



UNIVERSIDADE FEDERAL DE RORAIMA  
CENTRO DE CIÊNCIA E TECNOLOGIA  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO  
DCC603 - BANCOS DE DADOS II (2024.1)  
Prof. ACAUAN CARDOSO RIBEIRO

**Aluno:** Eduardo Henrique de Almeida Izidorio - 2020000315

## Relatório do Projeto: Livraria OneByte

- **Repositório GitHub**

O código-fonte do projeto está disponível no GitHub.

Link do Repositório GitHub: <https://github.com/EhoKira/livraria-onebyte.git>

- **Vídeo demonstrativo do Sistema**

Preparei também um vídeo demonstrativo do funcionamento do site, onde mostro as principais funcionalidades do sistema.

Link do vídeo no Youtube: <https://youtu.be/Ai6XdfarMcU>

### 1. Introdução

O projeto Livraria Onebyte visa criar uma plataforma web completa para a visualização, inserção e gerenciamento de livros de ciência da computação. A aplicação é um sistema full-stack, com funcionalidades que permitem a interação tanto pelo frontend quanto pelo backend, com armazenamento de dados em um banco de dados MongoDB.

### 2. Objetivo do Sistema

O principal objetivo do sistema é fornecer uma interface simples e eficiente para gerenciar informações de livros de ciência da computação. O sistema permite que os usuários visualizem uma lista de livros, insiram novos títulos, atualizem e excluam dados existentes.

### 3. Requisitos Principais

#### 3.1 Requisitos Funcionais:

- **Cadastro e Visualização de Livros:**

Os usuários podem visualizar uma lista de livros contendo informações como *id* e *nome*.

- **Inserção de Livros:**

O sistema permite a inserção de novos livros, vinculando-os automaticamente ao banco de dados com base no *id* e *nome* fornecidos.

- **Atualização e Exclusão de Livros:**

O usuário tem a capacidade de atualizar e excluir livros já cadastrados no sistema.

### 3.2. Requisitos Não Funcionais

- **Desempenho:**

O sistema deve responder às requisições de inserção, atualização e exclusão de livros em tempo hábil, com tempo de resposta máximo de 2 segundos por operação.

- **Segurança:**

A aplicação deve garantir que apenas usuários autenticados possam realizar operações de inserção, atualização e exclusão de livros. Além disso, as operações de CRUD devem ser protegidas contra ataques como *SQL Injection* (mesmo no contexto do MongoDB, com proteção contra injeções de código).

- **Usabilidade:**

O frontend deve ser intuitivo e fácil de usar, permitindo que qualquer usuário, mesmo sem conhecimento técnico, consiga navegar pela interface e realizar operações com facilidade.

### 4. Diagrama de Caso de Uso

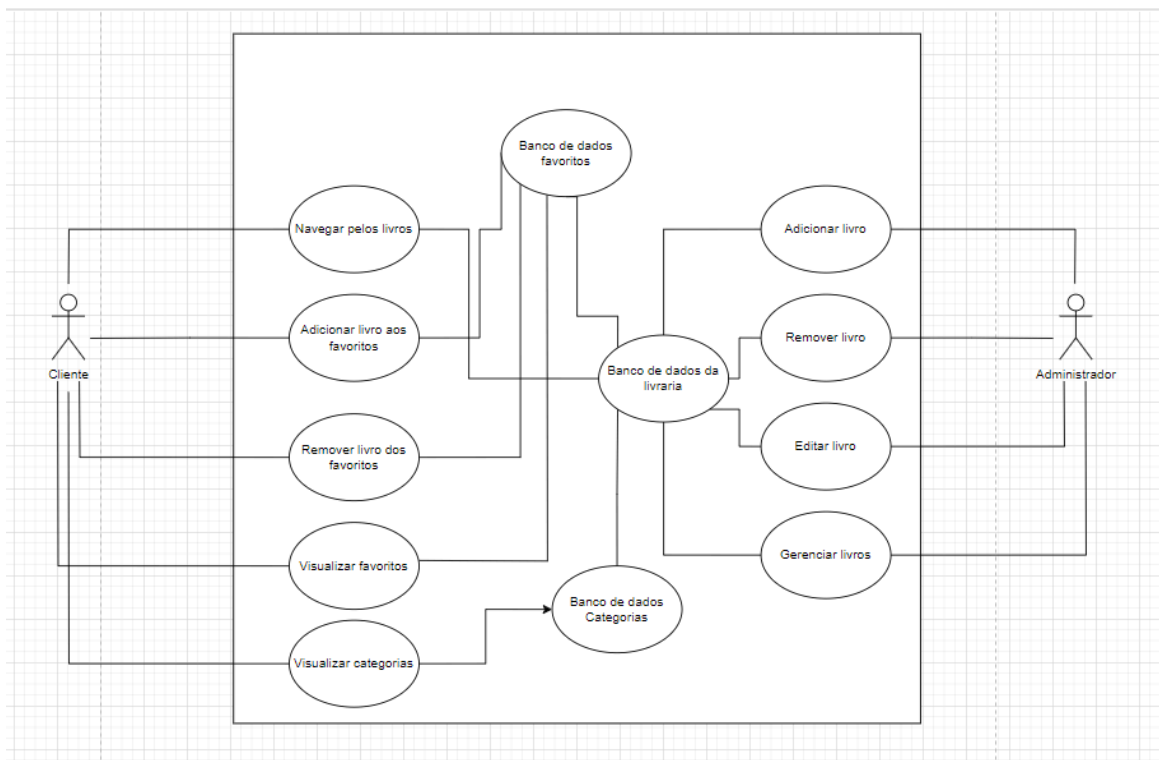
No diagrama de caso de uso, os principais atores são:

- **Usuário:**

Pode visualizar, adicionar, editar e excluir livros.

- **Sistema:**

Gerencia as interações do usuário com o banco de dados, garantindo que as operações de CRUD sejam executadas corretamente.



## 5. Ferramentas utilizadas

As ferramentas e tecnologias utilizadas no desenvolvimento do projeto incluem:

- **Node.js e Express** para o backend.
- **React** para o frontend.
- **MongoDB** como banco de dados.
- **Postman** para testes de requisições e integração entre frontend e backend.

## 6. Desafios

Durante o desenvolvimento do projeto, um dos principais desafios enfrentados foi a integração correta do banco de dados MongoDB com o backend, especialmente na inserção de novos registros. Houve problemas em que os dados estavam sendo salvos como **NULL** no banco de dados.