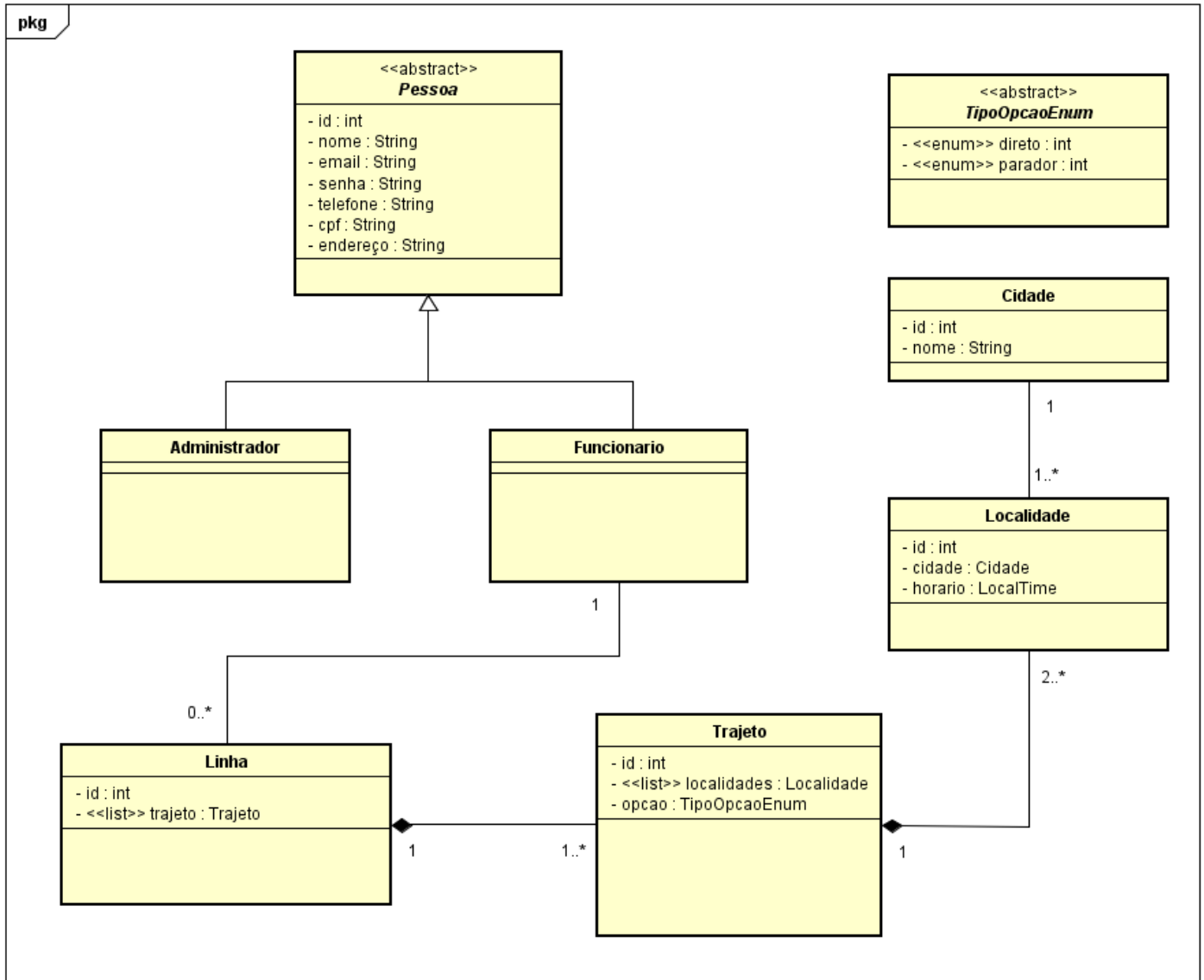


Tarefa 01

Aluno: Amarildo Jorge de Jesus Junior

Diagrama de Classe



Restrições

Pessoa

	Nome	Email	Senha	Telefone	CPF	Endereço
Aceita valor nulo?	NÃO	NÃO	NÃO	SIM	NÃO	SIM
Será permitido alteração futura?	SIM	SIM	SIM	SIM	SIM	SIM
Deverá ser único?	NÃO	SIM	SIM	SIM	SIM	NÃO
Padrão (regex)?	NÃO	NÃO	NÃO	NÃO	SIM	NÃO
Quantidade mínima e máxima de caracteres	3-100	10-100	10-100	10-100	12	10-100
Aceita valores espaço em branco ou não?	SIM	NÃO	NÃO	NÃO	NÃO	SIM

Pessoa/Funcionario

Objeto Funcionario deve estar relacionamento com o objeto Linha.	SIM
Número mínimo ou máximo de relacionamentos.	1-0..*
Objeto A deve estar relacionado com objeto B que possua em seu atributo X o valor Y.	Somente o ID

