

SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL
ESCOLA TÉCNICA SENAI DENDEZEIROS
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS



LUCAS GONÇALVES FRANÇA BATISTA

BookBud: Sistema de gestão de biblioteca.

Salvador
2023

Lucas Gonçalves

BookBud: Sistema de gestão de biblioteca.

Trabalho da matéria de Desenvolvimento de Sistemas 2 apresentado à professora Ítala Macedo, docente da Escola Técnica SENAI DENDEZEIRO, como requisito parcial para concluir a matéria. .

Orientador:

Salvador
2023

LISTAS DE ABREVIATURAS E SIGLAS

TERMO	DEFINIÇÃO
DAO	Data Access Object, em português Objeto de Acesso de Dados, é um padrão de aplicação que utiliza a persistência de dados, onde há a separação das regras de acesso ao banco de dados das regras de negócios.
DTO	Data Transfer Object, em português Objeto de Transferência de dados. : Representa o "Model". São objetos que definem a estrutura de dados.
MVC	Model View Controller, em português Modelo-Visão-Controle é um padrão de arquitetura de software que desacopla a interface do usuário (exibição/visão), os dados (o modelo) e a lógica do sistema (controlador).
RDBMS	Relational Database Management System - Sistema de Gerenciamento de Banco de Dados
SQL	Structured Query Language, em português linguagem de consulta estruturada, é uma linguagem desenvolvida para gerenciamento de bancos de dados relacionais.

LISTA DE FIGURAS

Figura 01 - Caso de uso 8
Figura 02 - Estrutura de pasta do projeto 11
Figura 03 - Modelo conceitual do banco de dados 12
Figura 04: Modelo lógico do banco de dados 12

LISTA DE TABELAS

Tabela 01 - Requisitos funcionais	8
Tabela 02 - Requisitos não funcionais	8
Tabela 03 - Regras de negócios	9

SUMÁRIO

LISTAS DE ABREVIATURAS E SIGLAS.....	3
LISTA DE FIGURAS.....	4
LISTA DE TABELAS.....	5
SUMÁRIO.....	5
1. INTRODUÇÃO.....	7
1.1. PROBLEMA.....	7
1.2. JUSTIFICATIVA.....	7
1.3. OBJETIVOS.....	7
1.4. PÚBLICO ALVO.....	7
2. LEVANTAMENTO DE REQUISITOS.....	8
2.1. REQUISITOS FUNCIONAIS.....	8
2.1.1. CASOS DE USO.....	8
2.2. REQUISITOS NÃO FUNCIONAIS.....	8
2.3. REGRA DE NEGÓCIOS.....	9
3. TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO DO SISTEMA.....	10
4. ESTRUTURA PARA DESENVOLVIMENTO DO SISTEMA.....	11
4.1. ESTRUTURA ANALÍTICA DO PROJETO (EAP).....	12
4.2. CRONOGRAMA.....	12
4.3. DIAGRAMA DE CASO DE USO.....	12
4.4. DIAGRAMA DE ATIVIDADE.....	12
5. ESTRUTURA DE BANCO DE DADOS.....	13
5.1. MYSQL.....	13
5.2. MODELO CONCEITUAL.....	13
5.3. MODELO LÓGICO.....	13
6. SEGURANÇA.....	14
CONCLUSÃO.....	15
O sistema de gerenciamento de livros atende às necessidades básicas de organizações e indivíduos que desejam manter um catálogo de livros. Sua facilidade de uso, segurança e desempenho o tornam uma ferramenta valiosa para a gestão de livros.....	
REFERÊNCIAS.....	16
Padrão ASP.NET MVC	
https://dotnet.microsoft.com/pt-br/apps/aspnet/mvc#:~:text=Model%20View%20Controller%20(MVC),obter%20uma%20separa%C3%A7%C3%A3o%20das%20preocupa%C3%A7%C3%B5es	
Diferença entre os patterns PO, POJO, BO, DTO e VO.....	

1. INTRODUÇÃO

O sistema de gerenciamento de livros foi desenvolvido com o objetivo de organizar, manter e gerir um catálogo de livros.

1.1. PROBLEMA

A falta de um sistema automatizado que permite gerenciar, buscar, atualizar e remover livros de um catálogo resulta em ineficiências, erros e demoras na obtenção de informações relevantes sobre os livros disponíveis.

1.2. JUSTIFICATIVA

Um sistema organizado facilita a administração de livros, a rápida recuperação de informações e a manutenção eficaz dos registros, evitando redundâncias e possíveis perdas.

1.3. OBJETIVOS

Desenvolver um sistema de gerenciamento que permita:

- Adicionar livros ao catálogo.
- Listar todos os livros disponíveis.
- Atualizar informações dos livros.
- Deletar livros.

