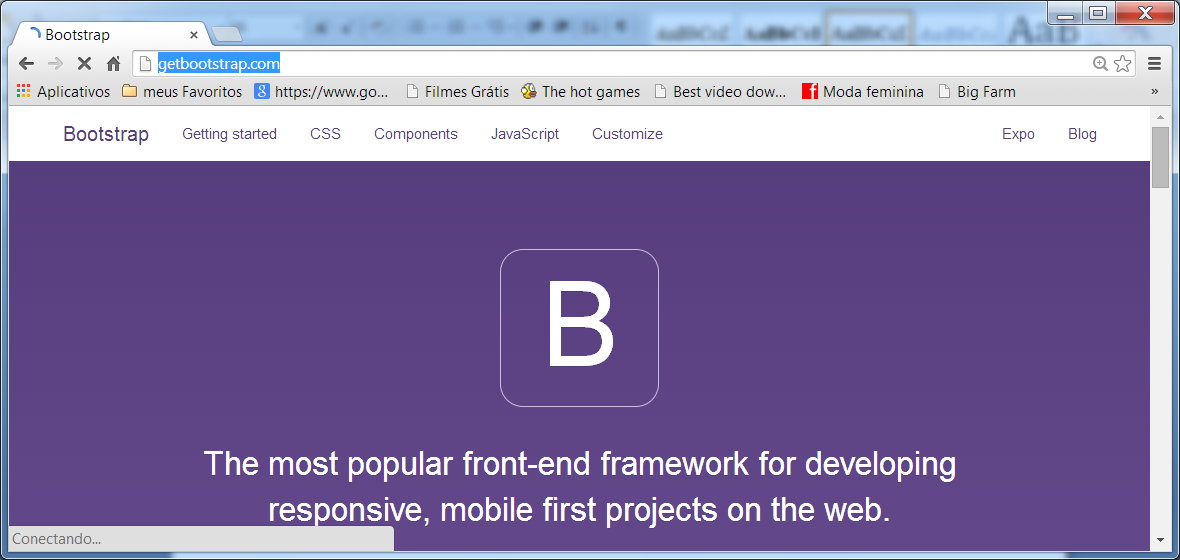
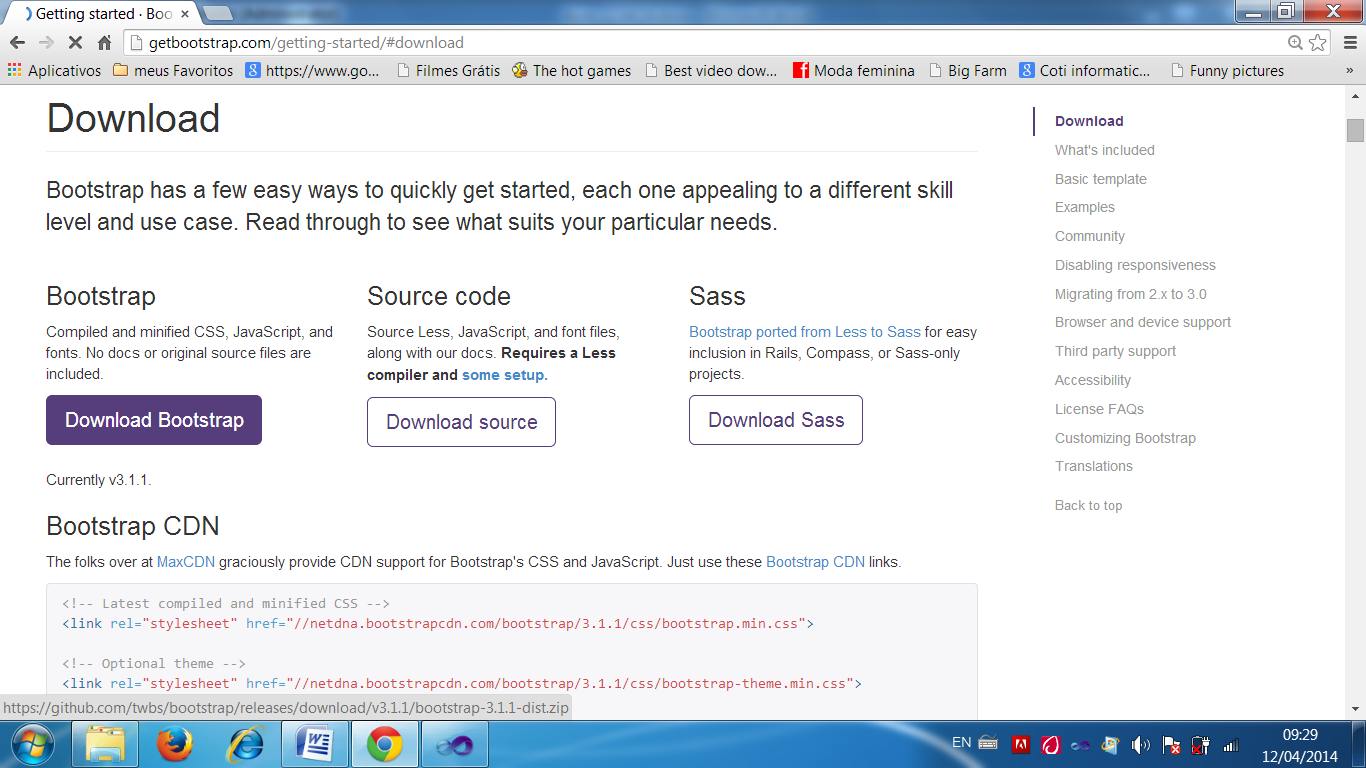


Bootstrap (http://getbootstrap.com/)

