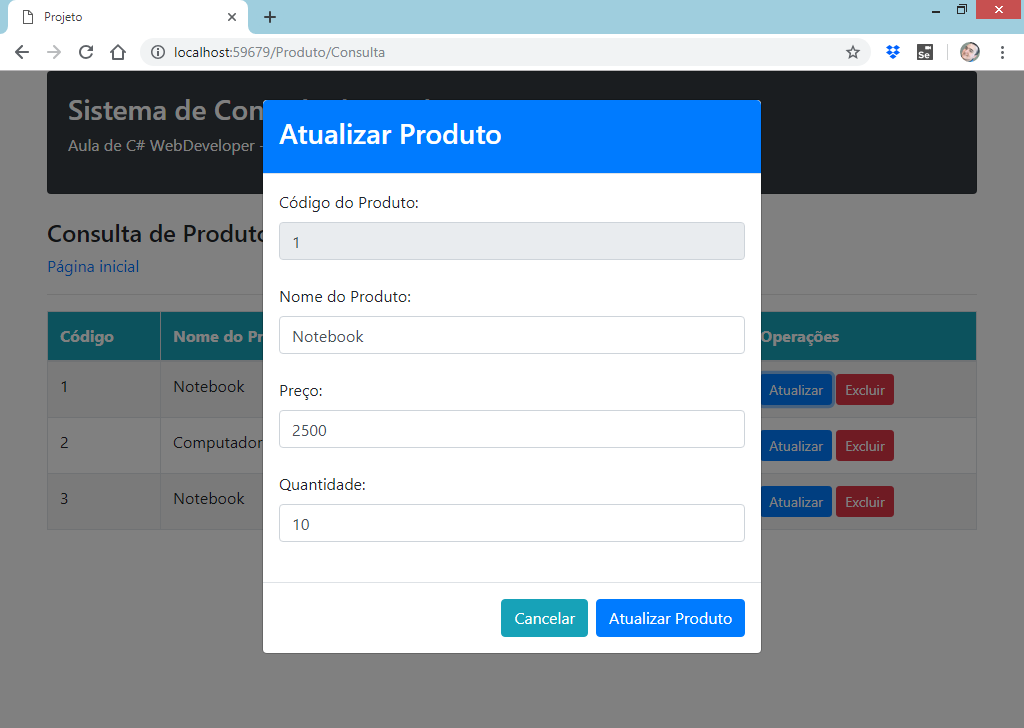
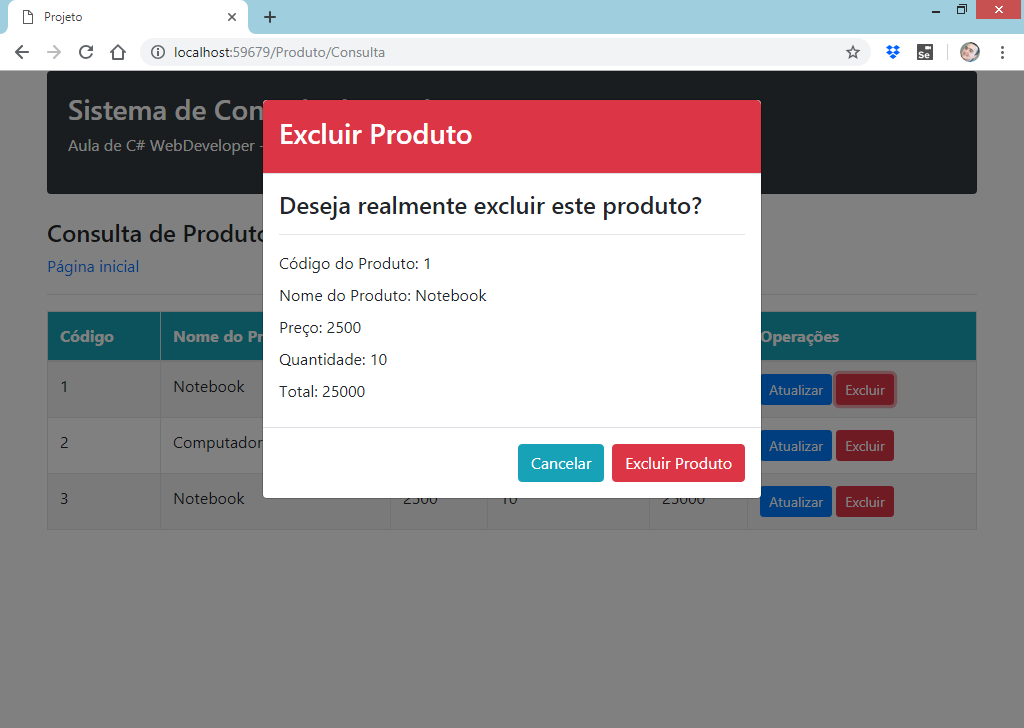
**Janelas modais de exclusão e edição:**





**Implementando o serviço de exclusão de produto:**

**Passo 1)** Criar um método **JsonResult** na classe   
**ProdutoController** para implementar a exclusão 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)

{

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);

}

}

//método para retornar a consulta de produtos para o jquery

public JsonResult ConsultarProdutos()

{

try

{

List<ProdutoConsultaViewModel> lista

= new List<ProdutoConsultaViewModel>();

//executar a consulta de produtos na camada de regras de negócio

ProdutoBusiness business = new ProdutoBusiness();

foreach(Produto produto in business.ConsultarTodos())

{

ProdutoConsultaViewModel model

= new ProdutoConsultaViewModel();

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

lista.Add(model); //adicionar na lista

}

return Json(lista); //retornando a lista

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

//retornando mensagem de erro..

return Json(e.Message);

}

}

//método pra consultar 1 produto pelo id

public JsonResult ConsultarProdutoPorId(int id)

{

try

{

ProdutoConsultaViewModel model = new ProdutoConsultaViewModel();

//buscar o produto pelo id

ProdutoBusiness business = new ProdutoBusiness();

Produto produto = business.ConsultarPorId(id);

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

return Json(model); //retornando o objeto model..

