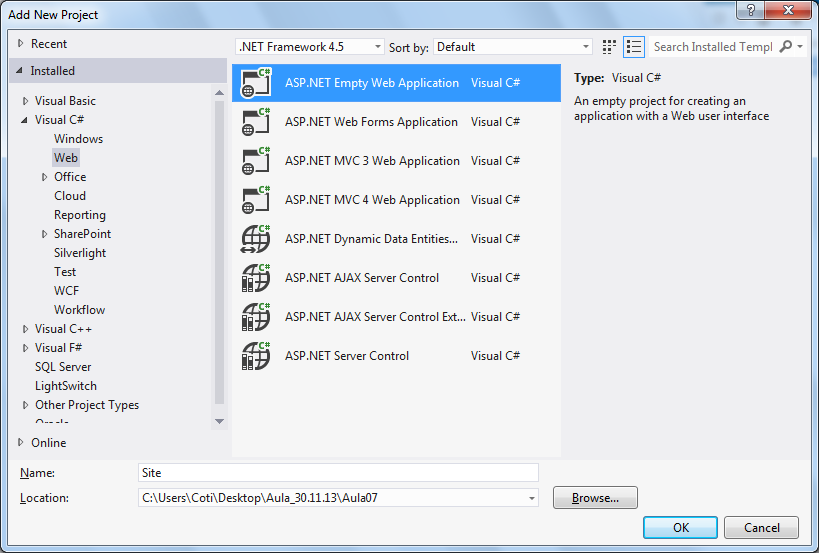
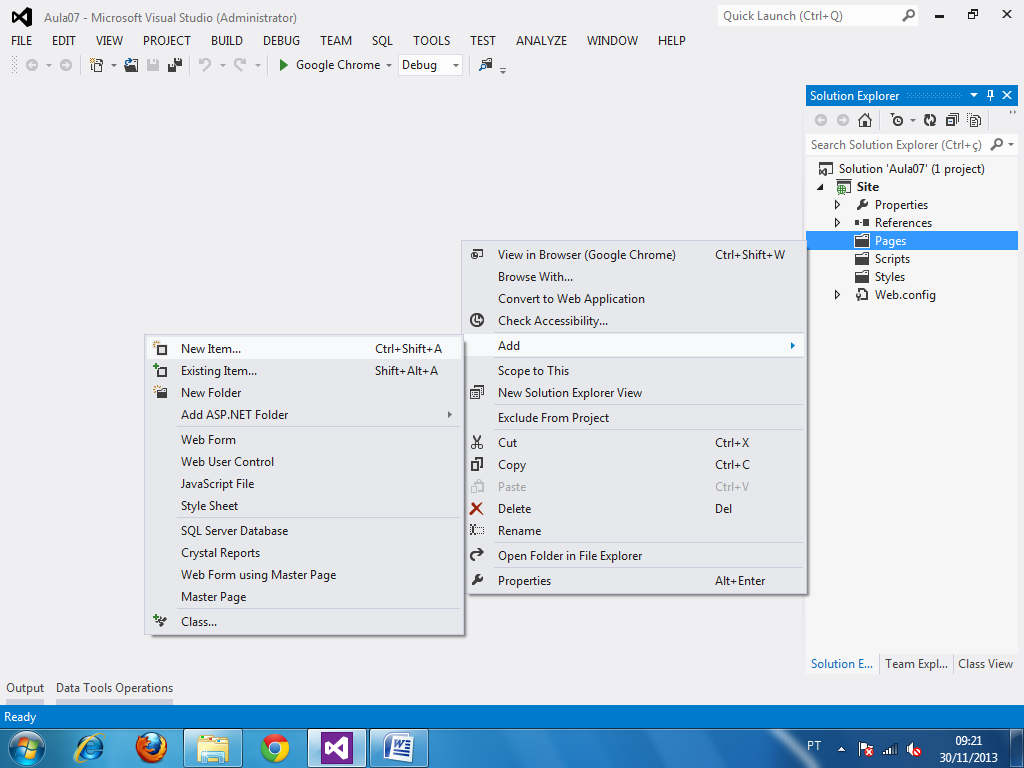
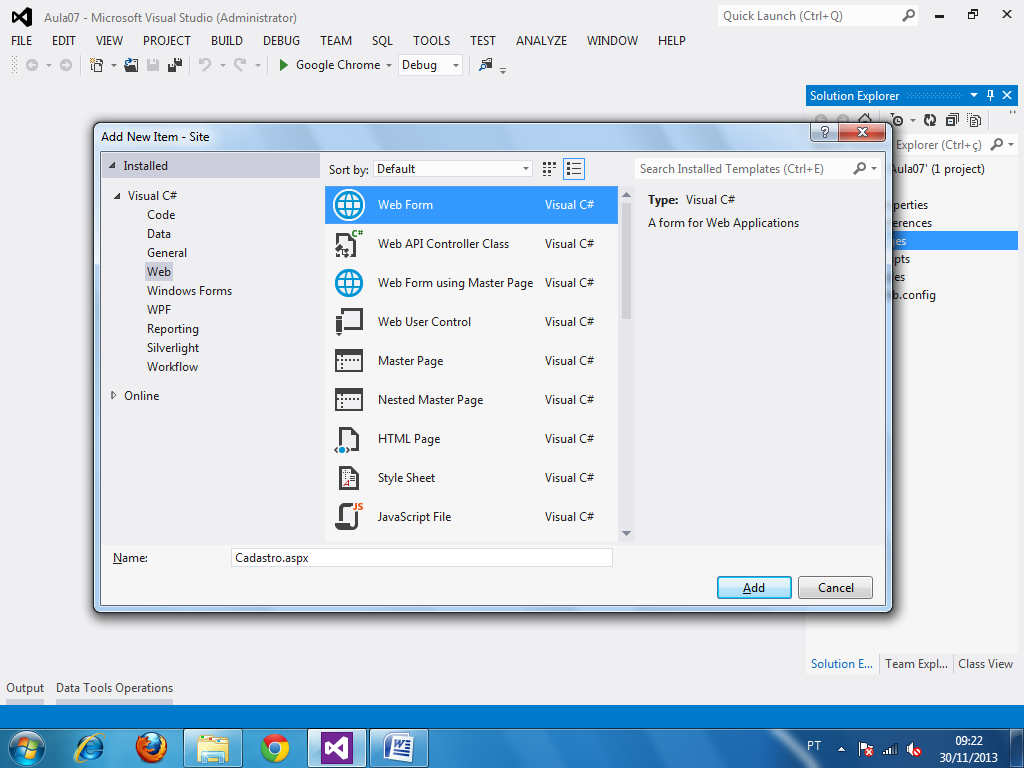


