

Modelo Conceitual

Análise de Sistemas e
Requisitos de Software II

Aula 8

Allan Rodrigo Leite

Modelo Conceitual

- Oferece uma visão das informações que são gerenciadas pelo sistema
 - Representação e transformação da informação
- O modelo conceitual enfatiza os aspectos estruturais do sistema
 - Quais são os conceitos envolvidos no sistema?
 - Como os conceitos estão relacionados?
 - Quais são as informações pertinentes a cada conceito?

Modelo Conceitual

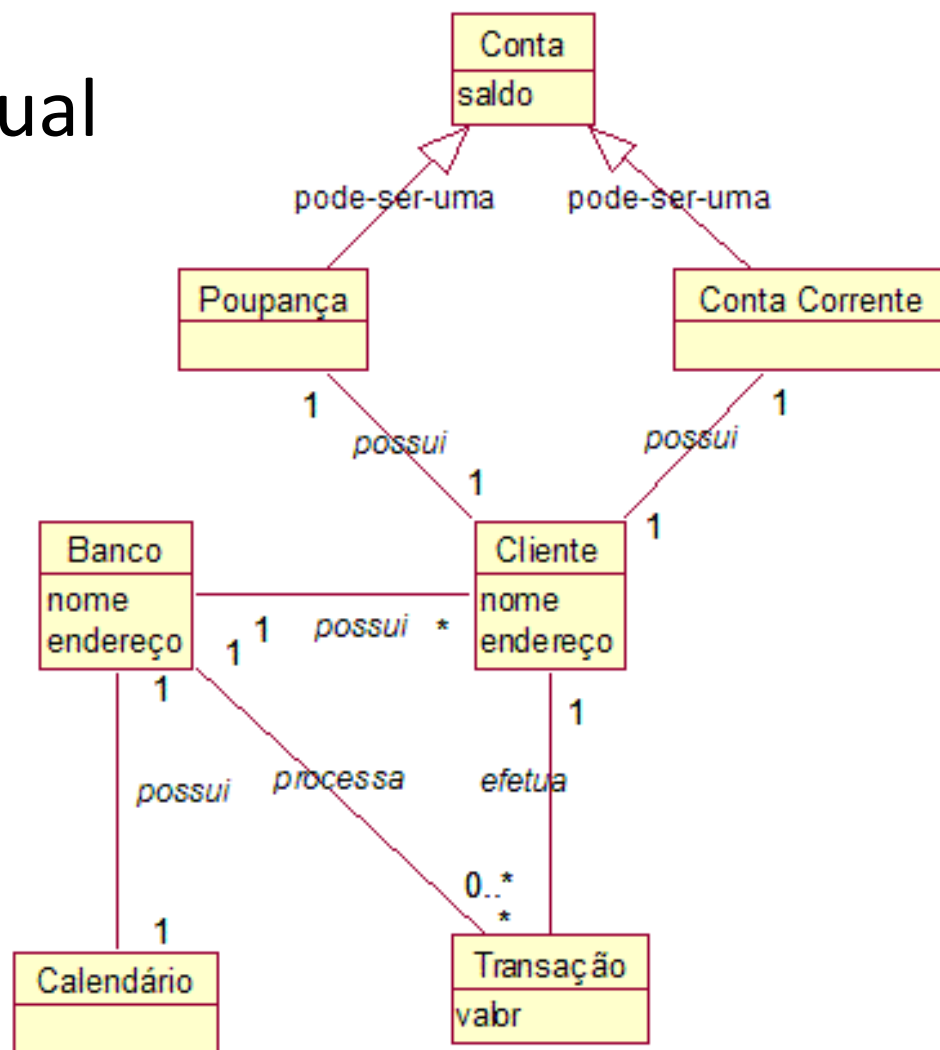
- Um modelo conceitual deve:
 - Ser independente de tecnologia
 - Apenas possuir conceitos pertinentes ao domínio do sistema em questão
 - Não podem conter referências à aspectos dinâmicos dos sistemas

Modelo Conceitual

- Oferece três perspectivas
 - Conceitual
 - Representa os conceitos do domínio relacionado
 - Especificação
 - Foco nas principais interfaces, métodos e associações
 - Implementação
 - Visão detalhista, contendo informações suficientes para o desenvolvimento