Galeria de componentes baseado em HTML5



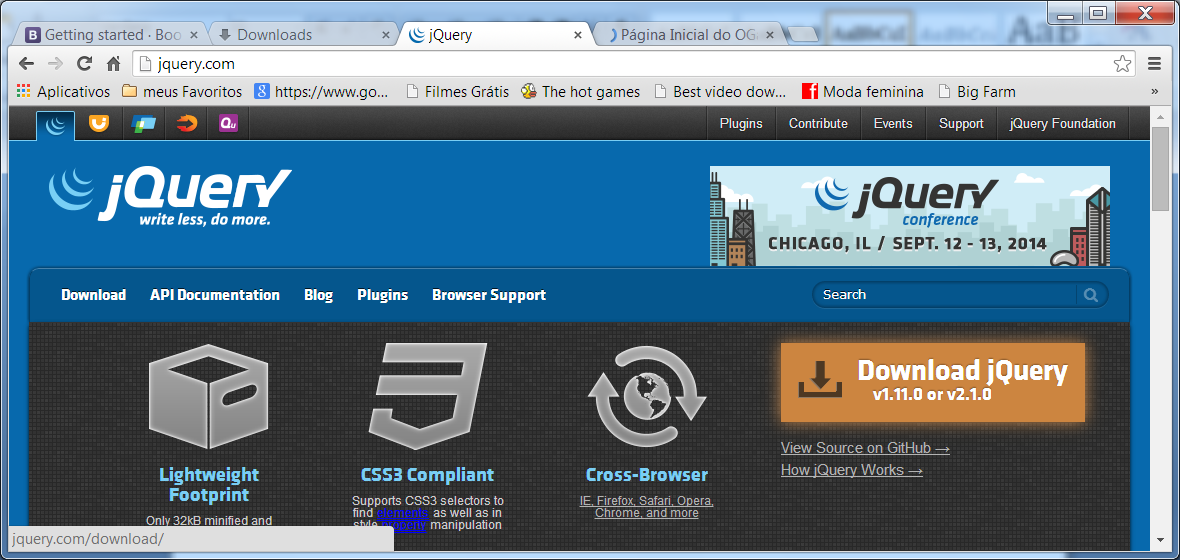


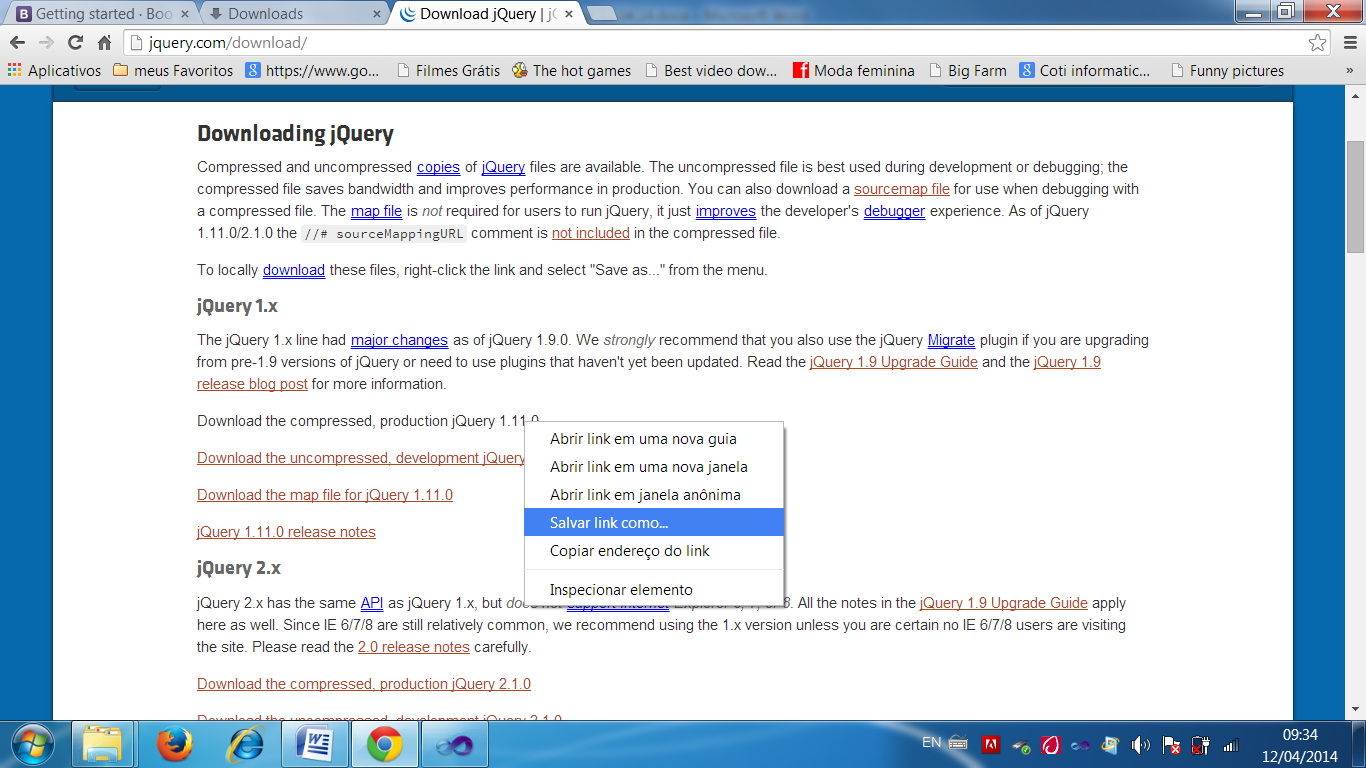
Incluindo no projeto

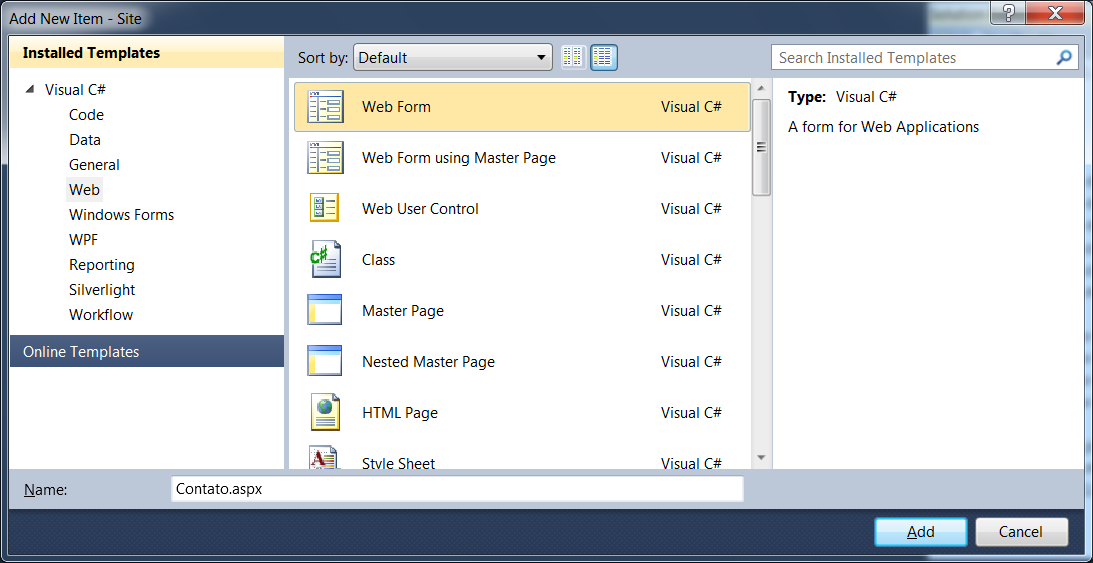
* css
* fonts
* js



<http://jquery.com/>







/Pages/Contato.aspx

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

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

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

<head runat="server">

<title></title>

<link href="../css/bootstrap.css" rel="stylesheet" type="text/css" />

<script src="../js/jquery-1.11.0.min.js" type="text/javascript"></script>

<script src="../js/bootstrap.js" type="text/javascript"></script>

</head>

<body>

<div class="well">

<h3>Projeto de Envio de Mensagens</h3>

<button class="btn btn-primary" data-toggle="modal" data-

target="#janela">

Escrever Email <span class="glyphicon glyphicon-envelope"></span>

</button>

<p>

<asp:Label ID="lblMensagem" runat="server" />

</p>

</div>

<!-- Janela de diálogo -->

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

<div class="modal-dialog">

<div class="modal-content">

<div class="modal-header">

<h4 class="modal-title">Formulário de Envio de Emails</h4>

</div>

<div class="modal-body">

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

<div class="form-group">

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

CssClass="form-control"

placeholder="Email para" />

</div>

<div class="form-group">

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

CssClass="form-control"

placeholder="Assunto da Mensagem" />

</div>

<div class="form-group">

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

CssClass="form-control"

placeholder="Texto da Mensagem"

TextMode="MultiLine" Height="100" />

</div>

<div class="form-group">

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

Text="Enviar Mensagem"

CssClass="btn btn-success"

OnClick="EnviarMensagem" />

</div>

</form>

</div>

<div class="modal-footer">

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

Fechar Formulário

</button>

</div>

</div>

</div>

</div>

</body>

</html>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace Site.Pages

{

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

{

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

{

}

protected void EnviarMensagem(object sender, EventArgs e)

{

lblMensagem.Text = "Mensagem enviada com sucesso.";

}

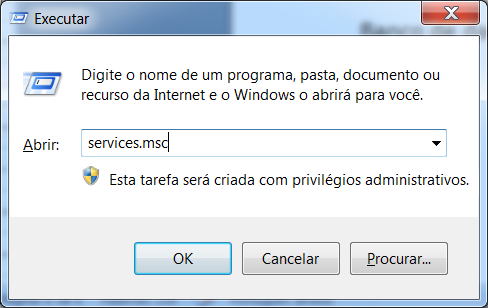
}

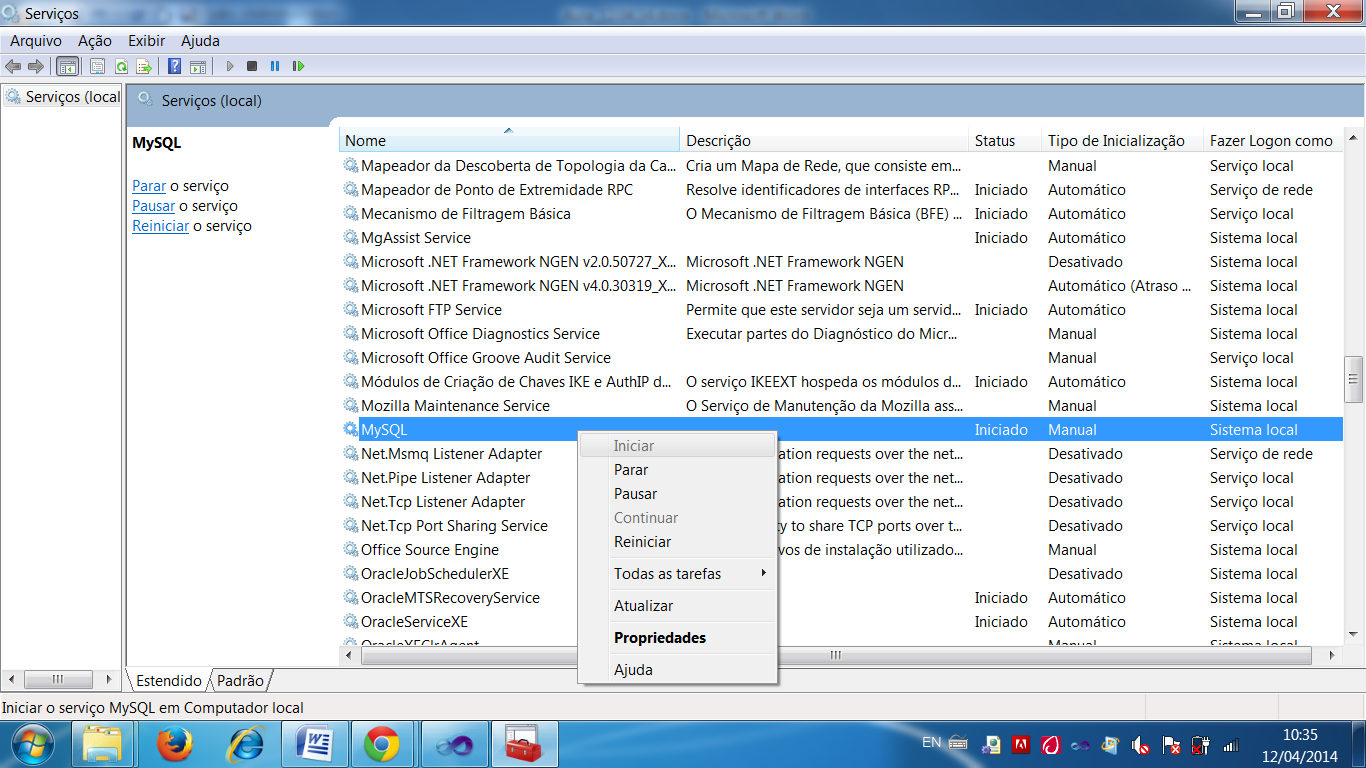
}

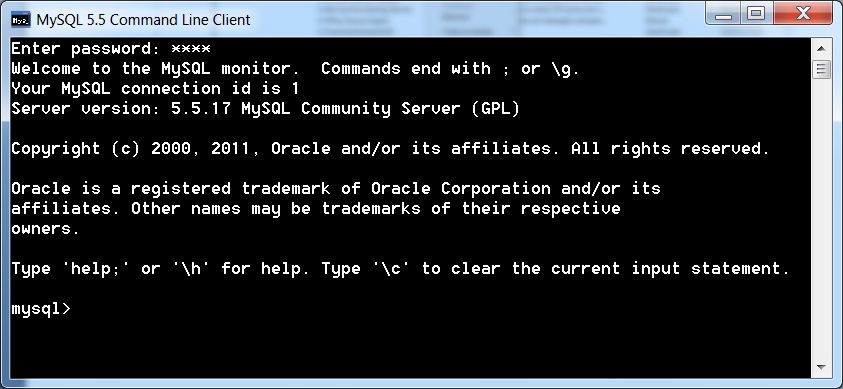


Banco de dados MySql

* Iniciando o serviço do MySQL...

