



**UNIVERSIDADE FEDERAL DE OURO PRETO
CAMPUS MORRO DO CRUZEIRO - OURO PRETO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

TRABALHO PRÁTICO II

BRUNA CRISTINA GONCALVES - 20.2.4123

ELISA ALVES VELOSO - 20.2.4006

IAN MARTINS ARANTES - 21.2.4178

LAVÍNIA FONSECA PEREIRA - 19.1.4170

LUCAS GOMES DOS SANTOS - 20.1.4108

BRUNA CRISTINA GONCALVES - 20.2.4123

ELISA ALVES VELOSO - 20.2.4006

IAN MARTINS ARANTES - 21.2.4178

LAVÍNIA FONSECA PEREIRA - 19.1.4170

LUCAS GOMES DOS SANTOS - 20.1.4108

TRABALHO PRÁTICO II

Trabalho prático apresentado à disciplina de
Programação Orientada a Objetos do curso
da Ciência da Computação

Professor:
Guillermo Camara Chavez

OURO PRETO, MG
2023

Conteúdo

1	Introdução	2
2	Objetivo Geral	2
3	Objetivos Específicos	2
4	Justificativa	3
5	Desenvolvimento	3
5.1	IDE Utilizada	3
5.2	Como operar o sistema	3
5.3	Unified Modeling Language (UML)	3
5.4	Conceitos utilizados	4
5.5	Estratégias utilizadas para o teste	5
5.6	Erros ocorridos	5
5.7	Dificuldades	5
5.8	O que não foi implementado	5
6	Implementação	6
6.1	DAO	6
6.2	Dados	6
6.3	Modelo	6
6.4	Principal	7
6.5	Telas	7
6.6	Exceções	7
7	Análise Geral	7
8	Impressões Gerais	8
9	Conclusão	8

1 Introdução

As bibliotecas sempre desempenharam um papel crucial na disseminação do conhecimento e na promoção da cultura e da educação. Com o avanço da tecnologia, tornou-se imperativo que essas instituições adotem soluções modernas para melhorar a eficiência de suas operações. Um dos aspectos essenciais na gestão bibliotecária é o empréstimo de livros, que demanda um sistema organizado e ágil para atender às necessidades dos usuários de maneira eficaz.

O desenvolvimento de um Sistema de Gerenciamento de Empréstimos de Livros em Bibliotecas representa uma resposta contemporânea aos desafios enfrentados por essas instituições. Tradicionalmente, a gestão manual de empréstimos resultava em dificuldades, como longos tempos de espera, erros de registro e dificuldade no rastreamento de livros. Além disso, a crescente demanda por serviços mais ágeis e convenientes por parte dos usuários modernos exige a adoção de soluções tecnológicas que otimizem esses processos.

Este relatório abordará o desenvolvimento e a implementação de um sistema automatizado de gerenciamento de empréstimos de livros em bibliotecas. Serão exploradas as funcionalidades-chave desse sistema, incluindo realizar o cadastro de funcionários, livros, usuários e empréstimos, além disso, deverá ser possível realizar o cadastro dos autores e também as categorias dos livros, como por exemplo, terror, ficção, infantil entre outros. Um livro poderá ser escrito por um ou mais autores e também poderá estar classificado como uma ou mais categorias. As operações de cadastro somente poderão ser realizadas por um usuário administrador identificado, o qual não precisa estar cadastrado no sistema, a operação de empréstimo somente poderá ser realizada por um funcionário a um usuário comum, o usuário comum poderá realizar a consulta de livros. . Além disso, será discutido sobre otimização do fluxo de trabalho dos bibliotecários e a melhoria da experiência do usuário.

2 Objetivo Geral

O objetivo principal deste projeto é desenvolver uma aplicação desktop utilizando a linguagem de programação Java, que possibilite o gerenciamento eficiente de empréstimos de livros em bibliotecas. A aplicação deve abranger desde o cadastro de funcionários, livros, usuários, autores e categorias, até a realização de empréstimos e consultas de livros disponíveis.

