

Gerador Automático de Planos de Aula com IA

Renan Rosa • Desenvolvedor Backend e IA

Desafio Técnico – Supabase Backend Dev 2025

Sumário

| | |
|-----------------------------|----|
| Sobre o Projeto | 1 |
| Tecnologias Utilizadas | 2 |
| Funcionalidades | 3 |
| Instalação e Execução | 4 |
| Modelo de IA Utilizado | 5 |
| Estrutura do Banco de Dados | 6 |
| Decisões Técnicas | 7 |
| Desafios e Soluções | 8 |
| Links Importantes | 9 |
| Autor | 10 |

Sobre o Projeto

O Gerador Automático de Planos de Aula utiliza IA generativa (Google Gemini) para criar planos pedagógicos personalizados a partir de informações simples inseridas pelo usuário. Cada plano é gerado automaticamente e armazenado no Supabase, podendo ser visualizado, baixado em PDF ou excluído pelo painel web.

Tecnologias Utilizadas

Frontend: HTML, CSS, JavaScript

Backend: Node.js, Express.js

Banco de Dados: Supabase

IA: Google Gemini API

PDF: jsPDF

Funcionalidades

- Geração automática de planos de aula com IA
- Salvamento e leitura no Supabase
- Listagem e exclusão de planos
- Exportação em PDF
- Interface simples e responsiva

Instalação e Execução

1. Clone o repositório GitHub
2. Instale dependências com `npm install`
3. Configure o arquivo `.env` com suas chaves
4. Execute o servidor local com `node server.js`

Modelo de IA Utilizado

Modelo escolhido: Gemini 1.5 Flash – rápido, eficiente e ideal para geração de conteúdo educacional estruturado em JSON.

Documentação: <https://ai.google.dev/gemini-api/docs>

Estrutura do Banco de Dados (Supabase)

Tabela: `planos_aula` (`id`, `titulo`, `tema`, `faixa_etaria`, `disciplina`, `duracao`, `nivel_dificuldade`, `observacoes`, `introducao`, `objetivo_bccc`, `passo_a_passo`, `rubrica_avaliacao`, `criado_em`)

Decisões Técnicas

- Backend: Node.js + Express
- Banco: Supabase (PostgreSQL com autenticação)
- IA: Google Gemini API
- Frontend: HTML/CSS/JS puro
- Hospedagem: Localhost (pronto para Render/Vercel/Railway)

Desafios e Soluções

- Parsing do JSON da IA resolvido com `JSON.parse()`
- Conexão Supabase usando `@supabase/supabase-js`
- Tratamento de erros com `try/catch`
- Renderização dinâmica via frontend

Links Importantes

GitHub: <https://github.com/dev-renanrosa/gerador-planos-aula>

Projeto Supabase:

<https://supabase.com/dashboard/project/ejdzafkedqbiiczogugp>

Aplicação Local: <http://localhost:3000>

Autor

Renan Rosa – Desenvolvedor Backend e IA

E-mail: dev.renanrosa@gmail.com

GitHub: github.com/dev-renanrosa