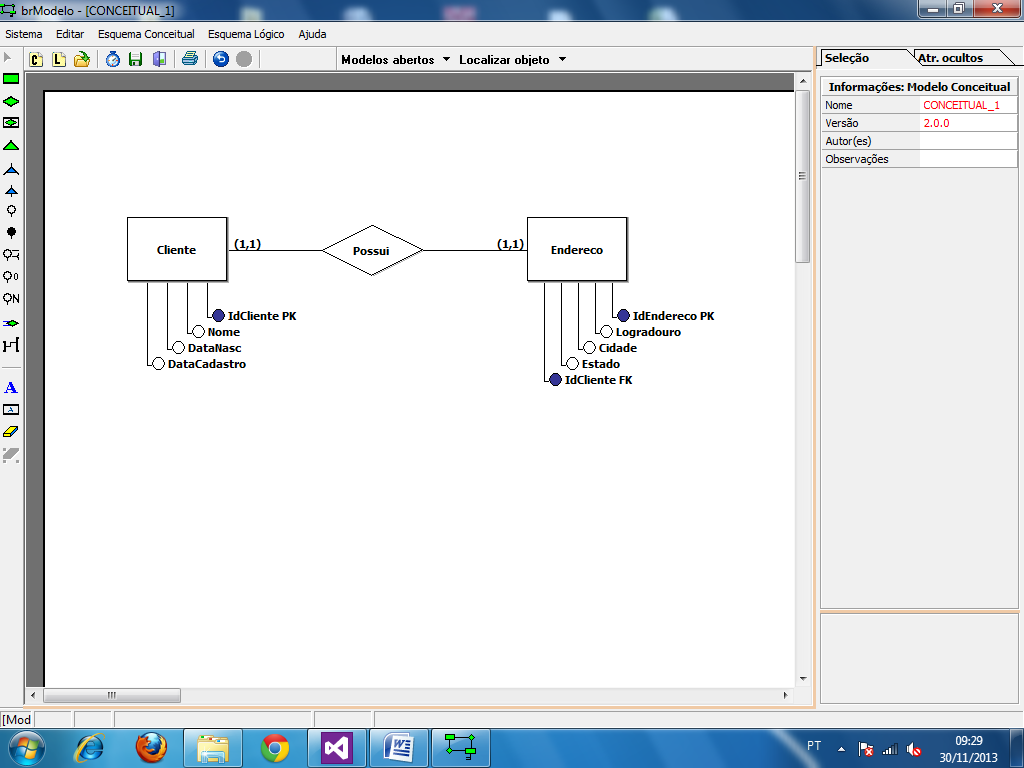
Criação do projeto Asp.Net







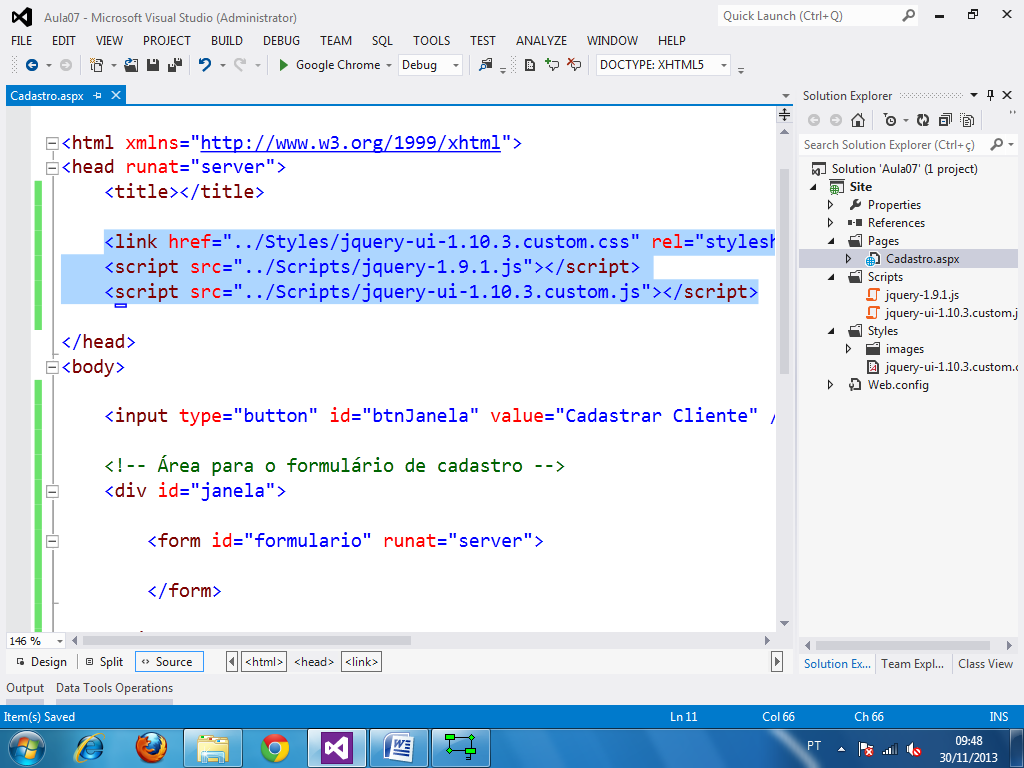
DER – Diagrama de Entidade / Relacionamento



JQuery User Interface

[www.jqueryui.com](http://www.jqueryui.com)

Conjunto de bibliotecas JavaScript para desenvolvimento de interface de aplicações. Ex: Interface de Usuário, Validação, etc...



<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Cadastro.aspx.cs" Inherits="Site.Pages.Cadastro" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link href="../Styles/jquery-ui-1.10.3.custom.css" rel="stylesheet" />

<script src="../Scripts/jquery-1.9.1.js"></script>

<script src="../Scripts/jquery-ui-1.10.3.custom.js"></script>

<script type="text/javascript">

//Executar quando a página estiver pronta

$(document).ready(

function () { //faça...

$("#janela").dialog(

{

modal: true,

width: 400,

autoOpen: false

}

);

$("#btnJanela").click( //quando o botão for 'clicado'

function () { //faça...

$("#janela").dialog("open"); //exibir o dialog

}

);

$("#abas").tabs(); //navegação em abas

$(".botao").button(); //formatação de design botão

//formatação de calendário para o campo

$(".data").datepicker(

{

dateFormat: 'dd/mm/yy',

changeYear: true,

changeMonth: true,

yearRange : "c-100:c" //c => ano atual (minimo/maximo)

}

);

}

);

</script>

</head>

<body>

<input type="button" id="btnJanela" class="botao"

value="Cadastrar Cliente" />

<!-- Área para o formulário de cadastro -->

<div id="janela" title="Formulário de Cadastro">

<form id="formulario" runat="server">

<p>

Para cadastrar um novo Cliente, preencha os dados abaixo:

</p>

<!-- Marcação HTML para navegação em abas -->

<div id="abas">

<ul>

<li><a href="#aba1">Dados do Cliente</a></li>

<li><a href="#aba2">Endereço</a></li>

</ul>

<div id="aba1">

Nome do Cliente: <br />

<asp:TextBox ID="txtNome" runat="server" />

<br /><br />

Data de Nascimento: <br />

<asp:TextBox ID="txtDataNasc" runat="server"

CssClass="data" />

</div>

<div id="aba2">

Logradouro: <br />

<asp:TextBox ID="txtLogradouro" runat="server" />

<br /><br />

Cidade: <br />

<asp:TextBox ID="txtCidade" runat="server" />

<br /><br />

Estado: <br />

<asp:TextBox ID="txtEstado" runat="server" />

</div>

</div>

<p>

<asp:Button ID="btnCadastro" runat="server"