3 Objetivos Específicos

Implementar a estrutura da aplicação, distribuindo as classes em pacotes de acordo com suas funções, como Principal, Modelo, DAO e Dados. Desenvolver a interface de usuário para o login de administradores, funcionários e usuários comuns. Criar a funcionalidade de cadastro de funcionários, livros, usuários, autores e categorias. Implementar o sistema de empréstimo de livros, assegurando que apenas funcionários possam realizar essa operação. Criar a funcionalidade de consulta de livros disponíveis para os usuários comuns.

4 Justificativa

A digitalização de processos é essencial para a adaptação das instituições às demandas modernas. No contexto das bibliotecas, a aplicação proposta visa eliminar processos manuais morosos, propensos a erros e pouco eficientes. Ao desenvolver um sistema de gerenciamento de empréstimos de livros, facilitamos tanto a vida dos bibliotecários, ao automatizar suas tarefas, quanto a experiência dos usuários, que poderão acessar informações de forma mais rápida e intuitiva.

Além disso, a aplicação permite maior controle sobre as operações realizadas na biblioteca, com registros precisos e acesso restrito a determinadas funcionalidades, garantindo a segurança e a integridade dos dados.

5 Desenvolvimento

5.1 IDE Utilizada

O IDE (Ambiente de Desenvolvimento Integrado) utilizado para criar esse sistema foi o NetBeans. O NetBeans é uma IDE de código aberto, altamente popular para desenvolvimento de software Java. Ela oferece um conjunto de ferramentas e recursos para facilitar o desenvolvimento de aplicativos, incluindo aplicativos desktop com interfaces gráficas. Ao criar uma interface gráfica para o sistema de gerenciamento de empréstimos de livros, o desenvolvedor teria usado componentes visuais fornecidos pelo NetBeans, como botões, caixas de texto, tabelas e outros elementos. Isso permite criar uma interface do usuário (UI) do sistema de forma visual, arrastando e soltando esses componentes na tela.

