

Laboratório 2 (Opcional)

Objetivos:

O objetivo deste laboratório é criar uma listagem de produtos a partir de um arquivo XML, com efeitos de tela utilizando JQuery

1. Crie um novo projeto chamado Cap03_02, do tipo Asp .Net Web Application (.Net Framework). Mantenha ele vazio, mas habilite as pastas para o padrão MVC.

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 proporcionam uma experiência de usuário avançada que inclui interações no lado do cliente.

Autenticação
Sem Autenticação
[Alterar](#)

Adicionar as pastas e as principais referências

- ☐ Formulários da Web
- ☒ MVC
- ☐ API Web

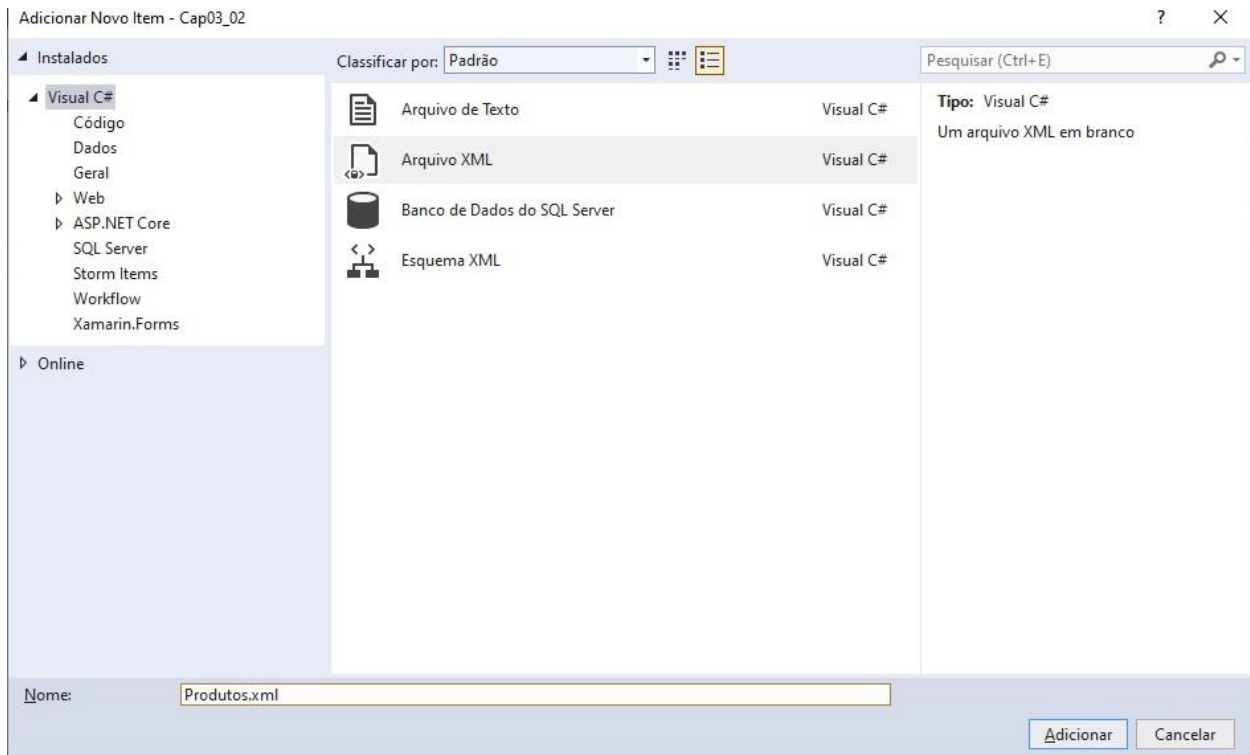
Avançado

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

WebApplication11.Tests

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

2. Na pasta App_Data, inclua um novo item, um arquivo XML, chamado Produtos:



3. Preencha o arquivo XML com o seguinte conteúdo (ele será nossa tabela do Banco de Dados):

```
<?xml version="1.0" encoding="utf-8" ?>
<Produtos>

  <Produto>
    <Codigo>1</Codigo>
    <Nome>Placa de Vídeo GTX 1050</Nome>
  </Produto>
  <Produto>
    <Codigo>2</Codigo>
    <Nome>HD SSD 1 TB</Nome>
  </Produto>
  <Produto>
    <Codigo>3</Codigo>
    <Nome>Memória RAM DDR4 512 MB</Nome>
  </Produto>
  <Produto>
    <Codigo>4</Codigo>
    <Nome>i5 9600 KF</Nome>
  </Produto>

</Produtos>
```

4. Crie um Controller chamado Produtos:

5. Altere no Controller Produtos a Action default:

```
public ActionResult ListaProdutos()
{
    return View();
}
```

6. Crie a View associada a esta Action:

7. O código HTML desta View será composto de 3 Divs, o bloco principal, outro onde ficará o título junto com o botão para expandir/ocultar e o último que irá guardar o conteúdo. Com isso em mente, monte a seguinte estrutura:

