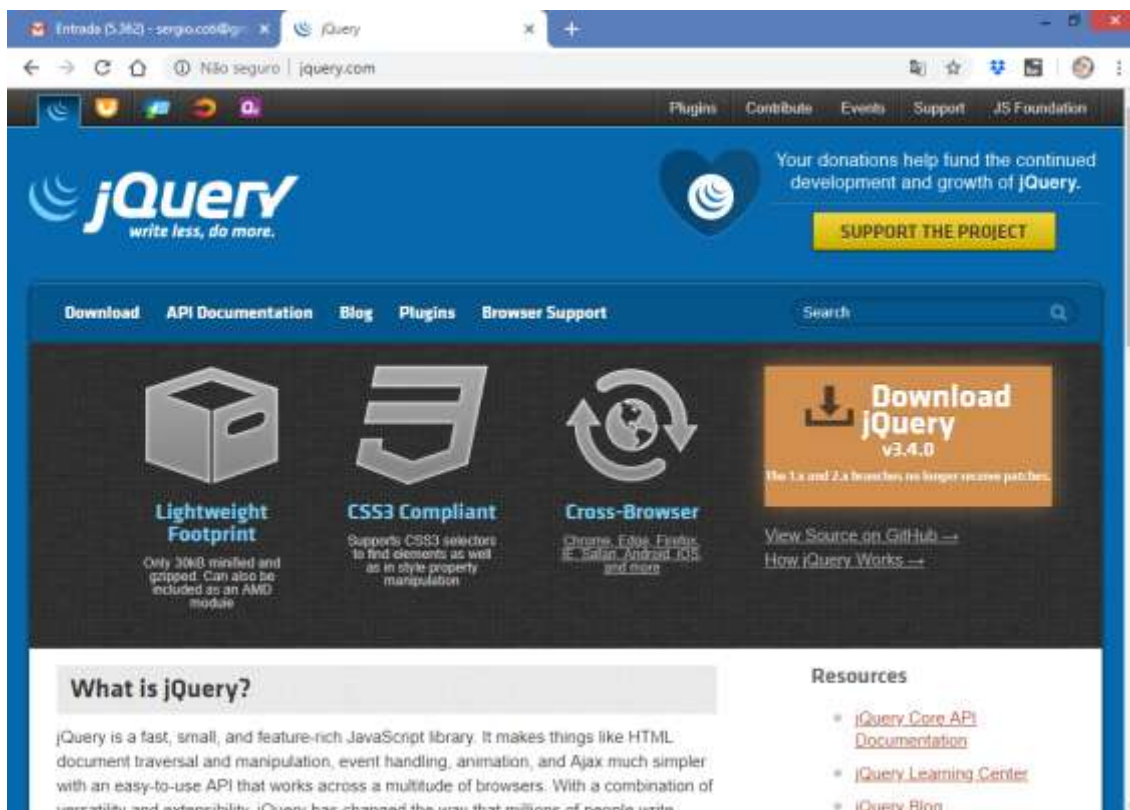


JQuery (<http://jquery.com/>)

Biblioteca para desenvolvimento baseado em linguagem JavaScript.

É composto de uma série de componentes que complementam e adicionam recursos ao JavaScript convencional.