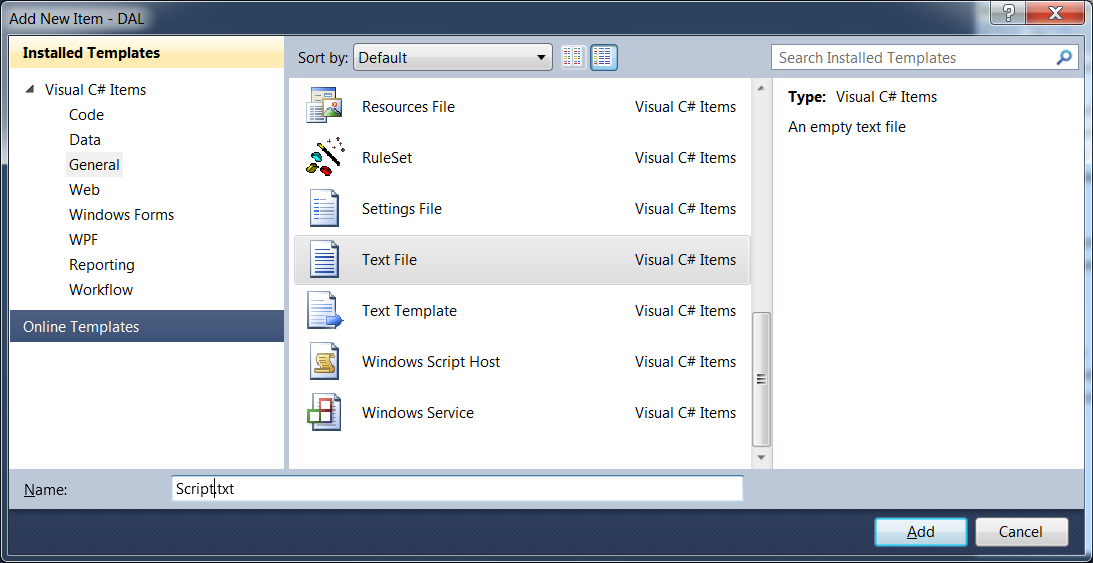








Criando o Script da base de dados...



drop database if exists aula;

create database aula;

use aula;

create table mensagem(

idmensagem integer auto\_increment,

destinatario varchar(50) not null,

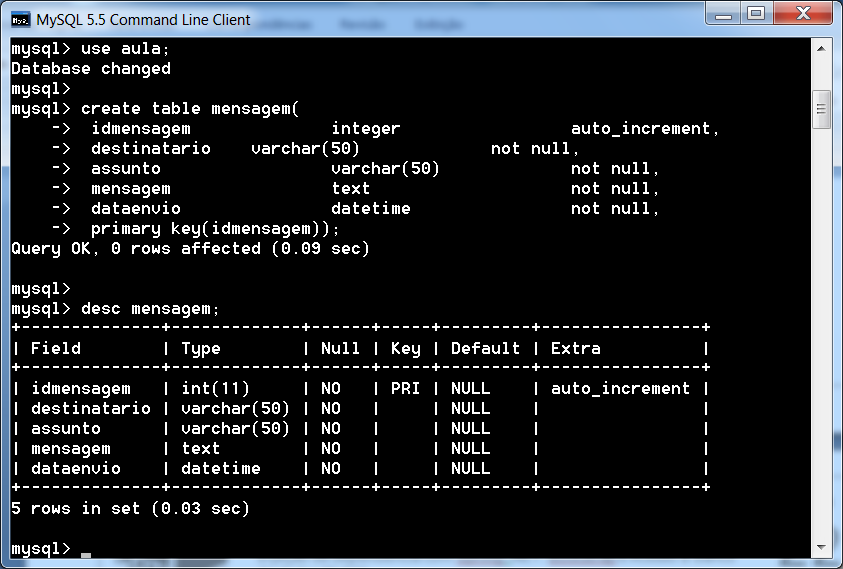
assunto varchar(50) not null,

mensagem text not null,

dataenvio datetime not null,

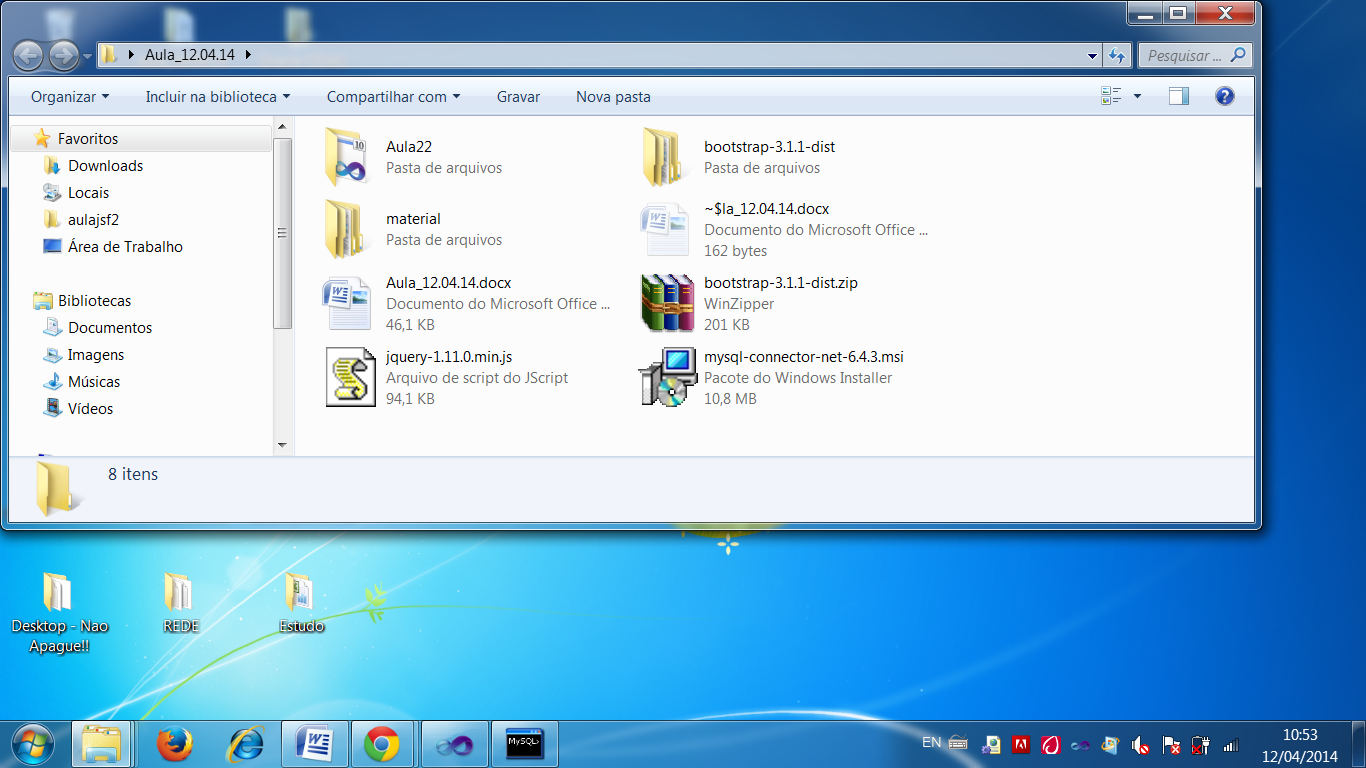
primary key(idmensagem));

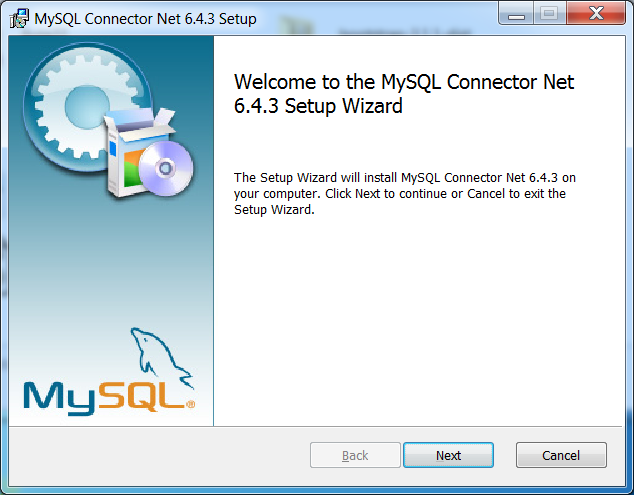
desc mensagem;