1.4. PÚBLICO ALVO

Bibliotecários, estudantes, professores e qualquer indivíduo ou organização interessada em manter um catálogo organizado de livros.

2. LEVANTAMENTO DE REQUISITOS

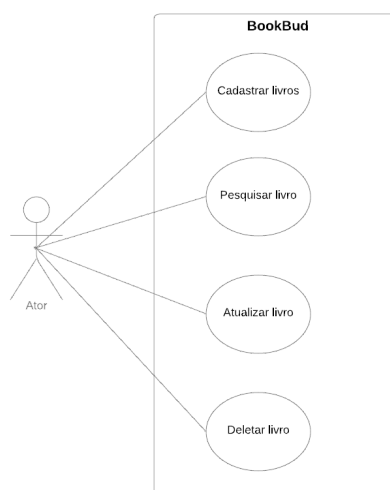
2.1. REQUISITOS FUNCIONAIS

Tabela 01 - Requisitos funcionais

RF 01	Registro de livros:: Permitir o cadastro de livros com título, autor, ano e ISBN.
RF 02	Listagem de livros: Exibir uma lista de todos os livros cadastrados.
RF 03	Pesquisa de livros: Permitir pesquisa de livros por ISBN, autor ou título.
RF 04	Atualização de livro: Permitir edição das informações de um livro.
RF 05	Exclusão de livros: Permitir a remoção de um livro usando seu ISBN.
RF 06	Autenticação de usuário: Possuir tela de login para acesso ao sistema.

2.1.1. CASOS DE USO

Figura 1 - Caso de uso



Fonte: Produção própria

2.2. REQUISITOS NÃO FUNCIONAIS

Tabela 02 - Requisitos não funcionais

RNF 01	A interface do sistema deve ser intuitiva e amigável.
RNF 02	O sistema deve proporcionar um desempenho rápido.
RNF 03	O sistema deve ter medidas robustas contra invasões.
RNF 04	Deve haver backup automático dos dados em intervalos regulares.

2.3. REGRA DE NEGÓCIOS

Tabela 03 - Regras de negócios

RN 01	Unicidade do ISBN	Cada livro no sistema "BookMaster" deve ter um ISBN único. Isso garante que cada livro seja identificável de forma exclusiva, evitando confusões ou conflitos de dados.
RN 02	Campos Obrigatórios	Ao cadastrar ou atualizar um livro no sistema, os campos de título, autor e ano de publicação são obrigatórios. Isso garante que os usuários tenham informações mínimas necessárias sobre cada livro.

3. TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO DO SISTEMA

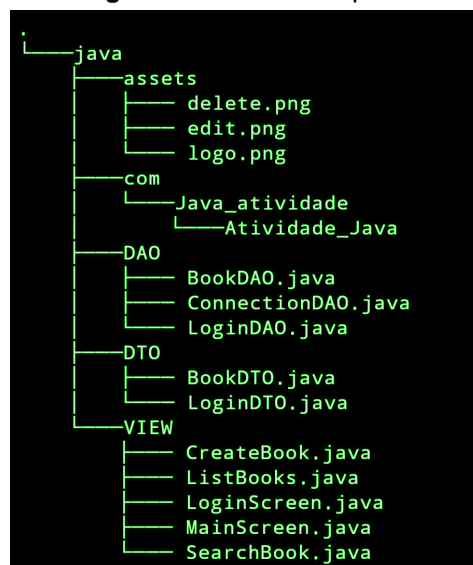
- Variação do padrão de Arquitetura: MVC (Model-View-Controller).
- Linguagem de programação: Java.
- Banco de dados: MySQL.
- Interface: Swing.

4. ESTRUTURA PARA DESENVOLVIMENTO DO SISTEMA

O sistema foi construído usando uma variação do padrão arquitetônico MVC (Model View Controller) . Em vez de um padrão MVC tradicional, optou-se por separar a lógica de acesso aos dados (leitura e gravação no banco de dados) em seus próprios objetos (DAOs). Esta escolha foi feita para isolar as responsabilidades, tornar o código mais organizado e reutilizável.

Abaixo está a estrutura de pastas do projeto:

Figura 2: Estrutura de pasta



Fonte: Produção própria

- **DAO (Data Access Object):** Atua como uma extensão do "Model", sendo responsável pela leitura e gravação de um DTO em um banco de dados ou outra fonte de persistência., ou seja, os objetos DAO estão atuando como Controller.
- **DTO (Data Transfer Object):** Representa o "Model". São objetos que definem a estrutura de dados.
- **VIEW:** Representa a visualização dos dados e onde a interação do usuário ocorre.