TipoOpcaoEnum

	direto	parador
Aceita valor nulo?	NÃO	NÃO
Será permitido alteração futura?	NÃO	NÃO
Deverá ser único?	SIM	SIM
Padrão (regex)?	NÃO	NÃO
Aceita valores negativos, positivos ou indiferente?	Positivos	Positivos
Faixa de valores permitidos	1-100	1-100
Valores multipols de X.	-	-

Linha

	Trajeto
Aceita valor nulo?	NÃO
Será permitido alteração futura?	SIM
Deverá ser único?	SIM
Padrão (regex)?	NÃO
Quantidade mínima e máxima de caracteres	10-100
Aceita valores espaço em branco ou não?	SIM
(Coleção) Mínimo ou máximo de elementos.	0 - 50
Objeto Linha deve estar relacionamento com o objeto Trajeto.	SIM
Número mínimo ou máximo de relacionamentos.	1-1..*
Objeto A deve estar relacionado com objeto B que possua em seu atributo X o valor Y.	SIM

Trajeto

	Localidade
Aceita valor nulo?	NÃO
Será permitido alteração futura?	SIM
Deverá ser único?	SIM
Padrão (regex)?	NÃO
Quantidade mínima e máxima de caracteres	10-100
Aceita valores espaço em branco ou não?	SIM
(Coleção) Mínimo ou máximo de elementos.	0 - 50
Objeto Trajeto deve estar relacionamento com o objeto Localidade.	SIM
Número mínimo ou máximo de relacionamentos.	1-2..*
Objeto A deve estar relacionado com objeto B que possua em seu atributo X o valor Y.	SIM

Localidade

	Cidade	Horario
Aceita valor nulo?	NÃO	NÃO
Será permitido alteração futura?	SIM	SIM
Deverá ser único?	SIM	NÃO
Padrão (regex)?	NÃO	NÃO
Quantidade mínima e máxima de caracteres	10-100	4
Aceita valores espaço em branco ou não?	SIM	NÃO
(Coleção) Mínimo ou máximo de elementos.	0 - 50	-
Objeto localidade deve estar relacionamento com o objeto Cidade.	SIM	-
Número mínimo ou máximo de relacionamentos.	1-1..*	-
Objeto A deve estar relacionado com objeto B que possua em seu atributo X o valor Y.	SIM	-

Cidade

	Nome
Aceita valor nulo?	NÃO
Será permitido alteração futura?	SIM
Deverá ser único?	SIM
Padrão (regex)?	NÃO
Quantidade mínima e máxima de caracteres	3-100
Aceita valores espaço em branco ou não?	SIM

Feedback – Tarefa 01

- + Diagrama
 - Colocar uma hierarquia onde a superclasse é Pessoa (classe abstrata) e duas subclasses Funcionário e Administrador.
 - O Administrador poderá cadastrar novos funcionários. E os funcionários poderão cadastrar linhas.
 - O Funcionário poderá gerenciar somente as linhas cadastradas por ele. Assim, classe Funcionário tem um relacionamento com a classe Linha.
 - Pense em mais alguns atributos, como telefones, documentos, endereços.
- + Restrições
 - É bom pensar nas restrições para cada atributo, por exemplo, você colocou que atributos de relacionamentos terão no mínimo 1 e máximo 2 relações. Porém no diagrama está diferente. Acho que as restrições devem ser pensadas para cada atributo, pois cada um tem suas peculiaridades.
 - Por exemplo, atributo nome da classe Pessoa: Não pode ser branco, nulo e deve ter no máximo 100 caracteres. Atributo Email da classe Pessoa: Não pode ser branco, nulo e deve ter no máximo 200 caracteres, além de ser único.
 - Favor revisar todas as restrições e mandar documento atualizado na próxima tarefa já com as classes implementadas.

Resposta Feedback – Tarefa 01

Realizado todas as correções solicitadas no diagrama e nas restrições o melhor que pude compreender. Diminuí a quantidade de caracteres para no máximo 100, pois acredito ser desnecessário para cadastrar uma linha mais caracteres.

Feedback – Tarefa 02

- Localidade não tem uma lista de cidades. Tem apenas uma cidade.
- Chegou a pesquisar um tipo mais adequado para horário? Veja esse: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalTime.html>

Resposta Feedback – Tarefa 02

- A lista de cidade foi alterada para apenas uma cidade e o relacionamento localidade com cidade ficou many to one, sendo many a localidade, com isso tive que inserir um atributo lista de localidades na classe cidade, pois o hibernate estava retornando erro e não rodava o projeto, pois esperava uma lista e não apenas um atributo simples do tipo localidade.
- A questão do horário tínhamos deixado como string para facilitar, em vista que é apenas informativo, porém dei uma pesquisada e vi que no java 8 se facilitou muito a utilização de datas e horas e creio que seja interessante ter implementado o localtime para se eu quiser fazer comparações futuras de horários. Alterado então horário para LocalTime.

