

reCaptcha

Logado no Gmail, acesse o site

<https://www.google.com/recaptcha/intro/index.html>

Clique em Get reCaptcha

Label

Informe um nome identificador

Exemplo: Luiz Guarino

Domains

Informe o seu domínio

Exemplo:

luizguarino.com

localhost

Muito importante adicionar o domínio localhost.

Anote as chaves Secret e Site

Site key

[sua site key]

Secret Key

[sua secret key]

Mostrando o widget

Pode ser mostrado de duas formas: automaticamente ou renderizado explicitamente.

Abrir a Views/Publico/Logar.cshtml

Adicionar ao final do arquivo o trecho

```
@section scripts{
    <script src="https://www.google.com/recaptcha/api.js" async defer></script>
}
```

Entre a senha e o submit, adicione o trecho

```
<div class="form-group">
    <label class="control-label col-md-2">
        Senha
    </label>
    <div class="col-md-10">
        <input type="password" name="senha" id="senha" class="form-control" />
    </div>
</div>

<div class="g-recaptcha" data-sitekey="SUA_SITE_KEY"></div>

<div class="form-group">
    <div class="col-md-offset-2 col-md-10">
        <input type="submit" value="Entrar" class="btn btn-default" />
    </div>
</div>
```

Adicione uma classe em Repositories

Classe CaptchaResponse.cs – essa classe irá receber a resposta da API do reCaptcha

```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
namespace WebSite1.Repositories
{
    public class CaptchaResponse
    {
        [JsonProperty("success")]
        public bool Success { get; set; }

        [JsonProperty("error-codes")]
        public List<string> ErrorCodes { get; set; }
    }
}
```

Agora devemos tratar a resposta do reCaptcha

A API retorna uma URL parecida com a seguir

https://www.google.com/recaptcha/api/siteverify?secret=your_secret&response=response_string&remoteip=user_ip_address

A resposta vem no formato

```
{
  "success": true|false,
  "error-codes": [...] // optional
}
```

Reescrever a Action (HttpPost) Logar

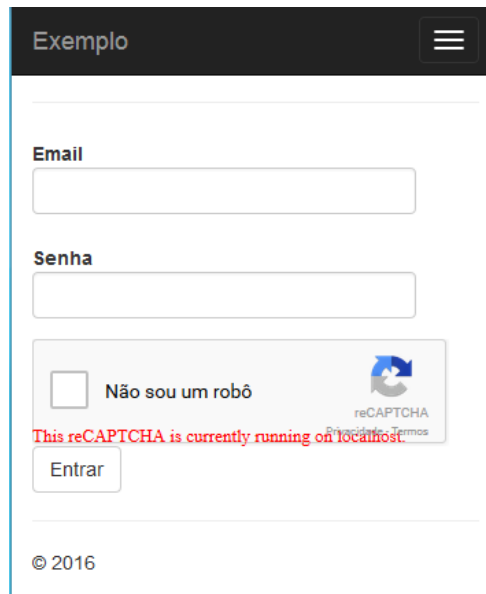
```
[HttpPost]
public ActionResult Logar(string email, string senha)
{
    //O código do e-mail está comentado apenas para não ficar enviando e-mail :)
    //GmailEmailService gmail = new GmailEmailService();
    //EmailMessage msg = new EmailMessage();
    //msg.Body = "Teste";
    //msg.IsHtml = false;
    //msg.Subject = "Assunto de teste";
    //msg.ToEmail = "teste@teste.com";
    //gmail.SendEmailMessage(msg);
    if (Funcoes.AutenticarUsuario(email, senha) == false)
    {
        ViewBag.Error = "Nome de usuário e/ou senha inválida";
        return View();
    }
    else
    {
        var response = Request["g-recaptcha-response"];
        //chave secreta que foi gerada no site
        const string secret = "YOUR KEY VALUE PAIR";
        var client = new WebClient();
        var reply =
            client.DownloadString(

string.Format("https://www.google.com/recaptcha/api/siteverify?secret={0}&response={1}",
secret, response));

        var captchaResponse = JsonConvert.DeserializeObject<CaptchaResponse>(reply);
        //Response false - devemos ver qual a mensagem de erro
        if (!captchaResponse.Success)
        {
            if (captchaResponse.ErrorCodes.Count <= 0) return View();
            var error = captchaResponse.ErrorCodes[0].ToLower();
            switch (error)
            {
                case ("missing-input-secret"):
                    ViewBag.Message = "The secret parameter is missing.";
                    break;
                case ("invalid-input-secret"):
                    ViewBag.Message = "The secret parameter is invalid or
malformed.";
                    break;
                case ("missing-input-response"):
                    ViewBag.Message = "The response parameter is missing.";
                    break;
                case ("invalid-input-response"):
                    ViewBag.Message = "The response parameter is invalid or
malformed.";
                    break;
                default:
                    ViewBag.Message = "Error ocured. Please try again";
                    break;
            }
            return View();
        }
        else
        {
            ViewBag.Message = "Valid";
            return RedirectToAction("Index", "Home");
        }
    }
}
```

Executar a aplicação

Acessar a URL <http://localhost:porta/Publico/Logar>



The screenshot shows a web application interface for a login page. At the top, there is a dark header with the text "Exemplo" on the left and a hamburger menu icon on the right. Below the header, there are two input fields: "Email" and "Senha". Below these fields is a reCAPTCHA widget with the text "Não sou um robô" and a checkbox. To the right of the checkbox is the reCAPTCHA logo and the text "reCAPTCHA". Below the reCAPTCHA widget is a red error message: "This reCAPTCHA is currently running on localhost". Below the reCAPTCHA widget is a button labeled "Entrar". At the bottom of the page, there is a copyright notice: "© 2016".

Referência

<https://developers.google.com/recaptcha/intro>