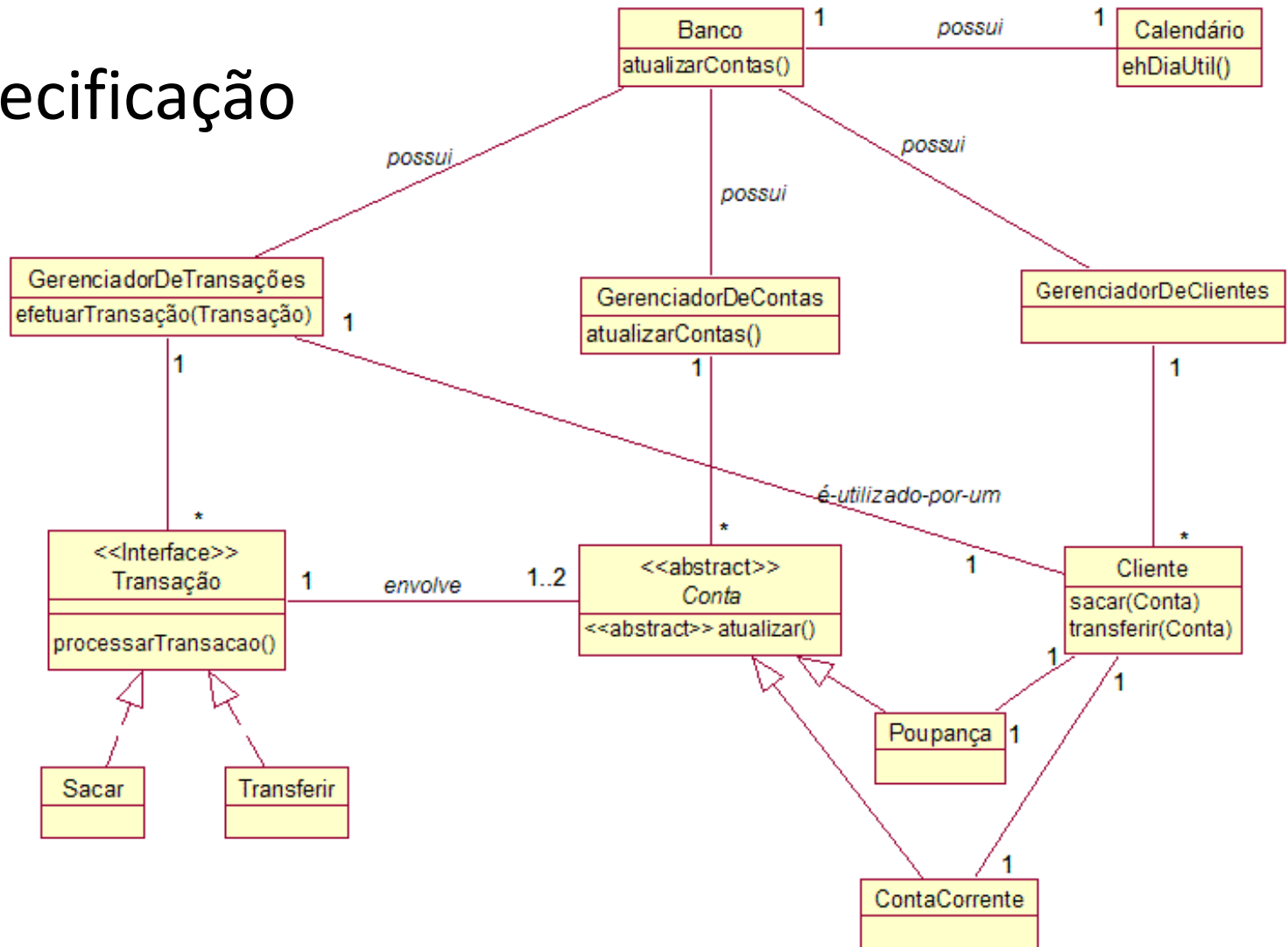
Modelo Conceitual

- Conceitual



Modelo Conceitual

- Especificação



Modelo Conceitual

- Implementação

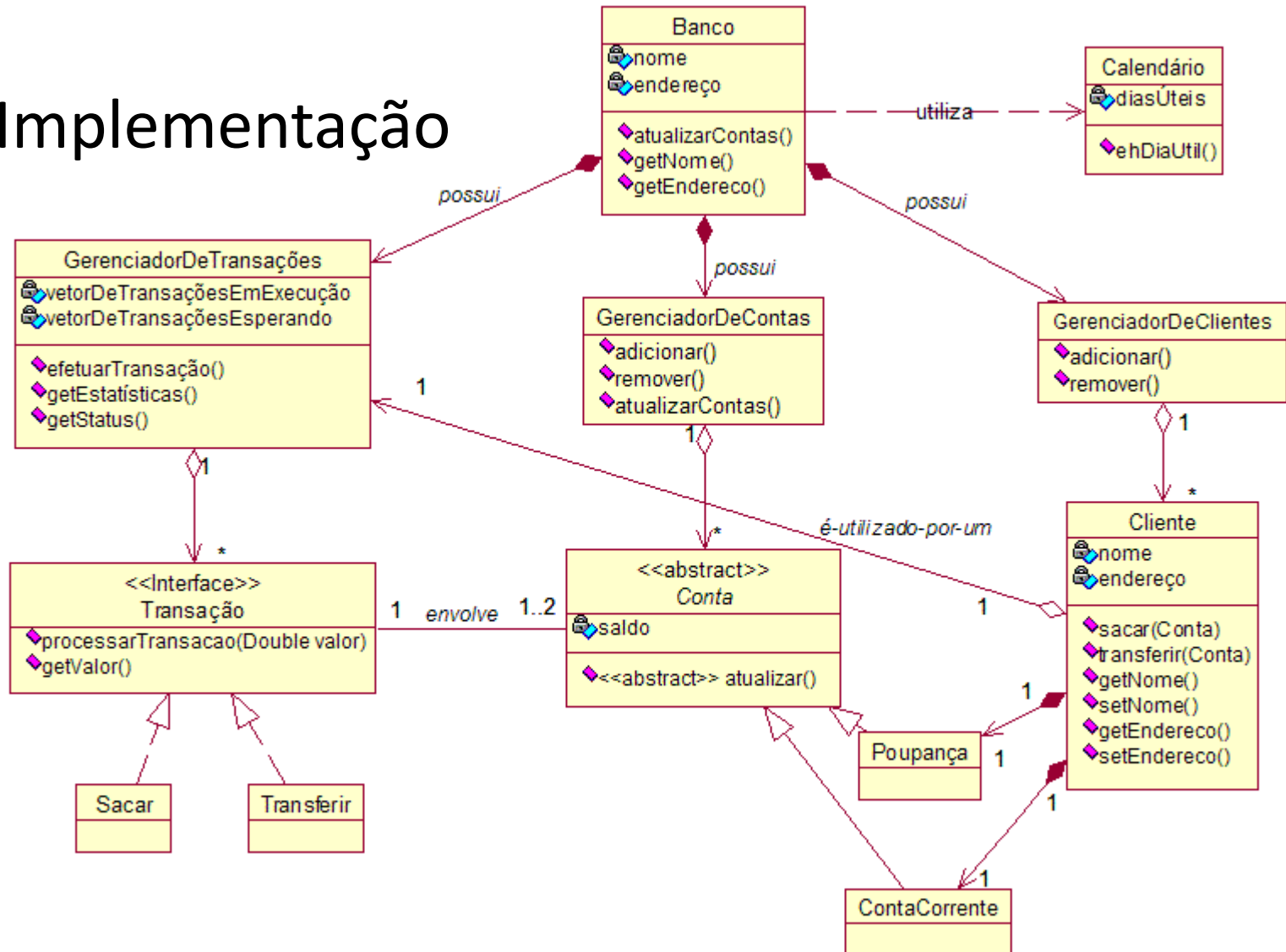


Diagrama de Classes

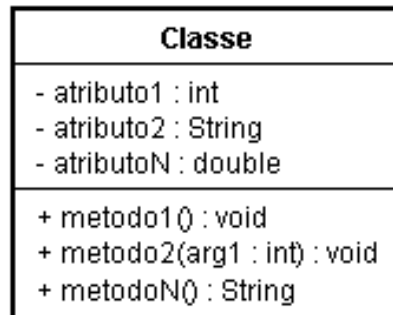
- A UML permite representar visualmente um modelo conceitual através do diagrama de classes
- Um diagrama de classes oferece uma representação da estrutura das classes e seus relacionamentos

Diagrama de Classes

- Atributos
 - Variáveis da classe
 - Os atributos definem a estrutura da classe
- Classes ou conceitos
 - Representação de informações complexas
 - Em geral agregam atributos que não pode ser descritos meramente por tipos simples
- Associações
 - Ligação entre conceitos
- Métodos
 - Comportamento de um conceito

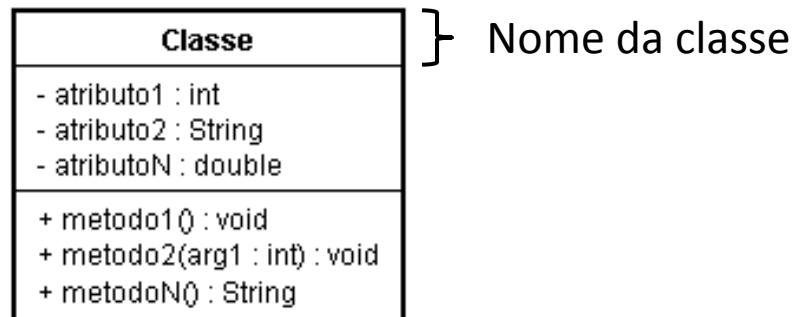
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:



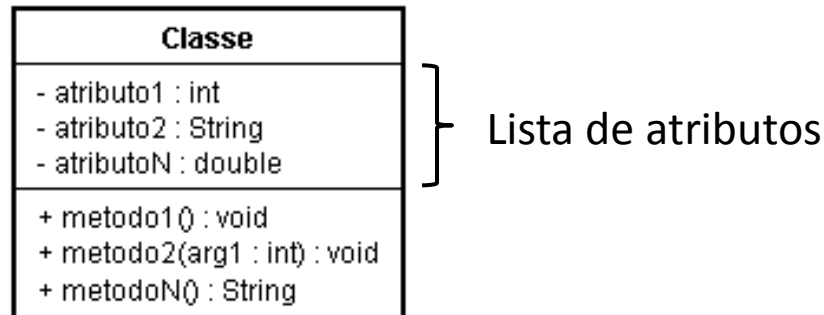
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe



Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos

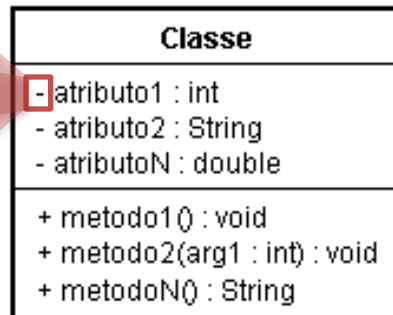


Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos

Modificador de acesso

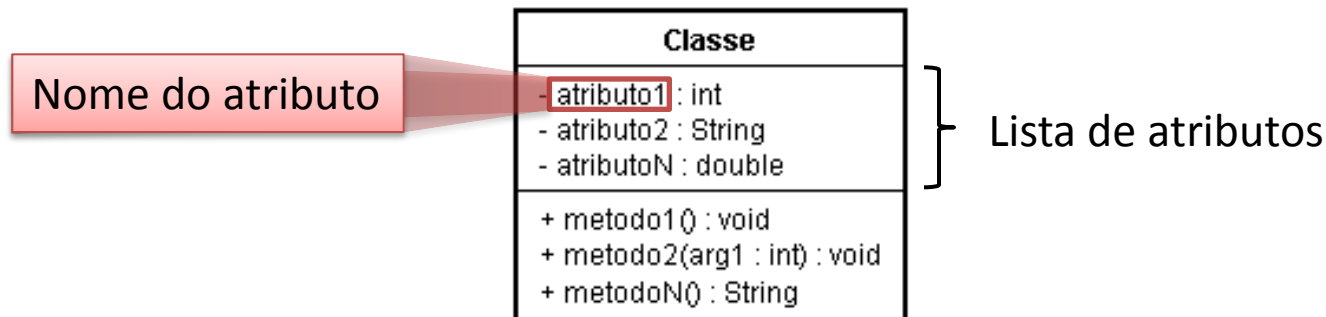
- : Privado
- + : Público
- # : Protegido
- ~ : Pacote



Lista de atributos

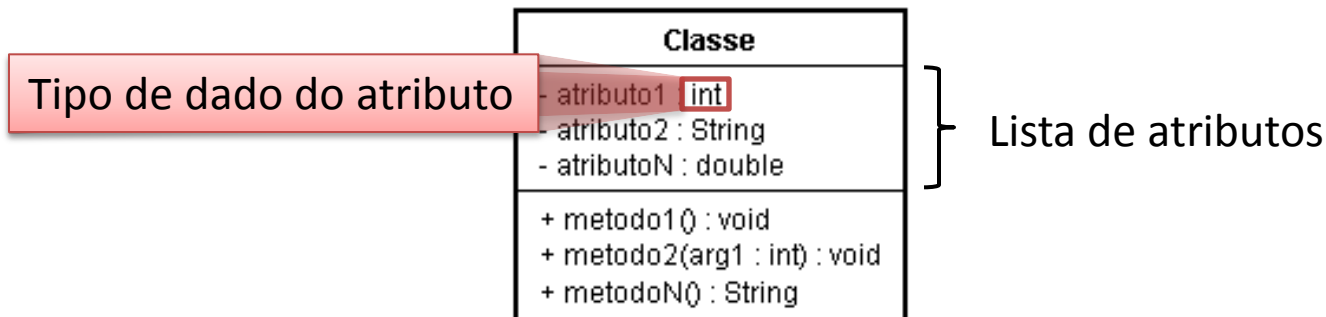
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos



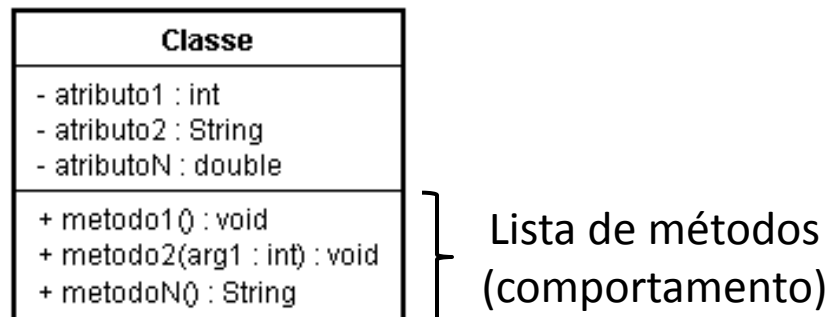
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos



Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos
 - Métodos

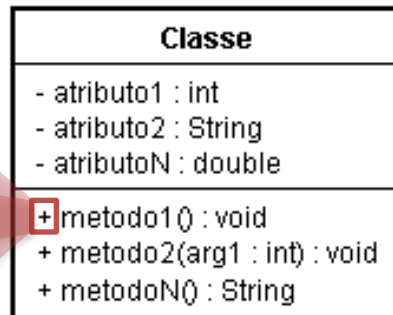


Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos
 - Métodos

Modificador de acesso

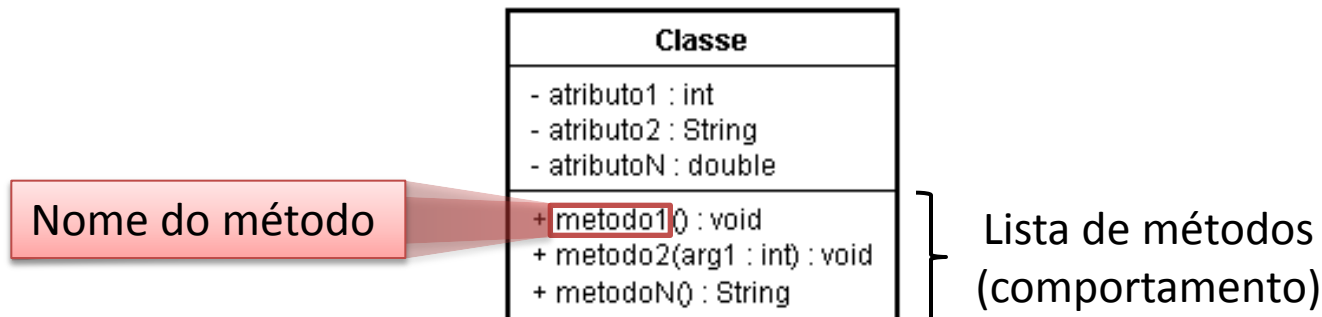
- : Privado
- + : Público
- # : Protegido
- ~ : Pacote



} Lista de métodos
(comportamento)

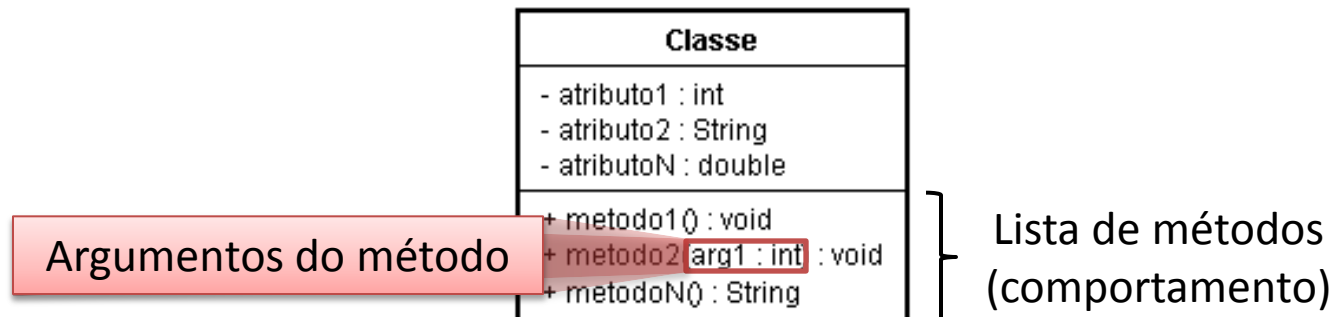
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos
 - Métodos



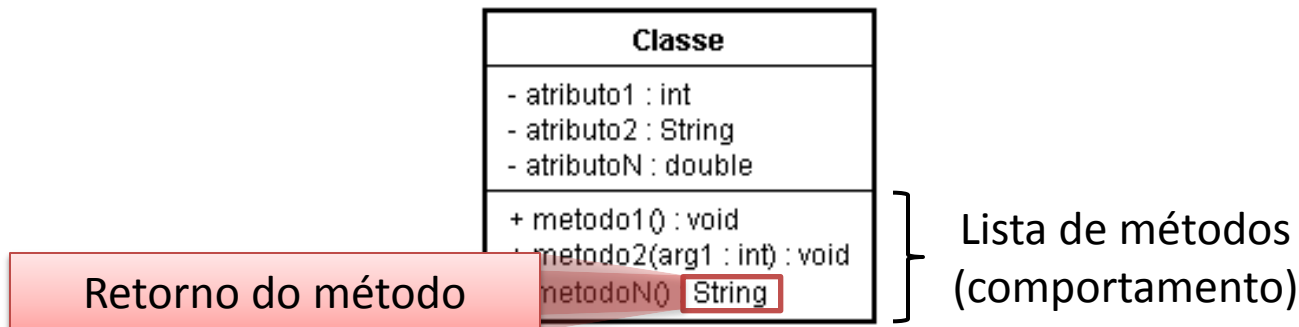
Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos
 - Métodos



Representação de uma Classe

- No diagrama de classes da UML, uma classe é representada por um retângulo dividido em três sessões:
 - Nome da classe
 - Atributos
 - Métodos

