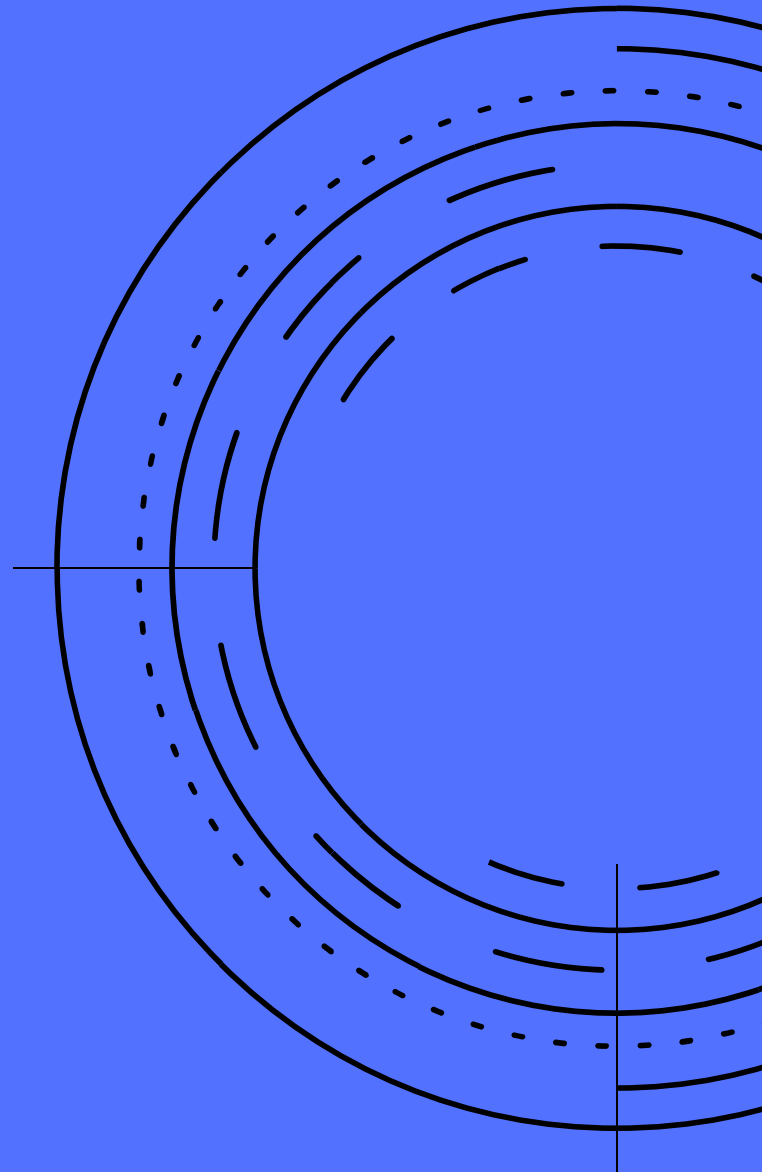
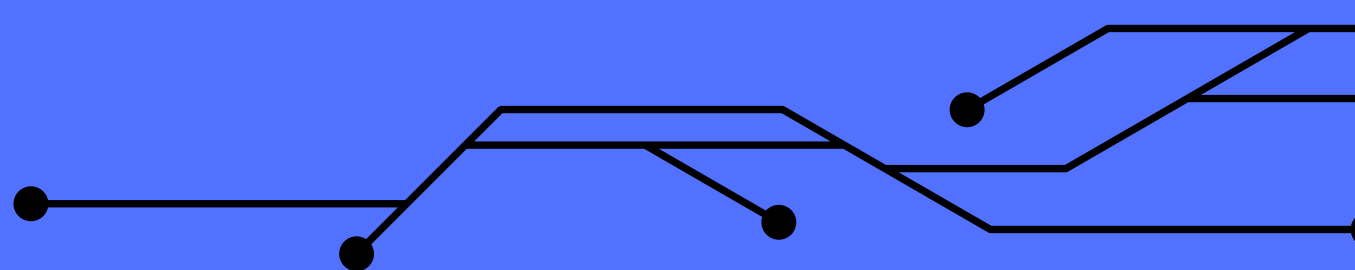




# API DE GESTÃO EDUCACIONAL COM FLASK

"SISTEMA CRUD PARA PROFESSORES, TURMAS E ALUNOS COM MYSQL, DOCKER, SWAGGER E DEPLOY"



**Autor: Guilherme de Freitas Fracasso  
Enzo Pelakoski Cavinato  
Instituição: Faculdade IMPACTA  
Curso: ADS  
Data: 17/03/2025**

## 1. INTRODUÇÃO

O objetivo deste projeto é apresentar o desenvolvimento de uma aplicação com foco na utilização de banco de dados MySQL, contêineres Docker e deploy na plataforma Render. A proposta teve como prioridade a integração de tecnologias modernas para garantir um ambiente de desenvolvimento eficiente e organizado.