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

return Json(e.Message); //mensagem de erro

}

}

**//método para excluir um produto**

**public JsonResult ExcluirProduto(int id)**

**{**

**try**

**{**

**ProdutoBusiness business = new ProdutoBusiness();**

**business.ExcluirProduto(id);**

**return Json("Produto excluído com sucesso.");**

**}**

**catch(Exception e)**

**{**

**Response.StatusCode = 500; //INTERNAL SERVER ERROR**

**return Json(e.Message); //mensagem de erro**

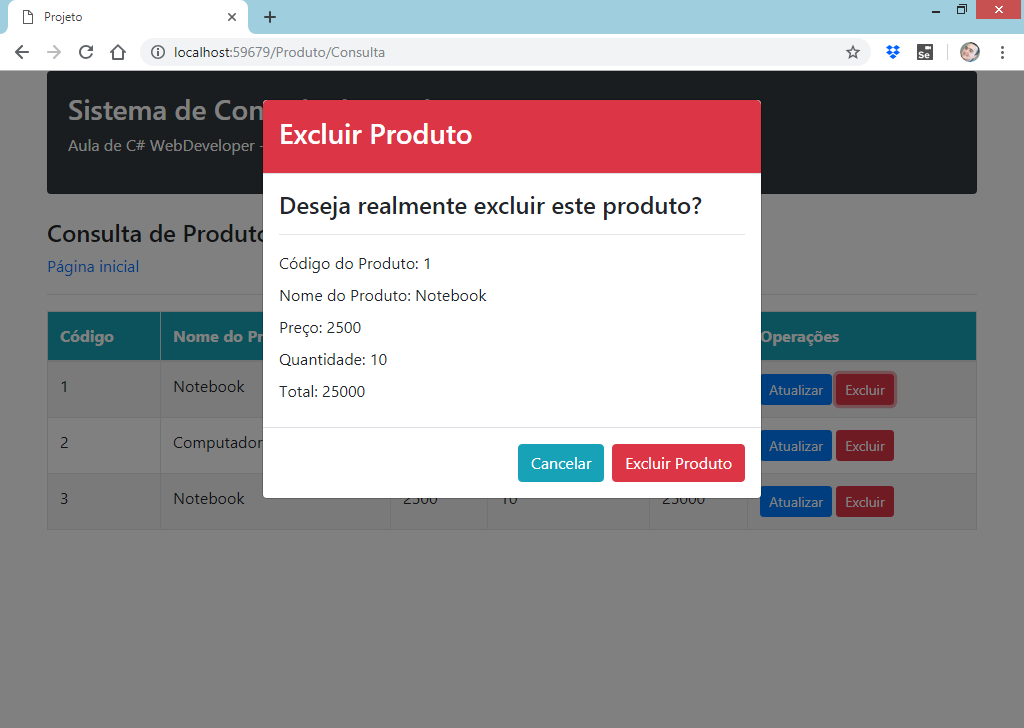
**}**

**}**

}

}

**Criando o evento para exclusão do produto:**



**Voltando para a página de consulta:**

@{

ViewBag.Title = "Consulta";

Layout = "~/Views/Shared/Layout.cshtml";

}

@section scripts{

<script>

**//função para excluir o produto**

**function excluirProduto() {**

**$.ajax({**

**type: "POST",**

**url: "/Produto/ExcluirProduto",**

**data: { id: $("#idproduto\_exclusao").html() },**

**success: function (msg) {**

**//imprimindo mensagem de sucesso..**

**$("#mensagem").html(msg);**

**//fechar a janela modal de exclusão..**

**$("#janelaexclusao").modal("hide");**

**//executar a consulta de produtos novamente**

**consultarProdutos();**

**},**

**error: function (e) {**

**console.log(e);**

**}**

**});**

**};**

//função para exibir os dados do produto pelo id

function obterProduto(idProduto) {

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutoPorId",

data: { id: idProduto },

success: function (obj) {

$("#idproduto\_edicao").val(obj.IdProduto);

$("#nome\_edicao").val(obj.Nome);

$("#preco\_edicao").val(obj.Preco);

$("#quantidade\_edicao").val(obj.Quantidade);

$("#idproduto\_exclusao").html(obj.IdProduto);

$("#nome\_exclusao").html(obj.Nome);

$("#preco\_exclusao").html(obj.Preco);

$("#quantidade\_exclusao").html(obj.Quantidade);

$("#total\_exclusao").html(obj.Total);

},

error: function (e) {

console.log(e);

}

})

}

//função para executar a consulta de produtos

function consultarProdutos() {

//requisição AJAX ao controller:

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutos",

data: {}, //vazio

success: function (lista) {

var conteudo = "";

$.each(lista, function (i, item) {

conteudo += "<tr>";

conteudo += "<td>" + item.IdProduto + "</td>";

conteudo += "<td>" + item.Nome + "</td>";

conteudo += "<td>" + item.Preco + "</td>";

conteudo += "<td>" + item.Quantidade + "</td>";

conteudo += "<td>" + item.Total + "</td>";

conteudo += "<td>";

conteudo += "<button onclick='obterProduto("+ item.IdProduto +")' class='btn btn-primary

btn-sm' data-target='#janelaedicao'

data-toggle='modal'>Atualizar</button>";

conteudo += "&nbsp;"; //espaço

conteudo += "<button onclick='obterProduto("+

item.IdProduto +")' class='btn btn-danger

btn-sm' data-target='#janelaexclusao'

data-toggle='modal'>Excluir</button>";

conteudo += "</td>";

conteudo += "</tr>";

});

$("#tabela tbody").html(conteudo);

},

error: function (e) {

console.log(e);

}

});

};

//quando a página for carregada, execute..

$(document).ready(function () {

consultarProdutos(); //executando a função

//criando um evento para realizar a exclusão..

$("#btnexclusao").click(function () {

excluirProduto(); //executando a função

});

});

</script>

}

<h4>Consulta de Produtos</h4>

<a href="/Home/Index">Página inicial</a>

<hr/>

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

<table id="tabela" class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th class="bg-info text-white">Código</th>

<th class="bg-info text-white">Nome do Produto</th>

<th class="bg-info text-white">Preço</th>

<th class="bg-info text-white">Quantidade</th>

<th class="bg-info text-white">Total</th>

<th class="bg-info text-white">Operações</th>

</tr>

</thead>

<tbody>

</tbody>

</table>

<!-- Janela modal para edição de produto -->

<div id="janelaedicao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-primary">

<h3 class="text-white">Atualizar Produto</h3>

</div>

<div class="modal-body">

<label>Código do Produto:</label>

<input type="text" id="idproduto\_edicao"

class="form-control" readonly="readonly"/>

<br/>

<label>Nome do Produto:</label>

<input type="text" id="nome\_edicao"

class="form-control" />

<br />

<label>Preço:</label>

<input type="text" id="preco\_edicao"

class="form-control" />

<br />

<label>Quantidade:</label>

<input type="text" id="quantidade\_edicao"

class="form-control" />

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button class="btn btn-primary">