5. ESTRUTURA DE BANCO DE DADOS

O banco de dados escolhido para o BookBud foi o MySQL.

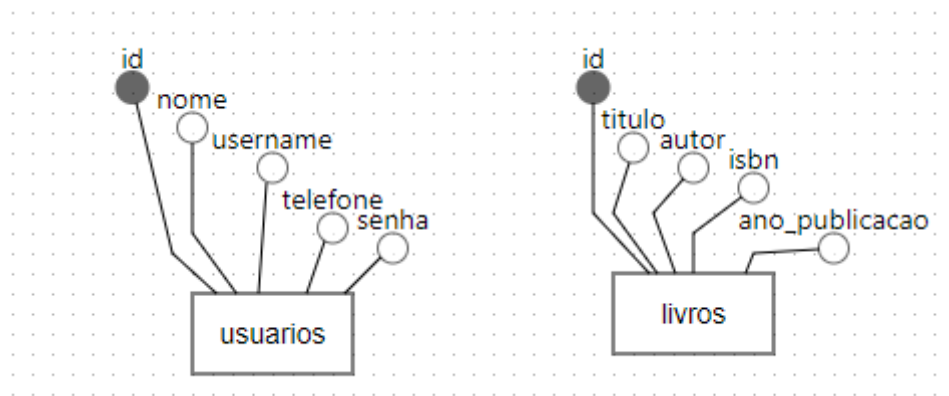
5.1. MYSQL

O MySQL é um sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto, que utiliza a linguagem SQL como interface. A escolha deste sistema de gerenciamento de banco de dados para o BookBud foi motivada pela capacidade técnica do desenvolvedor da aplicação BookBud.

5.2. MODELO CONCEITUAL

Modelo conceitual é uma representação de alto nível do esquema do banco de dados, que descreve a estrutura e as relações entre as diferentes entidades.

Figura 03: Modelo conceitual do banco de dados

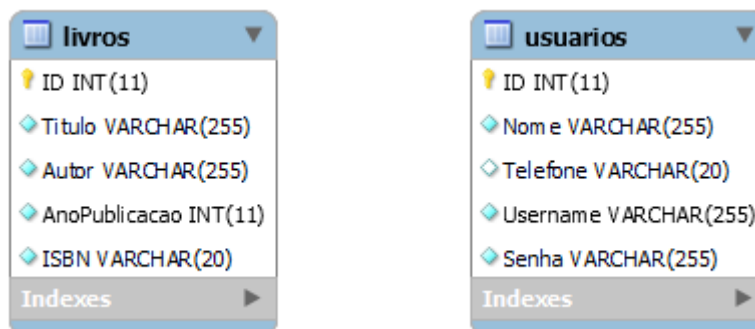


Fonte: Produção própria

5.3. MODELO LÓGICO

Após o modelo conceitual, passamos para a fase de criação do modelo lógico. Esta fase envolve a transformação das entidades e relações do modelo conceitual em tabelas, com suas chaves primárias e seus demais atributos.

Figura 04: Modelo lógico do banco de dados



Fonte: Produção própria

6. SEGURANÇA

- Sistema de login para acesso.
- Criptografia de senhas.

.

CONCLUSÃO

O sistema de gerenciamento de livros atende às necessidades básicas de organizações e indivíduos que desejam manter um catálogo de livros. Sua facilidade de uso, segurança e desempenho o tornam uma ferramenta valiosa para a gestão de livros.

REFERÊNCIAS

MICROSOFT. **Padrão ASP.NET MVC**. Disponível em: <[https://dotnet.microsoft.com/pt-br/apps/aspnet/mvc#:~:text=Model%20View%20Controller%20\(MVC\),obter%20uma%20separa%C3%A7%C3%A3o%20das%20preocupa%C3%A7%C3%B5es](https://dotnet.microsoft.com/pt-br/apps/aspnet/mvc#:~:text=Model%20View%20Controller%20(MVC),obter%20uma%20separa%C3%A7%C3%A3o%20das%20preocupa%C3%A7%C3%B5es)>. Acesso em: [data de acesso, por exemplo: 10 out. 2023].

RAMALHO, José. **Diferença entre os patterns PO, POJO, BO, DTO e VO**. DevMedia, [s.d.]. Disponível em: <<https://www.devmedia.com.br/diferenca-entre-os-patterns-po-pojo-bo-dto-e-vo/28162>>. Acesso em: [data de acesso].

WIKIPEDIA. **DAO - Objeto de Acesso de dados**. Disponível em: <https://pt.wikipedia.org/wiki/Objeto_de_acesso_a_dados>. Acesso em: [data de acesso].

HOSTINGER. **O que é MySQL**. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-mysql>>. Acesso em: [data de acesso].