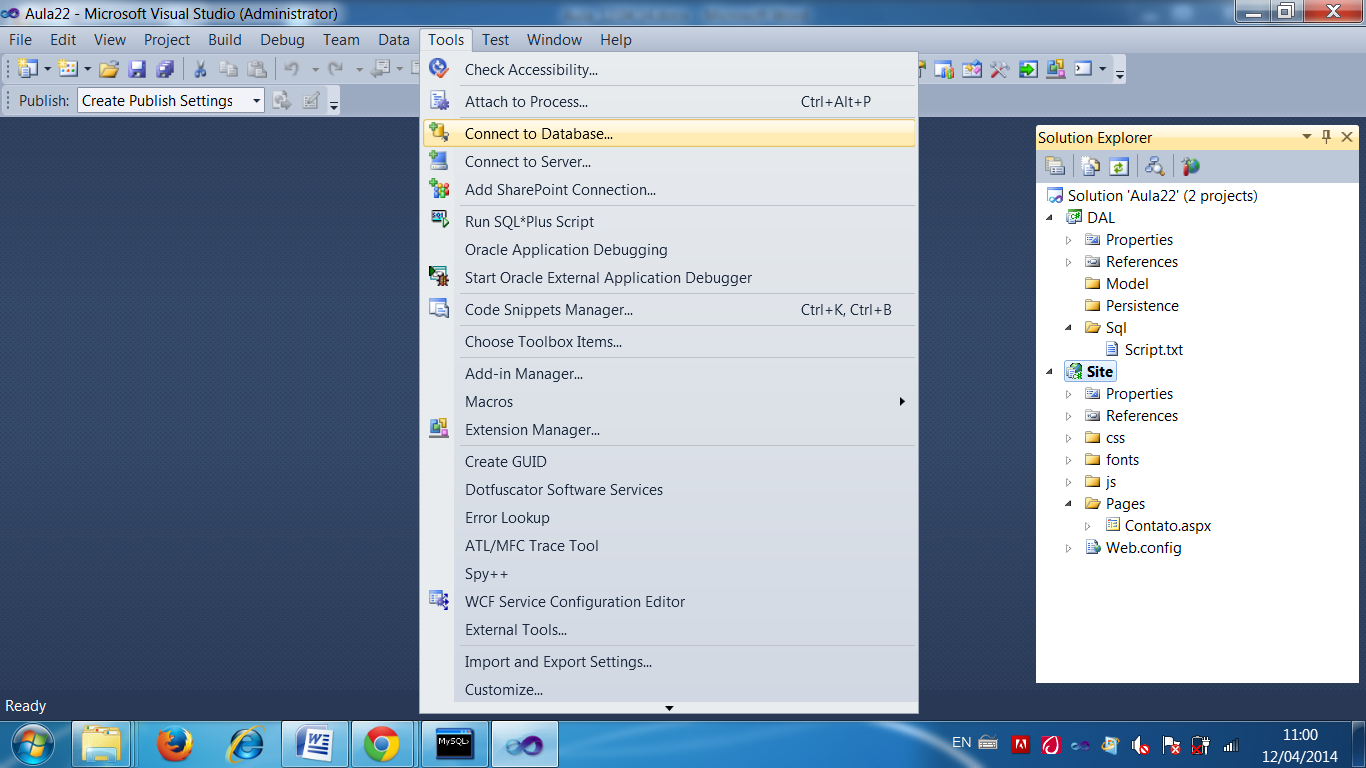


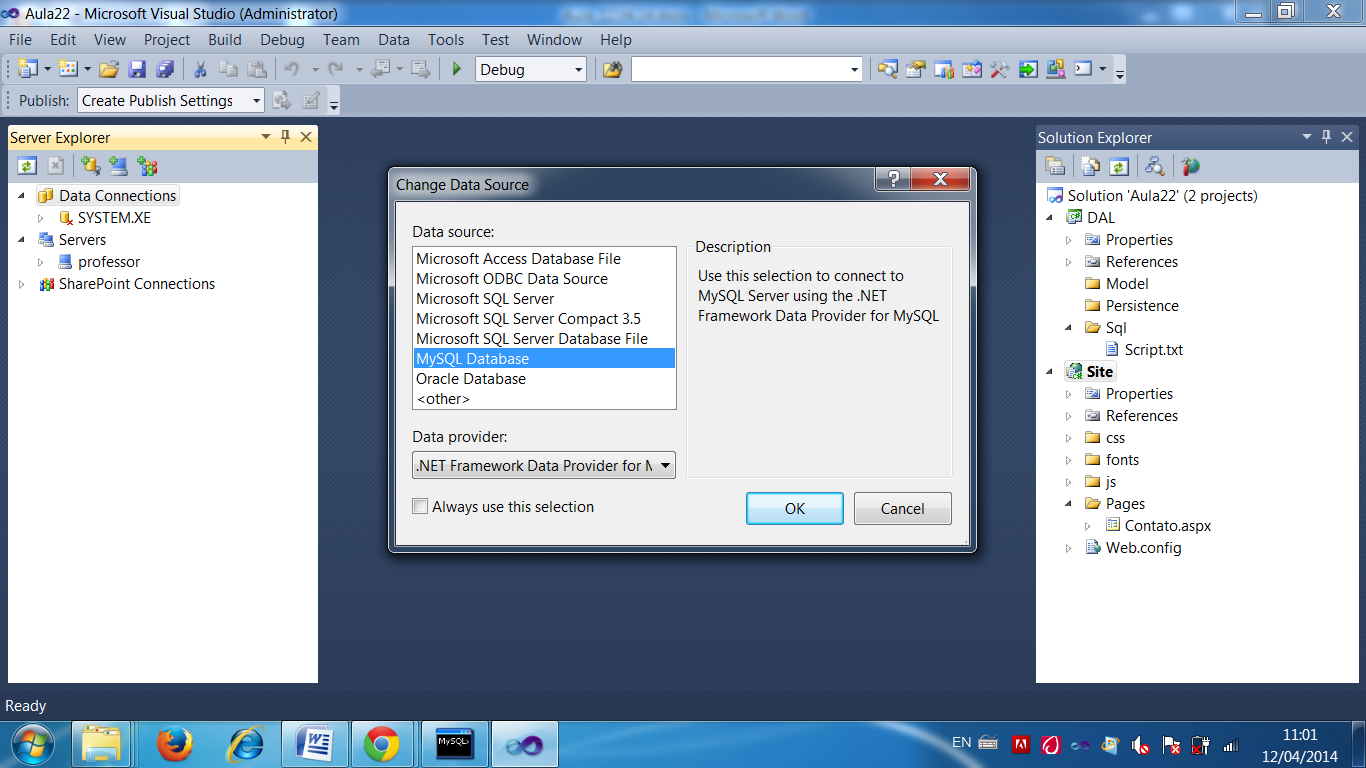
Conectando o .Net ao MySQL

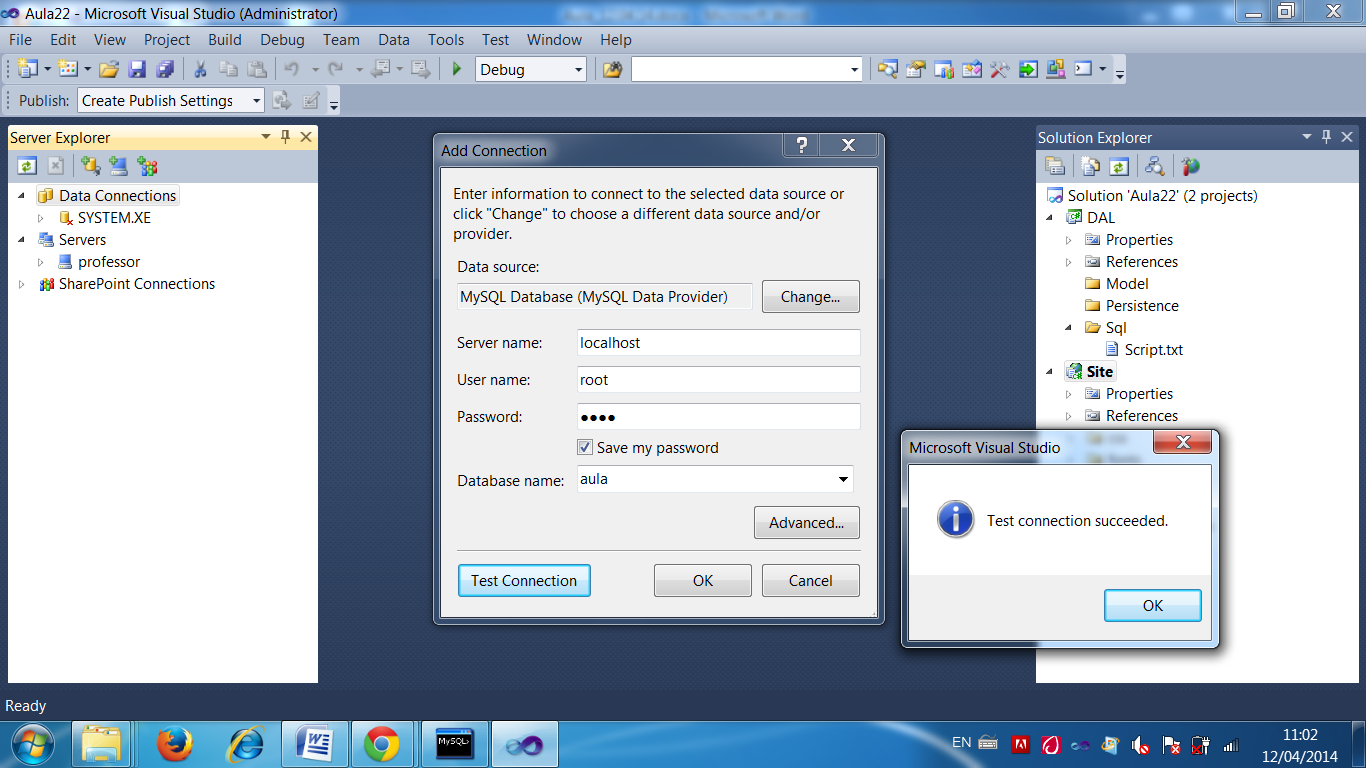
