o html original pelo deste slide

15

5.2.3.1 UTILIZAÇÃO DE LAYOUTS

5..NET CORE WEB

5.2 WEB SITE C/ LAYOUTS

Cada vista pode especificar a sua página de layout

```
@{ Layout = "~/Views/Shared/_MyLayout.cshtml"; }
```

```
@{ Layout = null; }
```

16

5.2.3.1 UTILIZAÇÃO DE LAYOUTS

5..NET CORE WEB

5.2 WEB SITE C/ LAYOUTS

1. Abrir o ficheiro Index da view Home

2. Inserir o path para o _MyLayout.cshtml logo abaixo da ViewData["Title"]

3. Repetir para o ficheiro Privacy.cshtml

Links

As views são injetadas aqui por baixo

4. Run & Try

Bem Vindo
Ao Fotoshopping
aqui vai encontrar as melhores fotos

17

5.2.3.2 HTML HELPERS

5..NET CORE WEB

5.1 – WEB SITE SIMPLES

```
<!DOCTYPE html>
<html>
    <meta name="viewport" content="width=device-width" />
    <head>
        <title>@ViewBag.Title</title>
    </head>
    <body>
        <table style="width:100%;">
            <tr>
                <td>Links</td>
                <td>As views entram aqui em baixo</td>
            </tr>
            <tr>
                <td></td>
                <td>
                    <p>@Html.ActionLink("Home", "Index")</p>
                    <p>@Html.ActionLink("Privacy", "Privacy")</p>
                </td>
            </tr>
            <tr>
                <td></td>
                <td>@RenderBody()</td>
            </tr>
        </table>
    </body>
</html>
```

HTML helper ActionLink

A classe **HtmlHelper** gera elementos de html usando objectos da classe model.

Liga (bind) o modelo a elementos de html para apresentar os valores das propriedades do modelo no elemento html

A vantagem dos html Helpers sobre as tags html é que os métodos dos html helpers são desenhados para facilitar o binding entre a view e o modelo de dados

18

5.2.3.2 HTML HELPERS			5..NET CORE WEB
			5.1 – WEB SITE SIMPLES
HtmlHelper	Strongly Typed HtmlHelpers	Html Control	
Html.ActionLink		Anchor link	
Html.TextBox	Html.TextBoxFor	Textbox	
Html.TextArea	Html.TextAreaFor	TextArea	
Html.CheckBox	Html.CheckBoxFor	Checkbox	
Html.RadioButton	Html.RadioButtonFor	Radio button	
Html.DropDownList	Html.DropDownListFor	Dropdown, combobox	
Html.ListBox	Html.ListBoxFor	multi-select list box	
Html.Hidden	Html.HiddenFor	Hidden field	
Password	Html.PasswordFor	Password textbox	
Html.Display	Html.DisplayFor	Html text	
Html.Label	Html.LabelFor	Label	
Html.Editor	Html.EditorFor	Generates Html controls based on data type of specified model property e.g. textbox for string property, numeric field for int, double or other numeric type.	

19

5.2.3.3 SECÇÃO FOOTER			5..NET CORE WEB
			5.2 WEB SITE C/ LAYOUTS
<pre><head> <meta charset="utf-8" /> <meta name="viewport" content="width=device-width, initial-scale=1.0" /> <title>@ViewData["Title"] - TP12</title> <link rel="stylesheet" href="~/css/site.css" /> </head> <table style="width:100%; border-collapse: collapse;"> <tr> <td>Links</td> <td style="text-align:center;">As views são injetadas aqui por baixo</td> </tr> <tr> <td></td> <td> <p>@Html.ActionLink("Home", "Index", "Home")</p> <p>@Html.ActionLink("Privacy", "Privacy", "Home")</p> </td> </tr> <tr> <td></td> <td>@RenderBody()</td> </tr> </table> <footer> @RenderSection("Footer", required: false) </footer> </body> </html></pre>	<p>1. Inserir na _MyLayout uma secção footer com @RenderSection denominada Footer</p>	<p>2. Inserir uma secção denominada Footer na Razor View Index.cshtml</p>	

20

5.2.3.1 UTILIZAÇÃO DE LAYOUTS

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

As vistas não necessitam obrigatoriamente de especificar o layout usado se recorrerem ao layout por omissão especificado no ficheiro _ViewStart.cshtml

• Basta inserir o layout por omissão no ficheiro ~/Views/_ViewStart.cshtml *@

```
@{
  Layout = "~/Views/Shared/_MyLayout.cshtml";
}
```

21

5.2.3.4 UTILIZAÇÃO DO _VIEWSTART

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

1. Abrir o ficheiro _ViewStart da pasta Views

2. Inserir o path para o _MyLayout.cshtml no bloco de c#

```
@{
  Layout = "~/Views/Shared/_MyLayout.cshtml";
}
```

3. Remover os path para o ficheiro _MyLayout.cshtml dos cabeçalhos de c# nos ficheiros index.cshtml e Privacy.cshtml

4. Run & Try aqui vai encontrar as melhores fotos

Bem Vindo
Ao Fotoshopping

Links
Home
Privacy

As views são injetadas aqui por baixo

22

5.2.4 SINTAXE RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

- O Razor view engine
 - Fornece e executa o código HTML das vistas
 - Disponibiliza vários html tag helpers para a geração do HTML;
- A linguagem Razor é uma sintaxe para templates de etiquetas (Template markup syntax)
 - Não é uma nova linguagem;
 - Editável com qualquer editor de texto
 - Tem um bom intellisense (built in no VisualStudio)
 - Facilmente testável por unidades (unit-test)

@DateTime.Now
NEW RAZOR

Na sintaxe Razor os blocos de código começam com o carácter @ e não necessitam que o bloco tenha um fecho explícito.