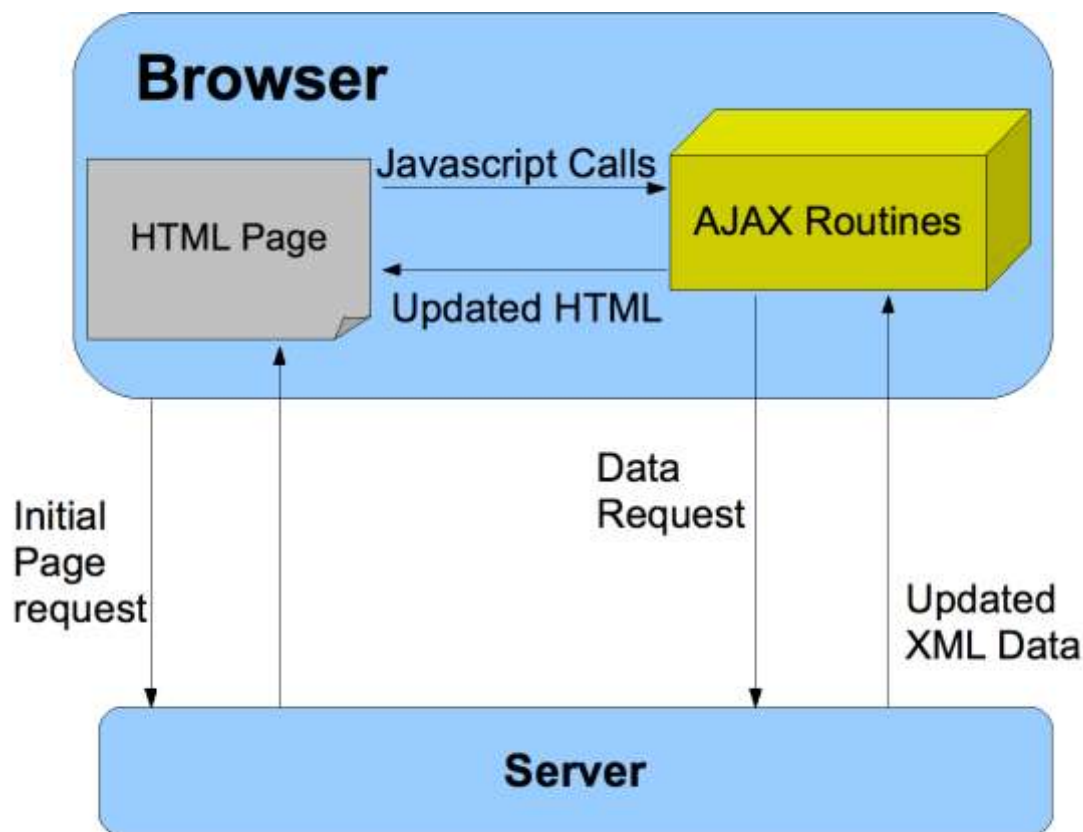


O uso do JavaScript em páginas web além de oferecer recursos para criação de componentes de usabilidade (UX), também permite aos desenvolvedores utilizarem o AJAX.

Ajax (**Assynchronous Javascript and XML**)

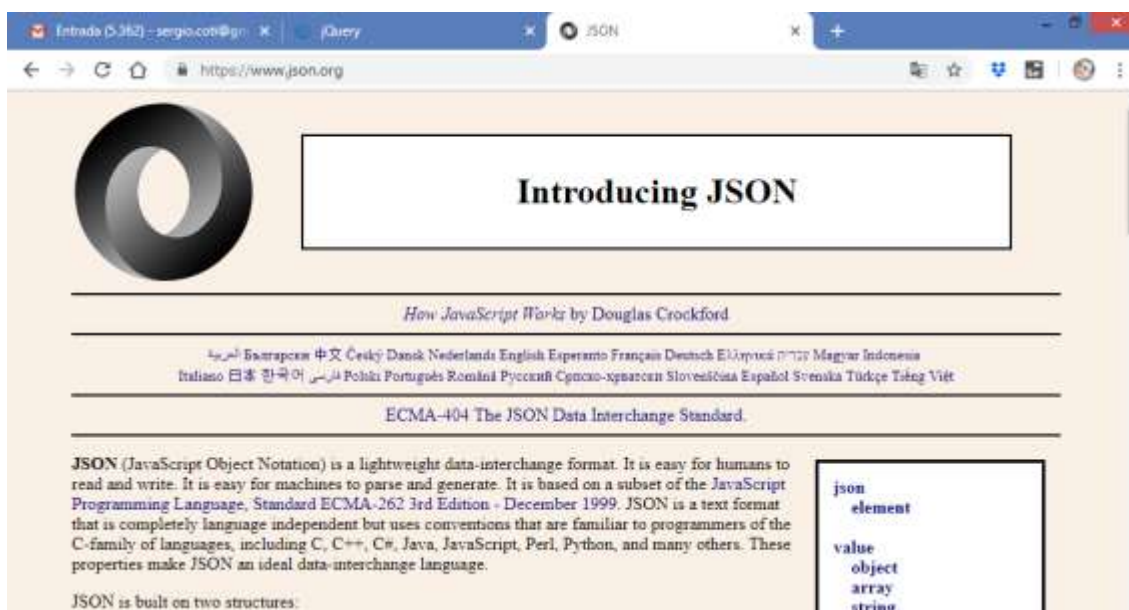
AJAX é um termo designado para definir o uso de Javascript para realizar a comunicação entre cliente (página) e servidor de forma assíncrona.

Em uma requisição AJAX, quando o usuário realiza um evento na página, uma função Java script é executada e esta função faz a chamada ao servidor e aguarda a resposta. Neste caso ao invés de toda a página ser renderizada (como em uma ação síncrona) somente uma parte da página HTML será renderizada conforme o código Java script definir.



O termo AJAX usa o XML como formato para envio e recebimento de dados entre o cliente e o servidor, porem, hoje a maioria das aplicações web tem abandonado o XML para usar no lugar dele o formato JSON.

JSON (<https://www.json.org/>)



Utilizando o JQuery para realizar o cadastro do produto:

Primeiro, precisamos definir um local dentro da página de Layout para inserir conteúdo JavaScript proveniente das demais páginas:

```
<!DOCTYPE html>

<html>
<head>
    <meta name="viewport" content="width=device-width" />
    <title>Projeto</title>

    <!-- Folhas de estilo CSS -->
    <link href="~/Content/bootstrap.min.css" rel="stylesheet" />

</head>
<body>

    <div class="container">

        <div class="card card-body bg-dark">
            <h3 class="text-white">Sistema de Controle de Produtos</h3>
            <p class="text-white">
                Aula de C# WebDeveloper - COTI Informática
            </p>
        </div>

        <br/>

        @RenderBody()

    </div>

    <!-- arquivos javascript -->
    <script src="~/Scripts/jquery-3.0.0.min.js"></script>
    <script src="~/Scripts/bootstrap.min.js"></script>

    @RenderSection("scripts", required : false)

</body>
</html>
```