Text="Cadastrar Cliente" CssClass="botao" />

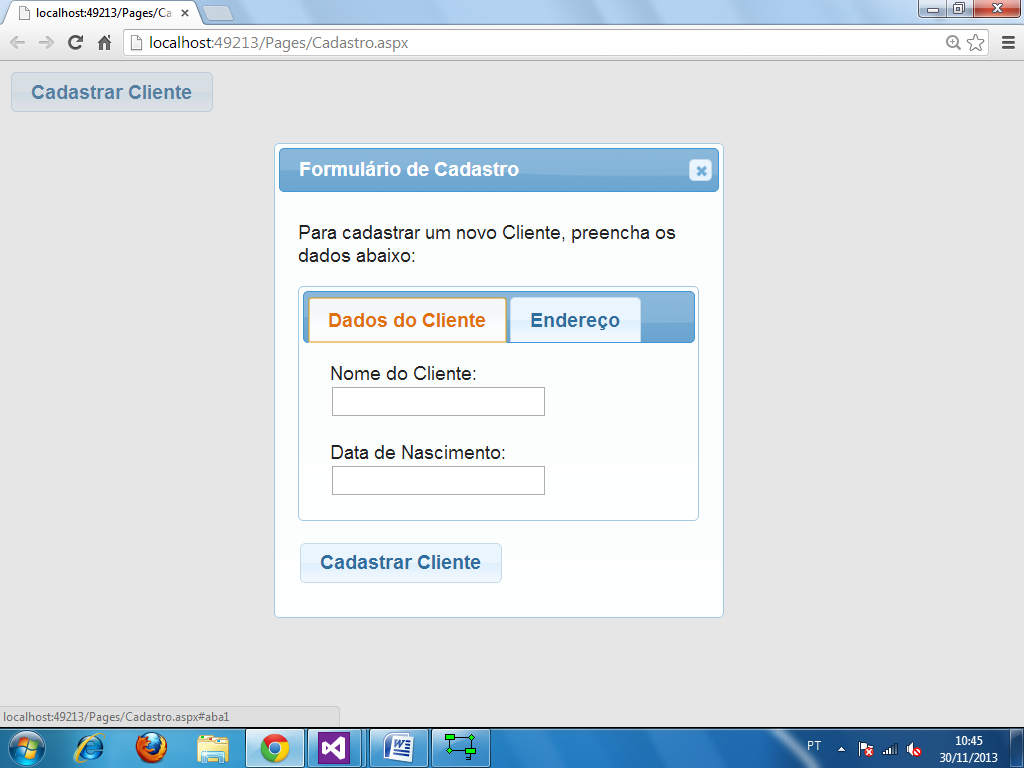
</p>

</form>

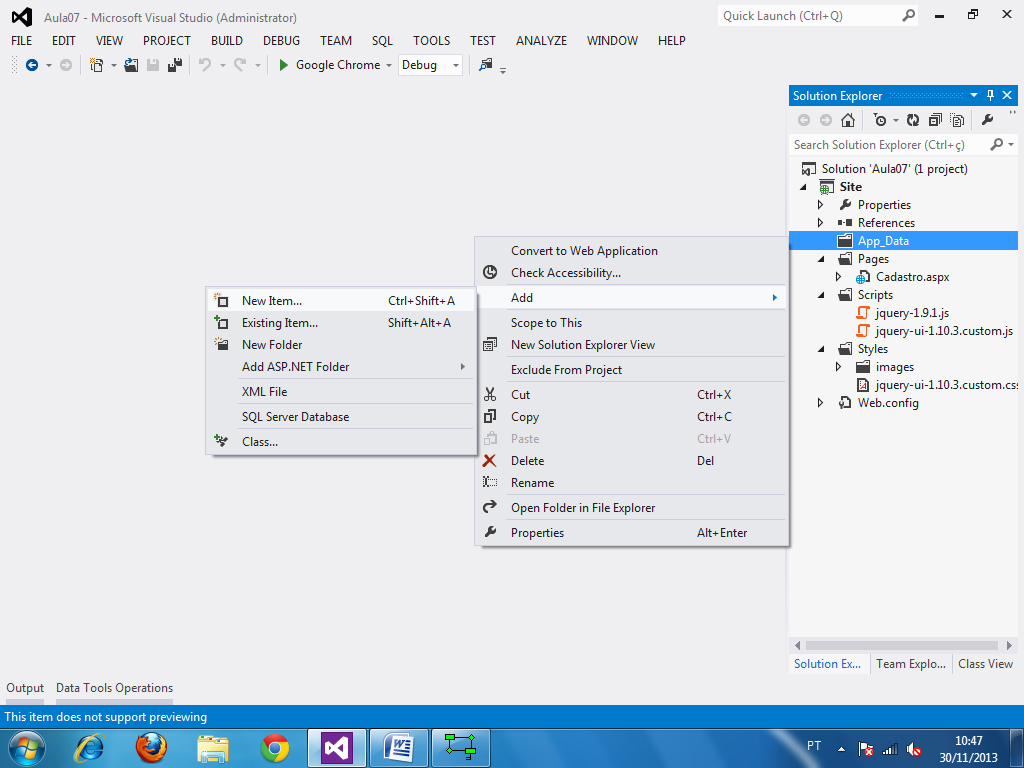
</div>

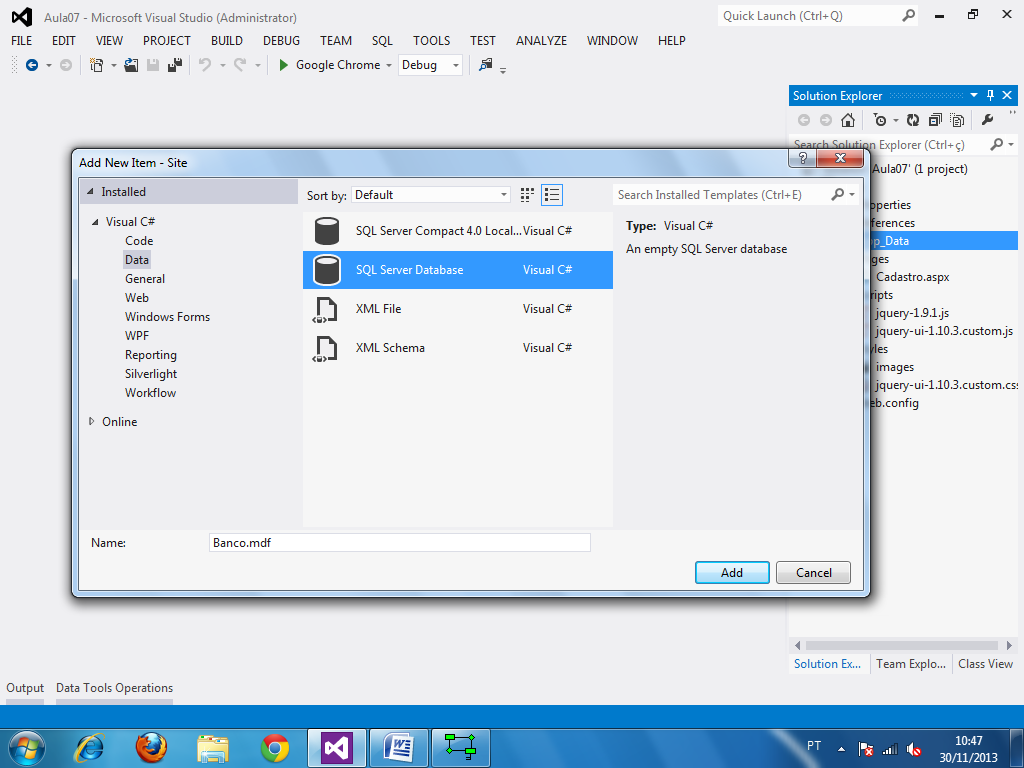
</body>

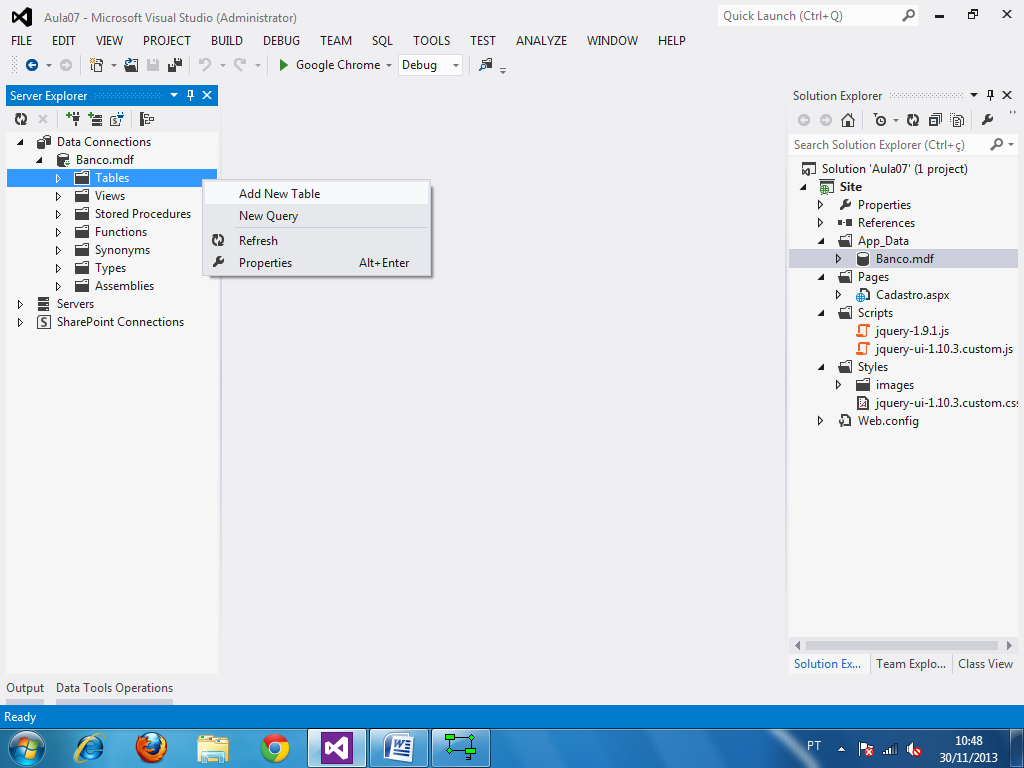
</html>











T-SQL linguagem padrão do SqlServer…

go

create table Cliente(

IdCliente integer identity,

Nome nvarchar(50) not null,

DataNascimento date not null,

DataCadastro datetime not null,

primary key(IdCliente))