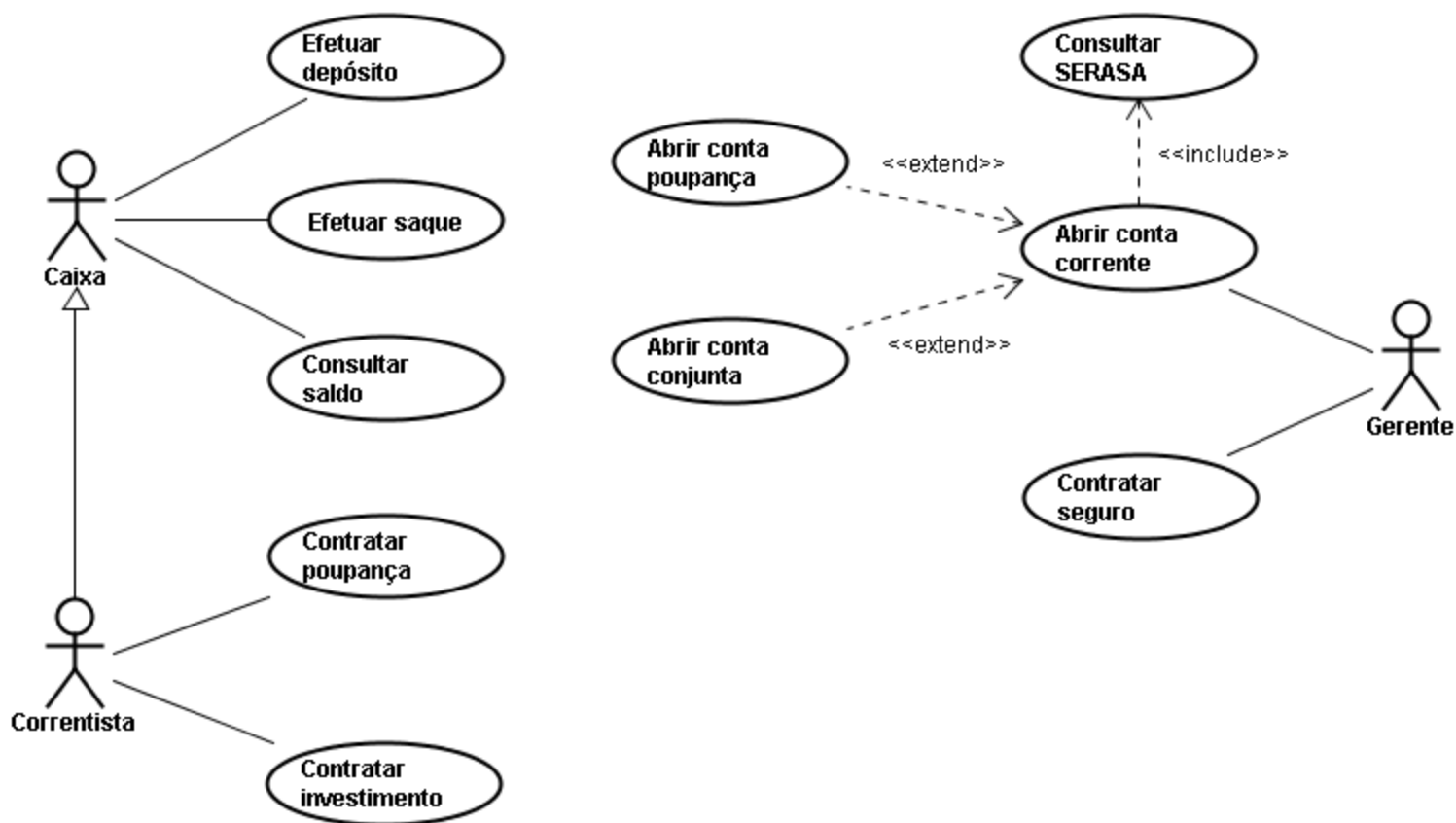


Como identificar conceitos?

Visão geral do sistema - gestão bancária

O sistema a ser desenvolvido deve ser capaz de gerenciar as principais atividades bancárias, permitindo ao correntista realizar operações como: abertura de conta corrente, consultar saldo, pagamento de boletos, saque, depósito, seguros, poupanças e fundos de investimentos. Apenas os gerentes podem realizar aberturas de contas e contratos de seguro. A abertura de conta é liberada mediante a consulta em serviços de proteção ao crédito. Já o correntista pode realizar as demais operações. As operação realizada pelo correntista deve computar pontos que serão utilizados para reduzir proporcionalmente o valor de suas tarifas bancárias.

Como identificar conceitos?



Como identificar conceitos?

- Examinar o conteúdo dos casos de uso expandidos do sistema
 - Substantivos, adjetivos e verbos são candidatos
- Também é possível identificar os atributos relacionado a cada conceito pelo texto dos casos de uso expandido

Fluxo principal:

1. O cliente se identifica e comunica ao gerente que deseja abrir uma conta corrente.
2. [EV] O gerente informa o CPF do cliente ao sistema.
3. [RS] O sistema realiza uma consulta nos sistemas de proteção ao crédito.
4. [EV] O gerente informa os demais dados pessoais do cliente ao sistema.
5. O cliente decide que tipo de conta corrente ele deseja abrir:
 - 5.1 Abrir conta corrente básica
 - 5.2 Abrir conta poupança
 - 5.3 Abrir conta conjunta
6. [RS] O cliente recebe o número de sua conta corrente.
7. [EV] O cliente informa uma senha de acesso para a conta corrente.
8. [EV] O gerente finaliza a abertura da conta corrente.

Variante 5.1: Abrir conta corrente básica

- 5.1.1 [RS] O sistema retorna os valores das tarifas referente a conta corrente básica.

Variante 5.2: Abrir conta poupança

- 5.2.1 O gerente questiona ao cliente qual será o valor mínimo a ser depositado mensalmente na poupança.
- 5.2.2 [EV] O gerente informa ao sistema o valor mensal a ser depositado na poupança.
- 5.2.3 [RS] O sistema retorna um extrato das correções dos últimos 12 meses.
- 5.2.4 [RS] O sistema retorna os valores das tarifas referente a conta poupança.

Variante 5.3: Abrir conta conjunta

- 5.3.1 O gerente solicita os dados do segundo titular da conta corrente.
- 5.3.2 [EV] O gerente informa os dados do segundo titular para o sistema.
- 5.3.3 [RS] O sistema retorna os valores das tarifas referente a conta conjunta.

Fluxo principal:

1. O cliente se identifica e comunica ao gerente que deseja abrir uma conta corrente.
2. [EV] O gerente informa o CPF do cliente ao sistema.
3. [RS] O sistema realiza uma consulta nos sistemas de proteção ao crédito.
4. [EV] O gerente informa os demaís dados pessoais do cliente ao sistema.
5. O cliente decide que tipo de conta corrente ele deseja abrir:
 - 5.1 Abrir conta corrente básica
 - 5.2 Abrir conta poupança
 - 5.3 Abrir conta conjunta
6. [RS] O cliente recebe o número de sua conta corrente.
7. [EV] O cliente informa uma senha de acesso para a conta corrente.
8. [EV] O gerente finaliza a abertura da conta corrente.

Variante 5.1: Abrir conta corrente básica

- 5.1.1 [RS] O sistema retorna os valores das tarifas referente a conta corrente básica.

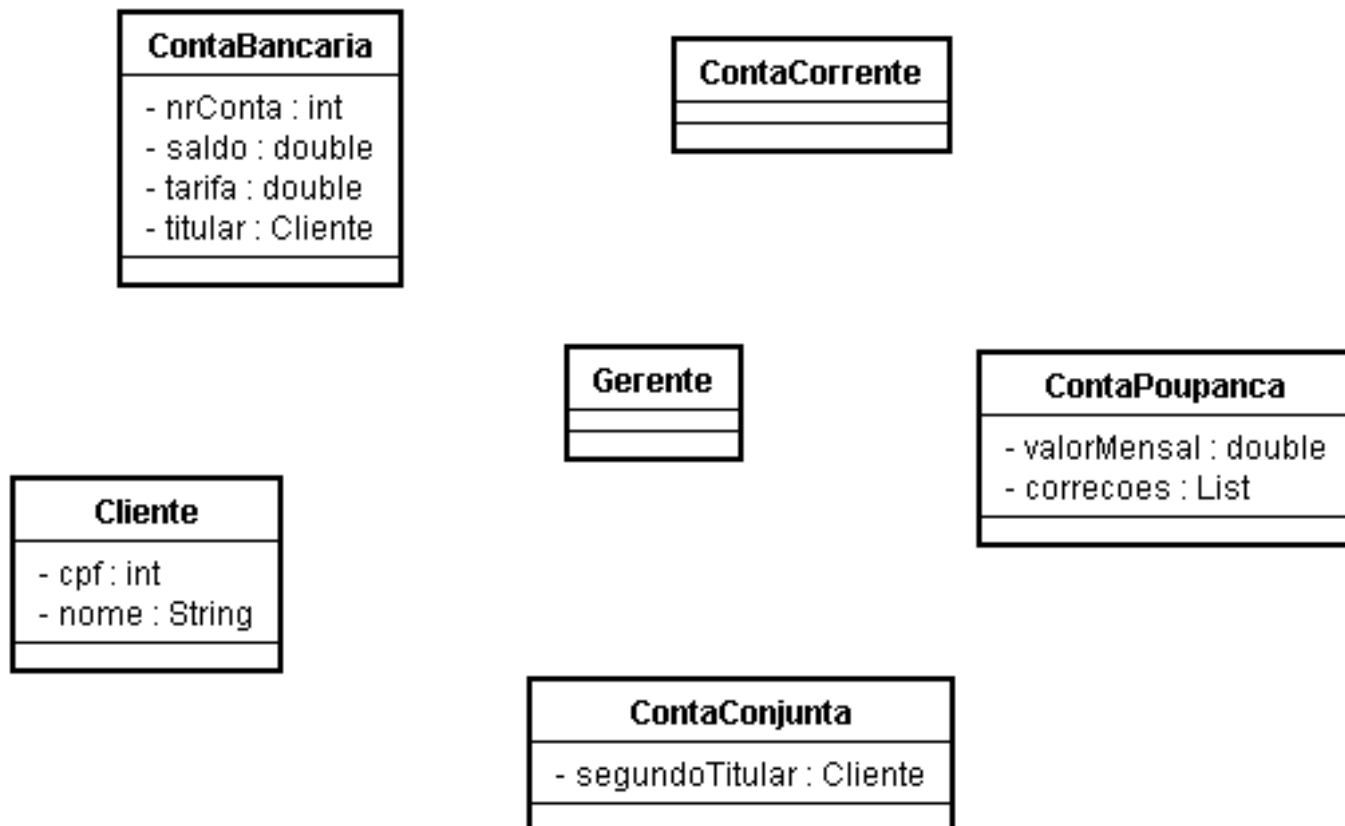
Variante 5.2: Abrir conta poupança

- 5.2.1 O gerente questiona ao cliente qual será o valor mínimo a ser depositado mensalmente na poupança.
- 5.2.2 [EV] O gerente informa ao sistema o valor mensal a ser depositado na poupança.
- 5.2.3 [RS] O sistema retorna um extrato das correções dos últimos 12 meses.
- 5.2.4 [RS] O sistema retorna os valores das tarifas referente a conta poupança.