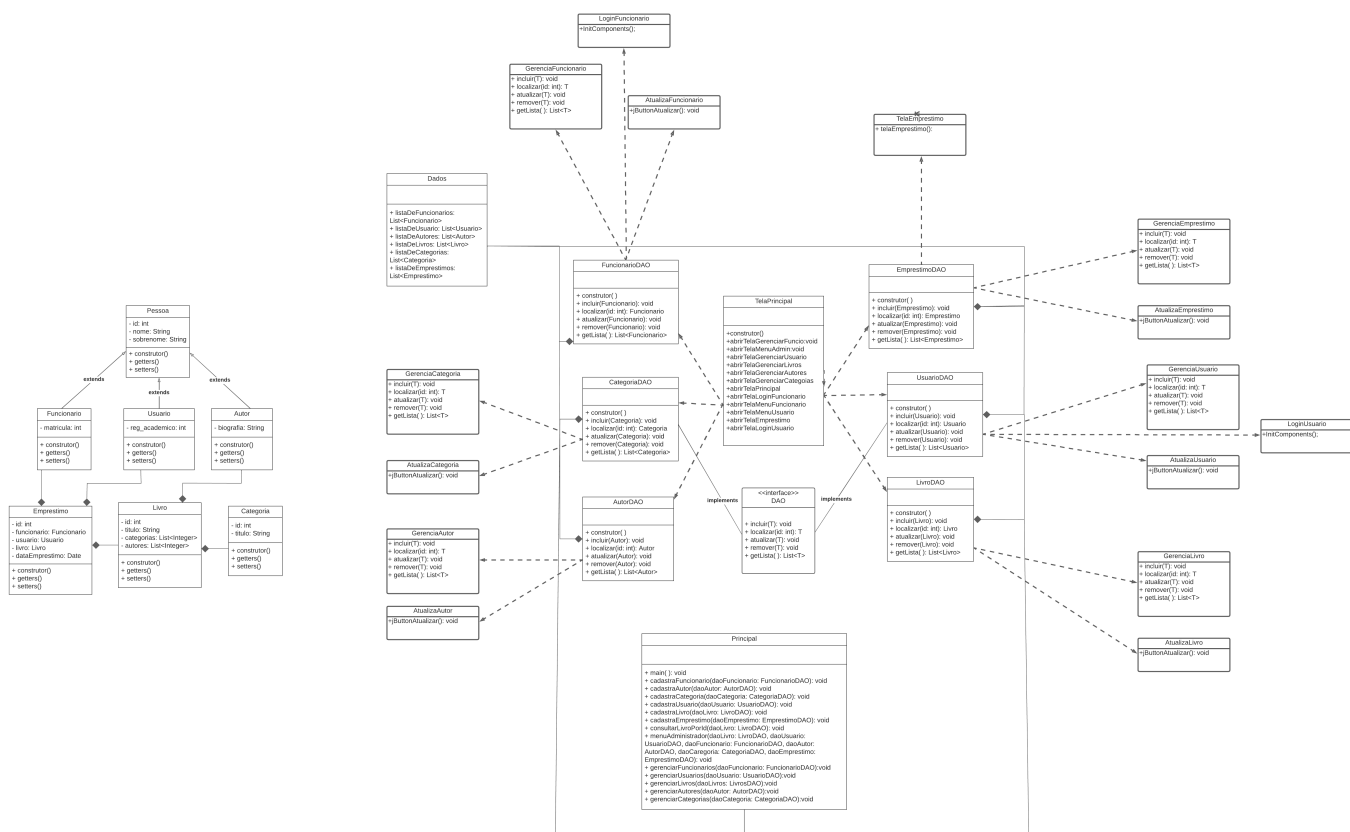
5.2 Como operar o sistema

Abra o NetBeans e carregue o projeto do sistema. Clique no botão "Executar" (geralmente representado por um ícone de seta verde) para iniciar o aplicativo. Dessa forma, abrirá a tela principal da interface gráfica do sistema de gerenciamento de empréstimos de livros.

5.3 Unified Modeling Language (UML)

Para proporcionar uma compreensão mais clara do diagrama UML, gostaria de compartilhar o link para o Lucidchart, onde criamos a representação da estrutura do sistema, incluindo as classes envolvidas e suas relações. Isso

permitirá uma visualização mais detalhada de como as classes estão organizadas e como estão interconectadas. O link vai ficar disponível junto a pasta compactada, devido a um problema de formatação do Overleaf que transforma em comentário quando utilizamos o símbolo por cento.



5.4 Conceitos utilizados

O trabalho envolveu a criação de uma aplicação desktop em Java utilizando o ambiente NetBeans, com foco no gerenciamento de empréstimos de livros em uma biblioteca. A disciplina de Programação Orientada a Objetos (POO) desempenhou um papel fundamental na concepção e implementação dessa aplicação, destacando vários conceitos chave: Classe e Objetos, encapsulamento, herança, polimorfismo, associações e relações, métodos e comportamentos, reutilização de código herança e interfaces.

Em suma, a disciplina de Programação Orientada a Objetos forneceu os conceitos e princípios necessários para a criação bem-sucedida da aplicação de gerenciamento de empréstimos de livros em uma biblioteca. Isso envolveu a modelagem de classes, definição de propriedades e comportamentos, uso de herança e interfaces, bem como a aplicação de boas práticas de programação orientada a objetos para criar uma aplicação coesa e modular.

5.5 Estratégias utilizadas para o teste

Foi possível desenvolver várias estratégias para garantir que a interface do usuário seja funcional, amigável e atenda aos requisitos. Aqui estão algumas estratégias utilizadas: Testes de Cadastro: Verifique se é possível cadastrar funcionários, livros, usuários e empréstimos corretamente. Validações de Entrada: Teste entradas válidas e inválidas nos campos. Fluxo de Empréstimo: Confirme que é possível realizar um empréstimo, verificando datas e informações associadas. Verificação de Livros: Avalie a capacidade de associar autores e categorias a livros, entre outras.