Na página de cadastro de produto:
/Views/Produto/Cadastro.cshtml

```
@{
    ViewBag.Title = "Cadastro";
    Layout = "~/Views/Shared/Layout.cshtml";
}

@section scripts{

    <script>
```

```
//quando a página for carregada faça..
$(document).ready(function () {

    //executar um evento quando o usuário
    //clique no botão de cadastro de produto
    $("#btncadastro").click(function () {

        //criar um objeto JSON para armazenar os dados
        //de cada campo da página de cadastro de produto
        var dados = {
            Nome: $("#nome").val(),
            Preço: $("#preco").val(),
            Quantidade: $("#quantidade").val()
        };

        //imprimindo o objeto JSON no console do navegador
        console.log(dados);

    });
});

</script>
}

<h4>Cadastro de Produtos</h4>
<a href="/Home/Index">Página inicial</a>
<hr/>

<div class="row">

    <div class="col-md-4">

        <label>Nome do Produto:</label>
        <input type="text" id="nome" class="form-control"
            placeholder="Nome do Produto"/>
        <br/>

        <label>Preço:</label>
        <input type="text" id="preco" class="form-control"
            placeholder="Preço"/>
        <br/>

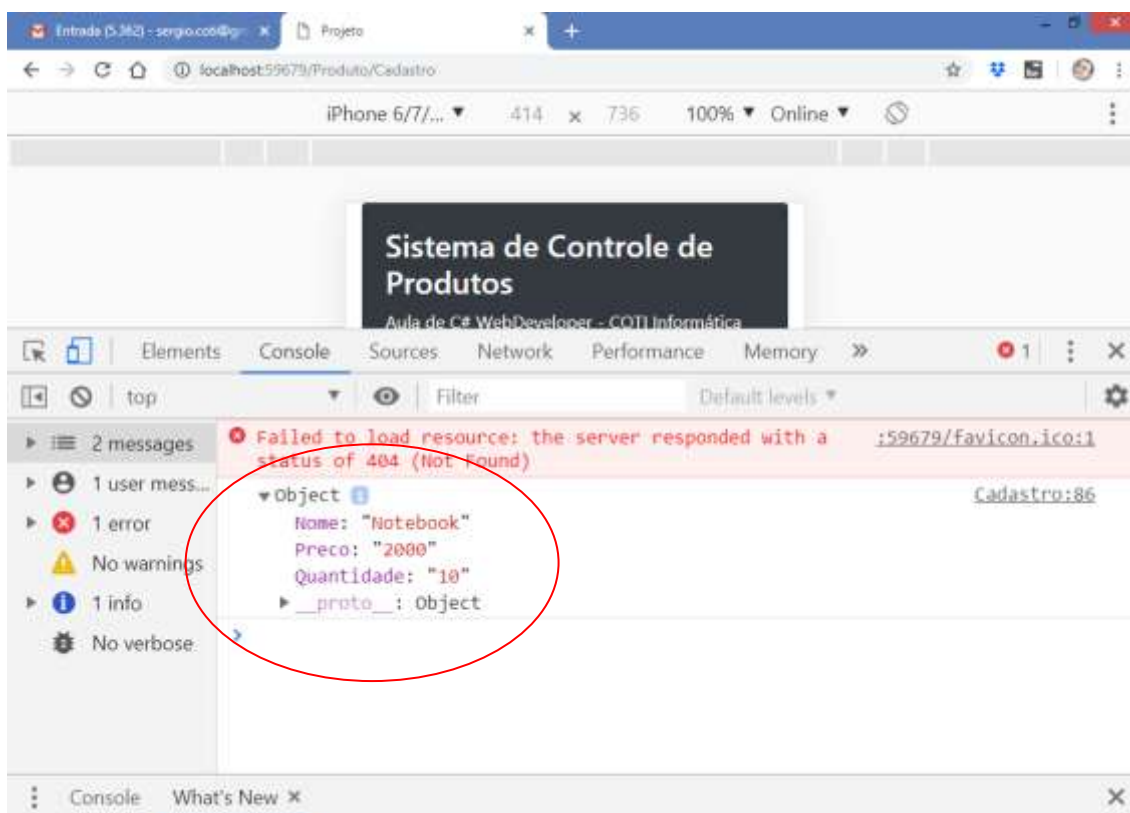
        <label>Quantidade:</label>
        <input type="text" id="quantidade" class="form-control"
            placeholder="Quantidade"/>
        <br/>

        <button id="btncadastro" class="btn btn-success">
            Cadastrar Produto
        </button>

    </div>

</div>
```

Visualizando os dados no Console:



Fazendo uma requisição AJAX para o controller:

```
@{
    ViewBag.Title = "Cadastro";
    Layout = "~/Views/Shared/Layout.cshtml";
}

@section scripts{

    <script>

        //quando a página for carregada faça..
        $(document).ready(function () {

            //executar um evento quando o usuário
            //clicar no botão de cadastro de produto
            $("#btncadastro").click(function () {

                //criar um objeto JSON para armazenar os dados
                //de cada campo da página de cadastro de produto

                var dados = {
                    Nome: $("#nome").val(),
                    Preço: $("#preco").val(),
                    Quantidade: $("#quantidade").val()
                };
            });
        });
    </script>
}
```

```
//requisição AJAX para o servidor
$.ajax({
    type: "POST",
    url: "/Produto/CadastrarProduto",
    data: dados,
    success: function () {

    },
    error: function (e) {
        $("#mensagem").html(e.responseText);
    }
});

});
})

</script>

}

<h4>Cadastro de Produtos</h4>
<a href="/Home/Index">Página inicial</a>
<hr/>

<div class="row">
    <div class="col-md-4">

        <label>Nome do Produto:</label>
        <input type="text" id="nome" class="form-control"
            placeholder="Nome do Produto"/>
        <br/>

        <label>Preço:</label>
        <input type="text" id="preco" class="form-control"
            placeholder="Preço"/>
        <br/>

        <label>Quantidade:</label>
        <input type="text" id="quantidade" class="form-control"
            placeholder="Quantidade"/>
        <br/>

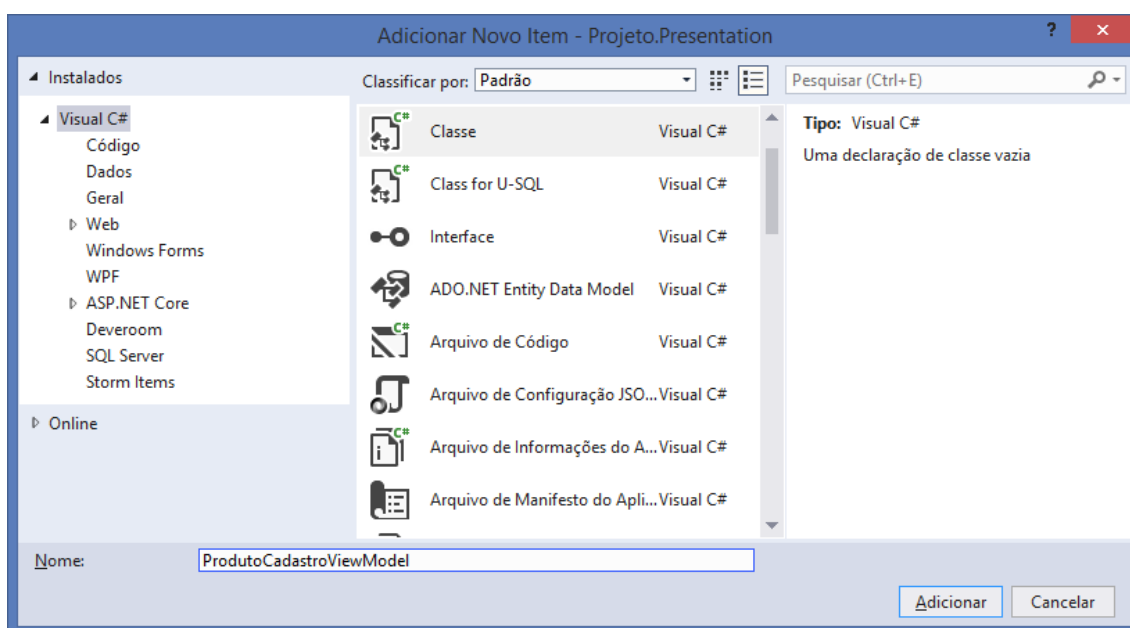
        <button id="btncadastro" class="btn btn-success">
            Cadastrar Produto
        </button>
        <br/>
        <br/>

        <div id="mensagem"></div>

    </div>
</div>
```



Criando uma classe de modelo para resgatar e validar os dados enviados pelo JQuery:



Regra: Os campos declarados na classe Model deverão ser iguais aos campos declarados no JSON:

```
var dados = {  
    Nome: $("#nome").val(),  
    Preco: $("#preco").val(),  
    Quantidade: $("#quantidade").val()  
};  
  
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.ComponentModel.DataAnnotations; //validações  
  
namespace Projeto.Presentation.Models  
{  
    public class ProdutoCadastroViewModel  
    {  
        [MinLength(4, ErrorMessage = "Informe no mínimo {1} caracteres.")]  
        [MaxLength(100, ErrorMessage = "Informe no máximo {1} caracteres.")]  
        [Required(ErrorMessage = "Por favor, informe o nome do produto.")]  
        public string Nome { get; set; }  
  
        [Required(ErrorMessage = "Por favor, informe o preço do produto.")]  
        public decimal Preco { get; set; }  
  
        [Required(ErrorMessage = "Por favor, informe a quantidade do produto.")]  
        public int Quantidade { get; set; }  
    }  
}
```

Criando o método na classe de controle para receber os dados enviados pela função AJAX do JQuery

JsonResult

Utilizado para métodos na classe de controle que recebem chamadas de funções AJAX baseadas em javascript

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.Mvc;  
using Projeto.Presentation.Models;  
  
namespace Projeto.Presentation.Controllers  
{  
    public class ProdutoController : Controller  
    {  

```



```
// GET: Produto/Cadastro
public ActionResult Cadastro()
{
    return View();
}

