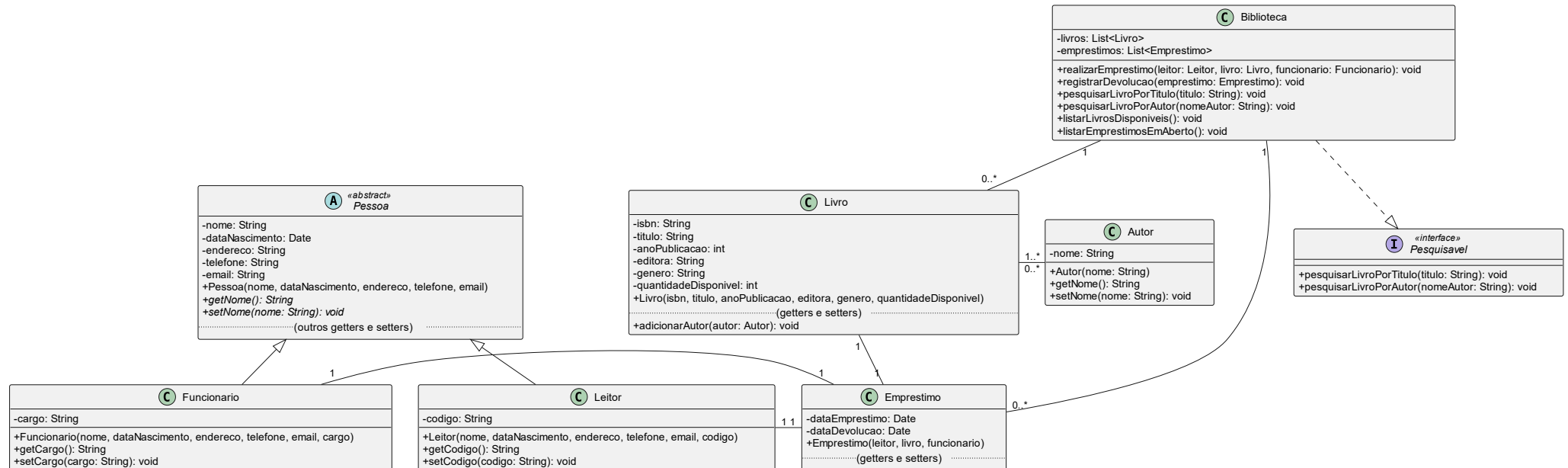


Anexo 1

Diagrama de Classe:



Enunciado:

Você foi contratado para desenvolver um sistema de gerenciamento de biblioteca em Java, utilizando os conceitos de orientação a objetos que aprendemos. O sistema deve permitir o cadastro de livros, autores, leitores e funcionários, além de realizar empréstimos de livros, registrar devoluções e gerar relatórios básicos.

Requisitos:

1. Classes e Interfaces:

- **Pessoa** (abstrata): Uma classe base para representar pessoas (leitores e funcionários).
 - Atributos: nome (String), dataNascimento (Date), endereco (String), telefone (String), email (String).
 - Métodos: getters e setters para todos os atributos, construtor que recebe nome, dataNascimento, endereco, telefone e email.
- **Leitor**: Representa um leitor.
 - Atributos: codigo (String).
 - Métodos: getters e setters para codigo, construtor que recebe nome, dataNascimento, endereco, telefone, email e codigo.
- **Funcionario**: Representa um funcionário.
 - Atributos: cargo (String).
 - Métodos: getters e setters para cargo, construtor que recebe nome, dataNascimento, endereco, telefone, email e cargo.
- **Livro**: Representa um livro.
 - Atributos: isbn (String), titulo (String), autores (List<Autor>), anoPublicacao (int), editora (String), genero (String), quantidadeDisponivel (int).
 - Métodos: getters e setters para todos os atributos, construtor que recebe isbn, titulo, autores, anoPublicacao, editora, genero e quantidadeDisponivel.
- **Autor**: Representa um autor.
 - Atributos: nome (String).
 - Métodos: getters e setters para nome, construtor que recebe nome.
- **Emprestimo**: Representa um empréstimo de livro.
 - Atributos: leitor (Leitor), livro (Livro), dataEmprestimo (Date), dataDevolucao (Date), funcionario (Funcionario).
 - Métodos: getters e setters para todos os atributos, construtor que recebe leitor, livro e funcionario.
- **Biblioteca**: Responsável por gerenciar os empréstimos e devoluções de livros.
 - Atributos: livros (List<Livro>), emprestimos (List<Emprestimo>).
 - Métodos: realizarEmprestimo(Leitor leitor, Livro livro, Funcionario funcionario), registrarDevolucao(Emprestimo emprestimo), pesquisarLivroPorTitulo(String titulo), pesquisarLivroPorAutor(String nomeAutor), listarLivrosDisponiveis(), listarEmprestimosEmAberto().
- Crie a seguinte interface:
 - **Pesquisavel**: Define os métodos pesquisarLivroPorTitulo(String titulo) e pesquisarLivroPorAutor(String nomeAutor).

