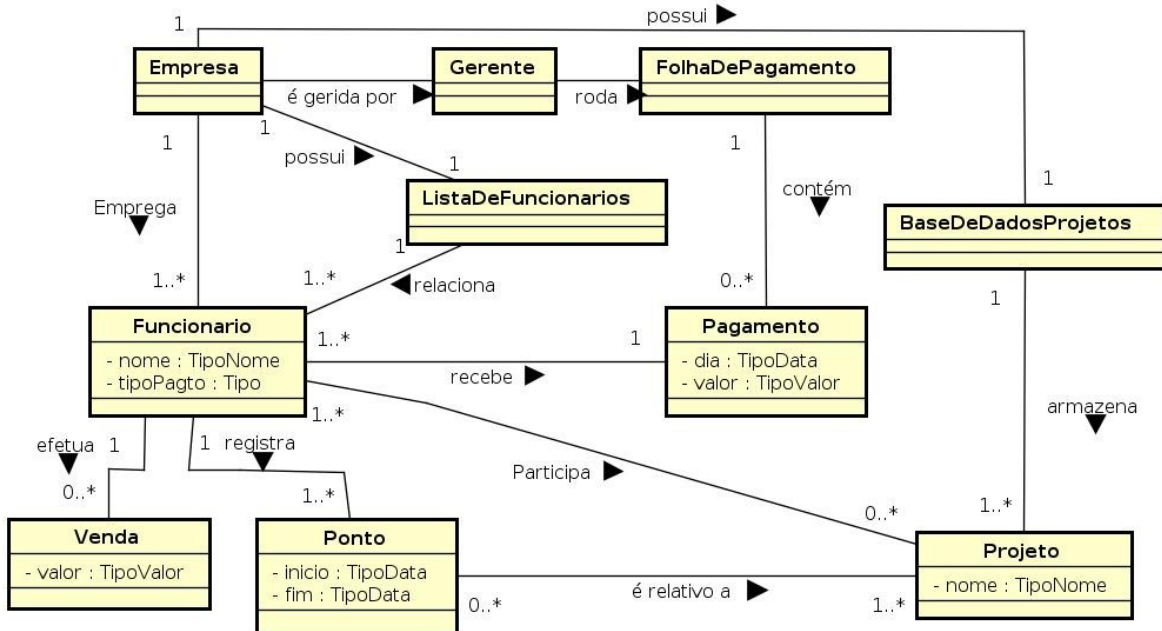


Exercício: Dados os os contratos de operação (operações que gerarem mudança de estado) dos Casos de uso Manter Informações de Empregado e Manter Cartão de Ponto e o modelo conceitual, crie os Diagramas de Sequência e Comunicação correspondentes (1 de cada para cada contrato).



powered by Astah

Operação: adicionarFuncionario(nome, tipo, endereço, inss, telefone, taxaHora, salario, taxaComissao, limiteHora)

Referências cruzadas: Caso de uso Manter Informações de Empregado

Pré-condições: Existe uma instância 'funcionários' de ListaDeFuncionários ativa.

Pós-Condições:

- uma nova instância 'func' de Funcionario é criada;
- nome é atribuído a func.nome;
- tipo é atribuído a func.tipo;
- endereço é atribuído a func.endereço;
- inss é atribuído a func.inss;
- telefone é atribuído a func.telefone;
- taxaHora é atribuído a func.taxaHora;
- salario é atribuído a func.salario;
- taxaComissao é atribuído a func.taxaComissao;
- limiteHora é atribuído a func.limiteHora;
- um número ID é gerado e atribuído a func.ID;
- entrega_pgto = "retirada" é atribuído a func.entrega_pgto;

func é adicionado à lista de funcionário.

Operação: buscarFuncionario(ID)

Referências cruzadas: Caso de uso Manter Informações de Empregado

Pré-condições: Existe uma instância 'funcionários' de ListaDeFuncionários ativa

Pós-Condições:

A instância 'func' de Funcionario com func.ID = ID é obtida de 'funcionários'.

Os dados de 'func' são enviados para a Tela.

Operação: alterarFuncionario(nome, tipo, endereço, inss, telefone, taxaHora, salario, taxaComissao, limiteHora)

Referências cruzadas: Caso de uso Manter Informações de Empregado

Pré-condições: uma instância 'func' de Funcionario está ativa;

Existe uma instância 'funcionários' de ListaDeFuncionários ativa.

Pós-Condições:

nome é atribuído a func.nome;

tipo é atribuído a func.tipo;

endereço é atribuído a func.endereço;

inss é atribuído a func.inss;

telefone é atribuído a func.telefone;

taxaHora é atribuído a func.taxaHora;

salario é atribuído a func.salario;

taxaComissao é atribuído a func.taxaComissao;

limiteHora é atribuído a func.limiteHora;

um número ID é gerado e atribuído a func.ID;

entrega_pgto = "retirada" é atribuído a func.entrega_pgto;

Operação: ExcluirFuncionario()

Referências cruzadas: Caso de uso Manter Informações de Empregado

Pré-condições: uma instância 'func' de Funcionario está ativa;

Existe uma instância 'funcionários' de ListaDeFuncionários ativa.

Pós-Condições:

func é marcado como removido.

Operação: recuperarCartaoPontoAtual()

Referências cruzadas: Caso de uso Manter Cartão Ponto.

Pré-condições: uma instância 'func' de Funcionario está ativa; Existe uma instância 'pontos' de ListaDePontos ativa;

Pós-Condições:

A instância 'pontoAtual' de Ponto atual é obtida de 'pontos'; se não houver, uma nova instância será criada;

Os dados de 'pontoAtual' são enviados para a Tela.

Operação: recuperarProjeto()

Referências cruzadas: Caso de uso Manter Cartão Ponto.

Pré-condições: uma instância 'func' de Funcionario está ativa; Existe uma instância 'projetos' de ListaDeProjetos ativa;

Pós-Condições:

Uma lista 'projs' de números de projetos é obtida a partir de 'projetos';

Os dados de 'projs' são enviados para a Tela.

Operação: insereDadosPonto(numero_projeto, horalni, horaFim)

Referências cruzadas: Caso de uso Manter Cartão Ponto.

Pré-condições: uma instância 'func' de Funcionario está ativa; Existe uma instância 'pontoAtual' de Ponto que representa o ponto atual ativa; Existe uma instância 'projetos' de ListaDeProjetos ativa; Existe uma instância 'pontos' de ListaDePontos ativa;

Pós-Condições:

uma instância 'proj' de Projeto com 'proj.ID' = ID é obtida de 'projetos';

'pontoAtual' é associado a 'proj';

horalni é atribuído a 'pontoAtual.horalni';

horaFim é atribuído a 'pontoAtual.horaFim';

A instância 'pontos' é salva na camada de persistência.