// GET: Produto/Consulta
public ActionResult Consulta()
{
    return View();
}

//método que será executado pela chamada AJAX feita na página
public JsonResult CadastrarProduto(ProdutoCadastroViewModel model)
{
    return Json(null);
}
}
```

Realizando o cadastro do Produto:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Projeto.Presentation.Models;
using Projeto.BLL;
using Projeto.Entities;

namespace Projeto.Presentation.Controllers
{
    public class ProdutoController : Controller
    {
        // GET: Produto/Cadastro
        public ActionResult Cadastro()
        {
            return View();
        }

        // GET: Produto/Consulta
        public ActionResult Consulta()
        {
            return View();
        }

        //método que será executado pela chamada AJAX feita na página
        public JsonResult CadastrarProduto(ProdutoCadastroViewModel model)
        {
            //verificar se os dados da model passaram nas regras de validação
            if(ModelState.IsValid)
            {
                try
                {
                    Produto produto = new Produto();
                    produto.Nome = model.Nome;
                    produto.Preco = model.Preco;
                }
            }
        }
    }
}
```

```

        produto.Quantidade = model.Quantidade;

        ProdutoBusiness business = new ProdutoBusiness();
        business.CadastrarProduto(produto);

        return Json($"Produto {produto.Nome},
                    cadastrado com sucesso.");
    }
    catch(Exception e)
    {
        return Json(e.Message);
    }
}
else
{
    //retornar as mensagens de erro de validação
    return Json(ModelState);
}
}
}
}
}

```

Voltando ao JQuery:

```
@{
```

```
    ViewBag.Title = "Cadastro";
    Layout = "~/Views/Shared/Layout.cshtml";
}
```

```
}
```

```
@section scripts{
```

```
<script>
```

```
    //quando a página for carregada faça..
    $(document).ready(function () {
```

```
        //executar um evento quando o usuário
        //clicar no botão de cadastro de produto
        $("#btncadastro").click(function () {
```

```
            //criar um objeto JSON para armazenar os dados
            //de cada campo da página de cadastro de produto
```

```
            var dados = {
                Nome: $("#nome").val(),
                Preco: $("#preco").val(),
                Quantidade: $("#quantidade").val()
            };

```

```
            //requisição AJAX para o servidor
            $.ajax({
                type: "POST",
                url: "/Produto/CadastrarProduto",
                data: dados,
                success: function (obj) { //promisse

```

```

        $("#mensagem").html(obj);
    },
    error: function (e) { //promisse
        $("#mensagem").html(e.responseText);
    }
});

});
})

</script>
}

<h4>Cadastro de Produtos</h4>
<a href="/Home/Index">Página inicial</a>
<hr/>

<div class="row">

    <div class="col-md-4">

        <label>Nome do Produto:</label>
        <input type="text" id="nome" class="form-control"
            placeholder="Nome do Produto"/>
        <br/>

        <label>Preço:</label>
        <input type="text" id="preco" class="form-control"
            placeholder="Preço"/>
        <br/>

        <label>Quantidade:</label>
        <input type="text" id="quantidade" class="form-control"
            placeholder="Quantidade"/>
        <br/>

        <button id="btncadastro" class="btn btn-success">
            Cadastrar Produto
        </button>
        <br/>
        <br/>

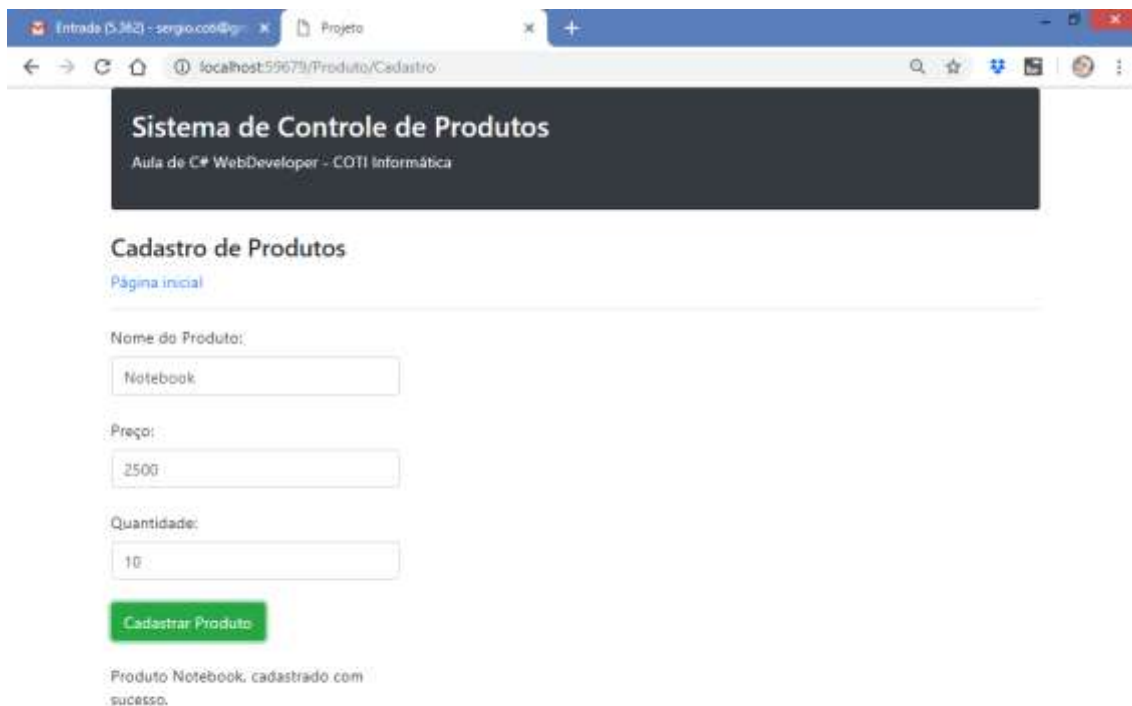
        <div id="mensagem"></div>

    </div>

</div>