2. Relacionamentos:

- **Herança**:
 - Leitor e Funcionario herdam da classe abstrata Pessoa.
- **Associação**:
 - Biblioteca possui uma lista de objetos Livro e uma lista de objetos Emprestimo.
 - Livro possui uma lista de objetos Autor.

- Empréstimo possui associações com Leitor, Livro e Funcionario.

3. Funcionalidades Básicas:

- **Empréstimo:** A classe Biblioteca deve ser capaz de realizar empréstimos de livros para leitores, verificando se o livro está disponível e se o leitor não possui outros empréstimos em aberto.
- **Devolução:** A classe Biblioteca deve permitir registrar a devolução de livros emprestados, atualizando a disponibilidade do livro e o status do empréstimo.
- **Pesquisa:** A classe Biblioteca deve implementar a interface Pesquisavel e fornecer métodos para pesquisar livros por título e por autor.
- **Listagens:** A classe Biblioteca deve fornecer métodos para listar livros disponíveis e empréstimos em aberto.

Exemplo de Uso para os testes:

Agora que você implementou as classes e interfaces do sistema de gerenciamento de biblioteca, é hora de testá-lo! Crie uma classe Main e siga os passos abaixo:

1. Crie Objetos:

- Crie objetos para representar livros (Livro), autores (Autor), leitores (Leitor) e funcionários (Funcionario), utilizando os construtores das classes.
- Crie um objeto Biblioteca.

2. Associe:

- Associe os autores aos livros que eles escreveram, utilizando o método add da lista de autores da classe Livro.
- Adicione os livros à biblioteca, utilizando o método add da lista de livros da classe Biblioteca.

3. Realize Empréstimos:

- Utilize o método realizarEmprestimo da Biblioteca para realizar empréstimos de livros para leitores.

4. Registre Devoluções:

- Utilize o método registrarDevolucao da Biblioteca para registrar a devolução de livros emprestados.

5. Realize Pesquisas:

- Utilize os métodos pesquisarLivroPorTitulo e pesquisarLivroPorAutor da Biblioteca para pesquisar livros.

6. Realize Listagens:

- Utilize os métodos listarLivrosDisponiveis e listarEmprestimosEmAberto da Biblioteca para listar livros e empréstimos.

7. Exiba os Resultados:

- Imprima no console os resultados das consultas realizadas conforme exemplo abaixo:

run:

Empréstimo realizado com sucesso!

Empréstimo realizado com sucesso!

Devolução registrada com sucesso!

Pesquisando livro por título 'Harry Potter e a Pedra Filosofal':

ISBN: 9780439554930

Título: Harry Potter e a Pedra Filosofal

Autores: J.K. Rowling

Ano de Publicação: 1997

Editora: Bloomsbury

Gênero: Fantasia

Quantidade Disponível: 5

Pesquisando livro por autor 'Machado de Assis':

ISBN: 9788525048682

Título: Dom Casmurro

Autores: Machado de Assis

Ano de Publicação: 1899

Editora: Companhia das Letras

Gênero: Romance

Quantidade Disponível: 2

Livros disponíveis:

ISBN: 9780439554930

Título: Harry Potter e a Pedra Filosofal

Autores: J.K. Rowling

Ano de Publicação: 1997

Editora: Bloomsbury

Gênero: Fantasia

Quantidade Disponível: 5

ISBN: 9780553103540

Título: A Guerra dos Tronos

Autores: George R.R. Martin

Ano de Publicação: 1996

Editora: Bantam Spectra

Gênero: Fantasia

Quantidade Disponível: 2

ISBN: 9788525048682

Título: Dom Casmurro

Autores: Machado de Assis

Ano de Publicação: 1899

Editora: Companhia das Letras

Gênero: Romance

Quantidade Disponível: 2

Empréstimos em aberto:

Emprestimo{Leitor=Maria Souza, Livro=A Guerra dos Tronos, Data de Empréstimo=04/06/2024,
Data de Devolução=Em aberto, Funcionário=Pedro Santos}

BUILD SUCCESSFUL (total time: 0 seconds)

Dicas:

- Utilize ArrayList para implementar as listas de livros, autores e empréstimos.
- Implemente a lógica dos métodos de acordo com os requisitos, incluindo tratamento de erros (por exemplo, verificar se o livro está disponível antes de tentar emprestá-lo).

Bom trabalho!!!
Prof. Alexandre Gomes