MySql Connector .Net

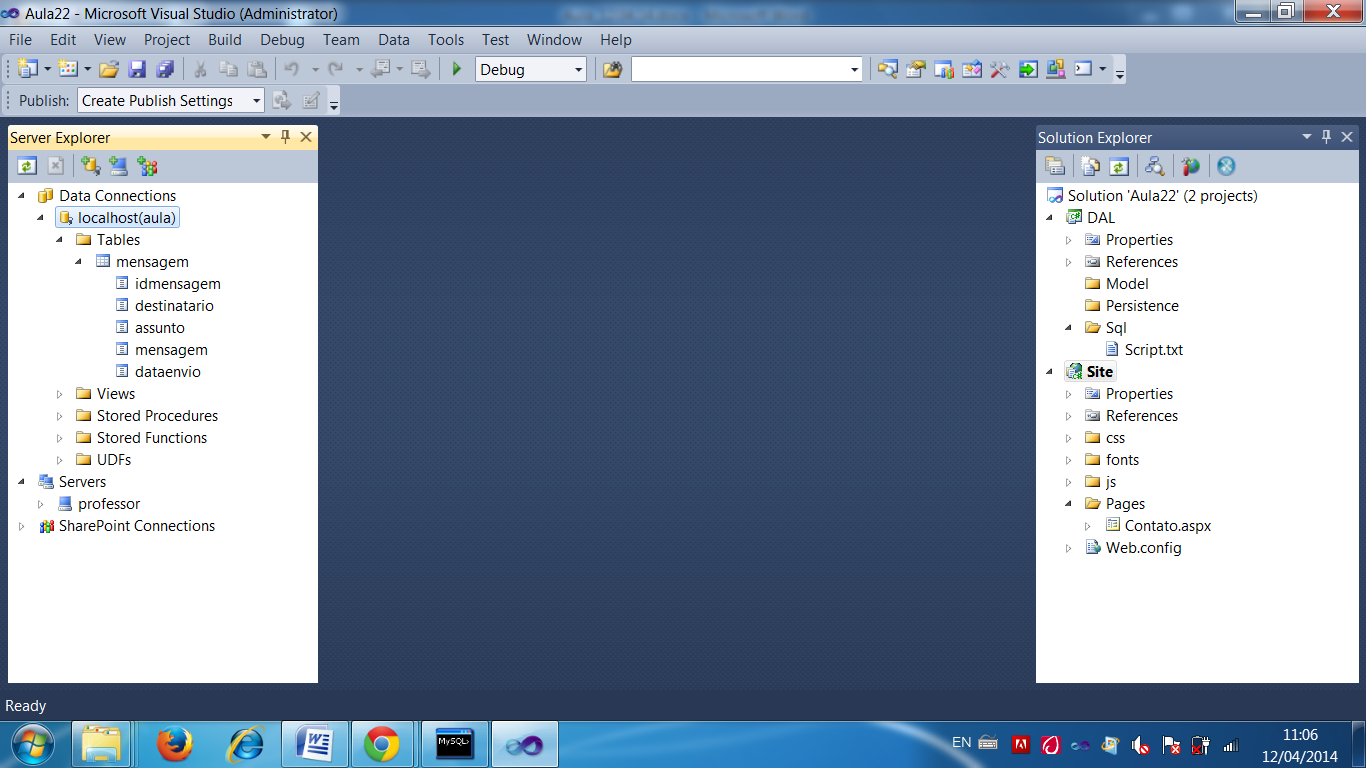




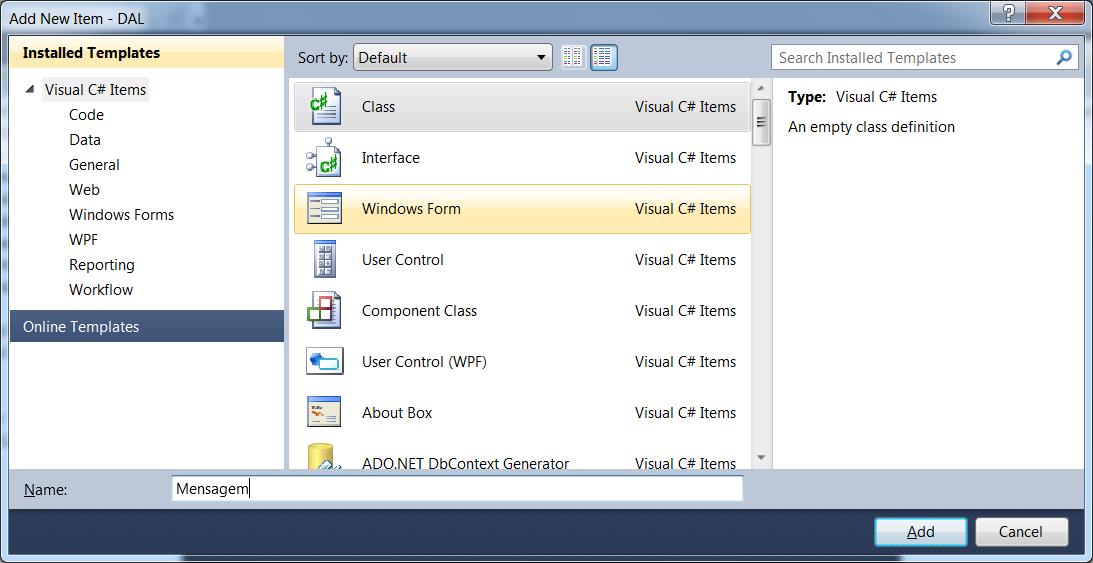








Modelagem da entidade...