Atualizar Produto

</button>

</div>

</div>

</div>

</div>

<!-- Janela modal para exclusão de produto -->

<div id="janelaexclusao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-danger">

<h3 class="text-white">Excluir Produto</h3>

</div>

<div class="modal-body">

<h4>Deseja realmente excluir este produto?</h4>

<hr/>

Código do Produto: <label id="idproduto\_exclusao"></label>

<br/>

Nome do Produto: <label id="nome\_exclusao"></label>

<br />

Preço: <label id="preco\_exclusao"></label>

<br />

Quantidade: <label id="quantidade\_exclusao"></label>

<br />

Total: <label id="total\_exclusao"></label>

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnexclusao" class="btn btn-danger">

Excluir Produto

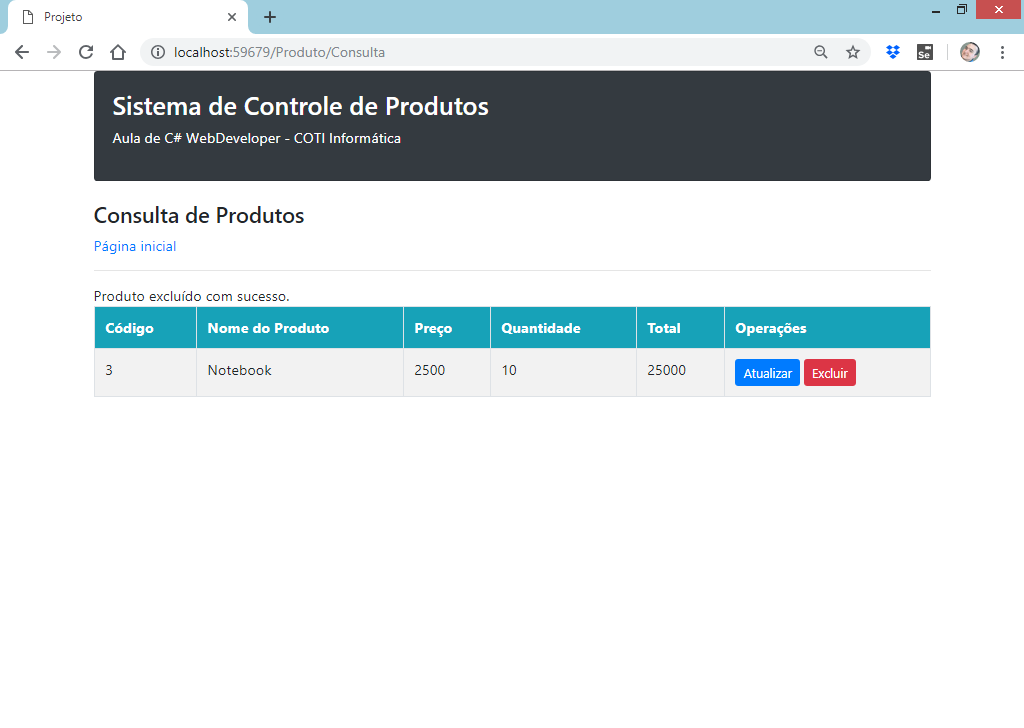
</button>

</div>

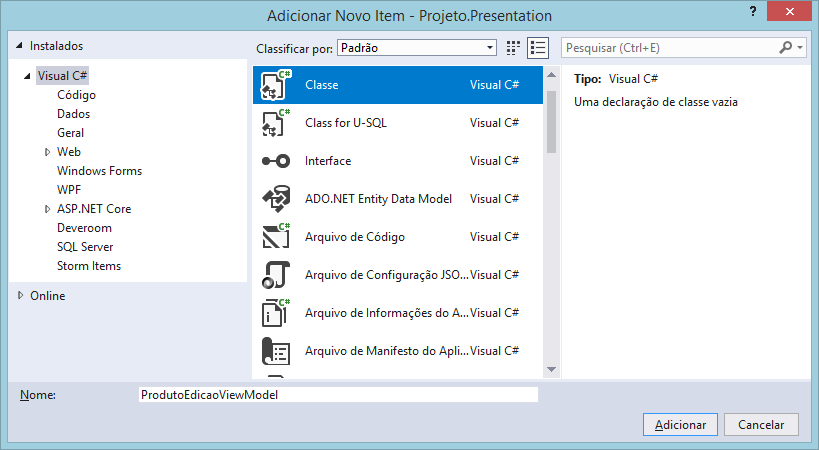
</div>

</div>

</div>



Criando uma classe ViewModel para implementar a edição do produto:



using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.ComponentModel.DataAnnotations;

namespace Projeto.Presentation.Models

{

public class ProdutoEdicaoViewModel

{

public int IdProduto { get; set; }

[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 executar a edição do produto:**

/ProdutoController.cs

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);

}

}

//método para retornar a consulta de produtos para o jquery

public JsonResult ConsultarProdutos()

{

try

{

List<ProdutoConsultaViewModel> lista

= new List<ProdutoConsultaViewModel>();

//executar a consulta de produtos na camada de regras de negócio

ProdutoBusiness business = new ProdutoBusiness();

foreach(Produto produto in business.ConsultarTodos())

{

ProdutoConsultaViewModel model

= new ProdutoConsultaViewModel();

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

lista.Add(model); //adicionar na lista

}

return Json(lista); //retornando a lista

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

//retornando mensagem de erro..

return Json(e.Message);

}

}

//método pra consultar 1 produto pelo id

public JsonResult ConsultarProdutoPorId(int id)

{

try

{

ProdutoConsultaViewModel model = new ProdutoConsultaViewModel();

//buscar o produto pelo id

ProdutoBusiness business = new ProdutoBusiness();

Produto produto = business.ConsultarPorId(id);

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

return Json(model); //retornando o objeto model..