```

Executando:



Sistema de Controle de Produtos
Aula de C# WebDeveloper - COTI Informática

Cadastro de Produtos
[Página inicial](#)

Nome do Produto:

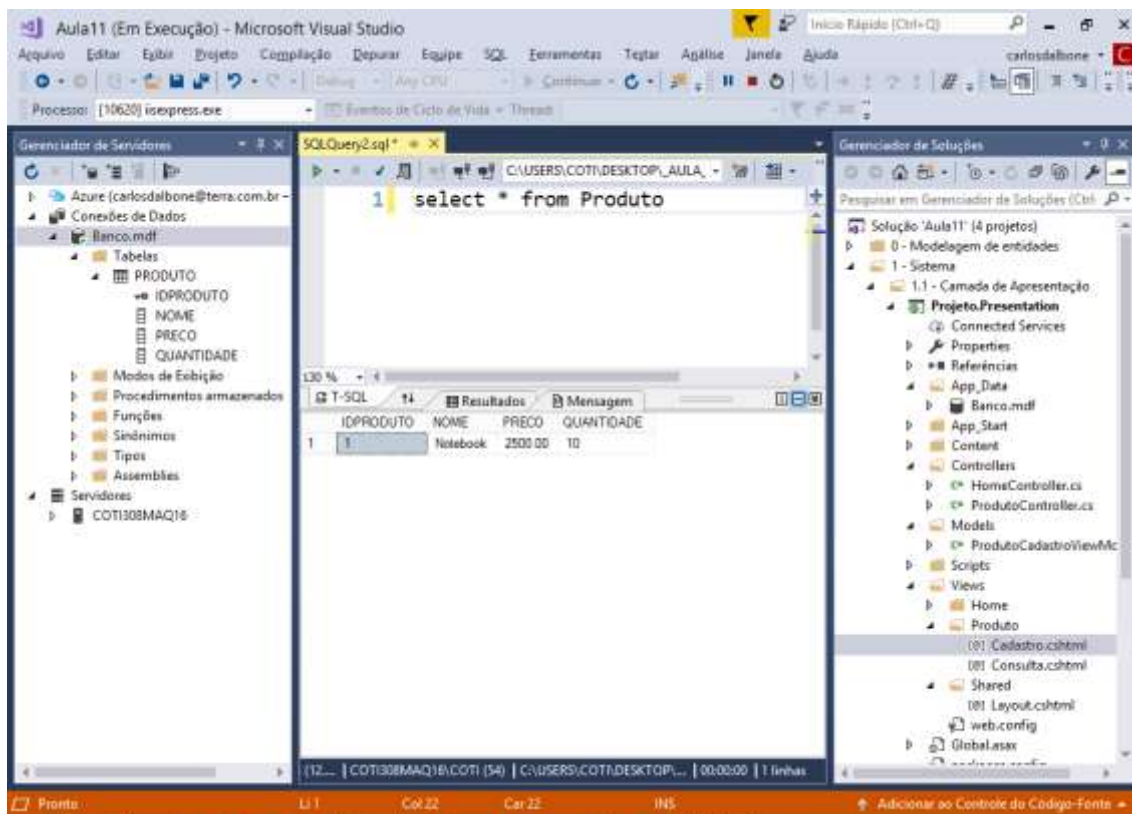
Preço:

Quantidade:

Cadastrar Produto

Produto Notebook, cadastrado com sucesso.

No banco de dados:



SQLQuery2.sql

```
1 select * from Produto
```

IDPRODUTO	NOME	PRECO	QUANTIDADE
1	Notebook	2500.00	10

Retornando as mensagens de erro de validação:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Projeto.Presentation.Models;
using Projeto.BLL;
using Projeto.Entities;

namespace Projeto.Presentation.Controllers
{
    public class ProdutoController : Controller
    {
        // GET: Produto/Cadastro
        public ActionResult Cadastro()
        {
            return View();
        }

        // GET: Produto/Consulta
        public ActionResult Consulta()
        {
            return View();
        }

        //método que será executado pela chamada AJAX feita na página
        public JsonResult CadastrarProduto(ProdutoCadastroViewModel model)
        {
            //verificar se os dados da model passaram nas regras de validação
            if(ModelState.IsValid)
            {
                try
                {
                    Produto produto = new Produto();
                    produto.Nome = model.Nome;
                    produto.Preco = model.Preco;
                    produto.Quantidade = model.Quantidade;

                    ProdutoBusiness business = new ProdutoBusiness();
                    business.CadastrarProduto(produto);

                    return Json($"Produto {produto.Nome},  
cadastrado com sucesso.");
                }
                catch(Exception e)
                {
                    Response.StatusCode = 500; //INTERNAL SERVER ERROR
                    return Json(e.Message);
                }
            }
        }
    }
}
```