using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace DAL.Model

{

public class Mensagem

{

public int IdMensagem { get; set; }

public string Destinatario { get; set; }

public string Assunto { get; set; }

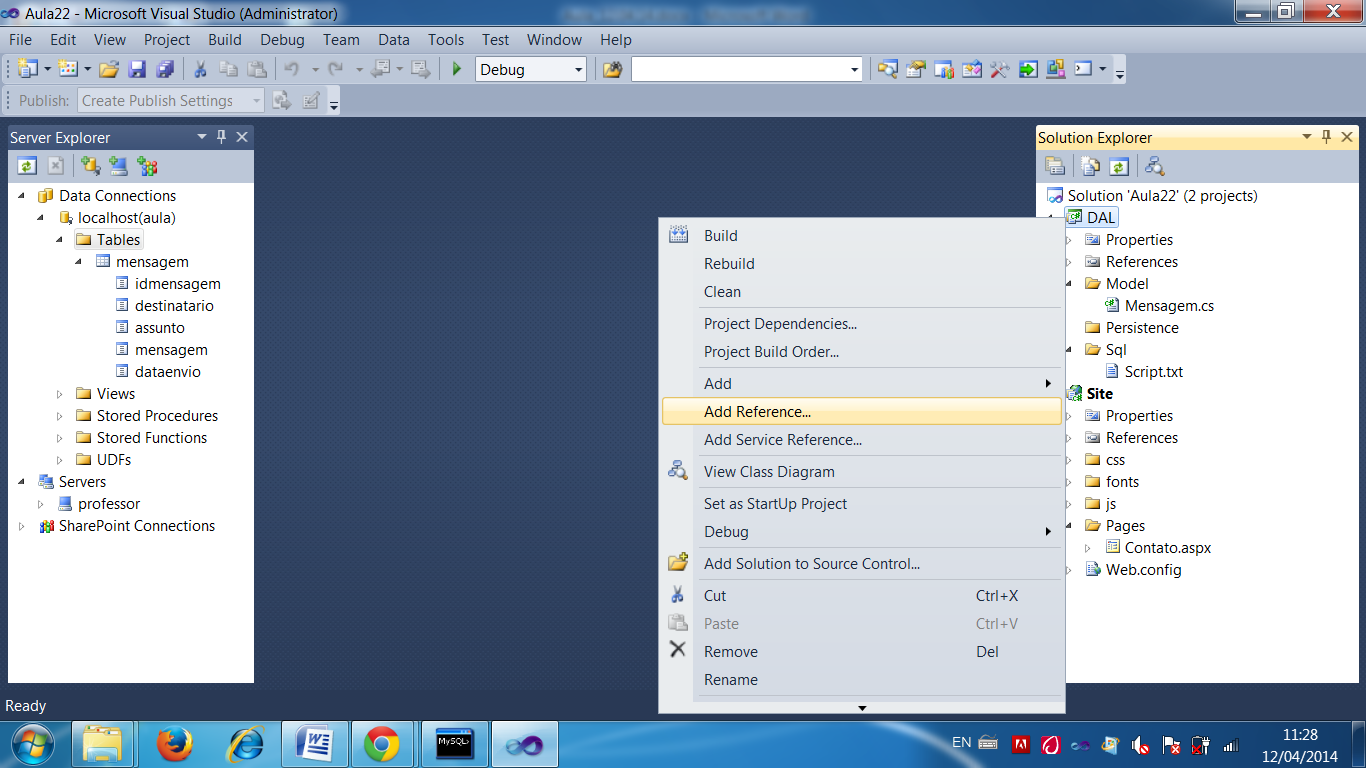
public string Texto { get; set; }

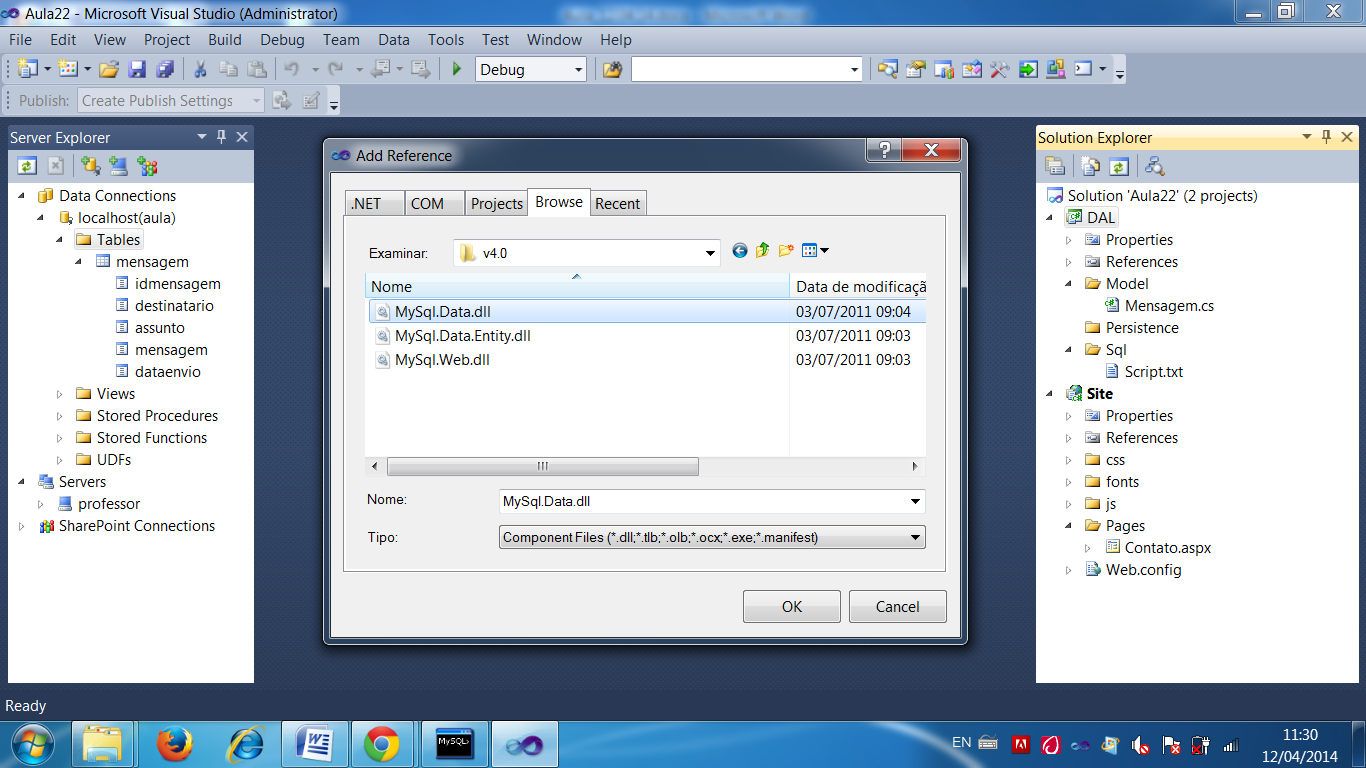
public DateTime DataEnvio { get; set; }

}

}

Adicionando referência na DAL para as   
bibliotecas do MySQL Connector





Classe de Conexão com o MySQL

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using MySql.Data.MySqlClient; //System.Data.SqlClient

namespace DAL.Persistence

{

public class Conexao

{

protected MySqlConnection Con;

protected MySqlCommand Cmd;

protected MySqlDataReader Dr;

protected void AbrirConexao()

{

Con = new MySqlConnection();

Con.Open();

}

protected void FecharConexao()

{

Con.Close();

}

}

}

<?xml version="1.0"?>

<configuration>

<system.web>

<compilation debug="true" targetFramework="4.0" />

</system.web>

**<connectionStrings>**

**<add**

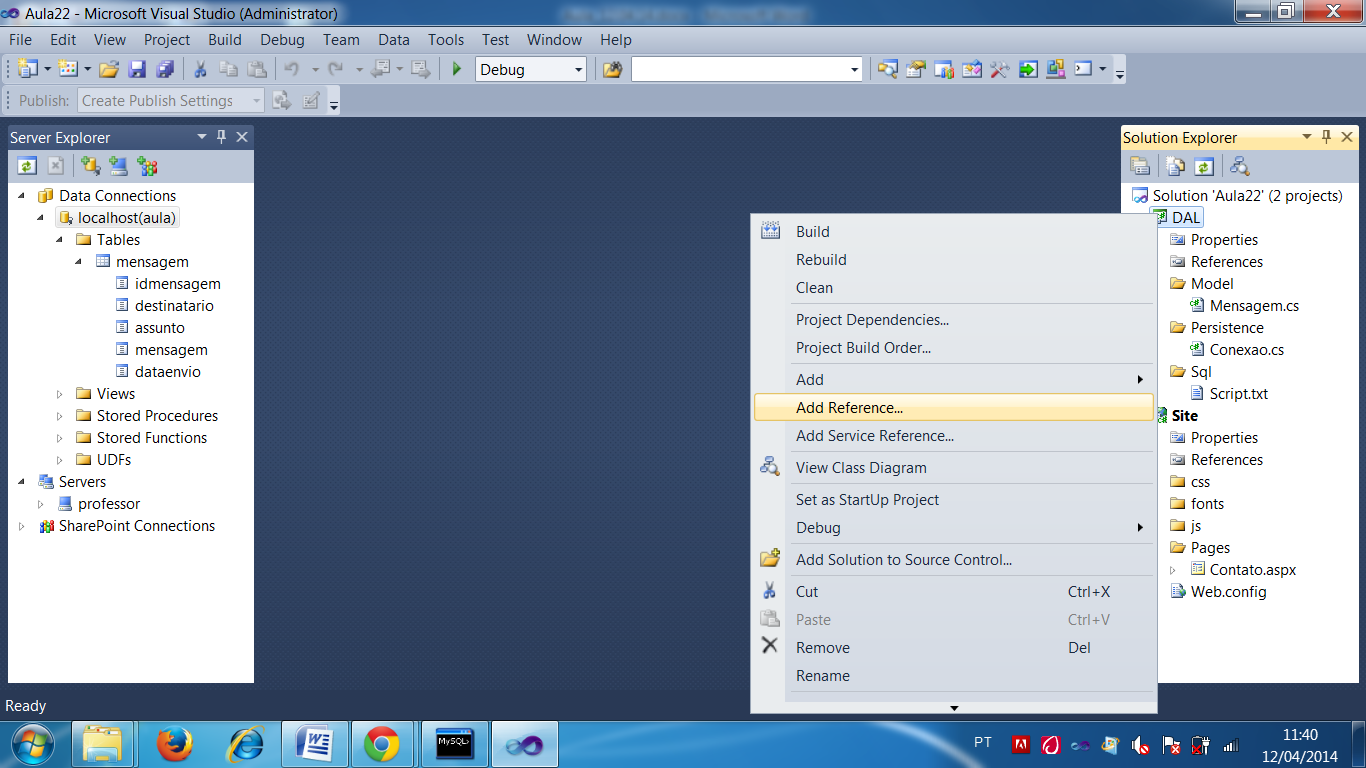
**name="mysql"**

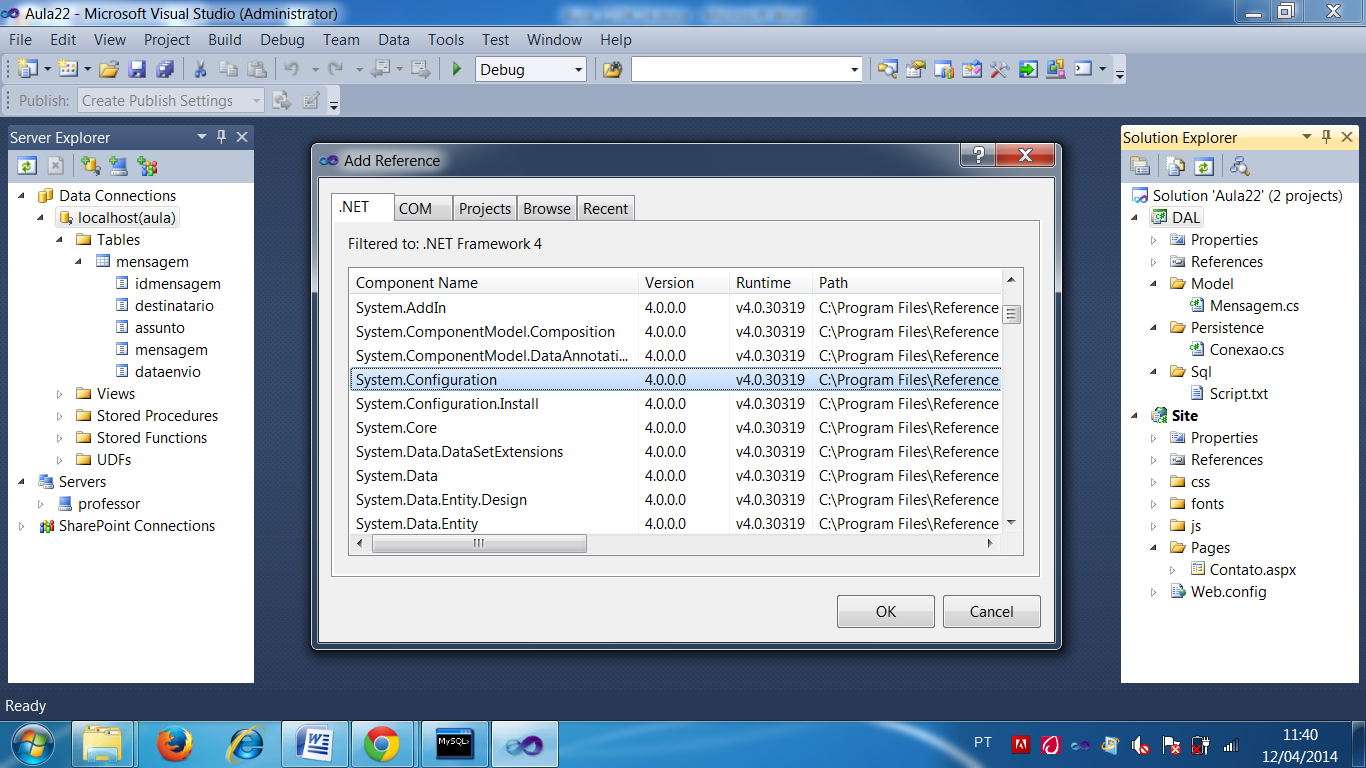
**connectionString="server=localhost;User Id=root;Persist Security Info=True;database=aula;password=coti"**

**/>**

**</connectionStrings>**

</configuration>





using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using MySql.Data.MySqlClient; //System.Data.SqlClient

using System.Configuration;

namespace DAL.Persistence

{

public class Conexao

{

protected MySqlConnection Con;

protected MySqlCommand Cmd;

protected MySqlDataReader Dr;

protected void AbrirConexao()

{

Con = new MySqlConnection(ConfigurationManager.