go

create table Endereco(

IdEndereco integer identity,

Logradouro nvarchar(50) not null,

Cidade nvarchar(50) not null,

Estado nvarchar(50) not null,

IdCliente integer not null unique,

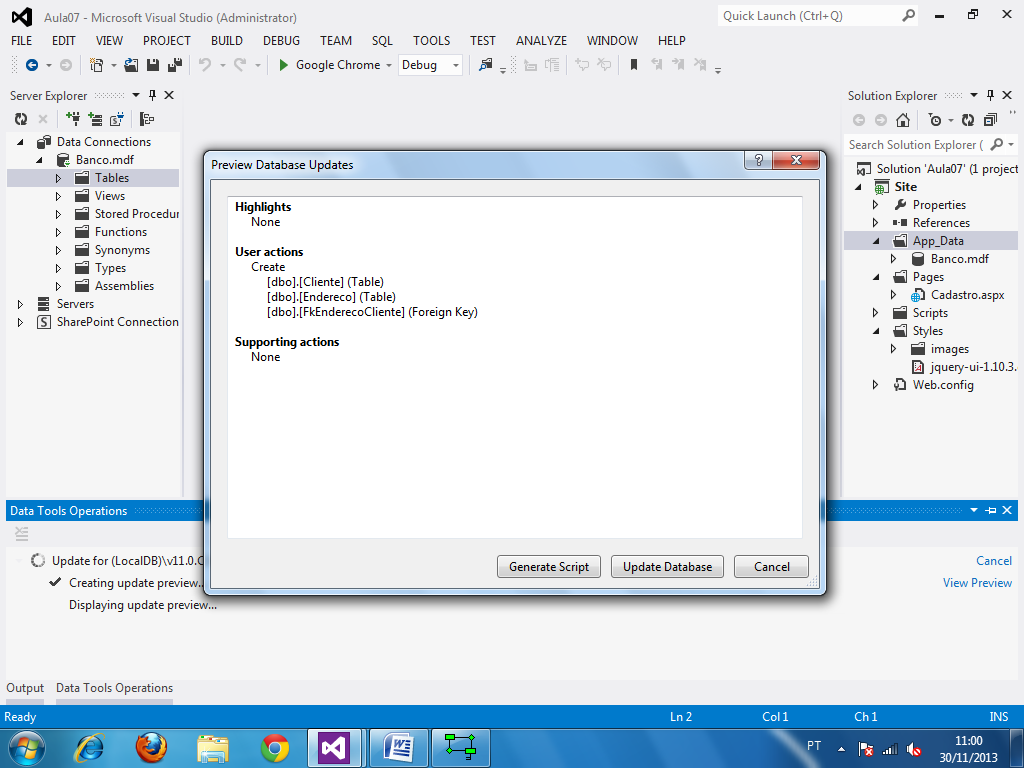
primary key(IdEndereco))

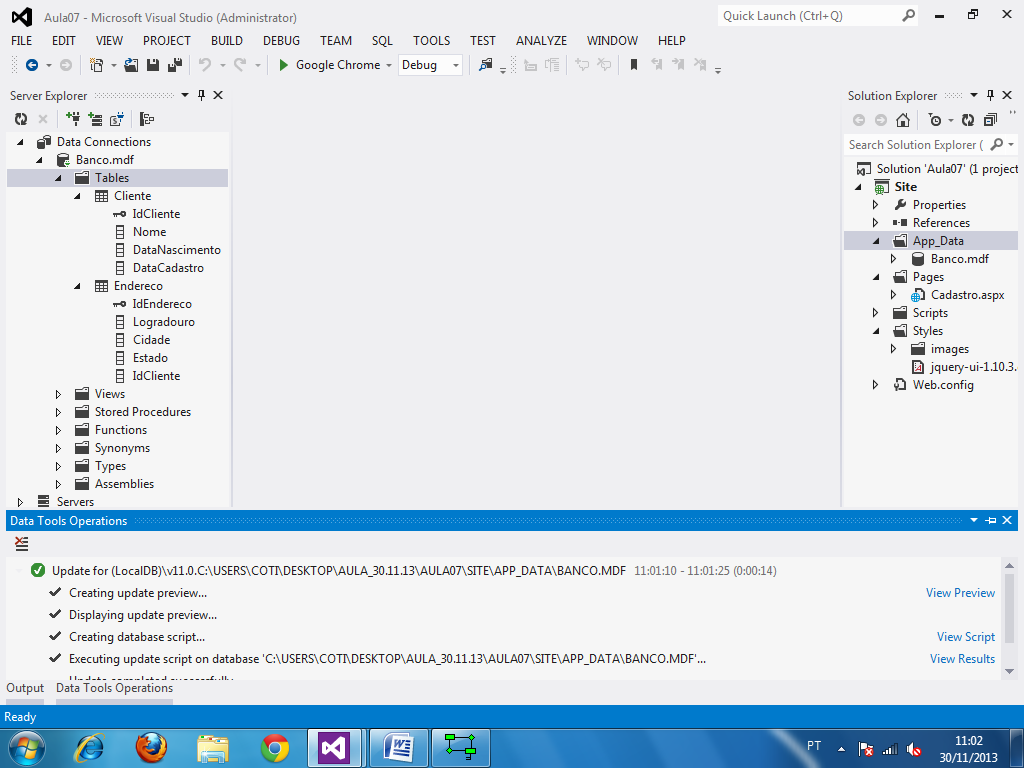
go

alter table Endereco add constraint FkEnderecoCliente

foreign key(IdCliente) references Cliente(IdCliente)

Executando...





Mapear a String de Conexão

\Web.config.xml

<?xml version="1.0"?>

<configuration>

**<connectionStrings>**

**<add**

**name="AULA"**

**connectionString="Data Source=(LocalDB)\v11.0;AttachDbFilename=C:\Users\Coti\Desktop\Aula\_30.11.13\Aula07\Site\App\_Data\Banco.mdf;Integrated Security=True"**