```

else
{
    //forçar o JsonResult a retornar um status de erro
    //para a função JQUERY...
    Response.StatusCode = 400; //BAD REQUEST
    //retornar as mensagens de erro de validação
    List<string> erros = new List<string>();
    foreach(var state in ModelState)
    {
        if(state.Value.Errors.Count > 0)
        {
            erros.Add(state.Value.Errors
                .Select(e => e.ErrorMessage)
                .FirstOrDefault());
        }
    }
    return Json(erros);
}
}
}
}

```

Voltando na página:

Exibindo as mensagens de validação na página:

```

@{
    ViewBag.Title = "Cadastro";
    Layout = "~/Views/Shared/Layout.cshtml";
}

@section scripts{

    <script>

        //quando a página for carregada faça..
        $(document).ready(function () {

            //executar um evento quando o usuário
            //clique no botão de cadastro de produto
            $("#btncadastro").click(function () {

                //criar um objeto JSON para armazenar os dados
                //de cada campo da página de cadastro de produto

                var dados = {
                    Nome: $("#nome").val(),
                    Preco: $("#preco").val(),
                    Quantidade: $("#quantidade").val()
                };

                //requisição AJAX para o servidor
                $.ajax({
                    type: "POST",
                    url: "/Produto/CadastrarProduto",
                    data: dados,
                    success: function (obj) { //promisse

```

```

        $("#mensagem").html(obj);

        //limpar os campos da página
        $("#nome").val("");
        $("#preco").val("");
        $("#quantidade").val("");
    },
    error: function (e) { //promisse

        //erro de validação
        if (e.status == 400) {

            //limpar o elemento 'mensagem'
            $("#mensagem").html("");

            console.log(e.responseJSON);
        }
        //erro interno do servidor
        else if (e.status == 500){

            $("#mensagem").html(e.responseText);
        }
    }
});
});
})
</script>
}

<h4>Cadastro de Produtos</h4>
<a href="/Home/Index">Página inicial</a>
<hr/>

<div class="row">
    <div class="col-md-4">

        <label>Nome do Produto:</label>
        <input type="text" id="nome" class="form-control"
            placeholder="Nome do Produto"/>
        <br/>

        <label>Preço:</label>
        <input type="text" id="preco" class="form-control"
            placeholder="Preço"/>
        <br/>

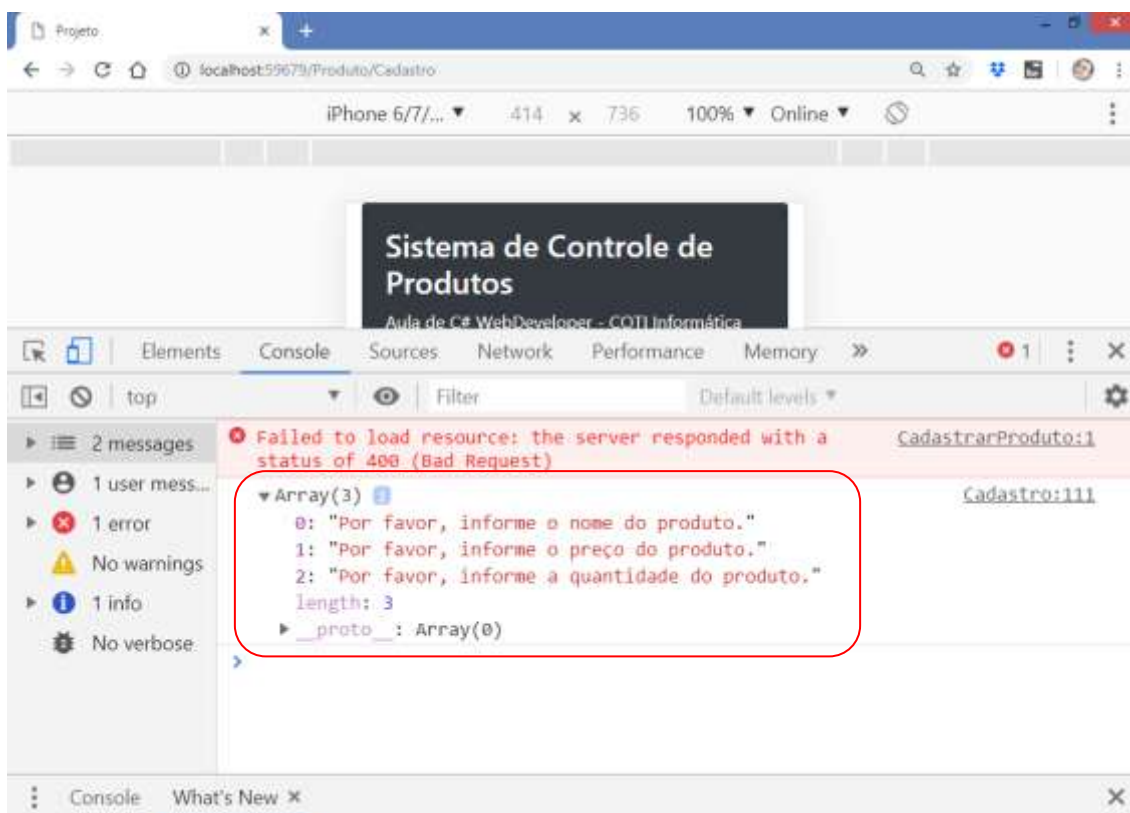
        <label>Quantidade:</label>
        <input type="text" id="quantidade" class="form-control"
            placeholder="Quantidade"/>
        <br/>

        <button id="btncadastro" class="btn btn-success">
            Cadastrar Produto
        </button>
        <br/>
        <br/>

        <div id="mensagem"></div>
    </div>
</div>

```


No console:



@{

```
ViewBag.Title = "Cadastro";
Layout = "~/Views/Shared/Layout.cshtml";
```

}

@section scripts{

<script>

//quando a página for carregada faça..