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

return Json(e.Message); //mensagem de erro

}

}

//método para excluir um produto

public JsonResult ExcluirProduto(int id)

{

try

{

ProdutoBusiness business = new ProdutoBusiness();

business.ExcluirProduto(id);

return Json("Produto excluído com sucesso.");

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

return Json(e.Message); //mensagem de erro

}

}

//método que será executado pela chamada AJAX feita na página

public JsonResult AtualizarProduto(ProdutoEdicaoViewModel model)

{

//verificar se os dados da model passaram nas regras de validação

if (ModelState.IsValid)

{

try

{

Produto produto = new Produto();

produto.IdProduto = model.IdProduto;

produto.Nome = model.Nome;

produto.Preco = model.Preco;

produto.Quantidade = model.Quantidade;

ProdutoBusiness business = new ProdutoBusiness();

business.AtualizarProduto(produto);

return Json($"Produto {produto.Nome},

atualizado 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);

}

}

}

}

**Executando a edição do produto na página de consulta:**

Criando a requisição AJAX

@{

ViewBag.Title = "Consulta";

Layout = "~/Views/Shared/Layout.cshtml";

}

@section scripts{

<script>

//função para atualizar os dados do produto

function atualizarProduto() {

//criar um objeto JSON para armazenar os dados

//de cada campo da página de cadastro de produto

var dados = {

IdProduto: $("#idproduto\_edicao").val(),

Nome: $("#nome\_edicao").val(),

Preco: $("#preco\_edicao").val(),

Quantidade: $("#quantidade\_edicao").val()

};

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

//requisição AJAX para o servidor

$.ajax({

type: "POST",

url: "/Produto/AtualizarProduto",

data: dados,

success: function (obj) { //promisse

//exibir mensagem

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

//fechando a janela

$("#janelaedicao").modal('hide');

//executando a consulta novamente..

consultarProdutos();

},

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);

}

}

});

};

//função para excluir o produto

function excluirProduto() {

$.ajax({

type: "POST",

url: "/Produto/ExcluirProduto",

data: { id: $("#idproduto\_exclusao").html() },

success: function (msg) {

//imprimindo mensagem de sucesso..

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

//fechar a janela modal de exclusão..

$("#janelaexclusao").modal("hide");

//executar a consulta de produtos novamente

consultarProdutos();

},

error: function (e) {

console.log(e);

}

});

};

//função para exibir os dados do produto pelo id

function obterProduto(idProduto) {

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutoPorId",

data: { id: idProduto },

success: function (obj) {

$("#idproduto\_edicao").val(obj.IdProduto);

$("#nome\_edicao").val(obj.Nome);

$("#preco\_edicao").val(obj.Preco);

$("#quantidade\_edicao").val(obj.Quantidade);

$("#idproduto\_exclusao").html(obj.IdProduto);

$("#nome\_exclusao").html(obj.Nome);

$("#preco\_exclusao").html(obj.Preco);

$("#quantidade\_exclusao").html(obj.Quantidade);

$("#total\_exclusao").html(obj.Total);

},

error: function (e) {

console.log(e);

}

})

}

//função para executar a consulta de produtos

function consultarProdutos() {

//requisição AJAX ao controller:

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutos",

data: {}, //vazio

success: function (lista) {

var conteudo = "";

$.each(lista, function (i, item) {

conteudo += "<tr>";

conteudo += "<td>" + item.IdProduto + "</td>";

conteudo += "<td>" + item.Nome + "</td>";

conteudo += "<td>" + item.Preco + "</td>";

conteudo += "<td>" + item.Quantidade + "</td>";

conteudo += "<td>" + item.Total + "</td>";

conteudo += "<td>";

conteudo += "<button onclick='obterProduto("+

item.IdProduto +")' class='btn btn-primary

btn-sm' data-target='#janelaedicao'

data-toggle='modal'>Atualizar</button>";

conteudo += "&nbsp;"; //espaço

conteudo += "<button onclick='obterProduto("+

item.IdProduto +")' class='btn btn-danger

btn-sm' data-target='#janelaexclusao'

data-toggle='modal'>Excluir</button>";

conteudo += "</td>";

conteudo += "</tr>";

});

$("#tabela tbody").html(conteudo);

},

error: function (e) {

console.log(e);

}

});

};

//quando a página for carregada, execute..

$(document).ready(function () {

consultarProdutos(); //executando a função

//criando um evento para realizar a exclusão..

$("#btnexclusao").click(function () {

excluirProduto(); //executando a função

});

//criando um evento para realizar a edição..

$("#btnedicao").click(function () {

atualizarProduto();

});

});

</script>

}

<h4>Consulta de Produtos</h4>

<a href="/Home/Index">Página inicial</a>

<hr/>

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

<table id="tabela" class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th class="bg-info text-white">Código</th>

<th class="bg-info text-white">Nome do Produto</th>

<th class="bg-info text-white">Preço</th>

<th class="bg-info text-white">Quantidade</th>

<th class="bg-info text-white">Total</th>

<th class="bg-info text-white">Operações</th>

</tr>

</thead>

<tbody>

</tbody>

</table>

<!-- Janela modal para edição de produto -->

<div id="janelaedicao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-primary">

<h3 class="text-white">Atualizar Produto</h3>

</div>

<div class="modal-body">

<!-- local para exibir as mensagens de erro de validação -->

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

<label>Código do Produto:</label>

<input type="text" id="idproduto\_edicao"

class="form-control" readonly="readonly"/>

<br/>

<label>Nome do Produto:</label>

<input type="text" id="nome\_edicao"

class="form-control" />

<br />

<label>Preço:</label>

<input type="text" id="preco\_edicao"

class="form-control" />

<br />

<label>Quantidade:</label>

<input type="text" id="quantidade\_edicao"

class="form-control" />

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnedicao" class="btn btn-primary">

Atualizar Produto

</button>

</div>

</div>

</div>

</div>

<!-- Janela modal para exclusão de produto -->

<div id="janelaexclusao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-danger">

<h3 class="text-white">Excluir Produto</h3>

</div>

<div class="modal-body">

<h4>Deseja realmente excluir este produto?</h4>

<hr/>

Código do Produto: <label id="idproduto\_exclusao"></label>

<br/>

Nome do Produto: <label id="nome\_exclusao"></label>

<br />

Preço: <label id="preco\_exclusao"></label>

<br />

Quantidade: <label id="quantidade\_exclusao"></label>

<br />

Total: <label id="total\_exclusao"></label>

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnexclusao" class="btn btn-danger">