Variante 5.3: Abrir conta conjunta

- 5.3.1 O gerente solicita os dados do segundo titular da conta corrente.
- 5.3.2 [EV] O gerente informa os dados do segundo titular para o sistema.
- 5.3.3 [RS] O sistema retorna os valores das tarifas referente a conta conjunta.

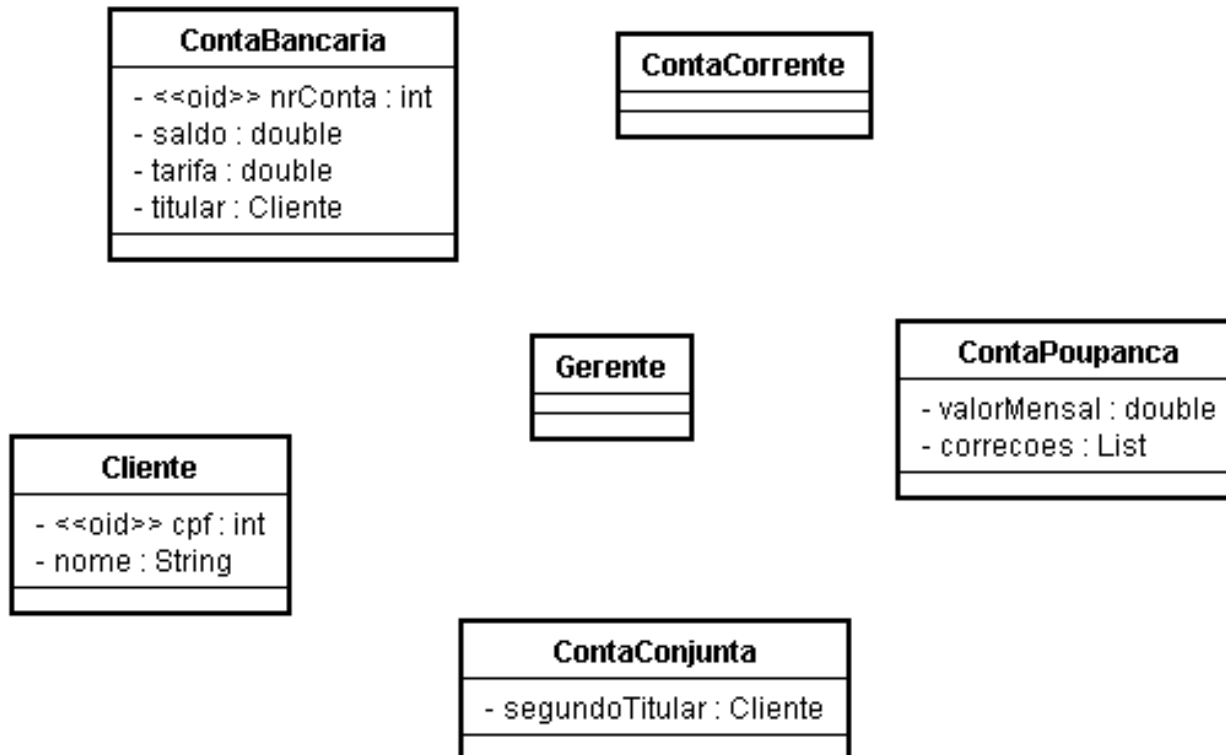
Diagrama de Classes



Como identificar um objeto?

- Todo objeto deve possuir um identificador exclusivo para diferenciá-lo dos demais
- O identificador é um atributo cujo valor é único
 - Não deve existir duas instâncias do mesmo conceito com o mesmo valor para este atributo
- Este atributo é chamado de Object ID (OID)
 - Usa-se o estereótipo <<oid>> para destacar o atributo que identifica exclusivamente o objeto

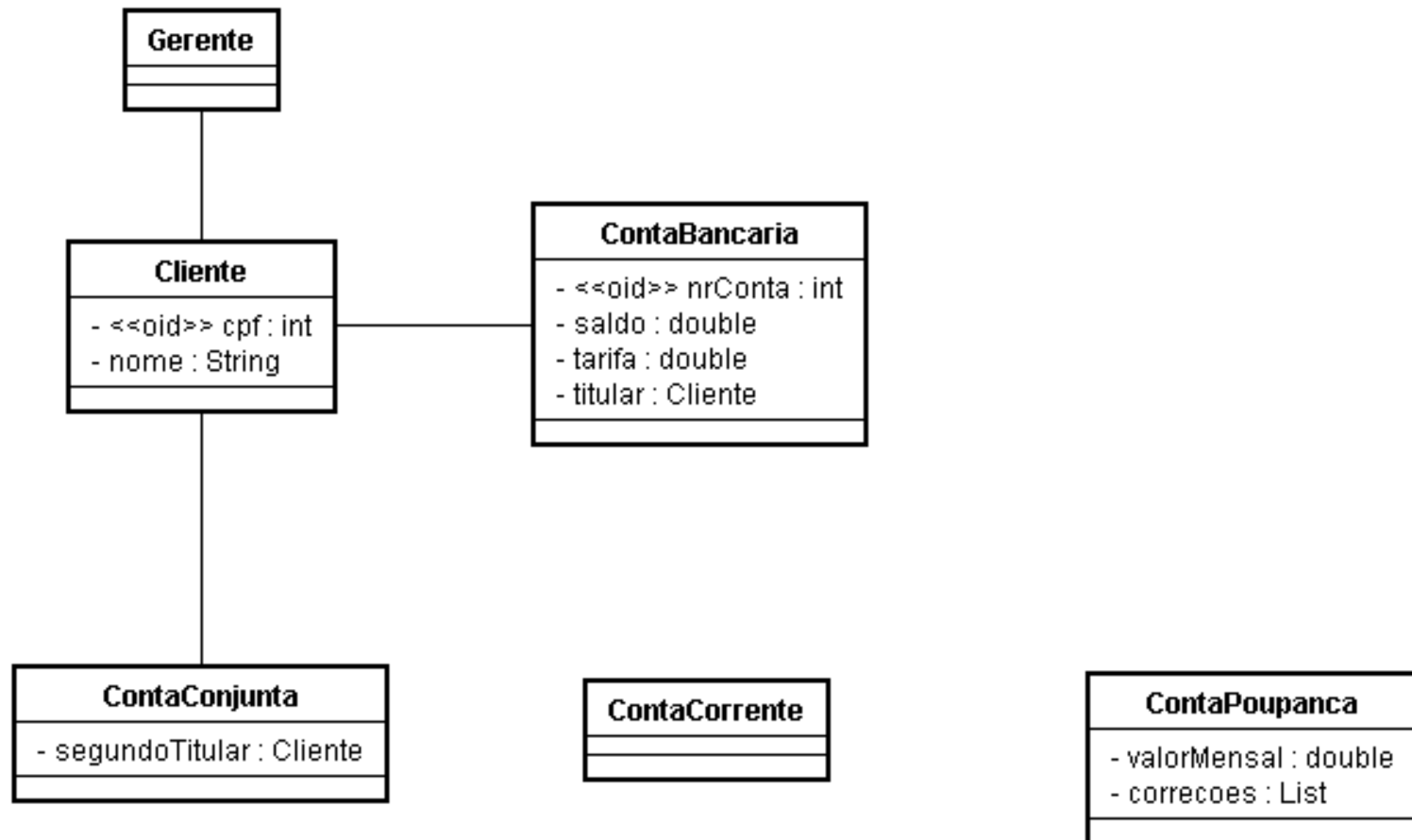
Diagrama de Classes



Associações entre Classes

- Um dos principais objetivos do diagrama de classes é a compreensão das associações entre as classes
- Geralmente uma associação complementa a informação de um objeto em um determinado instante
- Como identificar associações?
 - Localizar conceitos que não podem ser compreendidos isoladamente (depende de uma ligação para fazer sentido)

Diagrama de Classes



Associações entre Classes

- As associações possuem:
 - Nome
 - Descrição da associação
 - Sentido de leitura
 - Auxilia na interpretação da associação
 - Navegabilidade
 - Indica quem enxerga a associação
 - Multiplicidade
 - Indica regras de ocorrência da associação
 - Tipo
 - Significado da associação
 - Papel
 - Contextualiza a associação

