

Guia de Configuração

Supabase, PostgreSQL & API Integration

v1.0.0

</> Backend Guide



Instalações Realizadas

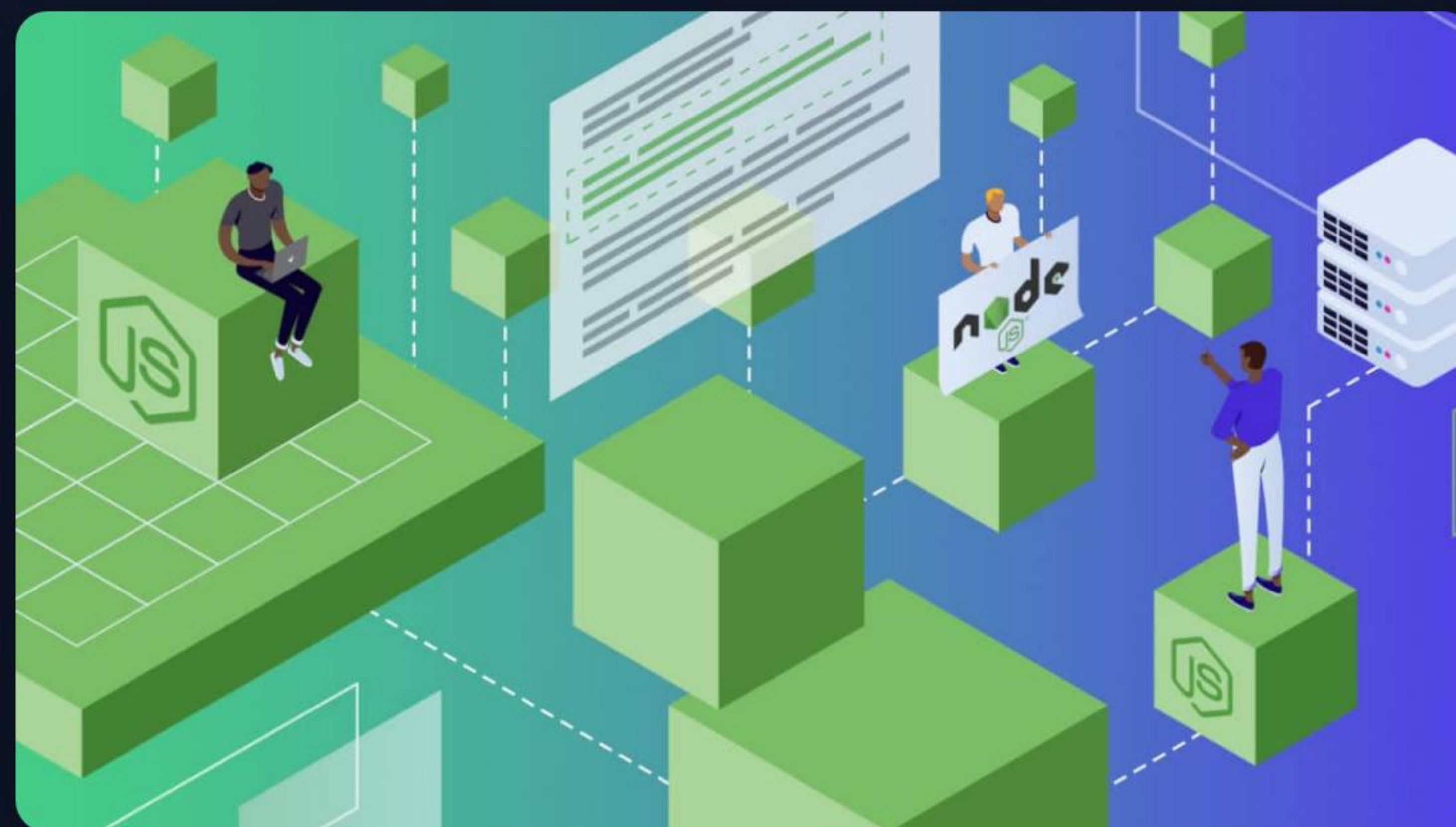
O ambiente foi preparado com as bibliotecas essenciais para conectar sua aplicação Node.js ao Supabase e gerenciar variáveis de ambiente com segurança.

✓ **@supabase/supabase-js**

Cliente oficial JavaScript para interação com o banco de dados e Auth.

✓ **dotenv**

Carrega variáveis de ambiente do arquivo `.env` para proteger suas credenciais.





Estrutura do Banco de Dados



Clientes

Armazena informações pessoais e de contato. E-mail único garantido.



Produtos

Catálogo de itens com controle de estoque, preço e categorização.



Pedidos

Registra transações vinculadas a clientes, com itens em JSONB.



