

Componente Teórica

60% da nota da prova escrita. Nota mínima de 8 valores

Para a pergunta 1, respostas erradas descontam metade da cotação.

1. Para as seguintes afirmações indique se são verdadeiras ou falsas (14 valores):
 - a. Um padrão de software é como uma receita de um bolo, segue-se os passos indicados fazendo pequenas adaptações ao gosto de cada um e a aplicação em desenvolvimento fica correcta.
 - b. Uma arquitectura de aplicação identifica principalmente o modelo de dados e a infra-estrutura física do sistema.
 - c. O padrão *Table Module* é adequado para aplicações com lógica de negócio pouco complexa
 - d. Numa aplicação construída tendo por base o modelo 3 camadas, a camada de acesso a dados deve esconder pormenores de implementação dos mecanismos de persistência da camada de lógica de negócio.
 - e. Na descrição de um padrão a secção de forças e contra-forças é documentação que nos ajuda a perceber se o padrão pode ou não ser aplicado ao nosso caso.
 - f. O padrão *Model-View-Controller* é um padrão de lógica de apresentação que separa a visualização dos dados da sua representação.
 - g. Usando o padrão *Row Data Gateway* existirá uma instância da classe para cada registo de uma tabela da BD. As classes que seguem este padrão possuem atributos correspondentes aos campos da tabela associada e métodos de manipulação de um registo na BD (insert, update, delete).
 - h. O padrão *Table Module* não deve ser usado em plataformas que não suportem o conceito de *Record set*.
 - i. Usando um *Domain Model* como padrão de lógica de negócio impede o uso de XML na aplicação.
 - j. As entidades de negócio devem ser sempre representadas como Data Sets em .NET e como *custom classes* em java. XML só se deve usar em web services.
 - k. O ASP, JSP ou PHP permitem a implementação do padrão Transform View mas de forma pouco natural já que a página de *template* não pode ter etiquetas XML.
 - l. Se a lógica de negócio a implementar numa aplicação for simples devemos sempre utilizar o padrão *Table Module*.
 - m. Entidades de negócio representadas em XML facilitam a sua utilização para transferência de dados entre aplicações.
 - n. O padrão *Lazy Load* deve ser usado sempre que se usa um *Domain Model* por forma a garantir que só uma instância da entidade de negócio está em memória.

2. Descreva em dois ou três parágrafos como é que uma aplicação de controlo e monitorização de sensores (ex, temperatura e nível de CO₂) poderia tirar partido do padrão *Observer* (6 valores).

Componente Prática

40% da nota da prova escrita. Nota mínima de 8 valores

3. Para o seguinte problema:

- Elabore esboços de ecrãs a utilizar para cada use case (2 valores)
- Identifique as operações de negócio necessárias para implementar cada um desses use cases; indique o nome, parâmetros com tipo e tipo de retorno para cada operação. No caso de usar DataSets indique qual a estrutura de tabelas e registos em cada DataSet (2 valores).
- Indique qual o padrão de lógica de negócio e de acesso a dados que utilizaria para implementar o sistema. Justifique a sua escolha (2 valores).
- Elabore o diagrama de sequência para implementar o use case “preencher folha de horas” (4 valores).
- Elabore o(s) diagrama(s) de classes indicando as operações (com assinatura completa) e atributos (se existentes) nas suas classes (5 valores)
- Implemente o código (apenas da lógica de negócio e de acesso a dados) correspondente ao diagrama de sequência da alínea d) (5 valores)

A sua empresa decidiu fazer um pequeno programa web para os funcionários preencherem a folha de horas semanal (cada funcionário pode indicar as várias actividades em que trabalhou em cada projecto durante a semana e quantas horas despendeu em cada uma). As actividades estão categorizadas previamente (ex., implementação de nova funcionalidade), bem como a lista de projectos existentes. Cada director de unidade, tem que aprovar a folha de horas dos seus subordinados. A informação constante das folhas de horas será utilizada posteriormente para debitar as contas dos clientes ou projectos associados.