```
$(document).ready(function () {
```

//executar um evento quando o usuário

//clicar no botão de cadastro de produto

```
$("#btncadastro").click(function () {
```

//criar um objeto JSON para armazenar os dados

//de cada campo da página de cadastro de produto

```
var dados = {
    Nome: $("#nome").val(),
    Preço: $("#preco").val(),
    Quantidade: $("#quantidade").val()
};
```

```
$("#erros").html(""); //limpar as mensagens de erro
```

//requisição AJAX para o servidor

```
$.ajax({
    type: "POST",
```

```

url: "/Produto/CadastrarProduto",
data: dados,
success: function (obj) { //promisse

    $("#mensagem").html(obj);

    //limpar os campos da página
    $("#nome").val("");
    $("#preco").val("");
    $("#quantidade").val("");
},
error: function (e) { //promisse

    //erro de validação
    if (e.status == 400) {

        //limpar o elemento 'mensagem'
        $("#mensagem").html("");

        //varrer as mensagens de erro
        var conteudo = "";
        $.each(e.responseJSON, function (i, msg) {
            conteudo += msg + "<br/>";
        });

        $("#erros").html(conteudo);
    }
    //erro interno do servidor
    else if (e.status == 500){

        $("#mensagem").html(e.responseText);
    }
    }
});
});

</script>
}

<h4>Cadastro de Produtos</h4>
<a href="/Home/Index">Página inicial</a>
<hr/>

<div class="row">
    <div class="col-md-4">

        <div id="erros" class="text-danger"></div>

        <label>Nome do Produto:</label>
        <input type="text" id="nome" class="form-control"
            placeholder="Nome do Produto"/>
        <br/>

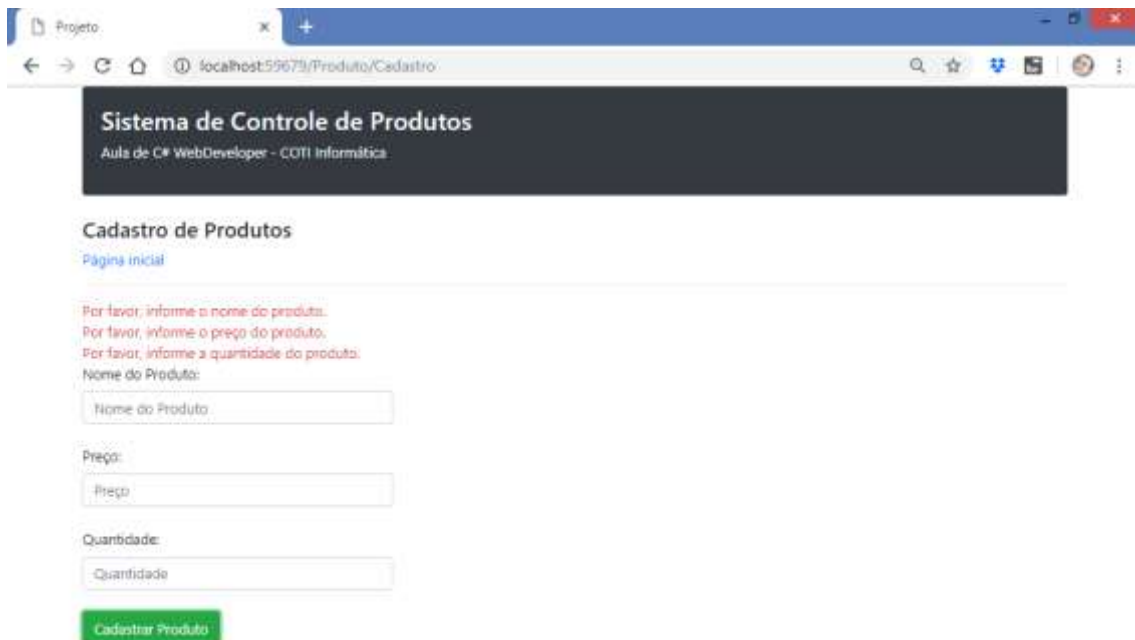
        <label>Preço:</label>
        <input type="text" id="preco" class="form-control"
            placeholder="Preço"/>
        <br/>
    
```

```
<label>Quantidade:</label>
<input type="text" id="quantidade" class="form-control"
      placeholder="Quantidade"/>
<br/>

<button id="btncadastro" class="btn btn-success">
  Cadastrar Produto
</button>
<br/>
<br/>

<div id="mensagem"></div>

</div>
</div>
```



Projeto

localhost:59679/Produto/Cadastro

Sistema de Controle de Produtos

Aula de C# WebDeveloper - COTI Informática

Cadastro de Produtos

[Página inicial](#)

Por favor, informe o nome do produto.
Por favor, informe o preço do produto.
Por favor, informe a quantidade do produto.

Nome do Produto:

Preço:

Quantidade:

Cadastrar Produto