Associações entre Classes

- Elementos de uma associação



Associações entre Classes

- Tipos de associações
 - Associação simples (não direcionada)
 - Associação direcionada
 - Herança
 - Agregação
 - Composição
 - Dependência

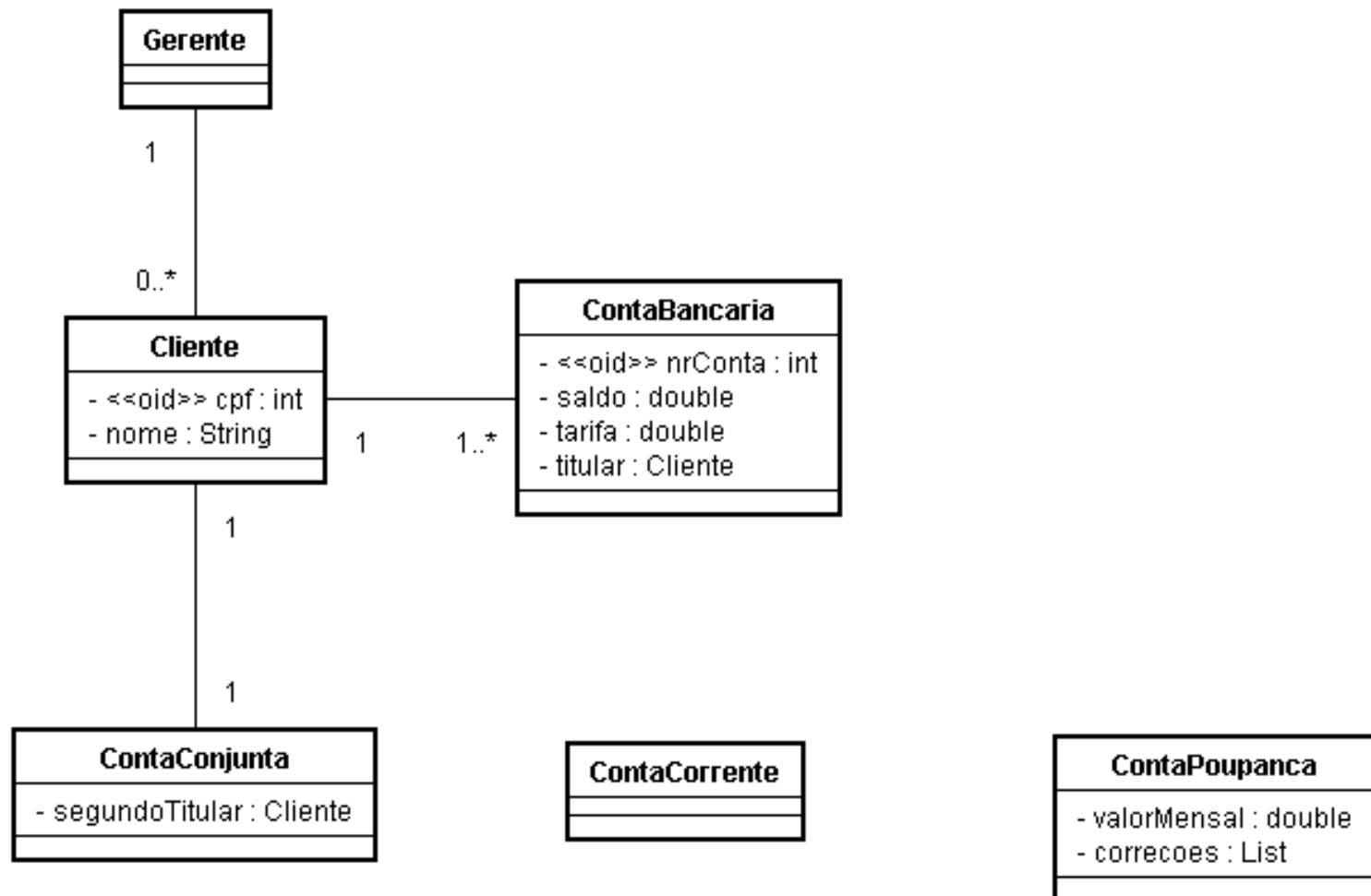
Multiplicidade da Associação

- É fundamental ser possível identificar os limites para a quantidade de elementos envolvidos nas associações
 - Esta informação remete à multiplicidade da associação
- A multiplicidade indica quantos objetos podem se associar
 - Sempre há um limite inferior
 - Pode haver um limite superior

Exemplos de Multiplicidade

- 1 – exatamente um
- 0 . . 1 – zero ou um
- * ou 0 . . * – de zero a infinito
- 1 . . * – de um a infinito
- 2 . . 5 – de dois a cinco
- 2, 5 – dois ou cinco
- 2, 5 . . 8 – dois ou de cinco a oito

Diagrama de Classes

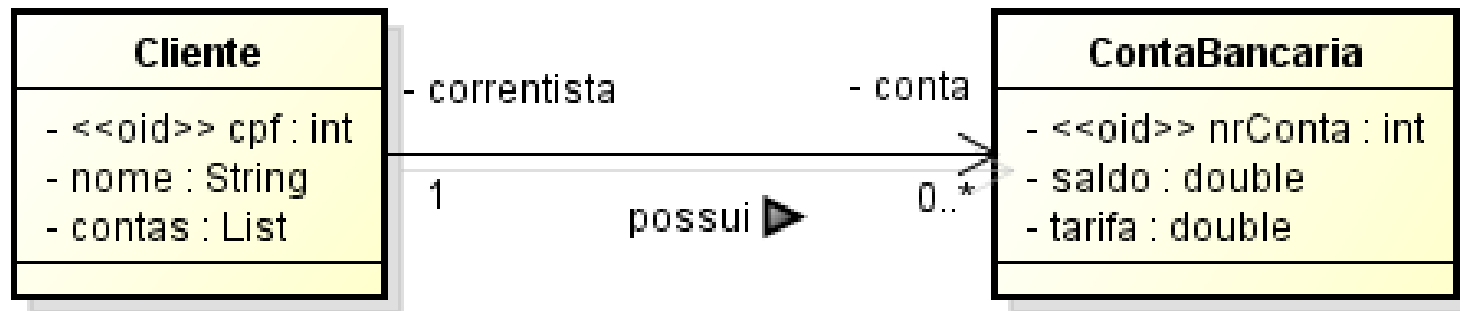


Direção das Associações

- Associações não-direcionais
 - Ocorre quando uma das classes associadas desconhece a existência da associação
- Associações direcionais
 - A associação é visível para as duas classes associadas

