



LABORATÓRIO DE DESENVOLVIMENTO DE SOFTWARE | 21179

Período de Realização

Decorre dia 16 de Junho de 2020

Data de Limite de Entrega

16 de Junho de 2020, até às 13:00 horas de Portugal Continental

Competências

O aluno deve demonstrar capacidades para:

- analisar um problema de desenvolvimento e estruturá-lo numa arquitetura modular;
- definir e implementar a arquitetura modular com abordagens de código que revelem as qualidades desejadas para um software que seja uma peça de engenharia: simplicidade, independência para fomentar a reutilização, acoplamento fraco com coesão forte, entre outras;
- tratar erros de forma independente do componente onde ocorrem e respeitando a arquitetura modular.

Trabalho a desenvolver

1.ª Parte [4 Valores]

O seguinte bloco de código é um exemplo de utilização da API Tiny JSON, que proporciona serviços de descodificação de strings em formato JSON (Adaptado de <https://github.com/zanders3/json/>)

```
1 public static class JsonTest
2 {
3     public static void Main(string[] args)
4     {
5         //Definir os valores a guardar
6         List<int> values = new List<int> { 1, 2, 3, 4, 5, 6 };
7
8         //Converter os valores em string em formato JSON
9         string json = values.ToJson();
10
11        //Guardar os valores em formato JSON num ficheiro.
12        File.WriteAllText("test.json", json);
13
14        //Ler esse ficheiro
15        string fileJson = File.ReadAllText("test.json");
16
17        //Extrair os valores da string em formato JSON
18        List<int> fileValues = fileJson.FromJson<List<int>>();
19    }
20 }
```

1a) Indique as linhas do código de referência onde há operações de input [0.5 valores]

1b) Indique as linhas do código de referência onde há operações de output. [0.5 valores]

Caso tenha tido dúvidas de interpretação do código que lhe afetaram a resposta, exponha-as no corpo da resposta.

2. Suponha que queria reescrever o código de referência segundo o estilo arquitetônico MVC. Indique como distribuiria pelos componentes as tarefas descritas nos comentários desse código. (Por ex., se considerar que a tarefa "//Converter os valores em string em formato JSON" (linha 8) corresponde ao Controller, escreva "C: 8", se acha que corresponde ao Model, escreva "M: 8", para indicar que corresponde à View, escreva "V: 8".)

2a) Segundo a abordagem de Krasner & Pope (1988) [0.3 valores]

2b) Segundo a abordagem de Curry & Grace (2008) [0.3 valores]

2c) Explique as dúvidas ou dilemas com que se debateu para responder às alíneas a) e b) e justifique as principais opções que tomou ao dar as suas respostas. [0.4 valores]

3. Em projetos de desenvolvimento de software distinguimos três tipos de defeitos: faltas, erros e falhas. Atendendo ao código de referência, imagine e exponha um defeito que possa ser constatado quando se executa o programa, um defeito que consista no estado incorreto do programa durante a execução, antes de se constatar a consequência desse estado, e um defeito que advenha do atual código-fonte. Para cada defeito, indique se é uma falta, um erro ou uma falha. [1 valor]

4. Dê exemplos de testes do tipo whitebox e do tipo blackbox para testar o código de referência. [1 valor]

2.ª Parte [8 Valores]

5. Em duas páginas, escreva um texto reflexivo onde incorpore os seguintes temas:

5a) Descrição e discussão do projeto desenvolvido ao longo dos e-fólios, identificando pontos de relação com os principais aspetos teóricos estudados ao longo do semestre. [3 valores]

NOTA: Para cada aspeto teórico, indique-o (por ex., as qualidades do desenvolvimento de software, os tipos de testes, as arquiteturas de software, etc.) e seja explícito na forma como relaciona esse aspeto teórico ao que fez efetivamente no seu projeto do semestre.

5b) Análise das abordagens por si seguidas relativamente à utilização de exceções personalizadas para gestão de erros, e aspetos que mudaria, em retrospectiva. [3 valores]

NOTA: Tenha o cuidado de nessa análise referir aspetos da distribuição de responsabilidades do lançamento e tratamento de exceções entre os componentes do estilo arquitetónico MVC e problemas concretos com que se deparou no âmbito da API com que trabalhou.

5c) Análise das suas anteriores práticas de trabalho em contraste com o paradigma MVC sobre o qual insistimos ao longo do semestre. [2 valores]

NOTA: Siga o seu processo de aprendizagem em relação a este paradigma específico e reflita sobre experiências passadas em que a abordagem tida tenha sido outra, e o processo/resultado pudesse beneficiar de uma abordagem diferente.

Critérios de avaliação e cotação

As cotações estão discriminadas ao longo do enunciado. Na avaliação do trabalho será tida em consideração a aplicação dos conceitos estudados ao longo do semestre, a clareza da linguagem, a correção dos processos e a capacidade de auto-reflexão.

Total: 12 pontos = 12 valores

Normas a respeitar

Deve compilar todo o seu E-fólio num PDF, não esquecendo a criação de um cabeçalho com a sua identificação.

Coloque todo o código necessário como texto no corpo do PDF.

Todas as páginas do documento devem ser numeradas.

Nomeie o ficheiro com o seu número de estudante, seguido da identificação do E-fólio, segundo o exemplo apresentado:
000000efolioA.

Deve carregar o referido ficheiro para a plataforma no dispositivo E-fólio A até à data e hora limite de entrega. Evite a entrega próximo da hora limite para se precaver contra eventuais problemas.

O ficheiro a enviar não deve exceder 8 MB.

Votos de bom trabalho!

Pedro Duarte Pestana