5.6 Erros ocorridos

Avaliando de maneira crítica o processo de implementação do projeto, é importante observar que ocorreram erros básicos durante o desenvolvimento, em parte devido à adaptação ao uso do ambiente NetBeans. Uma das principais complicações enfrentadas ao longo do processo foi a decisão de implementar integralmente a lógica do sistema antes de iniciar a criação da interface gráfica.

Em síntese, é fundamental reconhecer que a combinação entre a adaptação ao ambiente de desenvolvimento e a abordagem de implementação adotada geraram desafios substanciais, refletindo-se na integração da lógica e da interface gráfica. Identificar essas áreas problemáticas pode ser um ponto de partida valioso para aprimorar futuros projetos e evitar complicações similares.

5.7 Dificuldades

Encontramos desafios na transição do código implementado exclusivamente via terminal para a criação de interfaces gráficas. Inicialmente, desenvolvemos a lógica operacional do sistema através de comandos de terminal, priorizando a funcionalidade do software. Posteriormente, iniciamos o processo de criação das interfaces gráficas para melhorar a usabilidade do sistema. No entanto, essa migração apresentou dificuldades, uma vez que requereu uma adaptação significativa do código existente para a estrutura de interface gráfica. A abordagem de desenvolvimento baseada em terminal tende a se concentrar na lógica de linha de comando, enquanto as interfaces gráficas deram uma reorganização do código para incorporar elementos visuais interativos, como botões, campos de entrada, entre outros.

5.8 O que não foi implementado

Em resumo, o trabalho em questão foi concluído com êxito, demonstrando a completa implementação de todas as funcionalidades requeridas. A aplicação desktop desenvolvida em Java utilizando o ambiente NetBeans

abrangeu de maneira abrangente e precisa o gerenciamento de empréstimos de livros em uma biblioteca.

6 Implementação

A implementação é detalhada a seguir, em cada subseção, onde se apresenta de forma concisa a explicação correspondente a cada diretório.

A pasta "Source Packages" no NetBeans abriga o núcleo do código-fonte de um projeto, permitindo uma organização eficaz e modularização que são essenciais para o desenvolvimento de software bem estruturado.

6.1 DAO

A pasta "DAO" é uma abstração que separa a manipulação de dados da lógica de negócios. Isso torna a aplicação mais modular, facilita a manutenção e permite a troca da fonte de dados sem afetar a maior parte do código. Cada arquivo dentro da pasta "DAO" representa uma entidade ou uma tabela do banco de dados, e suas funções se concentram em executar operações específicas relacionadas a essas entidades.

6.2 Dados

A pasta "Dados" está contendo uma classe chamada "Dados", que contém listas estáticas para armazenar objetos de diferentes tipos de entidades relacionadas ao sistema de gerenciamento de empréstimos de livros em uma biblioteca.

6.3 Modelo

A pasta "Modelos" contém um conjunto de classes que representam diferentes entidades ou conceitos dentro do contexto de um sistema de gerenciamento de empréstimos de livros em uma biblioteca. Cada classe nessa pasta é projetada para encapsular os atributos e comportamentos associados a uma entidade específica, e inclui construtores, métodos getters e setters.

Cada uma dessas classes desempenham um papel importante na estrutura geral do sistema, permitindo a modelagem e o gerenciamento eficiente das informações relevantes para um sistema de gerenciamento de empréstimos de livros em uma biblioteca. Elas encapsulam os dados e comportamentos relacionados a cada entidade específica, seguindo princípios de orientação a objetos para manter o código organizado e modular. Os construtores, getters e setters são métodos essenciais para criar, acessar e modificar os atributos dessas classes de maneira controlada.