**/>**

**</connectionStrings>**

<system.web>

<compilation debug="true" targetFramework="4.5" />

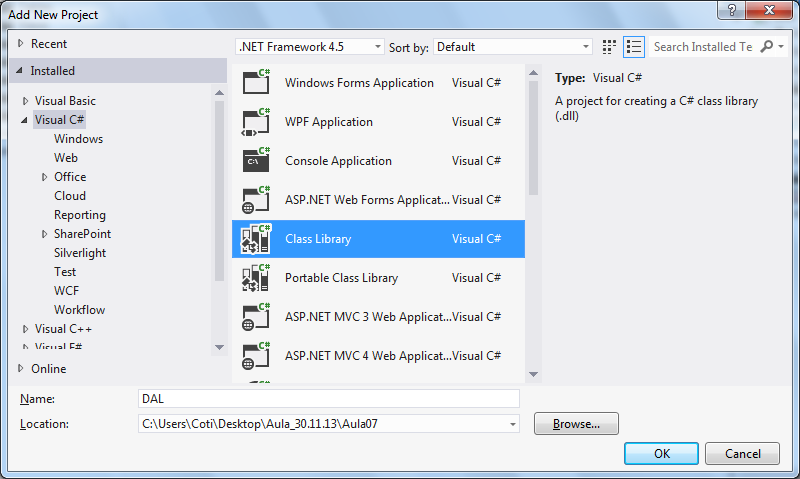
<httpRuntime targetFramework="4.5" />

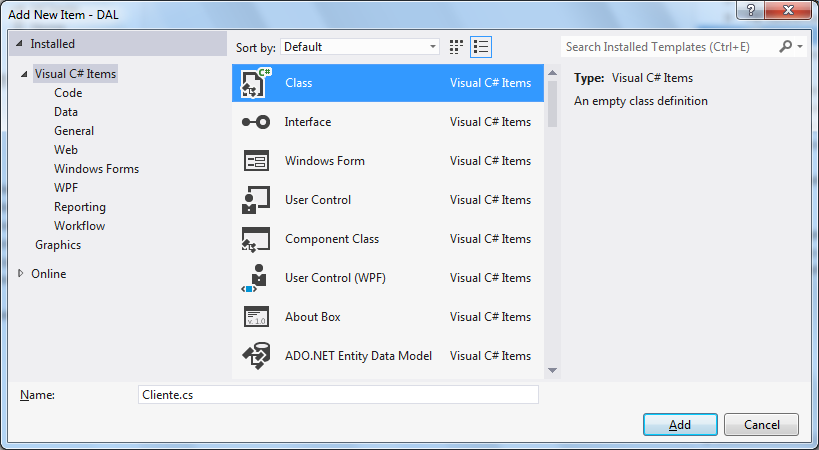
</system.web>

</configuration>

Data Access Layer

Camada de acesso a dados





using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DAL.Model

{

public class Cliente

{

public int IdCliente { get; set; }

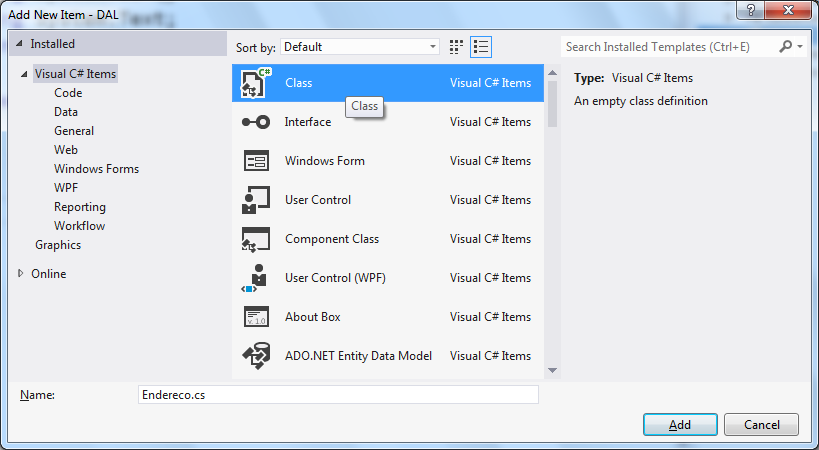
public string Nome { get; set; }

public DateTime DataNascimento { get; set; }

public DateTime DataCadastro { get; set; }

}

}



using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DAL.Model

{

public class Endereco

{

public int IdEndereco { get; set; }

public string Logradouro { get; set; }

public string Cidade { get; set; }

public string Estado { get; set; }

}

}

Definindo os relacionamentos

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DAL.Model

{

public class Cliente

{

public int IdCliente { get; set; }

public string Nome { get; set; }

public DateTime DataNascimento { get; set; }

public DateTime DataCadastro { get; set; }

//Relacionamento de Associação (1p1)

