

Documento do Projeto: Sistema de Gerenciamento de Alunos e Professores

1. Introdução

O desenvolvimento de sistemas de gestão acadêmica é fundamental para facilitar a administração de instituições educacionais, sobretudo em ambientes digitais. Este projeto consiste em uma aplicação web desenvolvida com Flask, cujo objetivo é permitir que professores possam gerenciar suas turmas, adicionar alunos por e-mail e, futuramente, compartilhar arquivos e fazer as simulações em VR nas turmas. A aplicação também oferece autenticação para alunos e professores, além de registrar ações importantes como login, criação de turmas e registros de novos usuários através de um sistema de logs. A aplicação busca proporcionar uma interface simplificada e eficiente para a gestão educacional, oferecendo funcionalidades de uso prático e intuitivo.

Objetivo Geral

Desenvolver uma aplicação web que permita a gestão de turmas e alunos por professores, com a possibilidade de adicionar alunos às turmas de forma fácil.

Objetivos Específicos

- Implementar autenticação para alunos e professores.
- Criar dashboards personalizados para cada perfil (professor e aluno).
- Permitir que professores criem e gerenciem suas turmas.
- Registrar as principais ações realizadas pelos usuários no sistema de logs.

2. Metodologia

O desenvolvimento do projeto foi conduzido utilizando o **Framework Flask**, um microframework em Python que possibilita o desenvolvimento ágil e modular de aplicações web. A aplicação segue uma arquitetura baseada em **blueprints**, o que facilita a organização do código e a modularização das rotas.

Tecnologias Utilizadas

- **Flask** para a criação da aplicação web.
- **Flask SQLAlchemy** como ORM para gerenciamento de banco de dados SQLite.
- **Werkzeug** para a geração e verificação de hashes de senha.
- **HTML/CSS** para a interface de usuário.
- **Blueprints** para a separação de rotas e funcionalidades em módulos.
- **SQLite** como banco de dados.

Estrutura de Dados

A estrutura de dados foi planejada para armazenar informações de professores, alunos, turmas e a relação entre alunos e turmas. O banco de dados contém as seguintes tabelas principais:

- **User:** Contém as informações dos usuários, diferenciando alunos de professores através do campo "role".
- **ClassM:** Contém os dados das turmas, associando-as a um professor.
- **ClassStudent:** Relação de muitos-para-muitos entre turmas e alunos, permitindo que cada aluno possa ser inscrito em várias turmas e cada turma tenha vários alunos.

Funcionalidades

- **Autenticação de Usuários:** Professores e alunos podem se registrar e realizar login na plataforma, com base em suas credenciais.
- **Criação e Gestão de Turmas:** Professores podem criar novas turmas e gerenciá-las, adicionando alunos por e-mail.
- **Dashboard:** Professores têm acesso a um painel de controle onde podem visualizar e gerenciar suas turmas. Alunos podem visualizar as turmas nas quais estão inscritos.
- **Sistema de Logs:** Todas as ações importantes, como login, registro de usuários e criação de turmas, são registradas em um arquivo de log, facilitando a auditoria.

Sistema de Logs

O sistema de logs foi implementado por meio da classe LogClass que grava todas as ações relevantes em um arquivo desenvolvimento_log.txt. Esse arquivo permite o acompanhamento detalhado de eventos, como tentativas de login, registros de novos alunos e professores, criação de turmas e adição de alunos.

3. Resultados

A aplicação foi testada e as funcionalidades implementadas se comportaram conforme o esperado. A seguir, são destacados os principais resultados:

Interface e Usabilidade

- **Autenticação:** Professores e alunos conseguem se registrar, realizar login e logout sem complicações.
- **Gerenciamento de Turmas:** Professores conseguem criar turmas e adicionar alunos a elas. A interface é simples e intuitiva, permitindo que um professor gerencie várias turmas de maneira prática.
- **Logs de Eventos:** O sistema de logs captura eficientemente os eventos principais, proporcionando um registro detalhado das atividades dos usuários. Logs de tentativas de login inválidas, registros de novos alunos e criação de turmas foram gerados corretamente.

Banco de Dados

- A aplicação faz uso de **SQLite** para armazenar os dados. As relações entre alunos, professores e turmas foram estabelecidas com sucesso, permitindo que cada aluno seja associado a múltiplas turmas e cada professor gerencie suas próprias turmas.

Escalabilidade e Manutenção

- A aplicação está estruturada para permitir fácil expansão. Com a separação das rotas em blueprints e a modularização da lógica de logs, funcionalidades futuras como o compartilhamento de arquivos e aulas poderão ser integradas sem impactar significativamente a base do código.

4. Conclusão

O sistema oferece uma interface clara para os professores criarem e gerenciarem turmas, além de adicionar alunos de maneira eficiente. A arquitetura modular permite a adição de novas funcionalidades sem grandes mudanças na base do código. O uso de logs garante que as ações dos usuários sejam rastreadas, aumentando a transparência e a auditabilidade do sistema.

Futuras Melhorias

- **Compartilhamento de Arquivos:** Uma próxima fase do projeto incluirá a possibilidade de professores compartilharem arquivos e materiais de aula com suas turmas.
- **Notificações:** A adição de um sistema de notificações para avisar alunos de novas turmas ou conteúdos.
- **Melhorias na Interface:** Futuramente, a interface poderá ser melhorada para incluir design mais moderno e intuitivo, facilitando ainda mais a experiência do usuário.

Este projeto é um ponto de partida para a construção de um sistema completo de gerenciamento acadêmico, com potencial para ser ampliado conforme as necessidades da LEPIC, como simulações VR, sistema de pontuação e compartilhamento de arquivos sobre a aula.