

## Análise Orientado a Objetos

### Diagramas de casos de uso, classes e atividades

Ma. Vanessa Matias Leite

1

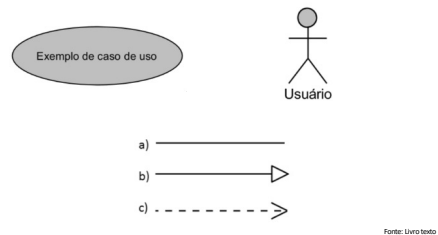
- Unidade de Ensino: 02
- Competência da Unidade: Compreender os diagramas de casos de uso, classes e atividades
- Resumo: Entender a aplicação e a modelagem dos diagramas de casos de uso, classes e atividades
- Palavras-chave: Diagramas de casos de uso; diagrama de classes; diagrama de atividades; UML
- Título da Teleaula: Diagramas de casos de uso, classes e atividades
- Teleaula nº: 02

2

## Modelagem de Casos de uso

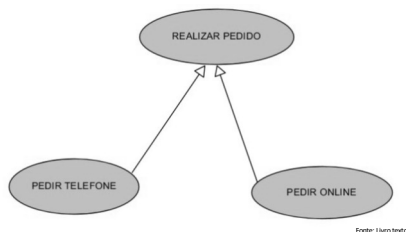
3

### Diagrama de casos de uso



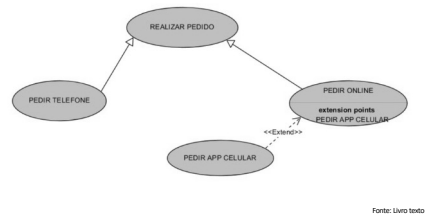
4

### Diagrama de casos de uso



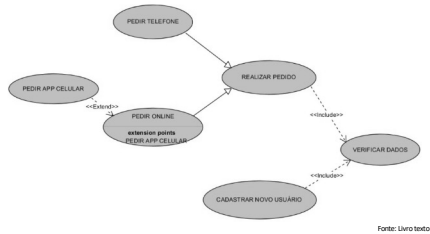
5

### Diagrama de casos de uso

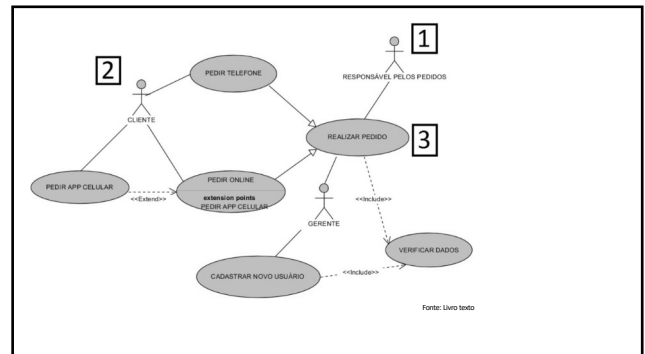


6

## Diagrama de casos de uso

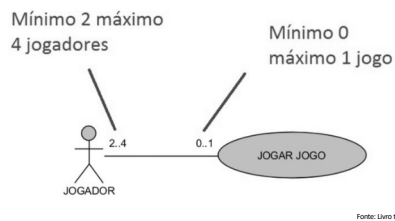


7



8

## Multiplicidade



9

## Diagrama de Casos de Uso - Prática

- Criar um caso de uso para sistema de controle bancário;
- Abrir conta comum, conta especial e conta poupança;
- Emitir saldo;
- Realizar saque e depósito;
- Encerrar a conta;

10

## Diagrama de Classes

11

## Conceitos básicos de orientação a objetos

- Atributos;
- Métodos;
- Classes;
- Objeto;

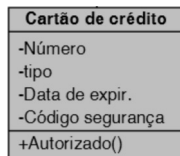
Cartão de crédito
-Número
-tipo
-Data de expir.
-Código segurança
+Autorizado()

Fonte: Livro texto

12

### Relacionamento do tipo encapsulamento

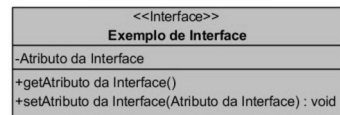
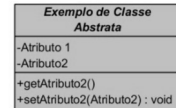
- Public (+)
- Private (-)
- Protected (#)



Fonte: Livro texto

13

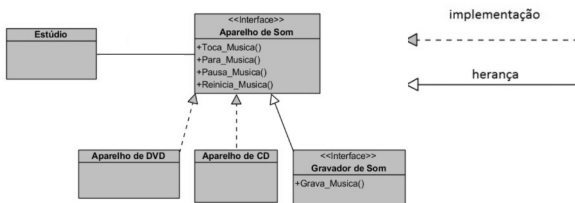
### Classe abstrata e interfaces



Fonte: Livro texto

14

### Tipo de relacionamento



Fonte: Livro texto

15

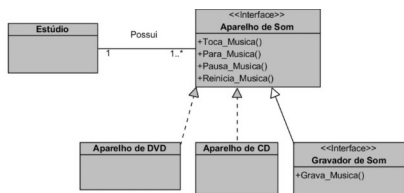
### Relacionamento de multiplicidade

- $n..m$  : significa n para m instâncias entre classes;
- $n..*$  : quando se utiliza o asterisco indica-se "muitos";
- 1 : exatamente uma instância associada.
- $1..*$  – uma ou mais instâncias que podem estar relacionadas.



