



CES-22 - Programação Orientada a Objetos

Documentação LAb 02 (Livraria)

Luís Henrique Zuin Ruiz

COMP 24

Professores

Dr. Edgar Yano

Dra. Karla Donato Fook

Instituto Tecnológico de Aeronáutica – ITA

Departamento de Computação

No quesito de armazenamento das informações, foram utilizados 4 bancos de dados: um de clientes, outro de autores, outro de livros e outro de compras. Inicialmente, essas informações são armazenadas dentro de uma lista. Uma implementação inicial para os métodos de inserção, alteração, exclusão e busca também foi feita. Nesse sentido, duas operações de busca foram criadas para o *database* de livros, permitindo que uma obra seja procurada por seu nome ou por seu autor.

Nesse contexto, optou-se por implementar as classes de Autor e Cliente como subclasses de Pessoa, uma vez que as informações de nome e email são compartilhadas.

Ademais, pensando no quesito de modificação futura do código, o imposto foi implementado como uma classe à parte, respeitando os princípios SOLID. O cálculo do imposto, por sua vez, é feito levando em conta a categoria na qual o produto se encaixa, que precisa respeitar uma regra de nomenclatura característica do estabelecimento.

Não obstante, no quesito de representação de um livro dentro do *software*, considerou-se que a classe Livro é uma subclasse da classe produto, o que permitirá a expansão do modelo de negócios do estabelecimento futuramente. Além disso, um dos parâmetros da classe Livro é um título, que armazena todas as informações que são comuns a todas as edições de um livro, como nome, gênero e autor. A estruturação completa das classes da aplicação pode ser visualizada no diagrama de classes UML abaixo:

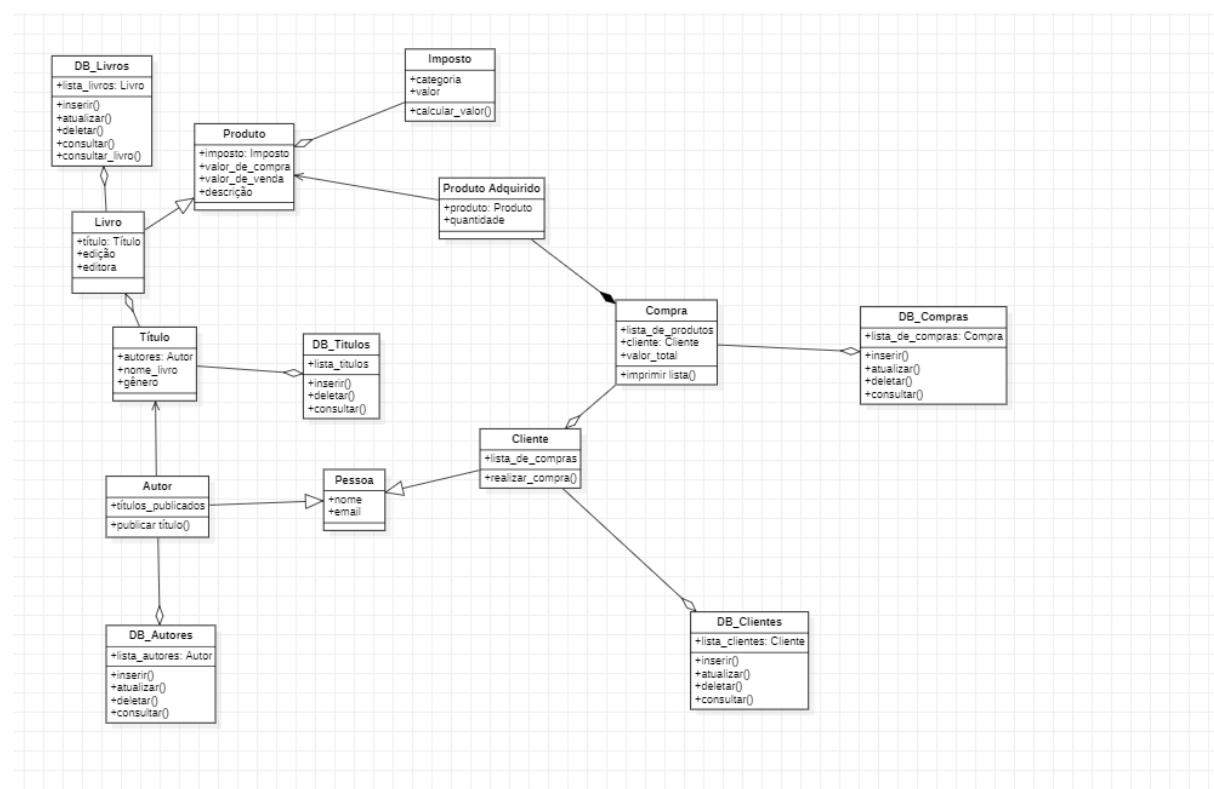


Figura 1: Diagrama de classes UML do projeto.

Para sua implementação, optou-se pela utilização apenas de um *backend*, de forma que as inserções de livros, autores, títulos e clientes são feitas no próprio código, através da função *main*. Com isso, a ideia é que a aplicação seja transformada em uma API e que uma interface de interação com o usuário - atendente da livraria - seja implementada. Um exemplo do funcionamento atual do código pode ser visto abaixo:

```
1 import implementation
2 DB_livros = implementation.DBLivros()
3 DB_titulos = implementation.DBTitulos()
4 DB_compras = implementation.DBCompras()
5 DB_autores = implementation.DBAutores()
6 DB_clientes = implementation.DBClientes()
7
8 DB_autores.inserir("Rick Riordan", "rick@gmail.com", []) #cadastro de autor
9 DB_titulos.inserir("Rick Riordan", "Percy Jackson", implementation.generos[1]) #cadastro de titulo
10 DB_livros.inserir("Percy Jackson", "Rick Riordan", implementation.generos[1], 3, "Intrínseca", 15, 30) #cadastro de livro
11 DB_livros.inserir("Percy Jackson", "Rick Riordan", implementation.generos[1], 4, "Intrínseca", 20, 35) #cadastro de livro
12 DB_livros.inserir("O nome do Vento", "Patrick Rothfuss", implementation.generos[1], 4, "Arquero", 20, 40) #cadastro de livro, titulo e autor
13 DB_livros.inserir("O temor do Sábio", "Patrick Rothfuss", implementation.generos[1], 3, "Arquero", 20, 40) #cadastro de livro
14 DB_autores.atualizar("Patrick Rothfuss", "patrick_rtf@gmail.com") #atualização de autor
15 DB_livros.inserir("A música do silêncio", "Patrick Rothfuss", implementation.generos[1], 2, "Arquero", 15, 25) #cadastro de livro
16 DB_livros.inserir("A crônica dos Kane", "Rick Riordan", implementation.generos[1], 1, "Intrínseca", 15, 30) #cadastro de livro
17 DB_clientes.inserir("Luis Zuin", "luiszuin@gmail.com") #cadastro de cliente
18 DB_clientes.inserir("Ana Laura Souza", "anasouza@gmail.com") #cadastro de cliente
19 DB_clientes.consultar("Luis Zuin").realizar_compra([DB_livros.consultar_livro("Percy Jackson", 4, "Intrínseca"), DB_livros.consultar_livro("O nome do Vento", 4, "Arquero")], [1, 1]) #realiza compra
20 print(DB_livros.consultar("Patrick Rothfuss")) #consulta livros por autor
21 print(DB_livros.consultar("Rick Riordan")) #consulta livros por autor
22 print(DB_clientes.consultar("Luis Zuin").lista_de_compras[0].valor_total) #consulta ao valor total de compra
23 DB_clientes.consultar("Luis Zuin").lista_de_compras[0].imprimir_lista() #imprimir pedido
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
2: \Users\Luis Zuin\anaconda3\python.exe" "D:/ITA/5° Semestre/CES-22/Livraria/main.py"
O nome do Vento', 'O temor do Sábio', 'A música do silêncio']
Percy Jackson', 'Percy Jackson', 'A crônica dos Kane']

Percy Jackson (1) - 35
nome do Vento (1) - 40

Process finished with exit code 0
```

Figura 2: Exemplo de funcionamento da aplicação criada, utilizando a própria IDE de Python como plataforma de teste.