```
<body>

    <div id = "Conteudo">

        <div id ="Titulo">

            <b>Listagem dos Produtos Disponiveis</b>

            <a href="#" id ="link_abre">Ver os Produtos</a>
```

```

</div>

<br/>

<div id ="Listagem">

    Produtos para Compra:

</div>

</div>

</body>

```

8. Precisamos incluir o acesso a biblioteca JQuery. Você pode baixar os arquivos de origem e incluir no seu projeto, ou fazer uma chamada a um Link publico, que é o que faremos aqui. Inclua essa chamada no Head de sua View:

```

<script src="http://ajax.googleapis.com/ajax/libs/
    jquery/1.10.1/jquery.min.js"></script>

```

9. Ainda na View, inclua a chamada para o Namespace que contém as classes que vão permitir manipular o XML:

```

@{
    Layout = null;
}
@using System.Xml;

```

10. Antes de montar os efeitos JQuery, devemos listar os produtos disponiveis a partir do arquivo XML. Inclua o código na DIV Listagem:

```

<div id ="Listagem">

    Produtos para Compra:<br /><br/>

    @{

        string sCaminho = Server.MapPath("~/App_Data/produtos.xml");

        XmlDocument xmlDoc = new XmlDocument();
        xmlDoc.Load(sCaminho);

        XmlNodeList Lista = xmlDoc.GetElementsByTagName("Produto");

        foreach (XmlNode Item in Lista)
        {
            string sCodigo = Item["Codigo"].InnerText;

```

```

        string sNome = Item["Nome"].InnerText;

        <div class="table">
            <div class="row">
                <div class="cell1">@sCodigo</div>
                <div class="cell2">@sNome</div>
            </div>
        </div>
    }
}

</div>