16

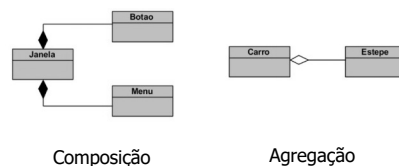
### Relacionamento de multiplicidade



Fonte: Livro texto

17

### Relacionamento: composição e agregação

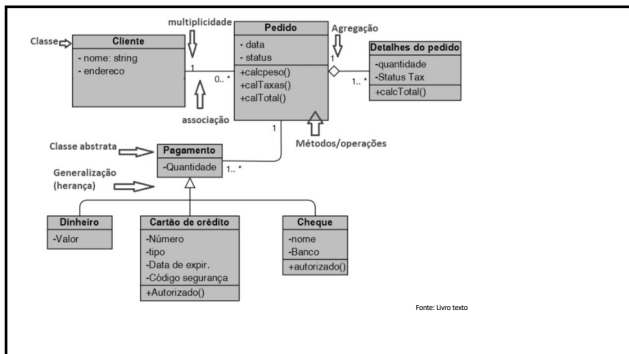


Composição

Agregação

Fonte: Livro texto

18



19

## Estereótipos

<<boundary>>: representa a interface do sistema;



<<control>>: controle da aplicação, serve de intermediário entre as classes definidas como <<boundary>> e <<entity>>

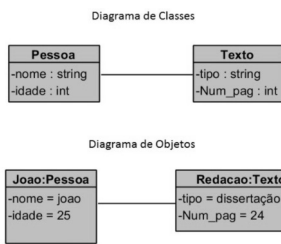


<<entity>>: classe entidade, contém informações geradas ou recebidas pelo sistema;



20

## Diagrama de Objetos



21

## Diagrama de Classes

22

- Diagrama de classes para um sistema bancário;
- Conta comum que deve ter alguns atributos: número da conta, data de abertura, senha, situação, saldo ...
- Essa classe também executará algumas ações: abrir conta, consultar conta, validar senha, verificar saldo ...
- Além da conta comum, é possível ter conta poupança e conta especial, que possuem os mesmos atributos e ações da conta comum, além dos seus próprios.

23

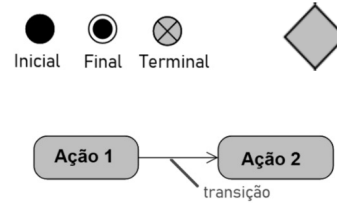
- Continuar o diagrama de classes da Conta bancário;
- Classe pessoa: armazena as informações gerais do cliente;
- Tanto pessoas físicas e jurídicas podem ter contas bancárias;
- Uma pessoa pode ter várias contas;
- Criar uma classe responsável por armazenar as transações ocorridas nas contas, por exemplo, tipo do movimento, data, hora, valor movimentado.

24

## Diagrama de Atividades

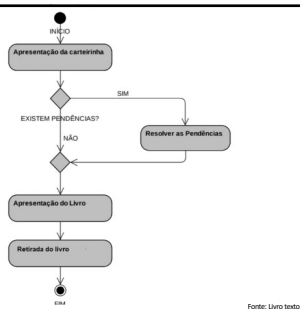
25

### Diagrama de atividades



Fonte: Livro texto

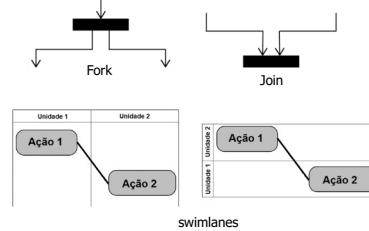
26



Fonte: Livro texto

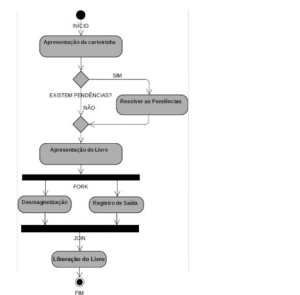
27

### Diagrama de atividades



swimlanes

28



Fonte: Livro texto

29

### Diagrama de Atividades- Prática

Criar um diagrama de atividades para "Consultar Saldo";

- Validar a conta;
- Validar a senha;
- Consultar e apresentar o saldo;

30

## Recapitulando

31

## Recapitulando

- Diagrama de caso de uso;
- Diagrama de classes;
- Diagrama de atividades;

32