**public Endereco Endereco { get; set; }**

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DAL.Model

{

public class Endereco

{

public int IdEndereco { get; set; }

public string Logradouro { get; set; }

public string Cidade { get; set; }

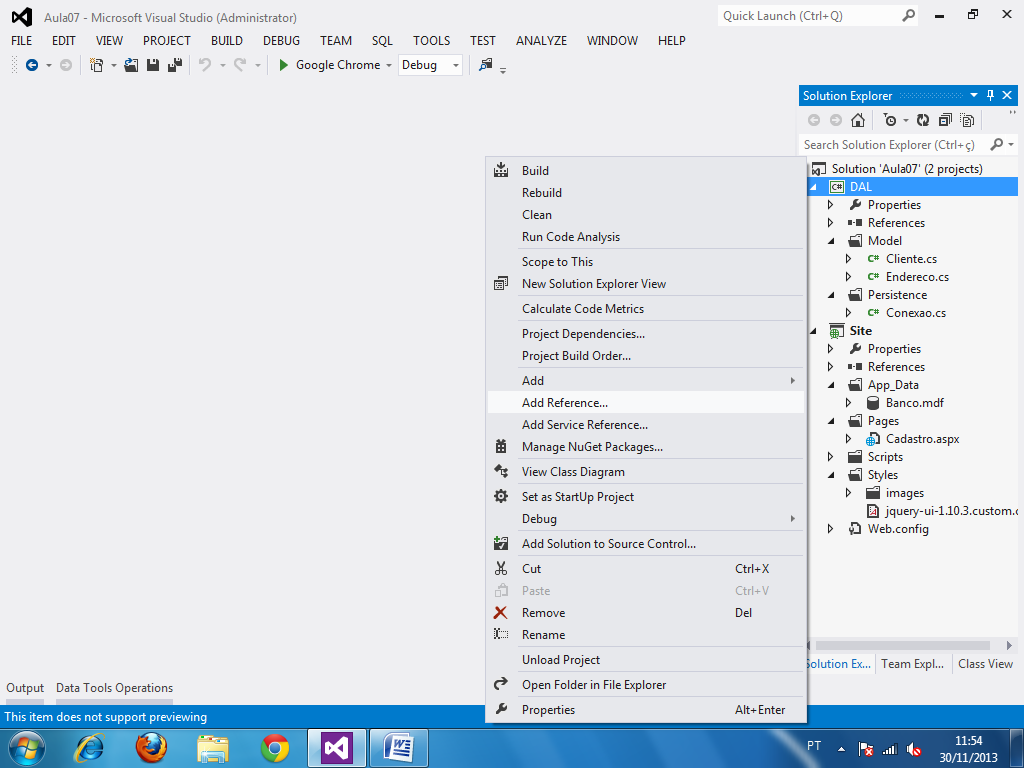
public string Estado { get; set; }

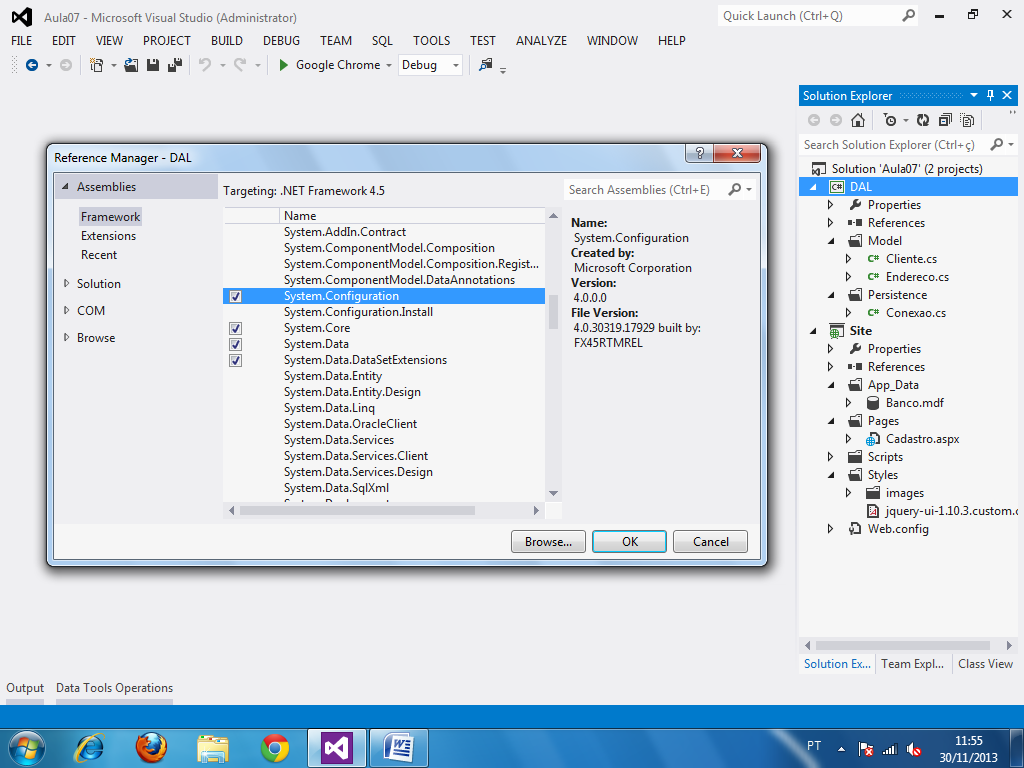
//Relacionamento de Associação

**public Cliente Cliente { get; set; }**

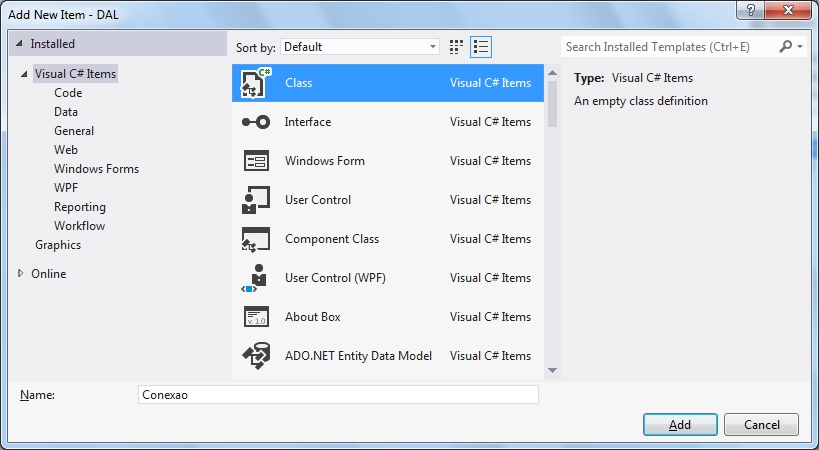
}

}





Criando a Classe de Conexão...



using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Data.SqlClient; //acesso ao sqlserver

using System.Configuration; //capturar o caminho da string de conexão mapeada

namespace DAL.Persistence

{

public class Conexao

{

protected SqlConnection Con;

protected SqlCommand Cmd;

protected SqlDataReader Dr;

protected void AbrirConexao()

{

Con = new SqlConnection();

Con.ConnectionString = ConfigurationManager.

ConnectionStrings["AULA"].ConnectionString;

Con.Open(); //conexão aberta

}

protected void FecharConexao()

{

Con.Close(); //conexão fechada

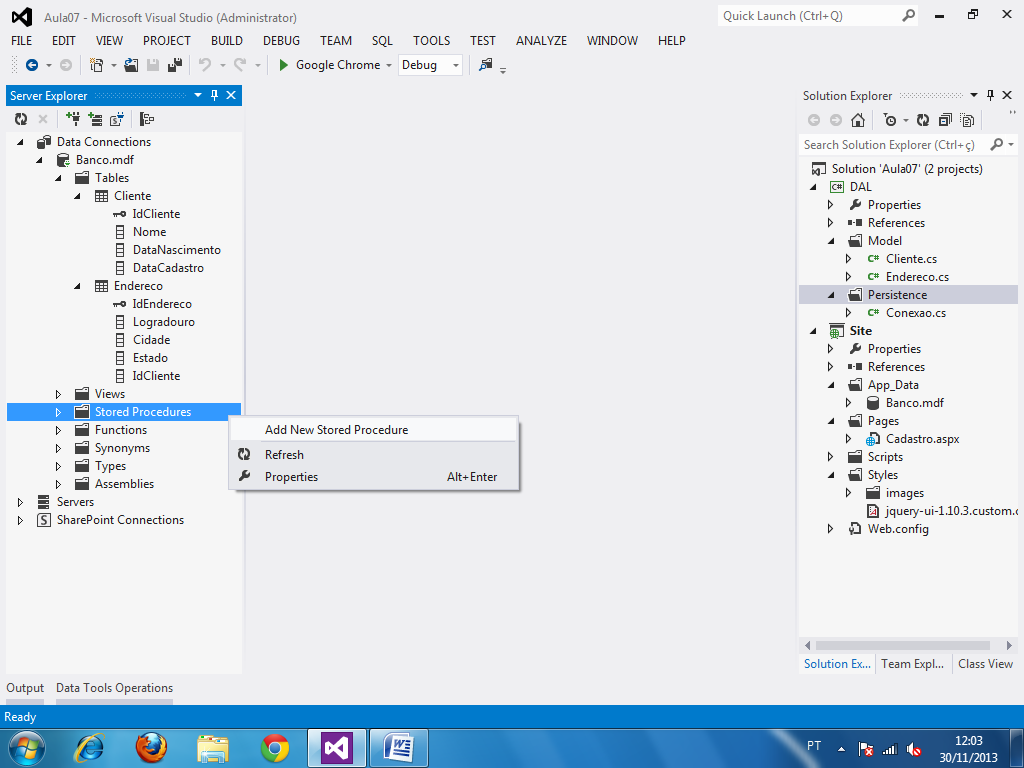
}

}

}

StoredProcedure

Programa escrito em linguagem T-SQL dentro do SqlServer. Utilizado para executar transações ou operações lógicas que envolvam as tabelas do banco de dados.



--StoredProcedure para cadastro de Cliente e Endereco

--Linguagem T-SQL (SqlServer)

create procedure CadCliente

@Nome nvarchar(50),

@DataNasc date,

@Logradouro nvarchar(50),

@Cidade nvarchar(50),

@Estado nvarchar(50)

as

declare

@IdCliente integer,

@DataCad datetime

begin

begin try --inicio do bloco try

begin transaction

--iniciando uma transação em banco de dados

--capturar a data atual do banco de dados

set @DataCad = (select CURRENT\_TIMESTAMP)

--gravar o cliente

insert into Cliente(Nome, DataNascimento, DataCadastro)

values(@Nome, @DataNasc, @DataCad)

--capturar o Id do Cliente que foi cadastrado (identity)

--IDENT\_CURRENT => resgata o ultimo Id gerado

--na tabela de Cliente

set @IdCliente = (select IDENT\_CURRENT('Cliente'))