6.4 Principal

A pasta "Principal" desempenha o papel de controladora principal do sistema. Ela interage com as classes DAO para acessar e manipular dados, além de oferecer uma estrutura de menus que permite a diferentes tipos de usuários (administrador, funcionário e usuário) realizar diversas operações, como cadastrar, consultar e gerenciar informações relacionadas a livros, empréstimos, autores, categorias, funcionários e usuários.

Inicialmente, realizamos a implementação completa utilizando o ambiente NetBeans. Em seguida, desenvolvemos as interfaces conforme planejado. Devido a essa abordagem, a parte central da nossa implementação agora contém diversas verificações. Em outras palavras, o código foi construído para funcionar tanto através do terminal quanto por meio da interface gráfica.

6.5 Telas

A pasta "Telas" contém as implementações das interfaces gráficas do sistema de gerenciamento de biblioteca. Cada tela é projetada para se comunicar com o código subjacente, executando ações específicas em resposta às interações dos usuários. Isso permite que a interface gráfica e a lógica do sistema funcionem de maneira integrada, fornecendo aos usuários uma experiência interativa e eficiente ao utilizar o software.

6.6 Exceções

A pasta "excecoes" contém uma classe chamada "AutenticacaoException" que estende a classe "Exception". Essa classe é usada para criar uma exceção personalizada que será lançada quando ocorrer um erro de autenticação no sistema.

A classe "AutenticacaoException" tem um construtor que chama o construtor da classe base (Exception) e fornece uma mensagem de erro específica relacionada a credenciais inválidas. Essa mensagem será exibida quando a exceção for capturada e tratada no código.

7 Análise Geral

O desenvolvimento dessa aplicação requer a implementação de várias classes interconectadas para realizar as operações desejadas. A utilização de diferentes pacotes para organizar essas classes facilita a manutenção e escalabilidade do sistema. A criação de um menu de interações e a autenticação garantem a usabilidade da aplicação, enquanto as operações simuladas dispensam a necessidade de persistência de dados além da sessão.

8 Impressões Gerais

O desenvolvimento dessa aplicação envolve desafios interessantes, desde a criação de uma interface intuitiva até a definição de lógicas complexas para cada operação. A necessidade de autenticação e a distribuição de permissões agrega um aspecto de segurança crucial ao sistema. Além disso, a aplicação é uma oportunidade de aplicar conceitos de programação orientada a objetos e estruturar um projeto de software de maneira eficiente.

9 Conclusão

Em conclusão, o presente trabalho apresentou a criação e desenvolvimento de uma aplicação desktop utilizando a linguagem de programação Java, com o suporte do ambiente de desenvolvimento NetBeans. A aplicação foi projetada com o propósito de efetuar o gerenciamento de empréstimos de livros em um contexto bibliotecário.

Através desta prática, foram implementadas funcionalidades essenciais para um sistema de gerenciamento bibliotecário eficiente. O software permitiu o cadastro detalhado de funcionários, livros e usuários, bem como a gestão dos empréstimos realizados. Além disso, a aplicação ofereceu a capacidade de cadastrar autores e categorias, enriquecendo a base de dados e facilitando a organização dos materiais presentes na biblioteca.

A flexibilidade da aplicação se destacou ao permitir que um livro fosse associado a um ou vários autores, assim como pudesse ser classificado em uma ou mais categorias. Isso possibilitou uma catalogação mais abrangente e precisa, tornando mais fácil para os usuários encontrarem livros que se adequassem aos seus interesses.

O uso do NetBeans como ambiente de desenvolvimento facilitou a criação da interface de usuário de forma intuitiva, contribuindo para uma experiência amigável e eficaz para os usuários da aplicação.

Em síntese, este projeto demonstrou a aplicação prática dos conceitos de programação Java na construção de um sistema de gerenciamento de biblioteca, oferecendo funcionalidades de cadastro, busca e controle de empréstimos. Através do desenvolvimento deste software, foi possível compreender a importância da organização e categorização adequada dos materiais bibliotecários, contribuindo assim para uma gestão mais eficiente e uma experiência mais positiva tanto para os funcionários quanto para os usuários da biblioteca.