

ATIVIDADE	FORMATIVA I - SITUAÇÃO PROBLEMA
<b>CONTEXTO e DESCRIÇÃO</b>	
<p>Nos dias atuais, a gestão educacional se tornou um desafio cada vez maior devido à complexidade das operações envolvidas no dia a dia das instituições de ensino. A administração de escolas e universidades exige uma coordenação eficiente de diversos aspectos, como o gerenciamento de professores, disciplinas, salas de aula e recursos pedagógicos. Para atender a essa necessidade crescente, muitas instituições têm adotado sistemas tecnológicos especializados para otimizar esses processos e garantir um ambiente organizado e funcional para todos os envolvidos no processo educacional.</p> <p>Um sistema de gerenciamento de professores e ambientes é uma ferramenta que permite a administração eficiente e integrada de diversos elementos importantes dentro de uma escola ou instituição de ensino. Esse sistema oferece aos gestores, como coordenadores e diretores, a capacidade de realizar o cadastro, controle e atualização de informações sobre os professores, suas respectivas disciplinas e os ambientes de aprendizagem disponíveis, como salas de aula ou laboratórios.</p> <p>Além disso, o sistema também facilita a reserva de ambientes para atividades pedagógicas, possibilitando que os responsáveis pela gestão das aulas planejem e aloquem recursos de maneira inteligente, evitando conflitos de horários e aproveitando ao máximo a infraestrutura da instituição. Por sua vez, os professores podem acessar o sistema para visualizar informações relevantes sobre suas disciplinas e os horários e locais das aulas, sem a necessidade de realizar alterações no sistema, o que proporciona mais agilidade e menos burocracia.</p> <p>Com a implementação de um sistema de gerenciamento de professores e ambientes, as escolas podem melhorar significativamente a organização interna, a comunicação entre os envolvidos no processo educacional e o uso eficiente dos recursos. Além disso, ao automatizar muitos processos administrativos, o sistema contribui para a redução de erros humanos e aumenta a transparência das operações.</p> <p>Neste cenário, surge a oportunidade de realizar a implementação de um novo sistema que permitirá aos gestores gerenciarem de maneira eficiente o cadastro de professores, Disciplinas e reservas de salas. O gestor será responsável por cadastrar, visualizar, atualizar e excluir informações relacionadas aos professores, suas Disciplinas e a reserva de ambientes, garantindo que todas as operações aconteçam de forma organizada e acessível.</p> <p>Por outro lado, os professores terão acesso ao sistema de forma mais restrita, podendo apenas visualizar as informações relacionadas às suas Disciplinas e aos ambientes reservados para suas aulas, sem a possibilidade de realizar alterações.</p> <p>Este sistema visa facilitar a gestão diária da instituição, proporcionando um ambiente mais organizado para o planejamento e execução das atividades educacionais, contribuindo para o aprimoramento contínuo do processo de ensino-aprendizagem.</p> <p><b>Objetivo do Projeto:</b></p> <p>Desenvolver um sistema de gerenciamento de professores, Disciplinas e reservas de ambientes, utilizando o Django para o back-end. O sistema permitirá que os Gestores possam cadastrar, visualizar, atualizar e excluir informações sobre professores, suas Disciplinas e as reservas de salas. Além disso, o sistema oferecerá funcionalidades de autenticação para garantir que apenas os gestores possam realizar essas operações, enquanto os professores terão acesso restrito à visualização das informações pertinentes às suas atividades.</p> <p><b>Requisitos do Sistema:</b></p> <p><b>1. Funcionalidades do Back-end:</b></p> <ul style="list-style-type: none"><li>○ <b>Cadastro de Professores:</b> Endpoint para permitir que os Gestores cadastrem novos professores no sistema, com suas informações essenciais como nome, e-mail, Disciplinas atribuídas, entre outros dados.</li><li>○ <b>Leitura de professores:</b> Endpoint para permitir que os Gestores visualizem a lista de todos os professores cadastrados, com a possibilidade de exibir os detalhes de cada um.</li><li>○ <b>Atualização de Professores:</b> Endpoint para permitir que os Gestores atualizem as informações de um professor já cadastrado, como dados de contato e Disciplinas</li></ul>	

atribuídas.