Feedback – Tarefa 03

- - @ElementCollection deve ser usado em coleções de tipos primitivos ou classes embedable.
- - Retirar anotação @ElementCollection do atributo linhas na classe Funcionario.
- - Em Linha.trajetos: retirar @ElementCollection
- - Em Trajeto.localidades: retirar @ElementCollection
- - No relacionamento bidirecional entre Localidade e Cidade, não foi definido o dono do relacionameto (quando se usa o mappedBy). Assim estão sendo representados no banco de dados dois relacionamentos. Definir quem é o dono da relação, na minha visão seria a localidade.

Resposta Feedback – Tarefa 03

- Sobre os Elementcollection's devo ter entendido a aula incorretamente, mas já corrigi, retirei todos, pois todos são do tipo da classe e nenhuma classe é embedable.
- Sobre o mappedby, erro foi corrigido e foi colocado na classe cidade, sendo o atributo cidade em localidade o dono da relação, assim como conversado com o senhor via msg no moddle.

OBS - Tarefa 04

Modificações:

- Os nomes podem ter no mínimo 3 caracteres, percebi que não há necessidade de ter um nome completo para funcionário ou administrador e cidades podem ter nomes pequenos também.

Feedback – Tarefa 04

- - Quando anotamos um atributo com @Valid, queremos que ao validar o objeto pai, seja feito a validação do objeto contido no atributo anotado. Nem sempre isso é necessário, principalmente quando o objeto contido no atributo já está cadastrado no BD e com isso, já validado.
- - Não vejo necessidade de @Valid em Localidade.cidade.
- - Não vejo necessidade de @Valid em Localidade.trajeto.
- - Não vejo necessidade de @Valid em Linha.funcionario.
- - Poderia usar uma @Length só tanto para min, quanto para max.
- - Colocaria @Valid em Linha.trajeto e em Trajeto.localidade. Pois geralmente será criado uma linha com trajetos e localidades e depois irá salvar a Linha, validando a Linha, bem como suas relações (trajetos e localidades).

Resposta Feedback – Tarefa 04

- Todas as correções da notação @Valid realizadas;
- Eu coloquei @Length um para min e outro para max, apenas para personalizar a mensagem para o usuário mesmo. Estou ciente que posso usar apenas 1.

OBS - Tarefa 05

- Criei na classe model.Linha o atributo nome, pois percebi que não existia.
- Alterei todos os Id's dos model de int para Long, para poder utilizar nos repositórios.
- Alterei na classe model.trajeto o atributo List<Localidade> localidades, colocando um cascade = CascadeType.ALL, orphanRemoval = true, para quando for excluída a Linha, tanto o trajeto como a localidade seja excluída tb. Ficando assim apenas as cidades para verificação na exclusão.

Feedback – Tarefa 05

- método "verificarExclusaoCidadeComLocalidade" não implementado. Sugiro criar um método do repositório de cidade que retorne à quantidade de localidades em que a cidade faz parte.

Resposta Feedback – Tarefa 05

- O método verificarExclusaoCidadeComLocalidade foi implementado de forma mais simples, sem criar um método no repositório.

Feedback – Tarefa 06

- Em "verificaCidadeEmLocalidade" basta esse método receber a cidade (obj) como parâmetro e colocar um if verificando de obj.localidades está vazio.
- O erro ao salvar uma Linha se deve ao fato de você ter colocado unique=true para o atributo opcao na classe trajeto. Troque para falso.

Resposta Feedback – Tarefa 06

- Alterado o parâmetro unique da linha para false.
- Alterado o método “verificaCidadeEmLocalidade” para receber o obj.

Feedback – Tarefa 07

- Em "verificaCidadeEmLocalidade", você está emitindo a exception se getLocalidade() estiver vazio, porém deveria ser a negação disso. A exceção deve ser lançada quando não estiver vazio. Aproveito para colocar uma observação de que o atributo localidade da classe Cidade deveria estar no plural, pois é uma lista.

Resposta Feedback – Tarefa 07

- Realmente a verificação da exception estava sem negação, foi uma falta de atenção minha assim como a lista em localidades. Agradeço o feedback, foram corrigidos os erros.

Feedback – Tarefa 08

- Linhas não permite a adição de trajetos e localidades.

Resposta Feedback – Tarefa 08

- Infelizmente não estou com tempo hábil para resolver o problema dos trajetos devido ao trabalho que consegui no final do semestre, mas não abandonarei o projeto, vou implementar os trajetos e localidades. Eu fico grato a todos os ensinamentos e feedbacks dado nesta disciplina.