Excluir Produto

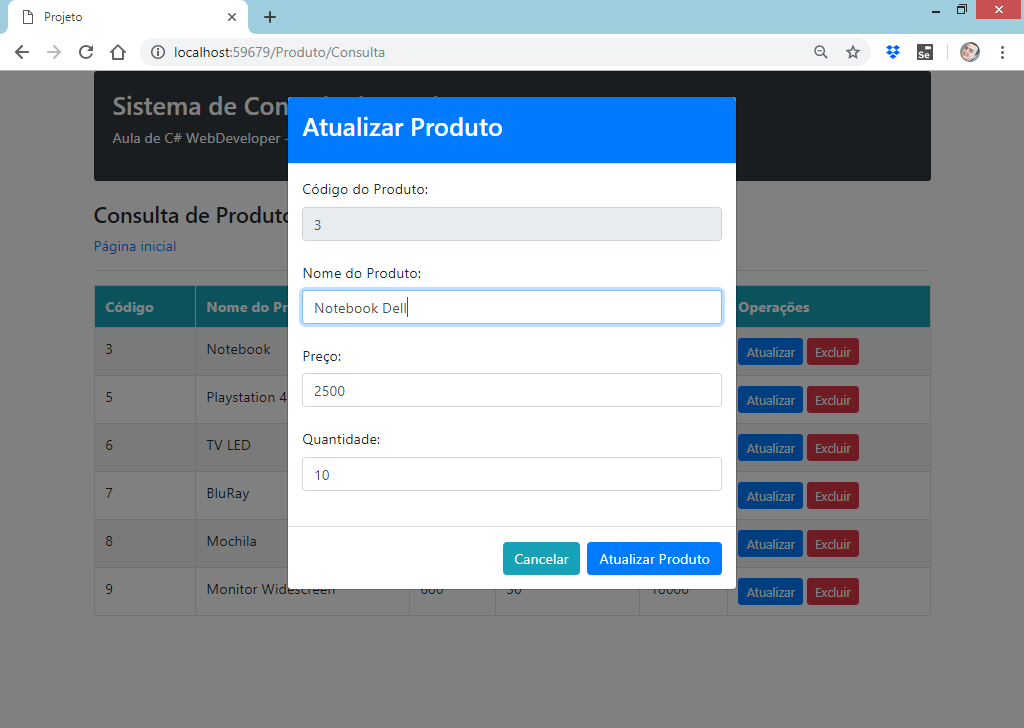
</button>

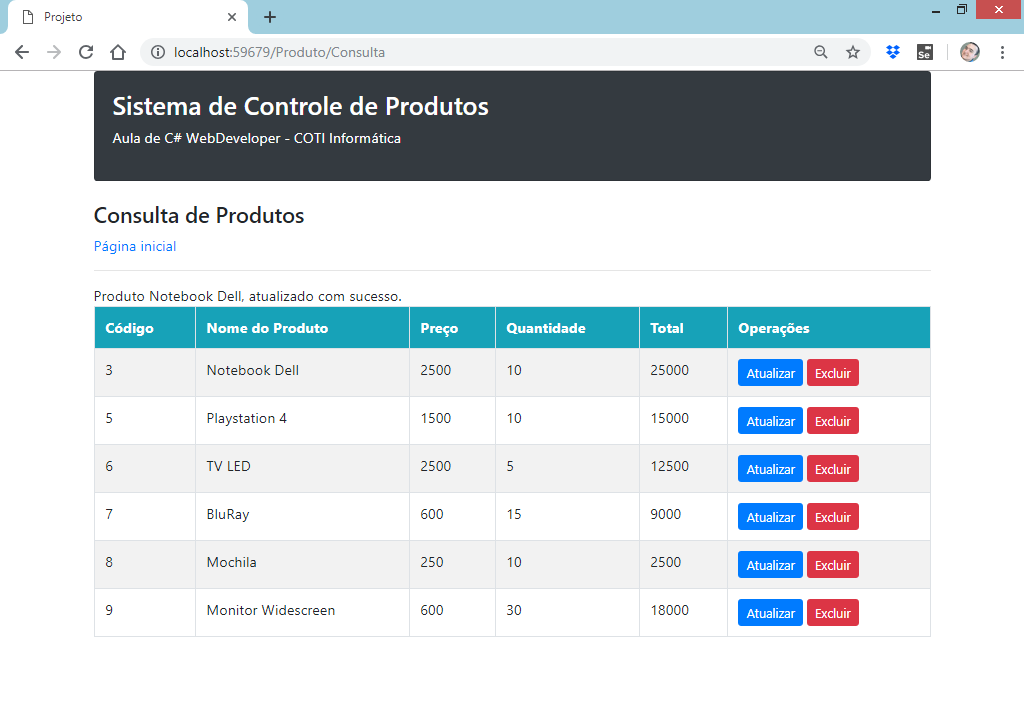
</div>

</div>

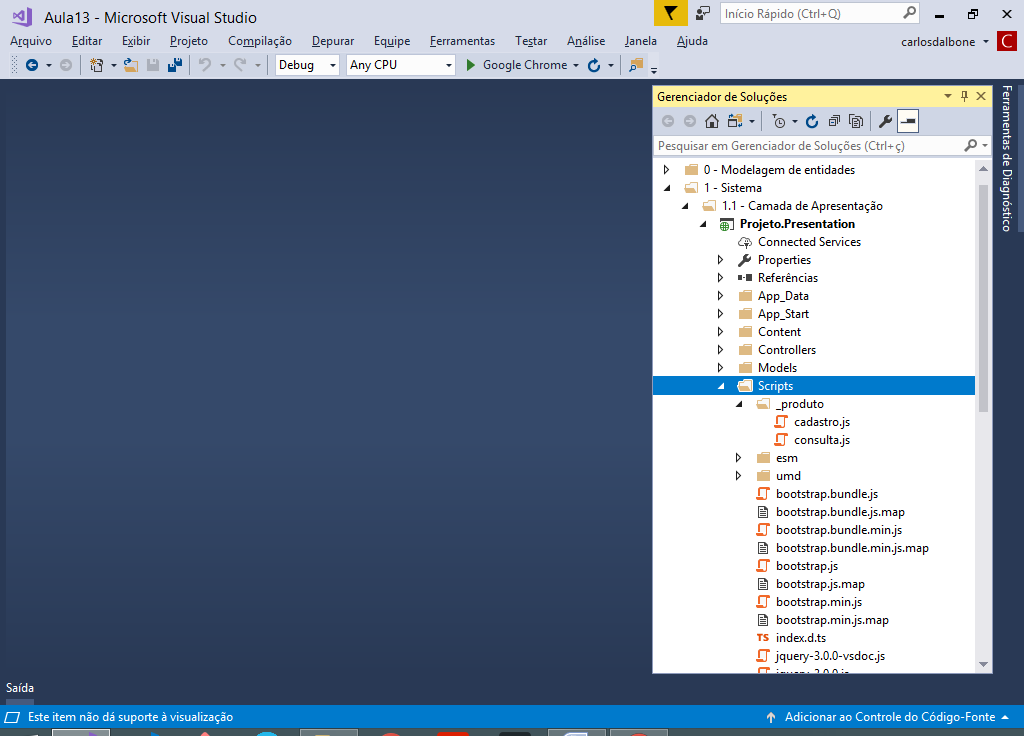
</div>

</div>





**Separando os arquivos JavaScript do projeto:**



/Scripts/\_produto/cadastro.js

//função para realizar o cadastro do produto

function cadastrarProduto() {

//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()