--gravar o endereço

insert into Endereco(Logradouro, Cidade, Estado,

IdCliente) values(@Logradouro, @Cidade, @Estado,

@IdCliente)

commit --confirma a transação

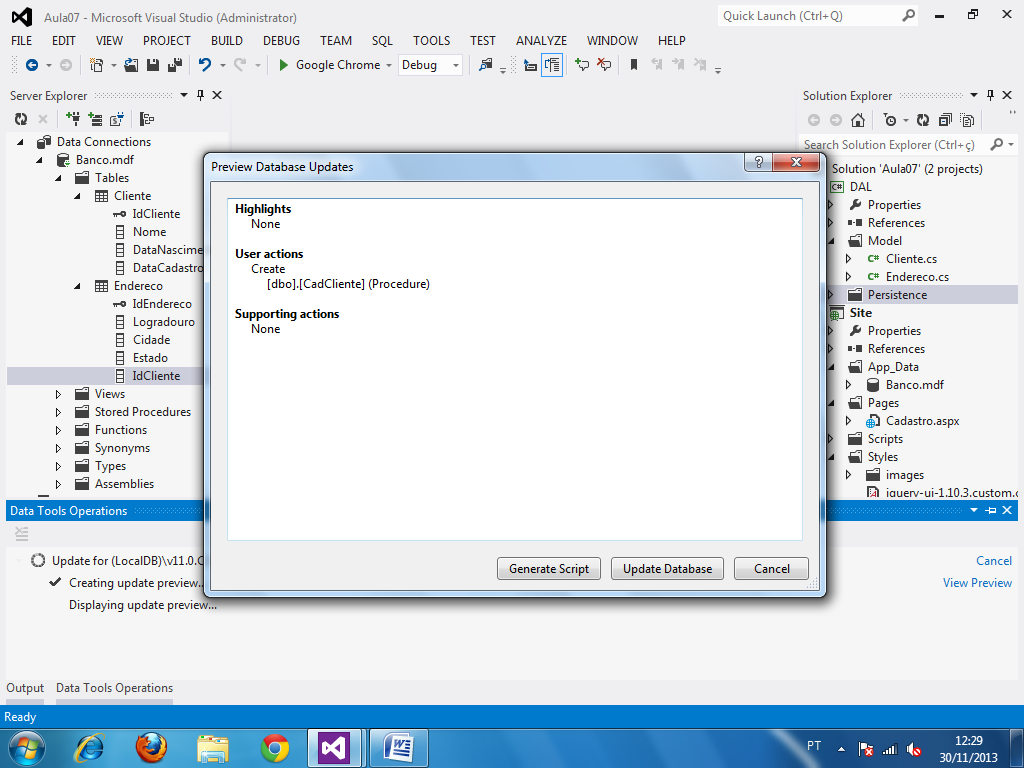
end try --final do bloco try

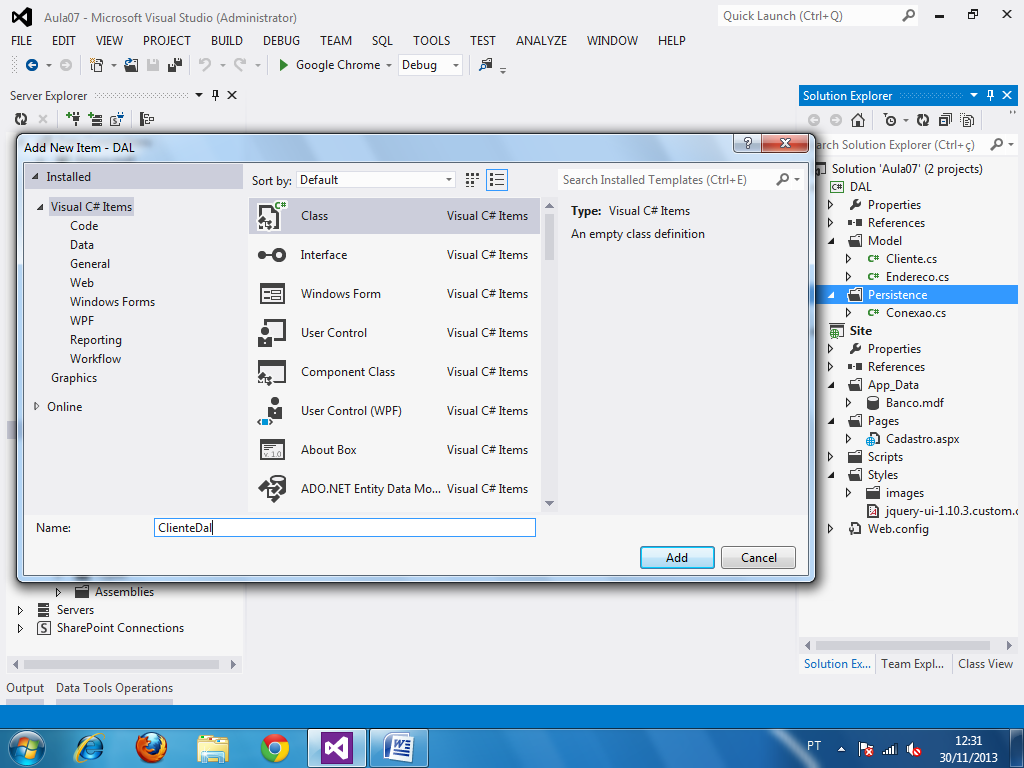
begin catch --inicio do bloco catch

rollback --desfazer a transação

end catch --final do bloco catch

end





using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Data; //storedprocedure

using System.Data.SqlClient; //sqlserver

using DAL.Model; //classes de modelo

namespace DAL.Persistence

{

public class ClienteDal : Conexao

{

//Método para executar a StoredProcedure

public void Salvar(Cliente c, Endereco e)

{

try

{

AbrirConexao();

Cmd = new SqlCommand("CadCliente", Con);

Cmd.CommandType = CommandType.StoredProcedure;

//interpretar o 'Cmd' como procedure

Cmd.Parameters.AddWithValue("@Nome", c.Nome);

Cmd.Parameters.AddWithValue("@DataNasc", c.DataNascimento);

Cmd.Parameters.AddWithValue("@Logradouro", e.Logradouro);

Cmd.Parameters.AddWithValue("@Cidade", e.Cidade);

Cmd.Parameters.AddWithValue("@Estado", e.Estado);

Cmd.ExecuteNonQuery(); //executar

}

catch (Exception ex)

{

throw new Exception("Erro: " + ex.Message);

}

finally

{

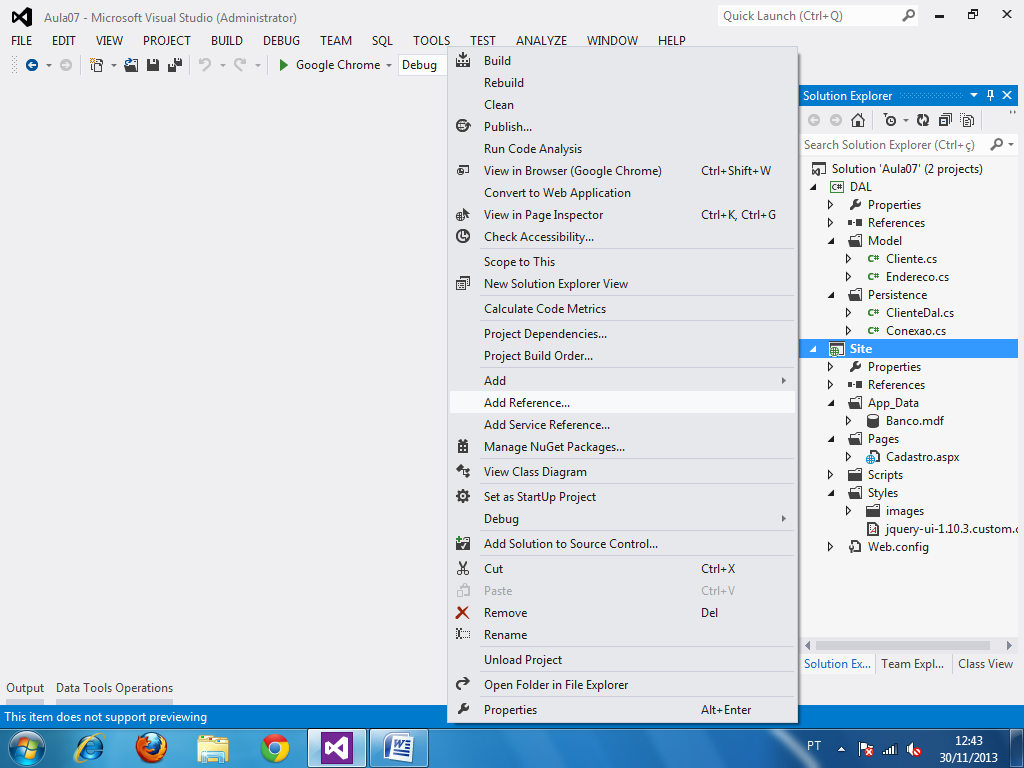
FecharConexao();