```

11. Inclua o seguinte Estilo abaixo da tag de Script do JQuery:

```

<style type="text/css">

    .table {
        display: table;
        border: solid;
        border-width: thin;
    }

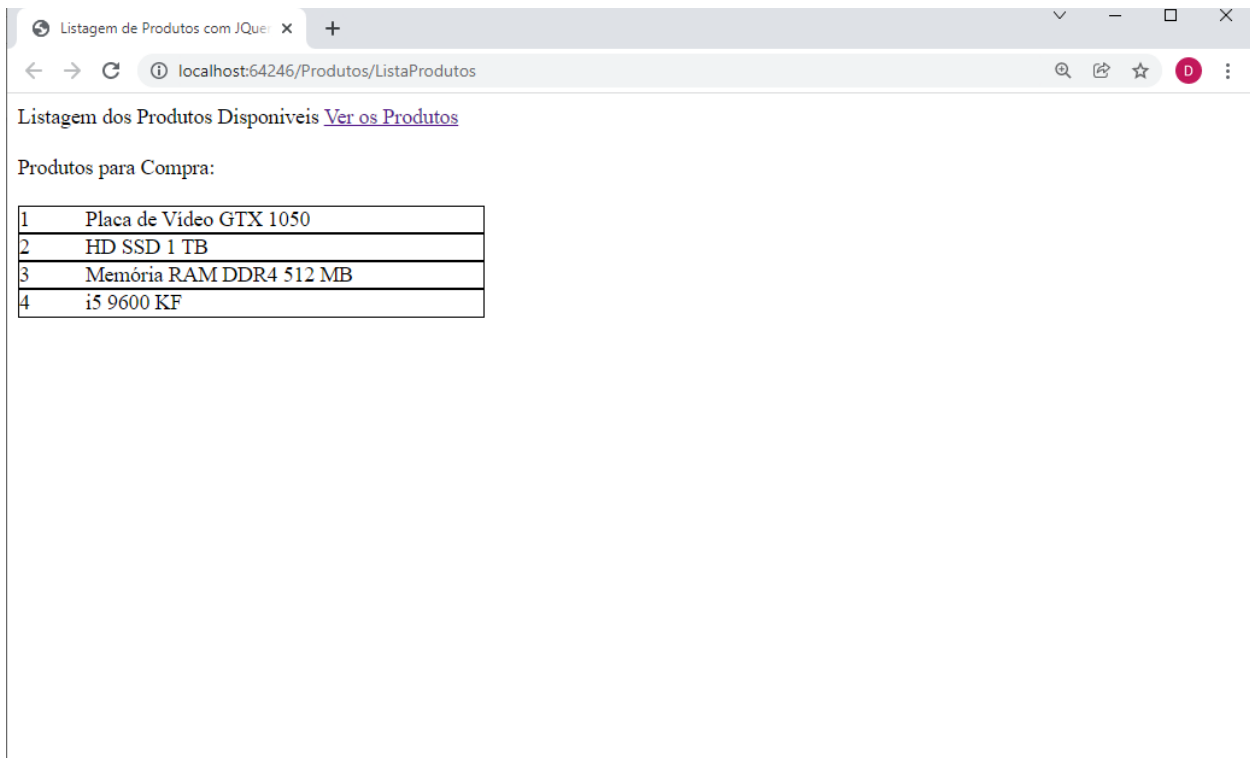
    .row {
        display: table-row;
    }

    .cell1 {
        display: table-cell;
        width: 50px;
    }

    .cell2 {
        display: table-cell;
        width: 300px;
    }
</style>

```

12. Teste e veja como está sua tela no momento:



13. Inclua esta nova seção de script JavaScript logo abaixo da declaração do JQuery.

```
<script type="text/javascript">

    // Verifica se o Documento já está carregado
    $(document).ready(function () {

        // Esconde a área de Listagem
        $("#Listagem").hide();

        // Define a função a ser executada ao clicar no Link
        $("#link_abre").click(function () {
            // Exibe a área de Listagem com efeito visual
            $("#Listagem").slideToggle("slow");
        });

    });

</script>
```

14. Inclua estes novos itens na área de Estilo já existente, formatando assim a tela.

```
#Conteudo {
    width: 450px;
    padding: 20px;
    background-color: #FC3;
}
```

```
#Titulo {
    padding: 10px;
    background-color: #FFF;
    font-weight: bold;
    font-size: large;
}

#link_abre {
    font-weight: normal;
    font-size: smaller;
}

a:link, a:visited {
    color: #000;
    border: #FF9900 1px solid;
    margin-left: 60px;
    padding: 10px;
    background-color: #FF9;
    text-decoration: none;
}

a:hover {
    color: #FFF;
    border: #FF9900 1px solid;
    margin-left: 60px;
    padding: 10px;
    background-color: #000;
    text-decoration: none;
}
```

15. Execute seu projeto e veja o funcionamento do JQuery.