};

$("#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);

}

}

});

};

//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 () {

cadastrarProduto();

});

})

/Scripts/\_produto/consulta.js

//função para atualizar os dados do produto

function atualizarProduto() {

//criar um objeto JSON para armazenar os dados

//de cada campo da página de cadastro de produto

var dados = {

IdProduto: $("#idproduto\_edicao").val(),

Nome: $("#nome\_edicao").val(),

Preco: $("#preco\_edicao").val(),

Quantidade: $("#quantidade\_edicao").val()

};

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

//requisição AJAX para o servidor

$.ajax({

type: "POST",

url: "/Produto/AtualizarProduto",

data: dados,

success: function (obj) { //promisse

//exibir mensagem

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

//fechando a janela

$("#janelaedicao").modal('hide');

//executando a consulta novamente..

consultarProdutos();

},

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);

}

}

});

};

//função para excluir o produto

function excluirProduto() {

$.ajax({

type: "POST",

url: "/Produto/ExcluirProduto",

data: { id: $("#idproduto\_exclusao").html() },

success: function (msg) {

//imprimindo mensagem de sucesso..

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

//fechar a janela modal de exclusão..

$("#janelaexclusao").modal("hide");

//executar a consulta de produtos novamente

consultarProdutos();

},

error: function (e) {

console.log(e);

}

});

};

//função para exibir os dados do produto pelo id

function obterProduto(idProduto) {

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutoPorId",

data: { id: idProduto },

success: function (obj) {

$("#idproduto\_edicao").val(obj.IdProduto);

$("#nome\_edicao").val(obj.Nome);

$("#preco\_edicao").val(obj.Preco);

$("#quantidade\_edicao").val(obj.Quantidade);

$("#idproduto\_exclusao").html(obj.IdProduto);

$("#nome\_exclusao").html(obj.Nome);

$("#preco\_exclusao").html(obj.Preco);

$("#quantidade\_exclusao").html(obj.Quantidade);

$("#total\_exclusao").html(obj.Total);

},

error: function (e) {

console.log(e);

}

})

}

//função para executar a consulta de produtos

function consultarProdutos() {

//requisição AJAX ao controller:

$.ajax({

type: "POST",

url: "/Produto/ConsultarProdutos",

data: {}, //vazio

success: function (lista) {

var conteudo = "";

$.each(lista, function (i, item) {

conteudo += "<tr>";

conteudo += "<td>" + item.IdProduto + "</td>";

conteudo += "<td>" + item.Nome + "</td>";

conteudo += "<td>" + item.Preco + "</td>";

conteudo += "<td>" + item.Quantidade + "</td>";

conteudo += "<td>" + item.Total + "</td>";

conteudo += "<td>";

conteudo += "<button onclick='obterProduto(" + item.IdProduto

+ ")' class='btn btn-primary btn-sm'

data-target='#janelaedicao'

data-toggle='modal'>Atualizar</button>";

conteudo += "&nbsp;"; //espaço

conteudo += "<button onclick='obterProduto(" + item.IdProduto

+ ")' class='btn btn-danger btn-sm'

data-target='#janelaexclusao'

data-toggle='modal'>Excluir</button>";

conteudo += "</td>";

conteudo += "</tr>";

});

$("#tabela tbody").html(conteudo);

},

error: function (e) {

console.log(e);

}

});

};

//quando a página for carregada, execute..

$(document).ready(function () {

consultarProdutos(); //executando a função

//criando um evento para realizar a exclusão..

$("#btnexclusao").click(function () {

excluirProduto(); //executando a função

});

//criando um evento para realizar a edição..

$("#btnedicao").click(function () {

atualizarProduto();

});

});

-----------------------

**Fazendo referência aos arquivos Javascript:**

@{

ViewBag.Title = "Cadastro";

Layout = "~/Views/Shared/Layout.cshtml";

}

**@section scripts{**

**<script src="~/Scripts/\_produto/cadastro.js"></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>

-----------------------------------

@{

ViewBag.Title = "Consulta";

Layout = "~/Views/Shared/Layout.cshtml";

}

**@section scripts{**

**<script src="~/Scripts/\_produto/consulta.js"></script>**

**}**

<h4>Consulta de Produtos</h4>

<a href="/Home/Index">Página inicial</a>

<hr/>

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

<table id="tabela" class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th class="bg-info text-white">Código</th>

<th class="bg-info text-white">Nome do Produto</th>

<th class="bg-info text-white">Preço</th>

<th class="bg-info text-white">Quantidade</th>

<th class="bg-info text-white">Total</th>

<th class="bg-info text-white">Operações</th>

</tr>

</thead>

<tbody>

</tbody>

</table>

<!-- Janela modal para edição de produto -->

<div id="janelaedicao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-primary">

<h3 class="text-white">Atualizar Produto</h3>

</div>

<div class="modal-body">

<!-- local para exibir as mensagens de erro de validação -->

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

<label>Código do Produto:</label>

<input type="text" id="idproduto\_edicao"

class="form-control" readonly="readonly"/>

<br/>

<label>Nome do Produto:</label>

<input type="text" id="nome\_edicao"

class="form-control" />

<br />

<label>Preço:</label>

<input type="text" id="preco\_edicao"

class="form-control" />

<br />

<label>Quantidade:</label>

<input type="text" id="quantidade\_edicao"

class="form-control" />

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnedicao" class="btn btn-primary">

Atualizar Produto

</button>

</div>

</div>

</div>

</div>

<!-- Janela modal para exclusão de produto -->

<div id="janelaexclusao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-danger">

<h3 class="text-white">Excluir Produto</h3>

</div>

<div class="modal-body">

<h4>Deseja realmente excluir este produto?</h4>

<hr/>

Código do Produto: <label id="idproduto\_exclusao"></label>

<br/>

Nome do Produto: <label id="nome\_exclusao"></label>

<br />

Preço: <label id="preco\_exclusao"></label>

<br />

Quantidade: <label id="quantidade\_exclusao"></label>

<br />

Total: <label id="total\_exclusao"></label>

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnexclusao" class="btn btn-danger">

