



Trabalho 01

Problema:

Implementar um sistema orientado a objetos em Java para permitir o controle de acesso de funcionários a um ambiente.

Escopo do desenvolvimento:

Um sistema é responsável por controlar o acesso dos funcionários à porta do setor financeiro de uma empresa.

O sistema deve permitir o cadastro de funcionários, contendo no mínimo: número de matrícula, nome, data de nascimento, telefone, salário e cargo.

Deve ser possível cadastrar os cargos dos funcionários, com código e nome, e também indicar para cada cargo: se permite acesso à porta do financeiro, quais horários permitem acesso ou se é um cargo gerencial (que permite acesso em qualquer horário). Se forem cargos sem acesso ou gerenciais, não se deve pedir o cadastro dos horários.

Ao tentar acessar a porta do setor financeiro, os funcionários são identificados pelo seu número de matrícula, que digitam em um terminal na entrada da porta. (Para viabilizar os testes, o sistema deve permitir a informação opcional do horário de acesso).

Para permitir a entrada dos funcionários, o sistema deve verificar primeiramente se a matrícula do funcionário é válida (se existe algum funcionário cadastrado com aquele número de matrícula) e, em caso positivo, verificar qual o cargo do funcionário. Alguns cargos de funcionários nunca podem entrar no financeiro, outros cargos permitem somente entrar na porta no horário comercial (08:00-12:00 e 14:00-18:00), outros cargos ainda devem poder entrar em horários especiais. Os funcionários que são gerentes podem entrar pela porta do setor financeiro em qualquer horário.

Se um funcionário tentar entrar na porta, digitar a sua matrícula e tiver seu acesso negado, deve ser emitida uma mensagem com o motivo da negação de acesso (matrícula não existe, não possui acesso, horário não permitido ou acesso bloqueado) e deve ser armazenado um registro da data e hora da tentativa de entrada negada. Deve ser registrado: data e hora da tentativa de acesso, matrícula do funcionário e motivo de negação de acesso.

Também deve ser possível pesquisar pelos registros de acesso negado. Neste caso o sistema deve permitir emitir um relatório de acesso negado, onde seja possível filtrar por: motivo de negação ou matrícula do funcionário.

Considere algumas regras:

1. Não podem existir dois funcionários cadastrados com o mesmo número de matrícula. Ao se tentar cadastrar um funcionário com número de matrícula já existente, o sistema deve impedir



e retornar mensagem de erro informando que já existe um funcionário com o mesmo número de matrícula.

2. Se um mesmo funcionário (mesmo número de matrícula) tentar acesso mais de três vezes em horários não permitidos, deve ser bloqueado o acesso daquele funcionário, emitindo uma mensagem na tela e registrando um evento de negação de acesso do tipo “acesso bloqueado”.

Restrições de escopo:

O sistema contempla somente algumas das funcionalidades de um sistema de controle de acesso, não abordando funcionalidades avançadas, de forma a facilitar a sua implementação.

Entrega:

Deve ser postado um único arquivo ZIP por equipe no Moodle até o dia **16/10/2017 às 18:00hs**, contendo:

- Código fonte completo do sistema orientado a objetos em Java.
- Figuras (JPG) contendo os diagramas de classes da solução implementada seguindo a notação UML 2. Deve ser elaborado um diagrama de classes por funcionalidade do sistema, englobando: controladores, classes de apresentação e as entidades.



Critérios de Avaliação:

1. Qualidade da solução do problema apresentado atendendo ao escopo do desenvolvimento (5,0 pontos), observando os seguintes cenários:
 - 1.1 Cadastrar funcionários (0,5 pontos).
 - 1.2 Cadastrar cargos (1,0 pontos).
 - 1.3 Realizar validação de acesso (1,5 pontos).
 - 1.4 Registrar evento de negação de acesso (1,0 ponto).
 - 1.5 Gerar relatório/pesquisa de negações de acesso (1,0 ponto).
2. Qualidade da notação UML no diagrama de classes e sua consistência com o código (1,0 ponto).
3. Documentação das operações das classes (exceto gets e sets) (0,5 pontos).
4. Utilização correta de interface e enum (0,5 pontos).
5. Utilização correta de dependência, associação, agregação e composição (1,0 ponto).
6. Utilização, de maneira adequada, de herança e polimorfismo (1,0 ponto).
7. Utilização, de maneira adequada, de tratamento de exceções (1,0 ponto).

Defesas dos Trabalhos:

A apresentação será realizada em laboratório uma equipe de cada vez, por sorteio, onde o professor irá questionar cada um dos membros da equipe quanto à implementação realizada e aos conceitos aplicados nos trabalhos. A nota de cada membro da equipe será individual, dependendo da sua participação individual no desenvolvimento do trabalho e na sua capacidade de explicar o código desenvolvido. Qualquer membro da equipe que não estiver presente na apresentação receberá 50% da média das notas da equipe no trabalho, desde que os demais membros da equipe confirmem que tenha efetivamente participado do desenvolvimento do trabalho.