Schema: Clientes e Produtos

Tabela Clientes

```
CREATE TABLE clientes (  
  id BIGSERIAL PRIMARY KEY,  
  nome VARCHAR(255) NOT NULL,  
  email VARCHAR(255) NOT NULL UNIQUE,  
  telefone VARCHAR(20),  
  dataCadastro DATE DEFAULT CURRENT_DATE,  
  created_at TIMESTAMP DEFAULT NOW()  
);
```

Tabela Produtos

```
CREATE TABLE produtos (  
  id BIGSERIAL PRIMARY KEY,  
  nome VARCHAR(255) NOT NULL,  
  categoria VARCHAR(100) NOT NULL,  
  preco DECIMAL(10, 2) NOT NULL,  
  estoque INTEGER DEFAULT 0,  
  created_at TIMESTAMP DEFAULT NOW()  
);
```




Schema: Pedidos e Índices

Tabela Pedidos

```
CREATE TABLE pedidos (  
  id BIGSERIAL PRIMARY KEY,  
  clienteId BIGINT REFERENCES clientes(id),  
  itens JSONB NOT NULL,  
  valorTotal DECIMAL(10, 2) NOT NULL,  
  status VARCHAR(50) DEFAULT 'pendente',  
  created_at TIMESTAMP DEFAULT NOW()  
);
```

Performance (Índices)

Índices criados para otimizar buscas frequentes por e-mail e relacionamentos de pedidos.

```
CREATE INDEX idx_pedidos_clienteId  
ON pedidos(clienteId);  
  
CREATE INDEX idx_clientes_email  
ON clientes(email);
```




Fluxo de Configuração



1. Dashboard

Acesse supabase.com e crie seu projeto. Copie a URL e a KEY.



2. SQL Editor

Execute os scripts de criação de tabelas diretamente no painel.



3. Variáveis

Configure o arquivo `.env` com as credenciais do projeto.

>_ Inicie o servidor localmente com: `npm start`

Segurança (RLS)

Row Level Security (RLS) é fundamental para proteger seus dados. Por padrão, tabelas novas no Supabase são privadas.

🛡️ Configuração Inicial

Para desenvolvimento, habilitamos acesso público. Em produção, substitua por regras autenticadas.

```
CREATE POLICY "Allow public access"  
ON clientes FOR ALL USING (true);  
  
-- Repita para produtos e pedidos
```





API: Clientes

MÉTODO	ENDPOINT	DESCRIÇÃO
GET	/api/clientes	Listar todos os clientes cadastrados
GET	/api/clientes/:id	Buscar dados detalhados de um cliente
POST	/api/clientes	Cadastrar novo cliente (JSON)
PUT	/api/clientes/:id	Atualizar dados cadastrais
DELETE	/api/clientes/:id	Remover um cliente do sistema



API: Produtos

MÉTODO	ENDPOINT	DESCRIÇÃO
GET	/api/produtos	Listar catálogo completo
GET	/api/produtos/:id	Consultar detalhes de um produto
POST	/api/produtos	Adicionar novo item ao estoque
PUT	/api/produtos/:id	Atualizar preço, estoque ou descrição
DELETE	/api/produtos/:id	Descontinuar produto



API: Pedidos

MÉTODO	ENDPOINT	DESCRIÇÃO
GET	/api/pedidos	Histórico geral de pedidos
GET	/api/pedidos/:id	Detalhes de um pedido específico
POST	/api/pedidos	Criar pedido (Vincula Cliente + Itens)
PUT	/api/pedidos/:id	Atualizar status (ex: 'concluido')
DELETE	/api/pedidos/:id	Cancelar ou remover pedido

✨ Vantagens da Arquitetura



Cloud Native

Dados seguros, acessíveis e com backup automático de desastres.



PostgreSQL

Poder de um banco relacional robusto com escalabilidade horizontal.



API Instantânea

Integração facilitada com Frontend através de APIs REST geradas.



Solução de Problemas Comuns

✓ Erro: URL/KEY obrigatórios

Verifique se o arquivo `.env` está na raiz e se as variáveis estão nomeadas corretamente.

✓ Erro: Table does not exist

Confirme se o script SQL foi executado com sucesso no "SQL Editor" do Supabase.

✓ Erro: Policy does not exist

O RLS deve ser habilitado manualmente na aba "Table Settings" e as políticas criadas.

✓ Dados não aparecem

Verifique se a política RLS permite leitura pública ("Allow Select/All").



Backend Configurado!

Sua API está pronta para consumo. Consulte a documentação oficial para recursos avançados como Autenticação e Realtime.



supabase.com/docs



Best Practices

Image Sources



<https://kinsta.com/wp-content/uploads/2021/05/what-is-node-js-featured-image-1200x675.jpg>

Source: kinsta.com



https://png.pngtree.com/png-clipart/20250205/original/pngtree-a-digital-shield-with-lock-symbol-surrounded-by-various-technology-and-png-image_20323402.png

Source: [pngtree.com](https://png.pngtree.com)



<https://cdn.sanity.io/images/3do82whm/next/b88e6888748d4209f8c6746ac5507c673b2a01e9-1440x800.svg>

Source: www.sanity.io