

Programação Orientada a Objetos (POO) - LESI|LESIN

1º Trabalho Prático

Luís G. Ferreira* & Eduardo Peixoto†

EST-IPCA
Barcelos

17 de outubro de 2023

Resumo

Este trabalho da Unidade Curricular (UC) de Programação Orientada a Objetos (POO) foca a análise de problemas reais simples e a aplicação do Paradigma Orientado a Objetos na implementação de possíveis soluções.

*Email: lufer@ipca.pt; Gab: 5

†Email: epeixoto@ipca.pt; Gab: 10

1 Motivação

Pretende-se que sejam desenvolvidas soluções em C# para problemas reais de complexidade moderada. Serão identificadas classes, definidas estruturas de dados e implementados os principais processos que permitam suportar essas soluções. Pretende-se ainda contribuir para a boa redação de relatórios.

2 Objetivos

- Consolidar conceitos basilares do Paradigma Orientado a Objectos;
- Analisar problemas reais;
- Desenvolver capacidades de programação em C#;
- Potenciar a experiência no desenvolvimento de software;
- Assimilar o conteúdo da Unidade Curricular.

3 Regras do jogo

- O trabalho deverá ser feito individualmente ou em grupo de duas pessoas;
- O trabalho deverá ser entregue em duas fases. Em cada fase o trabalho realizado deverá ser submetido um ficheiro compactado, na plataforma *moodle*, cujo nome deverá conter o número do aluno e número da fase.
Exemplo para a entrega da Fase 1 do aluno 1234:

– *trabalhoPOO_1234_Fase1.zip*

As datas de cada fase e as metas a atingir em cada uma, são:

- **Fase 1 (15-11-2023)**
 - Classes identificadas
 - Implementação essencial das classes
 - Estruturas de dados a utilizar
 - Relatório do trabalho desenvolvido até à data
 - Cumprimento dos prazos
- **Fase 2 (19-12-2023)**
 - Implementação final das classes e serviços
 - Aplicação demonstradora dos serviços implementados
 - Relatório final do trabalho realizado
 - Cumprimento dos prazos

4 Problemas a explorar

O tema de trabalho que o aluno pretenda explorar deve ser indicado na plataforma Onedrive. O aluno pode propor o seu próprio tema ou escolher uma das seguintes sugestões:

- (i) Gestão de obra de construção civil: sistema que permita controlar e gerir os custos de uma determinada obra.
Keywords: obras, materiais, armazéns, stocks, viaturas, serviços, mão de obra (contratada e subcontratada), orçamentos, documentos.
- (ii) Gestão de condomínios: sistema que permita fazer a gestão de condomínios de uma empresa.
Keywords: condomínios, condóminos (proprietários, inquilinos), permissões, quotas, despesas, gestão de pagamentos, agendamento de reuniões, atas, documentos
- (iii) Gestão de rendas/imóveis: sistema que permita a uma empresa/proprietário gerir os seus imóveis e as respetivas rendas mensais;
Keywords: senhorios, inquilinos, imóveis (apartamento, vivenda, terreno), recibos, contratos, despesas, documentos, distritos (código e descrição nas finanças), concelhos (código e descrição nas finanças), freguesias (código e descrição nas finanças), estados.
- (iv) Gestão de alojamentos turísticos: sistema que permita a gestão de alojamentos turísticos.
Keywords: registos, consultas, reservas, check-in, clientes, alojamentos.
- (v) Helpdesk: Sistema de gestão e apoio a assistências por telefone (call center).
Keywords: assistência, tipo de assistência, estado da assistência, operador, cliente, produtos, problemas conhecidos, tutoriais de resolução de problema, documentos, avaliação da assistência (e.g. se o problema foi resolvido e avaliação de 1 a 10).
- (vi) Comércio eletrónico: sistema que permita a gestão de uma loja online.
Keywords: produtos, categorias, garantias, stocks, clientes, campanhas, vendas, marcas.
- (vii) Gestão de jardim zoológico: sistema que permita a gestão das tarefas de um jardim zoológico.
Keywords: animais, informações, assistência veterinária, alimentação, calendários, tipos de comida, limpeza de jaulas, espetáculos, bilhetes.
- (viii) Gestão de hospital: sistema que permita gerir um hospital.
Keywords: staff, categorias, especialidades, consultas, camas, pacientes, diagnósticos, exames, custos.

5 Critérios de Avaliação

5.1 Qualidade do Relatório

- Estrutura, Clareza e Expressividade
- Capacidade de síntese e correção da escrita

5.2 Qualidade da solução desenvolvida

- Qualidade do código produzido: estrutura da solução, nome de ficheiros, uso de bibliotecas (DLL), Norma CLS.
- Organização e Implementação das Classes com recurso a Interfaces, Herança, classes abstractas, outras.
- Qualidade dos algoritmos aplicados
- Estruturas de dados exploradas
- Tratamento adequado de exceções
- Persistência de dados com recurso a ficheiros
- Programação por camadas (NTier, MVC, outras)
- Exploração de outras valências: Lambda Functions; LINQ; Windows Forms; WPF

6 Avaliação

- Fase 1 - 25% da nota final do trabalho (TP.C - Componente código)
- Fase 2 - 75% da nota final do trabalho (TP.C - Componente código)
- O trabalho será defendido individualmente - consultar metodologia e avaliação.

Bom trabalho.