

# Documentação do Banco de Dados - Sistema Escolar

## 1. Estrutura Atual do Sistema

Voce ja possui as seguintes tabelas no banco de dados:

- users: Tabela principal de autenticação, contendo nome, email, senha e tipo (admin, professor, aluno).
- cursos: Representa os cursos disponiveis no sistema.

## 2. Novas Tabelas Criadas e Relacionamentos

As seguintes tabelas foram adicionadas para completar a estrutura relacional do sistema:

- alunos: Ligada a tabela users por user\_id. Armazena dados pessoais do aluno.
- professores: Ligada a tabela users por user\_id. Armazena formacao e contato do professor.
- matriculas: Relaciona alunos e cursos com status da matricula e data.
- aulas: Cada aula pertence a um curso. Pode ser usada para agendamento e controle.
- frequencias: Marca presenca do aluno em uma aula especifica.
- notas: Guarda as avaliacoes do aluno em um curso, com valor, peso e data.
- mensagens: Comunicacao entre dois usuarios (remetente e destinatario).
- pagamentos: Controle financeiro por aluno e curso, com status e metodo de pagamento.

## 3. Ligacoes e Chaves Estrangeiras

- alunos.user\_id -> users.id
- professores.user\_id -> users.id
- cursos.professor\_id -> professores.id
- matriculas.aluno\_id -> alunos.id
- matriculas.curso\_id -> cursos.id
- aulas.curso\_id -> cursos.id
- frequencias.aluno\_id -> alunos.id
- frequencias.aula\_id -> aulas.id
- notas.aluno\_id -> alunos.id
- notas.curso\_id -> cursos.id
- mensagens.remetente\_id -> users.id
- mensagens.destinatario\_id -> users.id
- pagamentos.aluno\_id -> alunos.id
- pagamentos.curso\_id -> cursos.id

## 4. O que deve ser feito no Backend (ex: PHP, Node.js)

1. Criar rotas protegidas por autenticação JWT (login unico redireciona pelo tipo).
2. Implementar CRUDs para as tabelas: cursos, aulas, notas, mensagens, pagamentos.
3. Criar middlewares de autorizacao: aluno, professor, admin.

## **Documentação do Banco de Dados - Sistema Escolar**

4. Controlar permissões de acesso para cada tipo de usuário.
5. Relacionar modelos (ORM se estiver usando Sequelize, Eloquent, etc.).
6. Criar endpoints específicos para ações como marcar presença, enviar mensagem, registrar nota.

### **5. O que deve ser feito no Frontend**

1. Tela de login única com redirecionamento baseado no tipo de usuário.
2. Dashboards separados por tipo: aluno, professor, admin.
3. Telas específicas para:
  - Cursos e Aulas
  - Boletim e Presença (aluno)
  - Lançamento de notas e controle de frequência (professor)
  - Cadastro geral e controle financeiro (admin)
4. Tela de mensagens (chat interno) entre usuários.
5. Responsividade para uso em celulares e tablets.
6. Integração com backend via API REST.