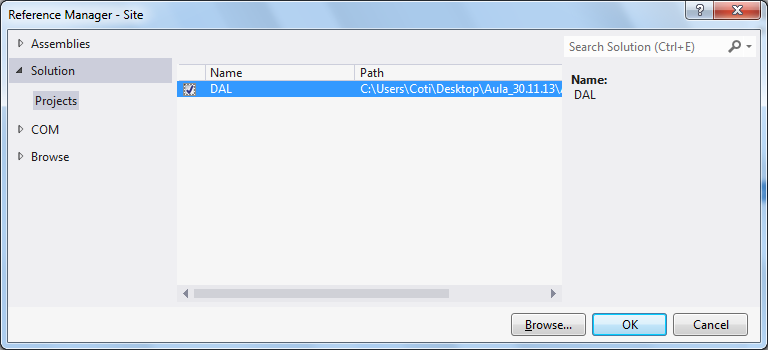
}

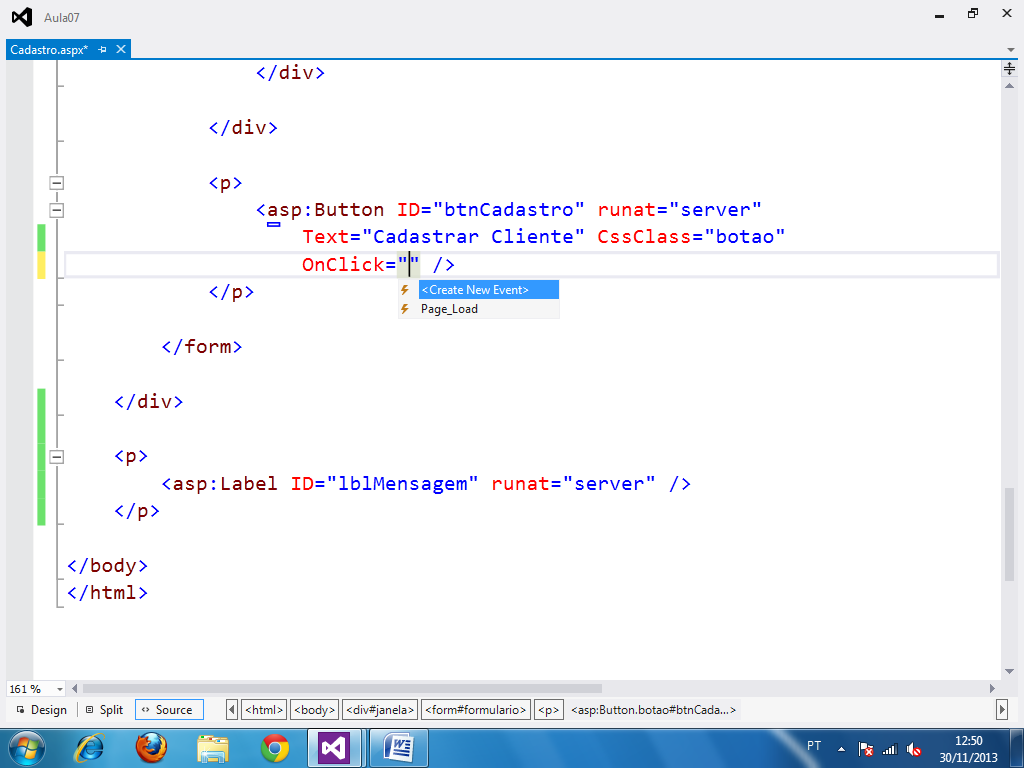
}

}

}







<asp:Button ID="btnCadastro" runat="server"

Text="Cadastrar Cliente" CssClass="botao"

**OnClick="btnCadastro\_Click"** />

CodeBehind

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using DAL.Model;

using DAL.Persistence;

namespace Site.Pages

{

public partial class Cadastro : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

**protected void btnCadastro\_Click(object sender, EventArgs e)**

{

try

{

Cliente cli = new Cliente();

cli.Nome = txtNome.Text;

cli.DataNascimento = Convert.ToDateTime(txtDataNasc.Text);

Endereco end = new Endereco();

end.Logradouro = txtLogradouro.Text;

end.Cidade = txtCidade.Text;

end.Estado = txtEstado.Text;

ClienteDal d = new ClienteDal();

d.Salvar(cli, end); //gravação

lblMensagem.Text = "Cliente cadastrado com sucesso.";

}

catch(Exception ex)

{

lblMensagem.Text = ex.Message; //mensagem de erro

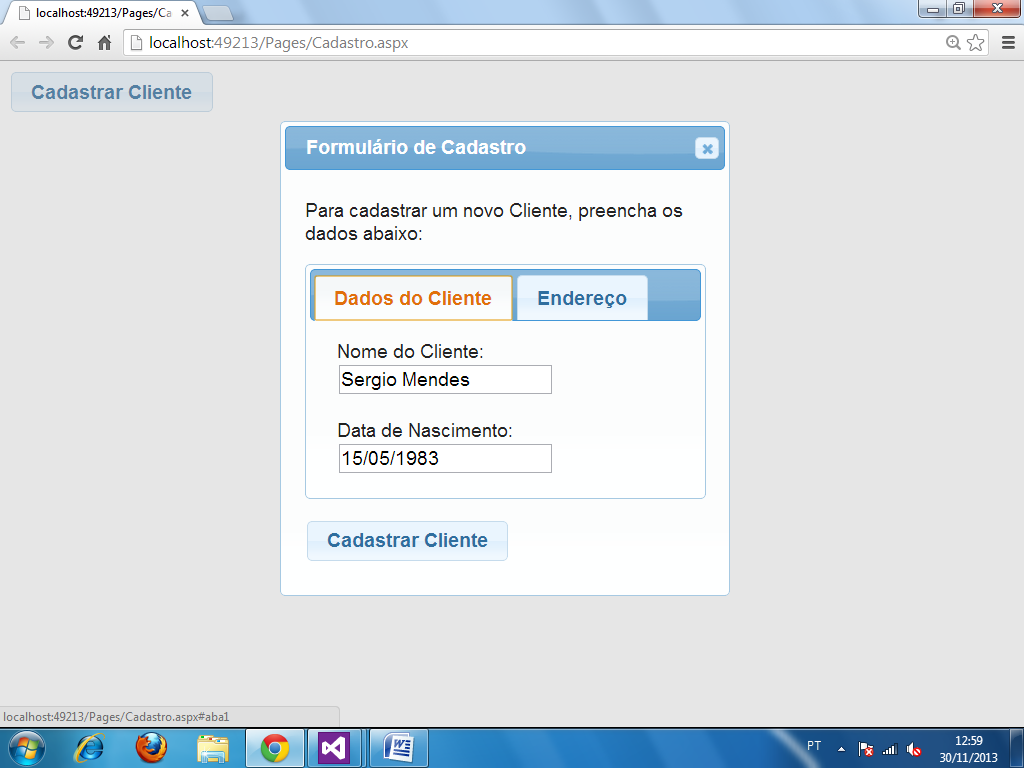
}

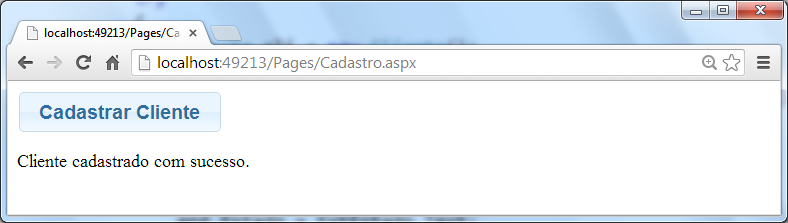
}

}

}

Executando...





No banco de dados...