ConnectionStrings["mysql"].ConnectionString);

Con.Open();

}

protected void FecharConexao()

{

Con.Close();

}

}

}

Classe de persistência...

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using MySql.Data.MySqlClient;

using DAL.Model;

namespace DAL.Persistence

{

public class MensagemDal : Conexao

{

//Método para gravar o registro da mensagem na tabela do MySql

public void Salvar(Mensagem m)

{

try

{

AbrirConexao();

Cmd = new MySqlCommand("insert into mensagem values(null,

@destinatario, @assunto, @mensagem, now())", Con);

Cmd.Parameters.AddWithValue("@destinatario", m.Destinatario);

Cmd.Parameters.AddWithValue("@assunto", m.Assunto);

Cmd.Parameters.AddWithValue("@mensagem", m.Texto);

Cmd.ExecuteNonQuery(); //executar

}

catch(Exception e)

{

throw new Exception("Erro ao salvar mensagem: " + e.Message);

}

finally

{

FecharConexao();

}

}

}

}

Implementando o Cadastro...

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 Contato : System.Web.UI.Page

{

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

{

}

protected void EnviarMensagem(object sender, EventArgs e)

{

try

{

Mensagem m = new Mensagem();

m.Destinatario = txtDestinatario.Text;

m.Assunto = txtAssunto.Text;

m.Texto = txtMensagem.Text;

MensagemDal d = new MensagemDal();

d.Salvar(m);

lblMensagem.Text = "Mensagem " + m.Assunto

+ ", enviada com sucesso.";

}

catch(Exception ex)

{

lblMensagem.Text = ex.Message;

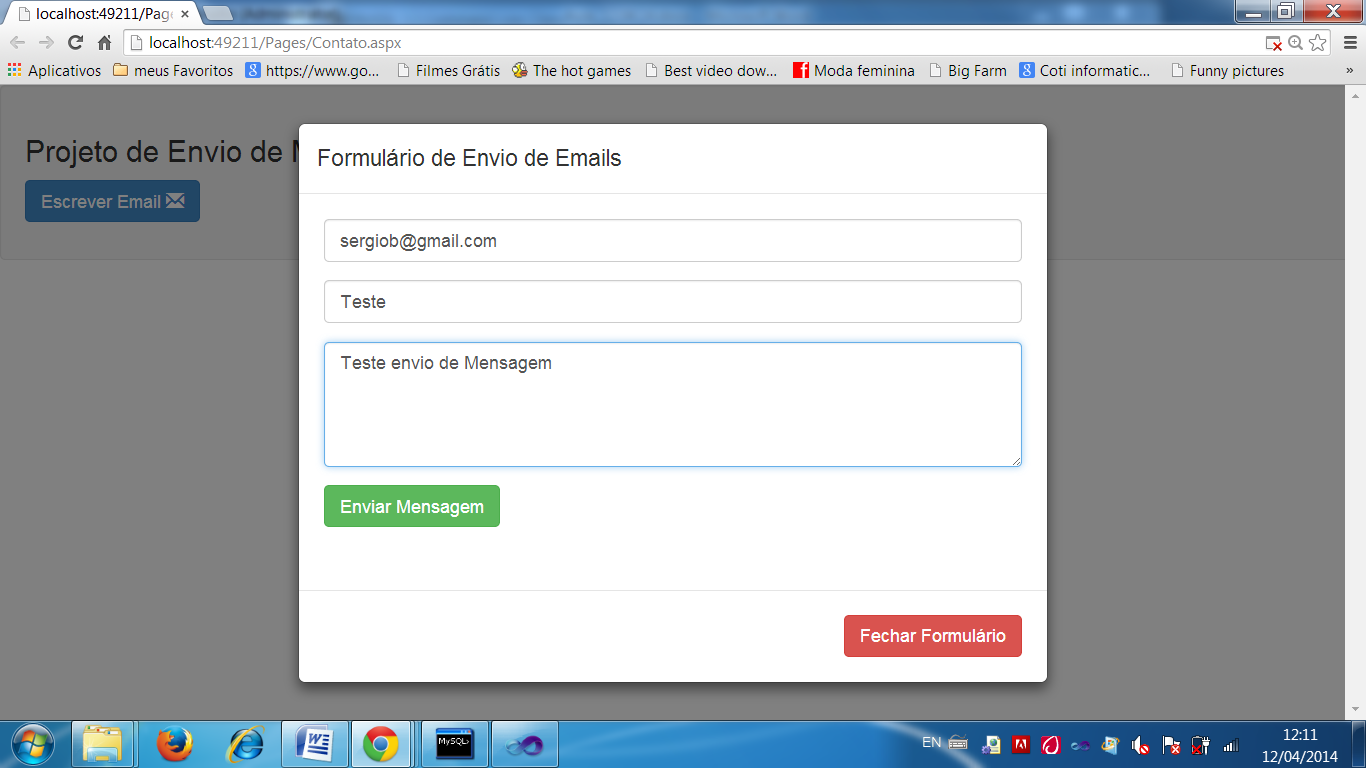
}

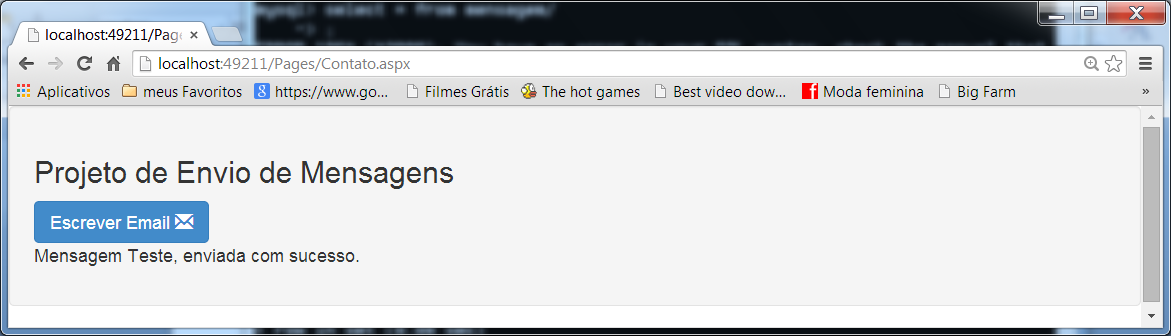
}

}

}

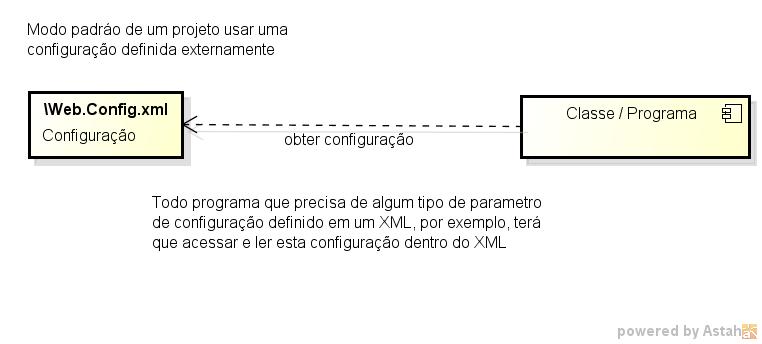
Executando...





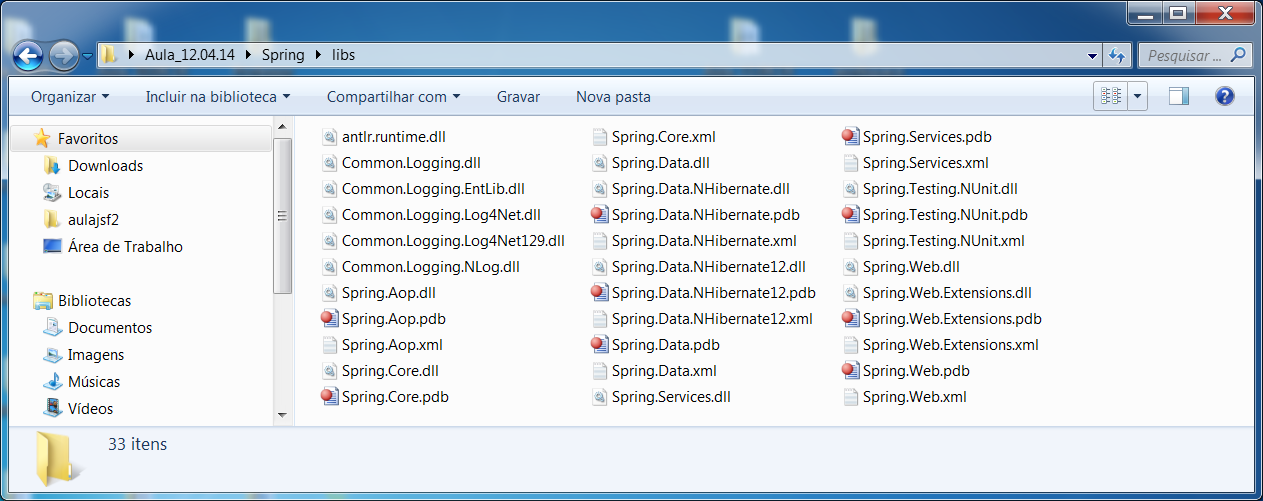
Spring .NET

Framework Spring é uma tecnologia desenvolvida pela comunidade Java e tem como principal função permitir que os programadores trabalhem com Injeção de Dependência em projetos orientados a objetos.