Excluir Produto

</button>

</div>

</div>

</div>

</div>

----------------------

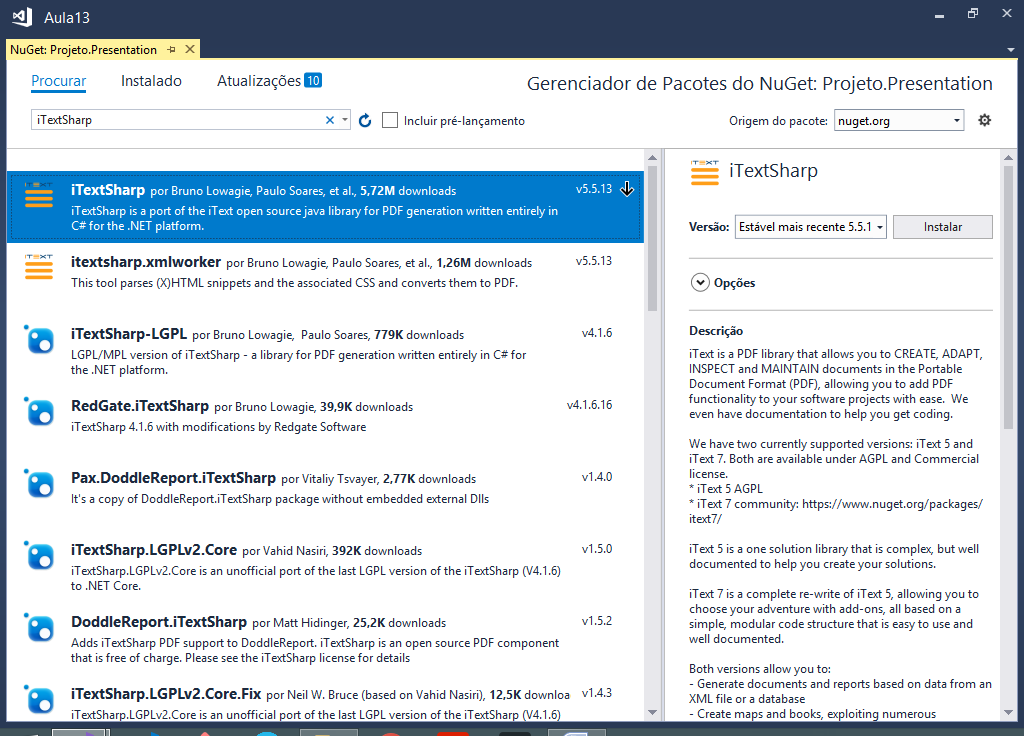
**Framework para geração de**

**relatórios PDF em aplicações .NET**

Instalando o iTextSharp

Gerenciador de pacotes do NuGet





**Incluindo um botão para gerar o relatório de produtos:**

@{

ViewBag.Title = "Consulta";

Layout = "~/Views/Shared/Layout.cshtml";

}

@section scripts{

<script src="~/Scripts/\_produto/consulta.js"></script>

}

<h4>Consulta de Produtos</h4>

<a href="/Home/Index">Página inicial</a>

<hr/>

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

**<div class="row">**

**<div class="col-md-12 text-right">**

**<a href="/Produto/Relatorio"**

**class="btn btn-warning btn-sm">**

**Gerar Relatório de Produtos**

**</a>**

**</div>**

**</div>**

<br/>

<table id="tabela" class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th class="bg-info text-white">Código</th>

<th class="bg-info text-white">Nome do Produto</th>

<th class="bg-info text-white">Preço</th>

<th class="bg-info text-white">Quantidade</th>

<th class="bg-info text-white">Total</th>

<th class="bg-info text-white">Operações</th>

</tr>

</thead>

<tbody>

</tbody>

</table>

<!-- Janela modal para edição de produto -->

<div id="janelaedicao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-primary">

<h3 class="text-white">Atualizar Produto</h3>

</div>

<div class="modal-body">

<!-- local para exibir as mensagens de erro de validação -->

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

<label>Código do Produto:</label>

<input type="text" id="idproduto\_edicao"

class="form-control" readonly="readonly"/>

<br/>

<label>Nome do Produto:</label>

<input type="text" id="nome\_edicao"

class="form-control" />

<br />

<label>Preço:</label>

<input type="text" id="preco\_edicao"

class="form-control" />

<br />

<label>Quantidade:</label>

<input type="text" id="quantidade\_edicao"

class="form-control" />

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnedicao" class="btn btn-primary">

