

TESTE DE ASPNET MVC (ÉPOCA NORMAL)

Regras para a realização do trabalho:

- O trabalho é **individual**.
- **Duração:** 2h.
- O autor do programa deve estar identificado nos comentários que precedem cada documento na zona do cabeçalho, contendo o nome completo e o número de aluno.
- **Todas as eventuais dúvidas na interpretação do enunciado deverão ser resolvidas por si próprio.** Caso detecte ambiguidades, omissões e/ou incorrecções, resolva-as em função da sua melhor interpretação. Caso considere necessário, explique a sua interpretação em linhas de comentário no programa. Tal será considerado aquando da avaliação.
- Os computadores pessoais **devem estar desligados da rede** local por cabo ou *wireless*.
- **A fraude é punida com zero na nota prática deste trabalho.**
- Os ficheiros resultantes (*.aspx, *.cs, *.js, base dados com o nome gsa.mdf e todos os outros ficheiros) do trabalho individual devem ser entregues ao docente no final da aula, comprimidos num ficheiro em formato zip com a seguinte sintaxe: `daw_aspnet_en_numAluno.zip`.
- Uma vez concluído o trabalho, o aluno deverá informar disso o Docente.

1) Implementação em ASP.NET MVC parcial do sítio web de gestão de avaliações do IPBeja

O sistema de gestão de avaliações (SGA) do IPBeja permite a docentes, alunos e outros intervenientes, gerir os dados das respetivas avaliações ocorridas nas UC, durante os dois semestres, do ano letivo.

- 1.1) [6 Valor] Criar um novo projecto ASP.NET MVC4|5|6 com o nome "GSA"
- Deve utilizar o motor de vista "Razor", a linguagem C#, e não necessita de criar nenhum teste unitário. Este projeto de desenvolvimento é baseado na abordagem "**DataBase First**". Pretende-se desenvolver um sistema que gere os dados de alunos, unidades curriculares (UC), e as classificações dos alunos.

A **base de dados** "gsa" deve ser criada no interior do próprio projeto Visual Studio, acessível na pasta "App_Data".

São necessárias implementar na base de dados as tabelas seguintes:

- a) **Aluno** (id, nome), **UC** (id, nome), **Classificacao** (id, id_aluno, id_uc, id_epoca,data, nota, obs), e **Epoca** (id, nome).

Os tipos dos campos devem ser os apropriados para os atributos em causa.

Implemente as classes do "model" com base nos procedimentos da abordagem "DataBase First".

1.2) [2 Valor] Criação de uma área

Crie a área “Alunos” que suporta a gestão dos resultados académicos dos alunos.

1.3) [3 Valor] Criação dos Controladores e Vistas das entidades

Criação dos mecanismos CRUD das entidades “*Aluno*”, “*UC*”, “*Classificacao*” e “*Epoca*”. (Antes de avançar compile a aplicação), dentro da área “Alunos”.

Nas respetivas interfaces da aplicação, devem ser criados de 3 registos de alunos, 3 registos de UC, 3 registos de épocas de avaliação (“Normal”, “Recurso”, e “Finalistas”), e 4 registos de classificações, que contenham pelo menos dois registos de UC e dois de alunos.

1.4) [1 Valor] Criação de hiper-ligações no menu principal

No menu principal devem ser criadas 3 hiper-ligações: “Aluno”, “UC”, “Classificações” e “Médias”. As três primeiras devem apontar para o respetivo controlador e para o método de ação “Index”. A última deve apontar o controlador e método de ação indicados na próxima questão (1.5).

1.5) [8 Valor] Média das classificações obtidas pelos alunos numa determinada UC

Adicionar um caso de uso que permita obter a média das classificações obtidas pelos alunos numa determinada UC, por época de avaliação, usando Web Services SOAP. Para tal devem ser desenvolvidos os seguintes elementos:

- b) Crie o projeto (WSAverage) que disponibiliza um Web Service SOAP. Implemente o método Average (int[] grades) que recebe um array com as classificações de uma dada UC, e devolve um float com a respetiva média. [3 Valor]
- c) Implemente no controlador “Classificacao” o(s) método(s) “Average(...)” que invocam as respetivas vistas, bem como o método web Average (int[] grades) enunciado no ponto anterior. [3 Valor]
- d) Crie a vista “AverageInput”, associada ao controlador “Classificacao”, que contém um formulário com os controlos necessários para inserir o nome da UC e a época de avaliação para a qual se pretende obter a média de classificações. [1 Valor]
- e) Crie a vista “AverageResult”, associada ao controlador “Classificacao”, que exibe o resultado da média de classificações de uma dada UC solicitada anteriormente. [1 Valor]

No fim da resolução deste exame DEVE ENTREGAR:

- Todo o código do sítio web desenvolvido e um script da base dados com o nome “gsa.sql”;
- Um ficheiro com o nome “readme.txt” que deve ter a **identificação do aluno** (nº / nome do autor / email), e que informe alguns aspectos relevantes no desenvolvimento e na utilização do trabalho e as **versões usadas do Visual Studio e da base dados SQL**.

Bom Trabalho!