23

5.2.5 LINGUAGEM RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

1. Abra a view PorTitulo.cshtml.
2. Declare e inicialize a variável someVar com o carácter '@'

```
@model A15_NomeApelido.Models.Foto
@{
  ViewData["Title"] = "PorTitulo";
  Layout = "~/Views/Shared/_Layout.cshtml";
  char someVar = '@';
}
```

@ Para valores (HTML Encoded)@

use @@ para valores (HTML encoded)

 Current time is: @DateTime.Now

 Not HTML encoded value: @Html.Raw(someVar)

3. Insira o código no final.
4. Run & Try

24

5.2.5 LINGUAGEM RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

```

@* @ {...} Para blocos de Código *@

{
    var fotoName = "North Sea";
    if (Model != null) {
        fotoName = Model.Titulo;
    }
    else if (ViewBag.FotoName != null) {
        fotoName = ViewBag.FotoName;
    }
    <p> A fotografia "@fotoName" foi adicionada à sua lista de compras</p>
}

```

3. Logo após o código do slide anterior, insira o código deste bloco
4. Run & Try

25

5.2.5 LINGUAGEM RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

```

public IActionResult ListarFotos()
{
    List<Foto> listaDeFotos = new List<Foto>();
    Foto f1 = new Foto{ Titulo = "North Sea", Codigo = "NS111"};
    Foto f2 = new Foto{ Titulo = "Turtle", Codigo = "TU222"};
    Foto f3 = new Foto { Titulo = "Sun Shining", Codigo = "SUSH333"};
    listaDeFotos.Add(f1);
    listaDeFotos.Add(f2);
    listaDeFotos.Add(f3);
    return View(listaDeFotos);
}

<p>@Html.ActionLink("Fotos", "ListarFotos")</p>

```

1. No controlador Home crie uma ação ListarFotos com este código

2. Adicione o ActionLink ao MyLayout.cshtml

3. Crie uma Razor View com View Name: ListarFotos Template: Details Model Class: Foto layout page: MyLayout

Add Razor View
View name: Index
Template: Details
Model class: Foto
Layout page: MyLayout

26

25

26

5.2.5 LINGUAGEM RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

```

@model IEnumerable<TP12.Models.Foto>
@{
    ViewData["Title"] = "Listar Fotos";
    Layout = "~/Views/Shared/_MyLayout.cshtml";
}


@if (Model.Count() == 0)
    {
        <p> Desculpe. não foram encontradas fotografias</p>
    }
    else
    {
        @: Lista de fotografias disponíveis:
        foreach (var foto in Model)
        {
            <br /><b>@foto.Codigo -- @foto.Titulo</b>
        }
    }


```

1. Substitua o código da View ListarFotos pelo código ao lado

As linhas de markup HTML podem ser inseridas em qualquer parte

Usamos @: para renderização de texto simples

27

5.2.5 LINGUAGEM RAZOR

5..NET CORE WEB
5.2 WEB SITE C/ LAYOUTS

```

@* Comentários em Razor *@

/*
 * Comentários multilinha
 que acordámos não usar em C# */
@* E como é com os "@" dos e-mails? *@
<p>
    O sinal que separa os nomes do e-mail dos domínios é: @<br/>
    E isto mostra como o razor pode ser espalhado:<br/>
    spam_them@gmail.com
</p>
@* Expressão de código explícita *@
<p>
    Preços actuais (0-10): @Model.Preco / 10.0 @* 6 / 10.0 *@
    Preços actuais (0-1): @Model.Preco / 10.0M @* 0.6 *@
    spam_them@Model.Preco @* spam_them@Model.Preco *@
    spam_them@Model.Preco @* spam_them6 *@

```

1. Adicione o código ao final do ficheiro PortTitulo.cshtml

PortTitulo

O Código da imagem Deserto é ABC000001
use @ para valores (HTML encoded)
Current time is: 07/12/2021 00:44:14
Not HTML encoded value: >

A fotografia "Deserto" foi adicionada à sua lista de compras
O sinal que separa os nomes de e-mail dos domínios é: @
Isto mostra como o razor pode ser espalhado:
spam_them@gmail.com

Preços actuais (0-10): 0 / 10.0
Preços actuais (0-1): 0 / 10.0M
spam_them@Model.Preco
spam_them@Model.Preco
spam_them@Model.Preco
spam_them6

2. Run & Try

28

27

28