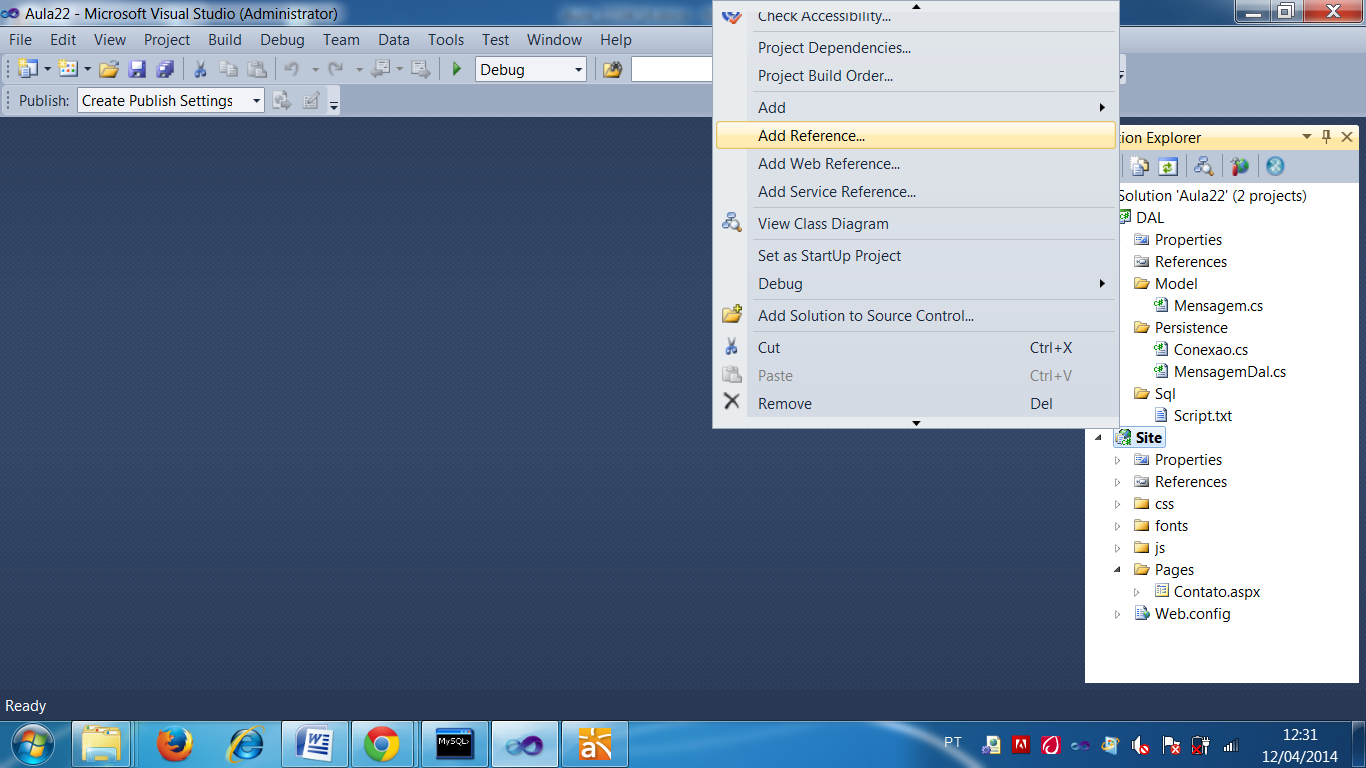




* Incluir as bibliotecas do Spring no Projeto...



Colar em: \Aula22\Site\bin





Escrever o programa para enviar   
a mensagem por Email...

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;

using System.Net;

using System.Net.Mail;

namespace Site.Pages

{

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

{

**//Atributos (encapsulamento implicito)**

**public string Conta { get; set; } //recebera um valor vindo de xml**

**public string Senha { get; set; } //recebera um valor vindo de xml**

**public string Smtp { get; set; } //recebera um valor vindo de xml**

**public int Porta { get; set; } //recebera um valor vindo de xml**

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

{

}

protected void EnviarMensagem(object sender, EventArgs e)

{

try

{

Mensagem m = new Mensagem();

m.Destinatario = txtDestinatario.Text;

m.Assunto = txtAssunto.Text;

m.Texto = txtMensagem.Text;

MensagemDal d = new MensagemDal();

d.Salvar(m);

//Envio do email

MailMessage msg = new MailMessage(Conta,

m.Destinatario); // de -> para

msg.Subject = m.Assunto;

msg.Body = m.Texto;

SmtpClient s = new SmtpClient(Smtp, Porta);

s.EnableSsl = true; //envio de email seguro

s.Credentials = new NetworkCredential(Conta, Senha);

s.Send(msg); //envio do email

lblMensagem.Text = "Mensagem " + m.Assunto

+ ", enviada com sucesso.";

}

catch(Exception ex)

{

lblMensagem.Text = ex.Message;

}

}

}

}

Injeção de dependência...

\Web.config.xml

<?xml version="1.0"?>

<configuration>

<configSections>

<!-- Spring -->

<sectionGroup name="spring">

<section name="context" type="Spring.Context.Support.

WebContextHandler, Spring.Web"/>

<section name="objects" type="Spring.Context.Support.

DefaultSectionHandler, Spring.Core"/>

<section name="parsers" type="Spring.Context.Support.NamespaceParsers

SectionHandler, Spring.Core"/>

</sectionGroup>

</configSections>

<!-- Spring -->

<spring>

<parsers>

</parsers>

<context>

<resource uri="config://spring/objects"/>

</context>

<objects xmlns="http://www.springframework.net"

xmlns:db="http://www.springframework.net/database">

<!-- Injetar na página Contato.aspx o valor dos atributos do CodeBehind -->

**<object type="/Pages/Contato.aspx">**

**<!-- Atributos -->**

**<property name="Conta" value="cotiexemplo@gmail.com"/>**

**<property name="Senha" value="@coticoti@"/>**

**<property name="Smtp" value="smtp.gmail.com"/>**

**<property name="Porta" value="587"/>**

**</object>**

</objects>

</spring>

<system.web>

<httpHandlers>

<add verb="\*" path="\*.aspx" type="Spring.Web.Support.

PageHandlerFactory, Spring.Web"/>

<add verb="\*" path="\*.asmx" type="Spring.Web.Services.WebServiceHandlerFactory,

Spring.Web"/>

</httpHandlers>

<httpModules>

<add name="SpringModule" type="Spring.Context.Support.

WebSupportModule, Spring.Web"/>

</httpModules>

<compilation debug="true" targetFramework="4.0" />

</system.web>

<connectionStrings>

<add

name="mysql"

connectionString="server=localhost;User Id=root;Persist Security

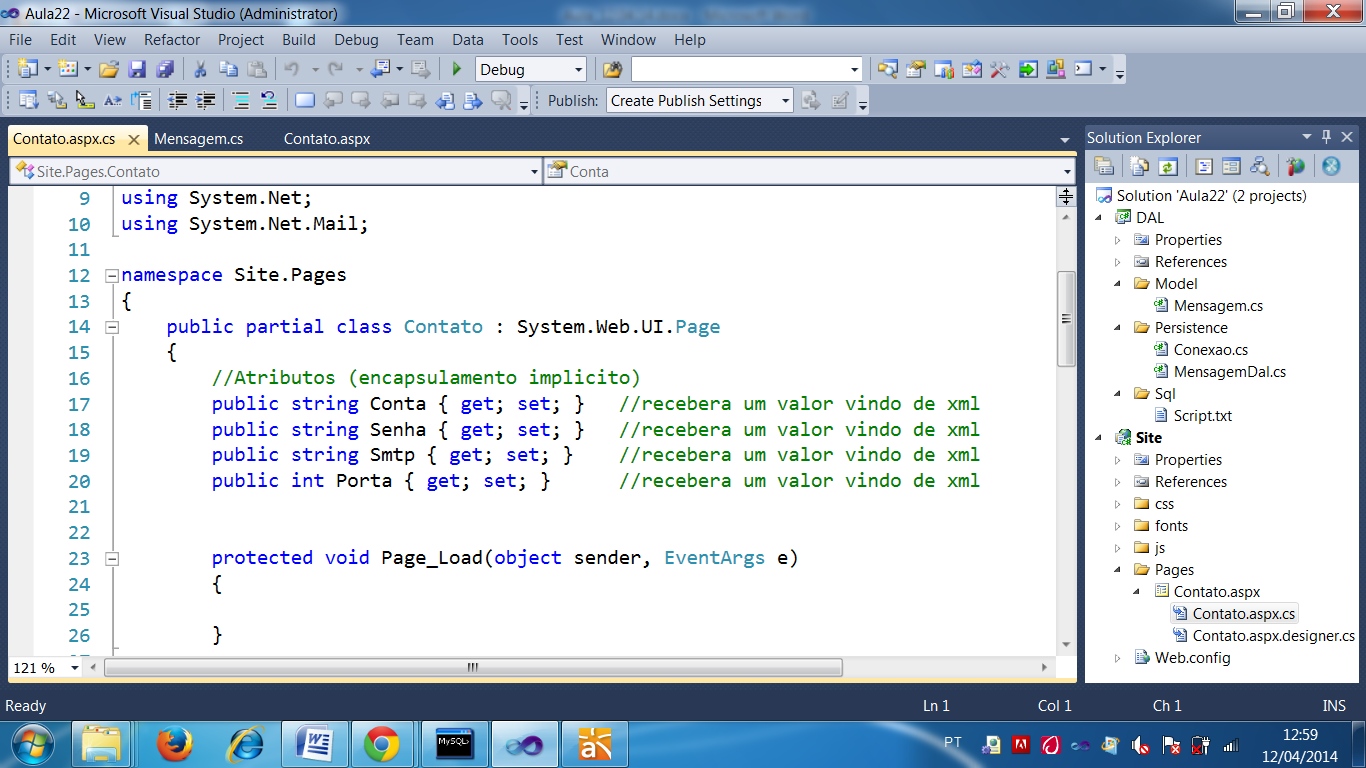
Info=True;database=aula;password=coti"

/>

</connectionStrings>

</configuration>





Executando…