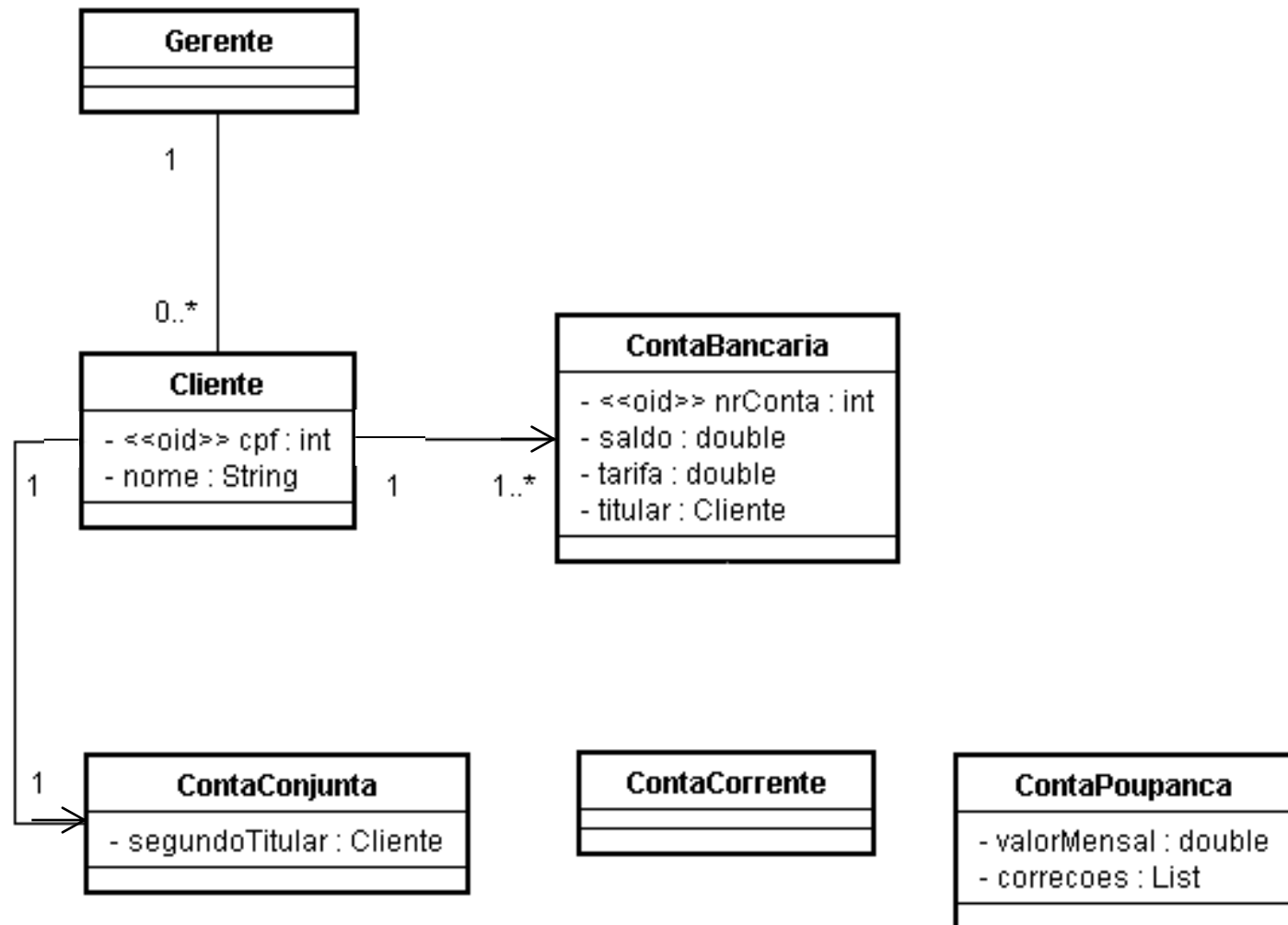
Direção das Associações

- Associação direcionada



O cliente sabe quem é correntista da conta bancária, porém, a conta bancária não sabe quem é o titular

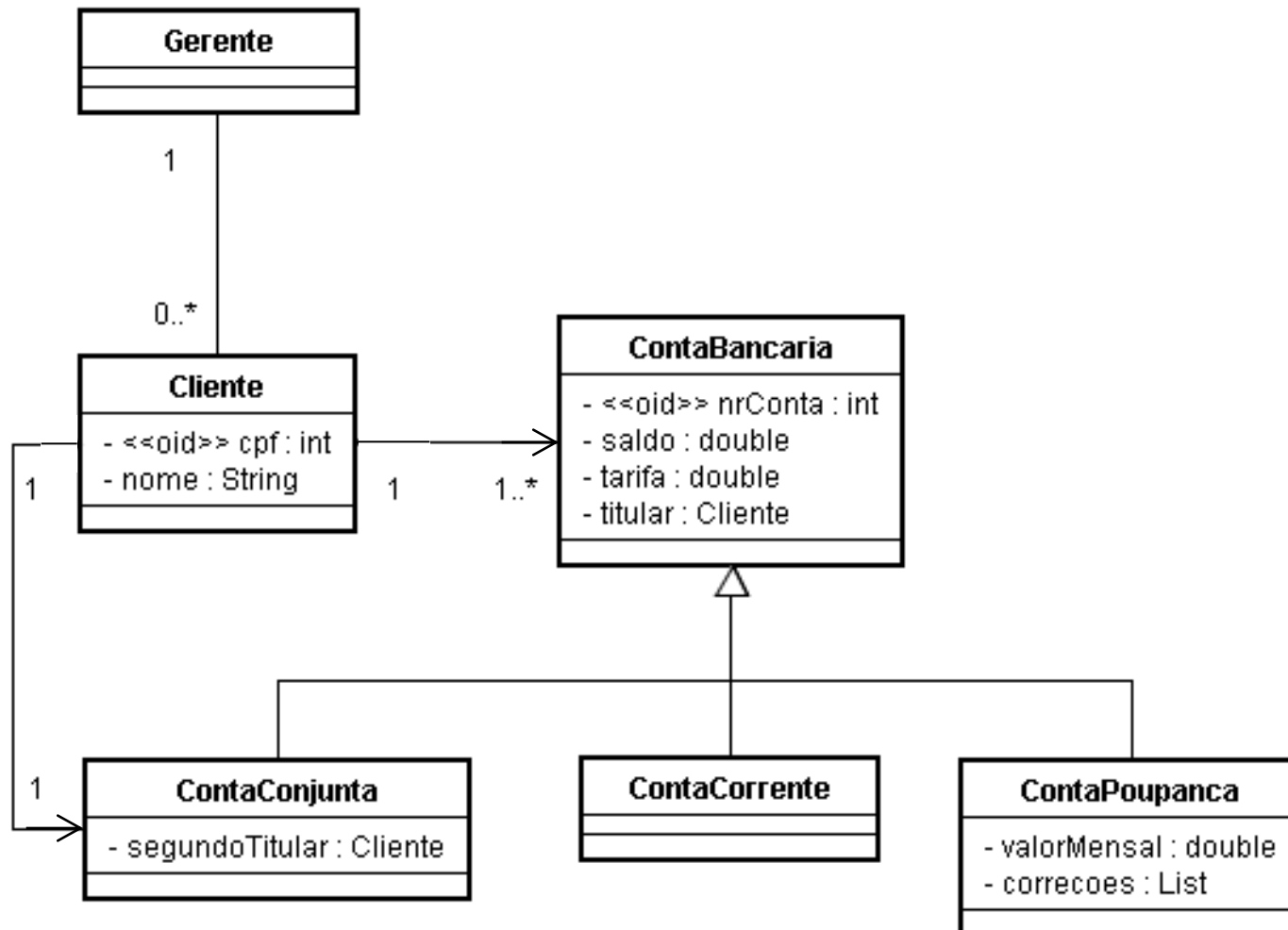
Diagrama de Classes



Herança

- Existe também outro tipo de associação, obtida a partir de uma relação de generalização e especialização das classes
 - Generalização (superclasse)
 - Especialização (subclasse)
- Esta associação é chamada de herança, cuja subclasse herda as características da superclasse

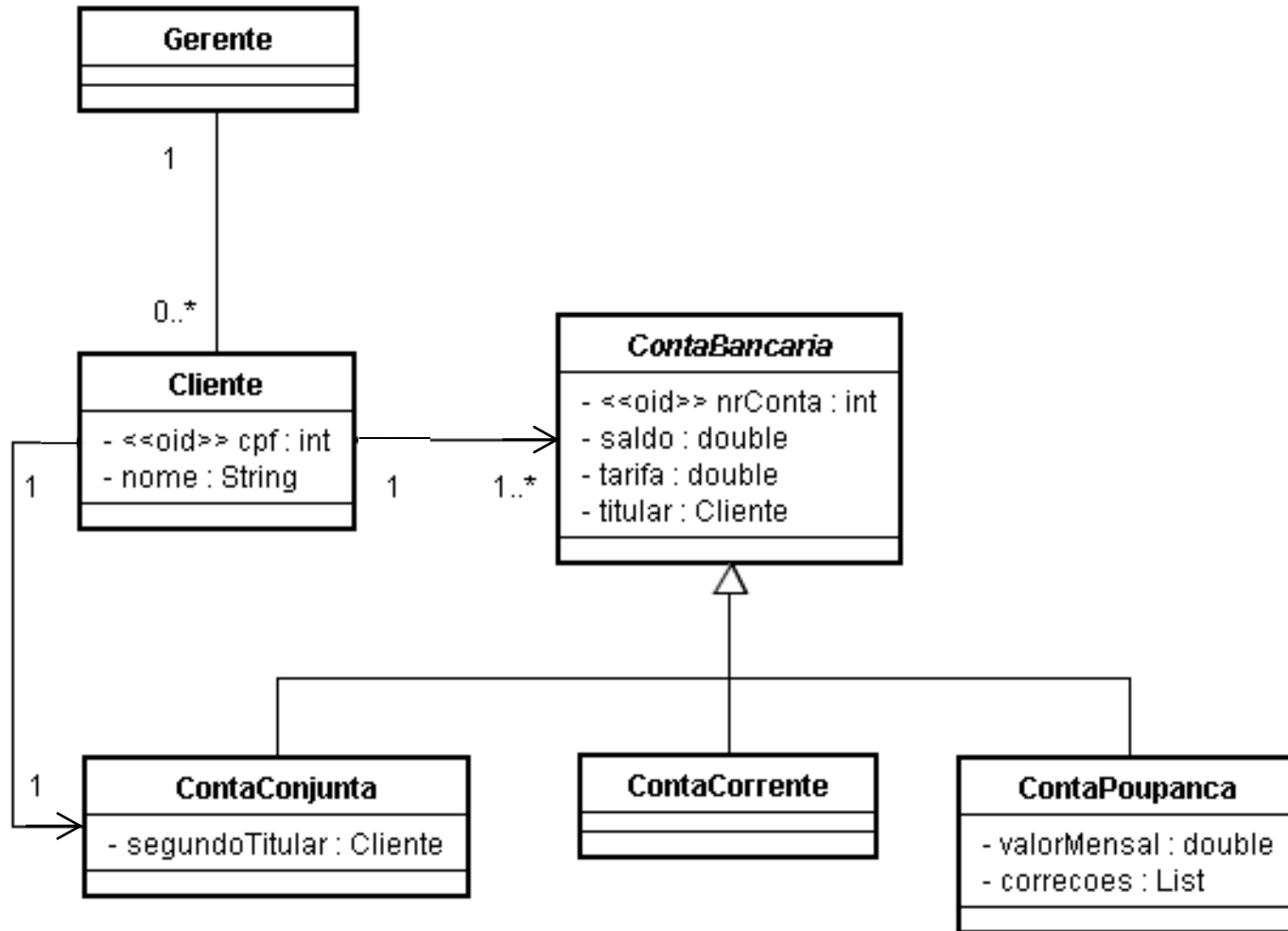
Diagrama de Classes



Classes Abstratas

- Uma classe é denominada abstrata quando representa um conceito abstrato
 - A classe abstrata não possui instâncias
- Usualmente a classe abstrata apresenta métodos abstratos
 - Apenas especifica a assinatura do método
 - A implementação do método deve ficar a cargo das subclasses da classe abstrata
 - Também chamadas de classes concretas

