

# Sistema Fullstack de Sistema de avaliação de Livros

---

Integração de Tecnologias Modernas para Gestão Digital

---

FERNANDA FERREIRA DOS REIS | FERNANDO OSÓRIO DE OLIVEIRA

GUILHERME DE ANDRADE BARBOSA | ÍTAO RODRIGUES NERI

JEYSSÉ VITORIA CAVALCANTE SILVA

**Curso de Engenharia de Software**

Professor: Jefferson Salomão Rodrigues

Brasília, Distrito Federal | 2025

# Objetivo e Escopo do Projeto

## OBJETIVO GERAL

Desenvolver um sistema fullstack de gerenciamento de biblioteca digital, utilizando React no frontend, Java com Spring Boot no backend, MySQL para armazenamento de usuários e autenticação, e MongoDB para gerenciamento de livros, empréstimos e demais registros operacionais, com foco em desempenho, escalabilidade e boa experiência do usuário.

## OBJETIVOS ESPECÍFICOS

- 1 Projetar uma interface web interativa e responsiva com acesso rápido às funcionalidades.
- 2 Implementar uma API RESTful em Spring Boot com segurança, modularidade e padronização.
- 3 Integrar MySQL e MongoDB de forma coerente, assegurando consistência e desempenho.
- 4 Desenvolver autenticação e autorização com diferentes níveis de acesso (Admin/Comum).
- 5 Criar módulos para cadastro, consulta, empréstimo e devolução de livros com histórico.
- 6 Garantir persistência e integridade dos dados com boas práticas de modelagem.

# Arquitetura Fullstack: Camadas Integradas

## CAMADA 1: APRESENTAÇÃO

### Frontend - React.js

Interface responsiva e interativa para o usuário, com componentes reutilizáveis, gerenciamento de estado e navegação intuitiva.

📱 **Tecnologia:** React.js, JavaScript, CSS Responsivo

↓ API REST (HTTP/JSON) ↓

## CAMADA 2: LÓGICA DE NEGÓCIOS

### Backend - Java + Spring Boot

Processamento de requisições, implementação de regras de negócio, autenticação com JWT, e orquestração da persistência de dados.

⚙️ **Tecnologia:** Java, Spring Boot, Spring Data JPA, JWT

↓ Consultas/Transações ↓

## CAMADA 3: PERSISTÊNCIA (HÍBRIDA)

### Bancos de Dados

#### MySQL

Usuários &  
Autenticação

#### MongoDB

Livros &  
Empréstimos

💾 **Bancos:** MySQL (Relacional) + MongoDB (NoSQL)

# Tecnologias Utilizadas

CAMADA	TECNOLOGIA	FUNÇÃO	JUSTIFICATIVA
FRONTEND	React.js	Interface responsiva e interativa para o usuário.	Alta performance, reutilização de componentes, facilidade de integração com APIs REST e ampla comunidade de suporte.
BACKEND	Java + Spring Boot	Lógica de negócios, API RESTful e orquestração de dados.	Robustez, segurança, produtividade, configuração automática, injeção de dependência e suporte integrado a segurança e APIs REST.
PERSISTÊNCIA	MySQL	Armazenamento de usuários e autenticação.	Banco relacional estruturado, ideal para dados consistentes, suporte a transações e integração nativa com Spring Data JPA.
PERSISTÊNCIA	MongoDB	Armazenamento de livros, empréstimos e histórico.	Banco NoSQL orientado a documentos, flexibilidade na modelagem, agilidade em consultas de grande volume e suporte a estruturas dinâmicas.

# Banco de Dados Relacional: MySQL

O MySQL armazena informações estruturadas de usuários e autenticação, garantindo consistência, segurança e integridade referencial através de um modelo relacional bem definido.

📋 TABELA: GRUPOS\_USUARIOS

<b>id</b>	UUID <b>PK</b>
<b>nome</b>	VARCHAR(100)
<b>descricao</b>	TEXT

👤 TABELA: USUARIOS

<b>id</b>	UUID <b>PK</b>
<b>nome</b>	VARCHAR(255)
<b>email</b>	VARCHAR(255) <b>UNIQUE</b>
<b>senha</b>	VARCHAR(255)
<b>data_cadastro</b>	TIMESTAMP
<b>ativo</b>	BOOLEAN
<b>grupo_id</b>	UUID <b>FK</b>

🔗 RELACIONAMENTO

**Tipo:** Um-para-Muitos (1:N) | **Descrição:** Um grupo pode ter vários usuários, mas cada usuário pertence a apenas um grupo.

✓ *Chave Estrangeira: usuarios.grupo\_id → grupos\_usuarios.id / Garante integridade referencial e facilita controle de acesso baseado em grupos.*

# Banco de Dados NoSQL: MongoDB

---

# Funcionalidades Principais do Sistema



1

## Gerenciamento de Acervo

Cadastro, edição, exclusão e consulta de livros com metadados completos (título, autor, gênero, ano, quantidade).



2

## Controle de Avaliação

Solicitação, controle de prazos, histórico de empréstimos e gerenciamento de devoluções com notificações.



3

## Sistema de Reservas

Reserva de livros indisponíveis com fila de espera automática e notificação de disponibilidade.



4

## Autenticação e Autorização

Login seguro com JWT, diferentes níveis de acesso (Administrador, Bibliotecário, Usuário Comum) e controle de permissões.



5

## Painel Administrativo

Gestão de usuários, geração de relatórios de empréstimos e reservas, controle de estoque em tempo real.



6

## Busca e Filtros

Pesquisa avançada por título, autor e gênero, com filtros e ordenação para facilitar navegação no acervo.

# Conclusão e Impacto

## RESULTADOS ALCANÇADOS

### Validação da Arquitetura Fullstack

#### ✓ INTEGRAÇÃO HÍBRIDA

MySQL e MongoDB funcionando de forma coerente e complementar.

#### ✓ API RESTFUL

Endpoints bem definidos com segurança JWT implementada.

#### ✓ INTERFACE RESPONSIVA

Frontend React com componentes dinâmicos e navegação intuitiva.

#### ✓ CONTROLE DE ACESSO

Autenticação e autorização com diferentes níveis de permissão.

## IMPACTOS E BENEFÍCIOS

### Vantagens da Solução Implementada



#### Desempenho

Consultas otimizadas em ambos os bancos de dados.



#### Escalabilidade

Arquitetura modular pronta para crescimento.



#### Segurança

Autenticação JWT e integridade de dados garantida.

O projeto validou com sucesso a eficácia de uma arquitetura fullstack moderna, demonstrando como a integração equilibrada de tecnologias relacionais e não relacionais resulta em uma solução eficiente, segura e facilmente mantível para ambientes bibliotecários reais.