A aplicação foi estruturada para utilizar o MySQL como sistema de gerenciamento de banco de dados, com o ambiente containerizado através do Docker, promovendo portabilidade e consistência no desenvolvimento.

O projeto foi desenvolvido por dois integrantes: Guilherme de Freitas Fracasso e Enzo Pelakoski Cavinato, que atuaram de forma igualitária em todas as etapas, desde a criação da base de dados até as tentativas de publicação da aplicação.

## 2. DESCRIÇÃO E ANÁLISE DO CASO

O projeto foi construído com base na utilização do banco de dados relacional MySQL, estruturando as tabelas e relações de forma adequada às necessidades da aplicação. A conexão com o banco foi testada com sucesso em ambiente local, utilizando contêineres Docker para isolar os serviços.

O Docker foi empregado para criar um ambiente completo e replicável, com contêineres separados para o backend e o banco de dados. Essa abordagem permitiu uma maior organização e independência entre os componentes do sistema.

Contudo, durante o processo de deploy na plataforma Render, surgiram desafios técnicos, especialmente na configuração do ambiente, integração com os contêineres e persistência dos dados. A aplicação ainda não foi disponibilizada online de forma estável.

## 3. IMPLEMENTAÇÃO OU PROCEDIMENTO

A implementação iniciou-se com a modelagem e criação do banco de dados MySQL. Foram desenvolvidas tabelas com chaves primárias e estrangeiras, conforme a lógica da aplicação.

Com a base de dados estruturada, partimos para a criação de um ambiente com Docker, utilizando arquivos Dockerfile e docker-compose.yml. O sistema funcionou corretamente em ambiente local, com comunicação eficaz entre os contêineres.

Na fase de deploy, configuramos o repositório e os comandos iniciais na Render. No entanto, enfrentamos dificuldades na exposição das portas corretas, na definição de variáveis de ambiente e na comunicação entre os serviços. A aplicação ainda se encontra em fase de ajustes para conclusão dessa etapa.

## 4. RESULTADOS

Até o momento, a aplicação está funcional localmente, com integração estável entre o backend e o banco de dados MySQL via Docker.

O uso de contêineres trouxe organização ao projeto e facilidade no desenvolvimento. Porém, a fase de deploy apresentou obstáculos que ainda estão sendo resolvidos, impedindo a disponibilização pública da aplicação até o presente momento.

## 5. CONCLUSÃO

O desenvolvimento local da aplicação foi bem-sucedido. As tecnologias escolhidas — MySQL e Docker — demonstraram ser eficazes para estruturar o projeto de forma organizada e escalável.

No entanto, o deploy na Render ainda está em andamento. Dificuldades com a configuração dos serviços, principalmente a integração com o contêiner de banco de dados, têm sido um ponto de atenção e aprendizado para os integrantes do grupo.

Mesmo com essas limitações, o projeto se mostra promissor e a equipe segue trabalhando para superar os desafios encontrados.

## 6. IMPACTO E CONEXÃO COM O MUNDO REAL

O uso de banco de dados MySQL e Docker reflete práticas reais no mercado de desenvolvimento de software. Esses recursos são amplamente utilizados em empresas e projetos profissionais.

Já a plataforma Render oferece um ambiente moderno e acessível para deploys, sendo uma alternativa viável para aplicações em pequena e média escala. Embora o processo ainda não tenha sido concluído com sucesso, o contato com esses desafios aproxima os integrantes da realidade de times de tecnologia.

## 7. DESAFIOS FUTUROS E MELHORIAS

Com base nas experiências e nas dificuldades enfrentadas, identificamos pontos importantes para melhoria:

- Finalizar o Deploy na Render: Resolver os problemas de configuração de rede, variáveis e persistência de dados.
- Logs e Diagnóstico: Implementar ferramentas de monitoramento e análise de erros para facilitar o diagnóstico no ambiente de produção.
- Melhorar a Documentação Técnica: Criar guias internos para ajudar na replicação do ambiente e facilitar a manutenção futura.
- Explorar Outras Alternativas de Deploy: Considerar o uso de outras plataformas como Railway, Vercel (para frontend) ou até uma VPS, caso as dificuldades com o Render persistam.