

Programação com Objetos

Projeto de Grupo 2021/2022

Pretende-se desenvolver um programa para gerir o processo de acesso aos parques de uma universidade. Cada parque tem um nome, o número de lugares, e um tipo. Existem três tipos de parques/acessos:

- a. Parques de acesso livre (€ 22,50)
- b. Parques de acesso condicionado sem lugar assegurado (€ 50)
- c. Parques com lugar assegurado (€ 100).

Os parques livres e os de acesso condicionado não têm lugar assegurado e podem ser frequentados por utilizadores que tenham adquirido qualquer tipo de acesso.

Os parques de acesso condicionado só podem ser frequentados por utilizadores que tenham solicitado o acesso a parques deste tipo.

Os parques com lugar assegurado garantem sempre lugar pelo que será necessário controlar, por parque, o número de lugares atribuídos. Quer isto dizer que, ao solicitar o acesso a um parque deste tipo, o utilizador terá obrigatoriamente que indicar qual o parque (de lugar assegurado) a que querará aceder.

A aplicação tem um administrador que tem como papel principal criar utilizadores do tipo Segurança, registar parques e tratar dos pedidos de acesso. O utilizador do tipo Segurança tem como papel principal o controlo das entradas (data, hora, viatura, parque) e saídas (data, hora, viatura) de viaturas nas instalações da Universidade. Um utilizador/utente pode ter até três viaturas (Matrícula, Marca, Modelo, Combustível). No entanto, se uma viatura já estiver no parque, mais nenhuma pode dar entrada.

Funcionalidades

A aplicação informática deverá ser composta por duas partes: a parte de back-end que suporta o funcionamento do sistema, e a parte de front-end que suporta a interface que permite a interação com o utilizador.

O back-end e o front-end deverão estar implementados em pacotes distintos, ou seja, todas as classes que digam respeito à interface deverão estar arrumadas num pacote e as classes respeitantes ao negócio, noutro.

A entrada e saída de dados deve ser realizada recorrendo a uma interface gráfica, assente numa ou mais janelas. O desenvolvimento da interface gráfica poderá ser baseado no uso das classes existentes nos packages AWT/Swing do JAVA. Pretende-se ainda que a aplicação tenha capacidade de persistência, ou seja, que tenha capacidade de guardar em ficheiro o estado do sistema num determinado momento e recuperá-lo quando pretendido. Refere-se que **não é permitida a utilização de bases de dados**. Ainda em relação à entrada e saída de dados importa referir que se pretende que o sistema tenha níveis de robustez adequados, ou seja, que seja capaz de se comportar

adequadamente em situações de erro (nomeadamente aquando da introdução de dados por parte dos utilizadores).

A seguir estão listadas as funcionalidades que deverão ser implementadas por tipo de ator/utilizador:

Utente

1. Registrar-se como utilizador
2. Autenticar
3. Ver/editar os dados de utilizador e viaturas
4. Solicitar a compra de um acesso ao parque (acesso livre, acesso condicionado ou lugar assegurado).
5. Ver o estado do seu pedido de acesso

Administrador

1. Autenticar-se como administrador
2. Adicionar/apagar utilizadores do tipo Segurança
3. Registrar parques
4. Listar os pedidos de acesso por estado (pendente, atribuído, recusado)
5. Listar os pedidos de acessos pendentes e conceder ou recusar os mesmos. Ao conceder o acesso a um parque com lugar assegurado, deverá ser automaticamente atribuído o número do lugar.
6. Listar os pedidos de acesso por estado (pendente, atribuído, recusado)
7. Calcular o valor total angariado de forma global e por tipo de parque
8. Mostrar a percentagem de viaturas por tipo de combustível

Segurança

1. Autenticar
2. Registrar a entrada de uma viatura
3. Registrar a saída de uma viatura
4. Saber as viaturas que estão atualmente nas instalações
5. Saber as viaturas que estiverem numa determinada data nas instalações
6. Saber as viaturas que estiverem numa determinada data e num determinado parque
7. Saber se um dado parque tem lugares disponíveis.
8. Verificar se uma data matrícula tem acesso a um determinado parque

Valorização

A aplicação dos conceitos e dos princípios fundamentais subjacentes ao paradigma de programação orientada aos objetos será amplamente valorizada.

A classificação dos trabalhos terá ainda em consideração aspetos como:

- Qualidade da execução do programa (com ou sem erros, validações, testes, ...)
- Qualidade da satisfação dos requisitos
- Facilidade de utilização do sistema
- Adequação da conceção e organização das classes
- Qualidade do código fonte apresentado
 - Correção (avaliação se o código está coerente com a especificação do enunciado e se não tem erros)
 - Design (avalia a forma como o código está escrito (eficiente, elegante, reutilizável))
 - Estilo (avalia a legibilidade do código (comentado, indentado, nomes de variáveis e funções adequadas,...))
- Desempenho do grupo no decorrer da apresentação

Datas e Formalidades de Entrega

O trabalho deve ser realizado em **grupos de 2 a 3 estudantes inscritos no mesmo turno PL**. O projeto, em formato zip, deve ser submetido no BlackBoard até **14 de janeiro de 2022**. Juntamente com o projeto deve ainda ser entregue um relatório básico (em formato pdf) incluindo uma tabela contendo todas as funcionalidades implementadas e não implementadas, e a documentação gerada (*javadoc*) relativa às classes criadas.