Atualizar Produto

</button>

</div>

</div>

</div>

</div>

<!-- Janela modal para exclusão de produto -->

<div id="janelaexclusao" class="modal fade">

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header bg-danger">

<h3 class="text-white">Excluir Produto</h3>

</div>

<div class="modal-body">

<h4>Deseja realmente excluir este produto?</h4>

<hr/>

Código do Produto: <label id="idproduto\_exclusao"></label>

<br/>

Nome do Produto: <label id="nome\_exclusao"></label>

<br />

Preço: <label id="preco\_exclusao"></label>

<br />

Quantidade: <label id="quantidade\_exclusao"></label>

<br />

Total: <label id="total\_exclusao"></label>

<br />

</div>

<div class="modal-footer">

<button class="btn btn-info" data-dismiss="modal">

Cancelar

</button>

<button id="btnexclusao" class="btn btn-danger">

Excluir Produto

</button>

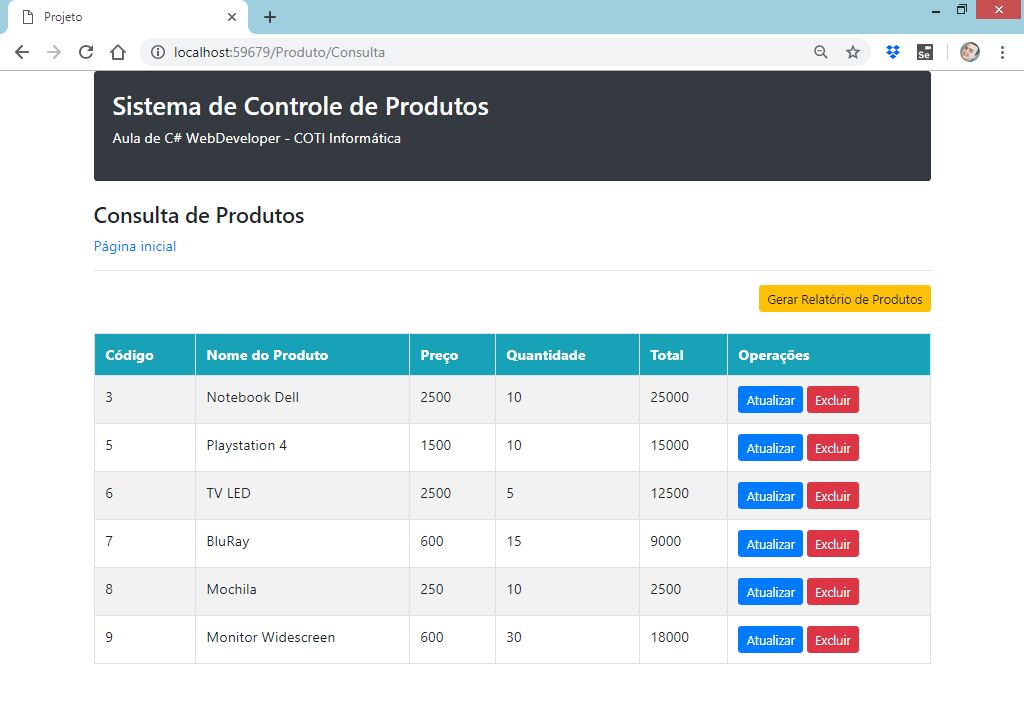
</div>

</div>

</div>

</div>

**Na página:**



Para gerar um relatório,   
iremos realizar 3 passos:

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;

using System.Diagnostics;

using System.Text;

using System.IO;

using iTextSharp.text;

using iTextSharp.text.pdf;

using iTextSharp.text.html.simpleparser;

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);

}

}

//método para retornar a consulta de produtos para o jquery

public JsonResult ConsultarProdutos()

{

try

{

List<ProdutoConsultaViewModel> lista

= new List<ProdutoConsultaViewModel>();

//executar a consulta de produtos na camada de regras de negócio

ProdutoBusiness business = new ProdutoBusiness();

foreach(Produto produto in business.ConsultarTodos())

{

ProdutoConsultaViewModel model

= new ProdutoConsultaViewModel();

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

lista.Add(model); //adicionar na lista

}

return Json(lista); //retornando a lista

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

//retornando mensagem de erro..

return Json(e.Message);

}

}

//método pra consultar 1 produto pelo id

public JsonResult ConsultarProdutoPorId(int id)

{

try

{

ProdutoConsultaViewModel model = new ProdutoConsultaViewModel();

//buscar o produto pelo id

ProdutoBusiness business = new ProdutoBusiness();

Produto produto = business.ConsultarPorId(id);

model.IdProduto = produto.IdProduto;

model.Nome = produto.Nome;

model.Preco = produto.Preco;

model.Quantidade = produto.Quantidade;

model.Total = produto.Preco \* produto.Quantidade;

return Json(model); //retornando o objeto model..

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

return Json(e.Message); //mensagem de erro

}

}

//método para excluir um produto

public JsonResult ExcluirProduto(int id)

{

try

{

ProdutoBusiness business = new ProdutoBusiness();

business.ExcluirProduto(id);

return Json("Produto excluído com sucesso.");

}

catch(Exception e)

{

Response.StatusCode = 500; //INTERNAL SERVER ERROR

return Json(e.Message); //mensagem de erro

}

}

//método que será executado pela chamada AJAX feita na página

public JsonResult AtualizarProduto(ProdutoEdicaoViewModel model)

{

//verificar se os dados da model passaram nas regras de validação

if (ModelState.IsValid)

{

try

{

Produto produto = new Produto();

produto.IdProduto = model.IdProduto;

produto.Nome = model.Nome;

produto.Preco = model.Preco;

produto.Quantidade = model.Quantidade;

ProdutoBusiness business = new ProdutoBusiness();

business.AtualizarProduto(produto);

return Json($"Produto {produto.Nome},

atualizado 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);

}

}

**//método para realizar o download do relatório em PDF..**

**//URL: Produto/Relatorio**

**public void Relatorio()**

**{**

**try**

**{**

**//PASSO 1) Escrever o conteúdo do relatório em formato HTML**

**StringBuilder conteudo = new StringBuilder();**

**conteudo.Append("<h2>Relatório de Produtos</h2>");**

**conteudo.Append($"Relatório gerado em: {DateTime.Now}");**

**conteudo.Append("<br/><br/>");**

**conteudo.Append("<table border='1' style='width: 100%'>");**

**conteudo.Append("<tr>");**

**conteudo.Append("<th>Código</th>");**

**conteudo.Append("<th>Nome do Produto</th>");**

**conteudo.Append("<th>Preço</th>");**

**conteudo.Append("<th>Quantidade</th>");**

**conteudo.Append("</tr>");**

**//buscar os produtos na base de dados**

**ProdutoBusiness business = new ProdutoBusiness();**

**List<Produto> lista = business.ConsultarTodos();**

**//percorrer a lista..**

**foreach(Produto produto in lista)**

**{**

**conteudo.Append("<tr>");**

**conteudo.Append($"<td>{produto.IdProduto}</td>");**

**conteudo.Append($"<td>{produto.Nome}</td>");**

**conteudo.Append($"<td>{produto.Preco}</td>");**

**conteudo.Append($"<td>{produto.Quantidade}</td>");**

**conteudo.Append("</tr>");**

**}**

**conteudo.Append("</table>");**

**conteudo.Append("<br/>");**

**conteudo.Append($"Quantidade de registros: {lista.Count}");**

**//PASSO 2) Utilizando o iTextSharp para transformar**

**//o conteudo HTML do passo 1 em um arquivo PDF**

**byte[] pdf = null;**

**MemoryStream ms = new MemoryStream();**

**TextReader reader = new StringReader(conteudo.ToString());**

**Document doc = new Document(PageSize.A4, 50, 50, 50, 50);**

**PdfWriter writer = PdfWriter.GetInstance(doc, ms);**

**HTMLWorker html = new HTMLWorker(doc);**

**doc.Open();**

**html.StartDocument();**

**html.Parse(reader);**

**html.EndDocument();**

**html.Close();**

**doc.Close();**

**pdf = ms.ToArray();**

**//PASSO 3) Utilizar o MVC para fazer o download do PDF**

**Response.Clear();**

**Response.ContentType = "application/pdf";**

**Response.AddHeader("content-disposition",**

**"attachment; filename=relatorio.pdf");**

**Response.Cache.SetCacheability(HttpCacheability.NoCache);**

**Response.BinaryWrite(pdf);**

**Response.End();**

**}**

**catch (Exception e)**

**{**

**Debug.WriteLine(e.Message);**

**}**

**}**

}

}

**Executando:**

