07

View de Resumo

Transcrição

Temos a *view* de carrinho exibindo os itens que selecionamos na tela do carrossel e, pelo fluxo da aplicação, continuaremos clicando no botão "Preencher Cadastro", seguindo à tela em que preencheremos os campos com nossos dados pessoais, ao fim do qual poderemos clicar no botão "Finalizar Pedido".

Vamos exibir o resumo como se já tivéssemos preenchido os dados e finalizado o cadastro. O que faremos é uma pequena modificação na *view* do carrinho, em que temos o botão "Preencher Cadastro" que será substituído por outro, "Finalizar Pedido".

Pausaremos a aplicação e, em `Carrinho.cshtml`, faremos a alteração, além de substituirmos o valor de `asp-action`:

```
<div class="pull-right">
    <a class="btn btn-success" asp-action="carrossel">
        Adicionar Produtos
    </a>
    <a class="btn btn-success" asp-action="resumo">
        Finalizar Pedido
    </a>
</div>
```

Voltando à tela de navegação, iremos atualizá-la com "F5", e teremos o botão com a devida alteração. Clicaremos nele e iremos à página de resumo do pedido, da maneira como ela foi criada pela nossa webdesigner.

> As informações são falsas, e os dados pessoais que ali se encontram, como e-mail, endereço de entrega, nome, telefone, itens comprados, apenas simulam uma tela deste tipo em contexto de vida real.

Começaremos a modificar a *view* de resumo da compra para exibirmos os dados do modelo dentro do nosso HTML. Acessaremos `PedidoController.cs` para localizar a *action* de resumo; pausaremos a aplicação e injetaremos nesta *view* o `pedido` a ser exibido, o qual precisará ser obtido de algum lugar.

```
public IActionResult Resumo()
{
    Pedido pedido = pedidoRepository.GetPedido();
    return View(pedido);
}
```

Consultaremos o nosso repositório de pedido com `GetPedido()` e passaremos à nossa *view* de resumo da compra. Agora que estamos passando um modelo para dentro da *view*, teremos que modificá-la para recebermos estes dados. Em `Resumo.cshtml`, declararemos o tipo deste modelo, incluindo a diretiva também: `@model Pedido`.

Feito isso, começaremos a substituir as informações que estão na página pelos dados que vêm do modelo, como o número do pedido, por exemplo, cujo `123` trocaremos por uma expressão C#:

```
@model Pedido;
<h3>Resumo do Pedido</h3>

<div class="panel panel-default">
    <div class="panel-body">
        <div class="row">
            <div class="col-md-12">
                <h3>Nº do Pedido: @(Model.Id)</h3>
            </div>
        </div>
    </div>
</div>
```

Mais abaixo, encontramos informações pessoais como nome, telefone, e-mail e afins. Por enquanto, ignoraremos estes dados por ainda não estarmos preenchendo o cadastro. No fim do código, há outras informações, como nome do item e a quantidade correspondente.

Colapsaremos as linhas referentes aos produtos do carrinho, no caso, três, que no momento se encontram fixos na *view* e precisam ser trocados por **itens dinâmicos**. Para tal, manteremos apenas a primeira `div`, que deixaremos em um laço `foreach` para podermos varrer os itens do pedido e exibi-los na tela.

Para cada item em `Model.Itens`, será exibido um item dinamicamente, sendo assim substituiremos o nome do produto por `@item.Produto.Nome, e a quantidade por @item.Quantidade`. O código ficará da seguinte forma:

```
@foreach (var item in Model.Itens)
{
    <div class="row">
        <div class="col-md-10">
            <div>@item.Produto.Nome</div>
        </div>
        <div class="col-md-2">
            <div class="pull-right">@item.Quantidade</div>
        </div>
    </div>
}
```

Com isto, temos a *view* preparada. Rodaremos a aplicação novamente e criaremos um novo pedido, isto é, colocaremos os produtos no carrinho mais uma vez. Na *view* de carrossel, selecionaremos o primeiro, o segundo, o terceiro e o quarto produto. Na página de resumo do pedido, clicaremos em "Finalizar Pedido".

Na página a que somos redirecionados, temos o número do pedido, trazido do banco de dados por meio do repositório, e os quatro produtos selecionados por nós estão listados conforme esperado, com suas respectivas quantidades, que vieram do nosso modelo.