Diagrama de Classes



Agregação e Composição

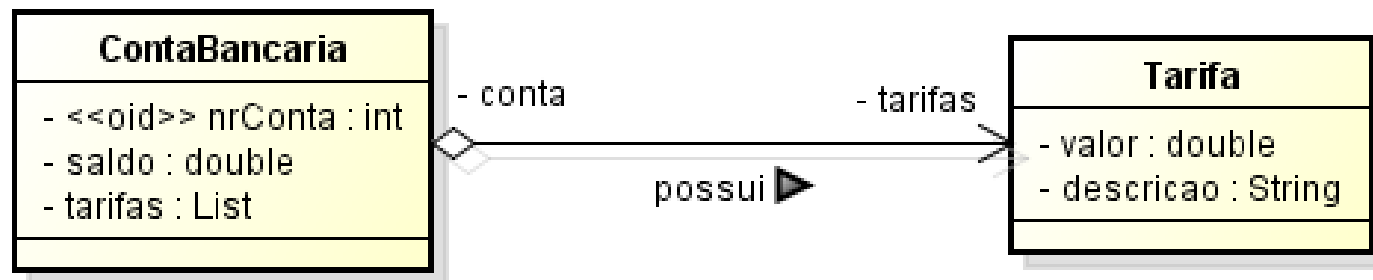
- Algumas associações podem ser consideradas mais fortes do que outras
 - No sentido de que o objeto é composto por outro
- Se há exclusividade na associação, isto é, o objeto não pode ser parte de nenhum outro objeto, a associação é chamada de composição
- Caso contrário, a associação é considerada uma agregação

Como Identificar uma Agregação ou Composição

- Regras
 - Apenas se aplica para relacionamento todo-parte
 - Caso o objeto parte depende exclusivamente do todo para existir então:
 - A relação é de composição
 - Caso o objeto parte não faz sentido ou não possui utilidade sem o todo então:
 - A relação é de agregação
 - Caso contrário:
 - A relação é simples

Agregação e Composição

- Agregação



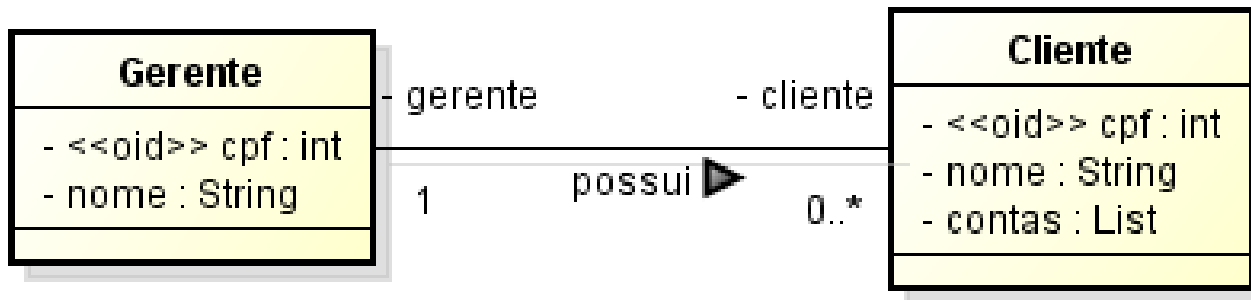
Agregação e Composição

- Composição



Agregação e Composição

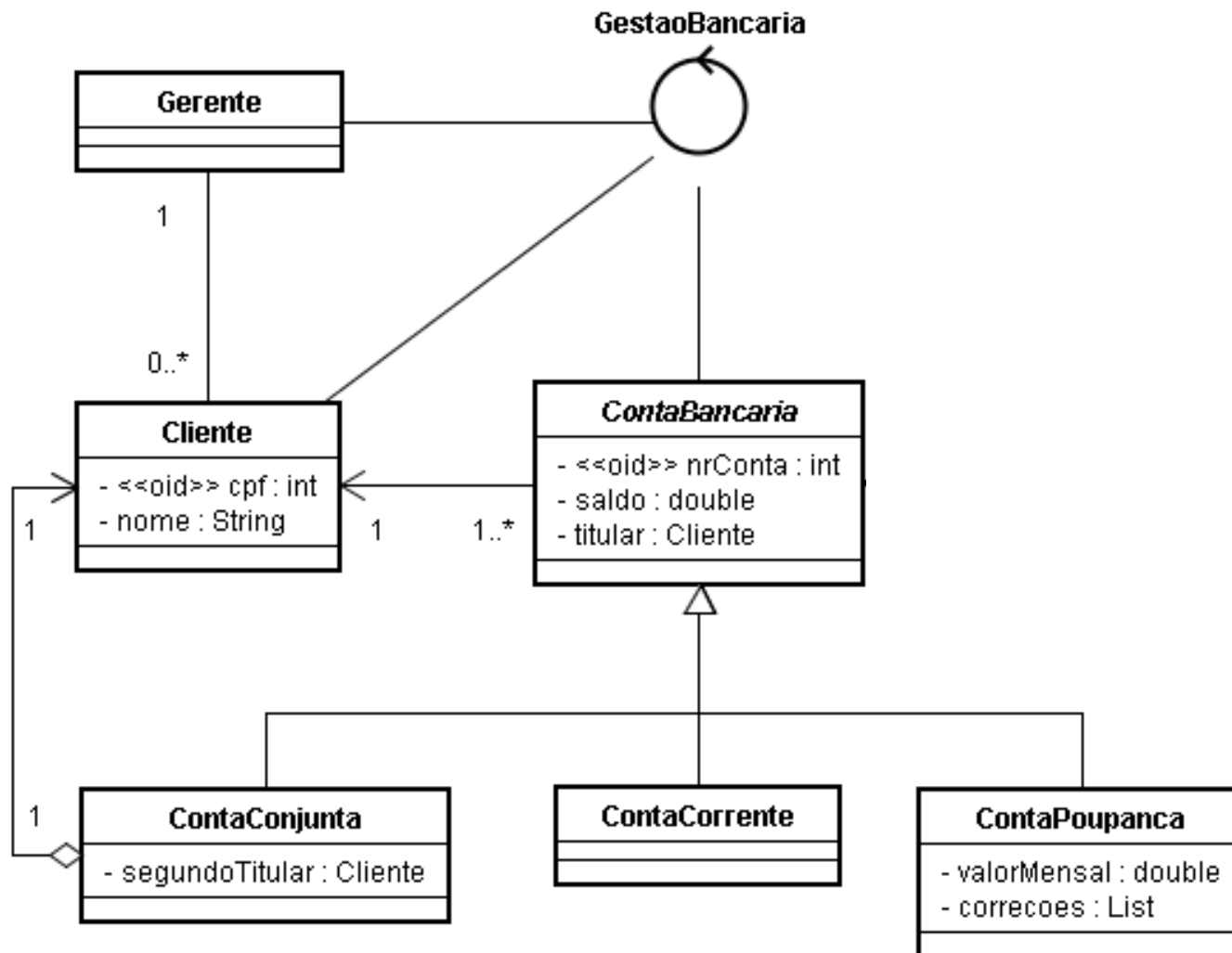
- Associação simples



Classe Controladora

- No diagrama de classe, é desejável apresentar uma classe que representa todo o sistema
 - Esta classe corresponde ao controlador-fachada
- É associado ao controlador todas as classes que são ditas como independentes
 - São conceitos que podem ser compreendidos sem estar associado a outros
 - Usualmente equivalem ao cadastros do sistema

Diagrama de Classes



Modelo Conceitual

Análise de Sistemas e
Requisitos de Software II

Aula 8

Allan Rodrigo Leite