C# WebDeveloper Avançado

Upload de arquivos com AngularJS e WebApi

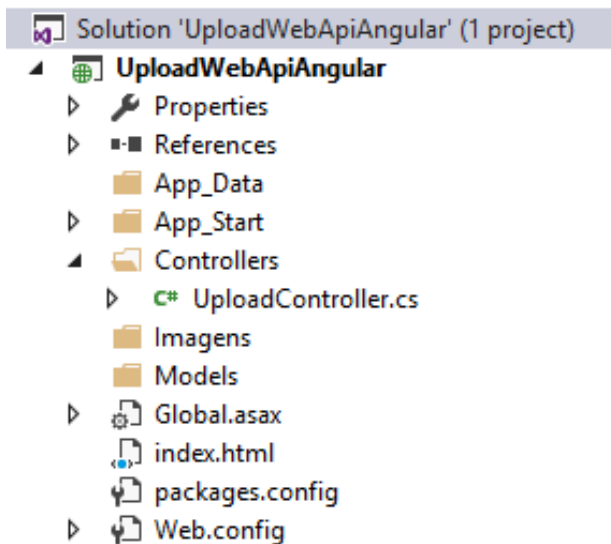Estrutura do projeto:

Página **index.html**

```html
<!DOCTYPE html>

<html>

<head>
    <title>Projeto</title>

    <link rel="stylesheet" type="text/css"
    href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" />
    <script
    src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.4/angular.min.js">
    </script>

    <script type="text/javascript">
        angular.module('projeto', [])
            .directive('ngFiles', ['$parse', function ($parse) {

                function fn_link(scope, element, attrs) {
                    var onChange = $parse(attrs.ngFiles);
                    element.on('change', function (event) {
                        onChange(scope, { $files: event.target.files });
                    });
                };

                return {
                    link: fn_link
                }
            }])
            .controller('uploadController', function ($scope, $http) {

                var dados = new FormData();
                $scope.getTheFiles = function ($files) {
                    angular.forEach($files, function (value, key) {
                        dados.append(key, value);
                    });
                };
```

```javascript
            $scope.uploadFiles = function () {

                var request = {
                    method: 'POST',
                    url: '/services/upload/gravar',
                    data: dados,
                    headers: {
                        'Content-Type': undefined
                    }
                };

                $http(request)
                    .success(function (d) {
                        $scope.mensagem = d;
                    })
                    .error(function (e) {
                        $scope.mensagem = e;
                    });
            }
        });

    </script>
</head>

<body class="container" ng-app="projeto">

    <h3>Exemplo - Upload de Arquivos em AngularJS com WebApi</h3>
    <hr />

    <div class="well col-md-3" ng-controller="uploadController">

        <label>Selecione o Arquivo desejado:</label> <br />
        <input type="file" id="file1" name="file" multiple
            accept=".jpg" class="form-control"
            ng-files="getTheFiles($files)" />

        <br />

        <input type="button" ng-click="uploadFiles()"
            value="Enviar para o Servidor"
            class="btn btn-primary btn-block" />

        <br /><br />

        {{mensagem}}

    </div>
</body>
</html>
```

UploadController.vb

```vb
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Net.Http.Headers;
using System.Threading.Tasks;
```

```csharp
using System.Web;
using System.Web.Http;

namespace UploadWebApiAngular.Controllers
{
    [RoutePrefix("services/upload")]
    public class UploadController : ApiController
    {
        [HttpPost]
        [Route("gravar")]
        public async Task<HttpResponseMessage> Post()
        {
            if (!Request.Content.IsMimeMultipartContent())
            {
                throw new HttpResponseException(
                    HttpStatusCode.UnsupportedMediaType);
            }

            try
            {
                var pasta = HttpContext.Current.Server.MapPath("/Imagens/");
                var provider = new CustomMultipartFormDataStreamProvider
                            (pasta, ".jpg");

                await Request.Content.ReadAsMultipartAsync(provider);

                return Request.CreateResponse(HttpStatusCode.OK,
                    "Upload realizado com sucesso.");
            }
            catch (Exception ex)
            {
                return Request.CreateResponse(
                    HttpStatusCode.Forbidden, ex.Message);
            }
        }
    }


    //classe para gerar o nome do arquivo...
    class CustomMultipartFormDataStreamProvider
            : MultipartFormDataStreamProvider
    {
        private string _extensao;

        public CustomMultipartFormDataStreamProvider
            (string path, string extensao)
            : base(path)
        {
            _extensao = extensao;
        }

        public override string GetLocalFileName(HttpContentHeaders headers)
        {
            return Guid.NewGuid().ToString() + _extensao;
        }
    }
}
```
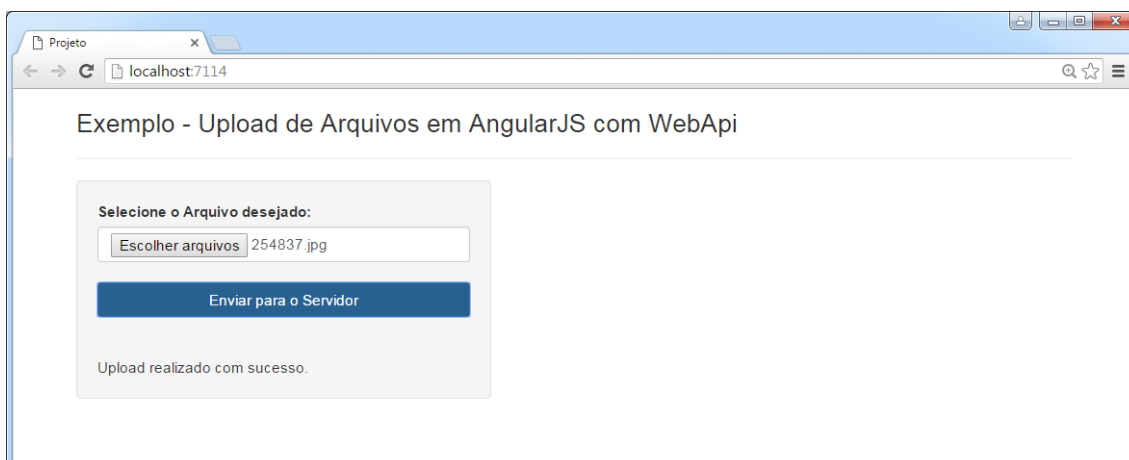
Testando:

Na pasta /Imagens