

Proposta de projeto JAVA / POO

Ver modelo UML com as classes, as associações entre as classes, os atributos e os métodos a serem criados.

CLASSES

1. Implementar classe conta – conforme diagrama UML.
2. Implementar classe conta corrente – conforme diagrama UML.
 - a. Os métodos debitar e creditar devem, respectivamente, subtrair do saldo e adicionar ao saldo o valor recebido.
3. Implementar a classe conta poupança – conforme diagrama UML.
 - a. O método debitar deve subtrair do saldo o valor recebido.
 - b. O método debitar deve adicionar ao saldo o valor recebido mais o percentual do valor recebido correspondente à taxa de bônus.
4. Implementar o repositório de correntista – conforme diagrama UML e código de referência.
5. Implementar o repositório de conta – conforme diagrama UML e código de referência.
6. Implementar o mediator de correntista – conforme o diagrama UML, as funcionalidades (abaixo) e o código de referência.
7. Implementar o mediator de conta – conforme o diagrama UML, as funcionalidades (abaixo) e o código de referência.
8. Implementar a tela do sistema – conforme o diagrama UML e o código de referência.

FUNCIONALIDADES (compõem o menu da tela).

9. Incluir correntista
 - a. Ler os dados do correntista do teclado.
 - b. Validar CPF (tem que ser maior que zero) e nome (tem que estar preenchido). Não é permitido haver CPF duplicado na lista de correntistas.
 - c. Instanciar um objeto do tipo correntista.
 - d. Incluir tal objeto no repositório que contém a lista de correntistas.
10. Incluir conta corrente ou conta poupança.
 - a. Ler do teclado a opção de cadastrar conta corrente ou conta poupança.
 - b. Ler os dados de uma conta corrente do teclado (número da agência, número da conta, CPF do correntista e tarifa. OU
 - c. Ler os dados de uma conta poupança do teclado (número da agência, número da conta, CPF do correntista e taxa de bônus).
 - d. Buscar o correntista pelo CPF informado.
 - e. Validar CPF (tem que existir na lista de correntistas), número da agência (tem que ser maior que zero) e número da conta (tem que ser maior que zero), tarifa OU taxa de bônus (ambas têm que ser maior ou igual a zero). Não pode existir na lista de contas a combinação agência + número duplicada.
 - f. Instanciar um objeto do tipo conta corrente ou conta poupança.
 - g. Incluir tal objeto no repositório que contém a lista de contas.
11. Debitar
 - a. Ler do teclado número da agência, o número da conta e valor a ser debitado.
 - b. Validar o valor a ser debitado (deve ser maior que zero).
 - c. Buscar a conta pelos números da agência e da conta informados.
 - d. Se a conta for encontrada, realizar a operação de débito na conta. A operação de débito só deve ser realizada se o saldo da conta for maior ou igual ao valor a ser debitado.

12. Creditar

- a. Ler do teclado número da agência, o número da conta e valor a ser creditado.
- b. Validar o valor a ser creditado (deve ser maior que zero).
- c. Buscar a conta pelos números da agência e da conta informados.
- d. Se a conta for encontrada, realizar a operação de crédito na conta.

13. Listar contas ordenadas por saldo em ordem crescente.

- a. Consultar todas as contas do repositório.
- b. Ordenar o resultado por saldo em ordem decrescente.
- c. Mostrar no console a lista de contas (saldo, agência e número).

14. Listar correntistas ordenados por nome em ordem crescente.

- a. Consultar todos os correntistas do repositório.
- b. Ordenar o resultado por nome em ordem crescente.
- c. Mostrar no console a lista de correntistas (nome e cpf).