- **Exclusão de Professores:** Endpoint para permitir que os Gestores excluam professores do sistema.
- **Criação de Disciplinas:** Endpoint para permitir que os Gestores cadastrem novas Disciplinas e as vinculem aos professores correspondentes.
- **Leitura de Disciplinas:** Endpoint para permitir que os Gestores visualizem todas as Disciplinas cadastradas, incluindo informações sobre os professores responsáveis.
- **Atualização de Disciplinas:** Endpoint para permitir que os Gestores atualizem os detalhes das Disciplinas, como nome, carga horária e professor responsável.
- **Exclusão de Disciplinas:** Endpoint para permitir que os Gestores excluam Disciplinas do sistema.
- **Criação de Reservas de Ambiente:** Endpoint para permitir que os Gestores realizem reservas de salas para os professores, incluindo informações como data, horário e sala reservada.
- **Leitura de Reservas de Ambiente:** Endpoint para permitir que os Gestores visualizem as reservas de ambientes, com os detalhes da sala, horário e professor responsável.
- **Exclusão de Reservas de Ambiente:** Endpoint para permitir que os Gestores excluam reservas de ambientes.
- **Autenticação e Autorização:** Implementar um sistema de autenticação simples para garantir que apenas os Gestores possam realizar ações de cadastro, atualização e exclusão. Além disso, deve ser garantido que os professores tenham apenas acesso para visualizar as informações sobre as Disciplinas e as reservas de salas às quais estão vinculados.

## 2. Funcionalidades Adicionais:

- **Visualização de Reservas por Professores:** Endpoint para permitir que os professores visualizem as salas que foram reservadas para suas aulas, incluindo data e horário da reserva.
- **Visualização de Disciplinas por Professores:** Endpoint para permitir que os professores visualizem as Disciplinas a que estão atribuídos, com informações como nome da Disciplinas, carga horária e dados do professor responsável.

## Modelo de Dados:

### Professor:

- NI
- Nome
- E-mail
- Telefone
- Data de nascimento
- Data de contratação
- Disciplinas atribuídas (relacionamento com a tabela de Disciplinas)

### Disciplinar:

- Nome
- Curso
- Carga horária
- Descrição
- Professor responsável (relacionamento com a tabela de Professores)

### Reserva de Ambiente:

- Data início
- Data término
- Período ( Manhã, Tarde e Noite )
- Sala reservada
- Professor responsável (relacionamento com a tabela de Professores)
- Disciplina associada (relacionamento com a tabela de Disciplinas)

## Etapas do projeto:

### 1. Configuração do Ambiente de Desenvolvimento:

- Instalar e configurar o Django e Django Rest Framework.
  - Configurar o banco de dados MySQL.
  - Criar um projeto e uma aplicação no Django.
  - Configurar o ambiente para permitir o desenvolvimento de APIs RESTful utilizando Django Rest Framework.
- 2. Desenvolvimento do Back-end:**
- Criar o modelo de dados para Professores, Disciplinas e Reservas de Ambiente.
  - Implementar os endpoints de CRUD (Create, Read, Update, Delete) para Professores, Disciplinas e Reservas de Ambiente.
  - Implementar a autenticação utilizando JWT ou outra abordagem simples para garantir que apenas os Gestores possam realizar ações de CRUD.
  - Implementar lógica de autorização para garantir que professores tenham acesso apenas à visualização dos dados relevantes.
- 3. Configuração do Banco de Dados MySQL:**
- Instalar e configurar o MySQL.
  - Configurar o Django para usar o banco de dados MySQL.
  - Criar as migrações e aplicar as migrações para criar as tabelas no banco de dados MySQL.
- 4. Desenvolvimento de Funcionalidades no Back-end:**
- Implementar endpoints para o cadastro, visualização, atualização e exclusão de Professores, Disciplinas e Reservas de Ambiente.
  - Desenvolver endpoints de visualização específicos para professores (exibição das Disciplinas atribuídas e reservas realizadas).
  - Implementar mecanismos de validação e verificação de dados nos endpoints.
- 5. Implementação de Autenticação e Autorização:**
- Implementar autenticação baseada em tokens (JWT) para garantir que apenas usuários autenticados possam acessar os endpoints protegidos.
  - Configurar a autorização para garantir que apenas os Gestores possam cadastrar, editar ou excluir Professores, Disciplinas e Reservas de Ambiente.
- 6. Testes e Validação:**
- Testar o fluxo completo da aplicação para garantir que os endpoints de CRUD estão funcionando corretamente para Professores, Disciplinas e Reservas de Ambiente.
  - Realizar testes de autenticação e autorização para garantir que o sistema está corretamente protegendo os dados.
  - Validar a consistência dos dados no banco e a comunicação entre o back-end e as APIs.
  - Garantir que o sistema de reserva de ambientes esteja funcionando corretamente, com as informações sendo exibidas de forma adequada para os professores.
- 7. Documentação:**
- Documentar os endpoints da API utilizando ferramentas como Swagger ou Postman.
  - Criar documentação detalhada sobre a instalação e configuração do sistema.