IA-KPI – Documentação de Arquitetura e Contexto do Projeto

# 1. Visão Geral do Projeto

O IA-KPI é um sistema web que conecta bases de dados empresariais (principalmente MySQL), sincroniza tabelas selecionadas e permite criar, visualizar e customizar indicadores de performance (KPIs) por setor (Financeiro, Comercial, Produção etc). A arquitetura é pensada para BI self-service, flexível e totalmente personalizável, com capacidade de relacionamento de tabelas e visualização amigável.

# 2. Stack e Estrutura

- Backend: FastAPI (Python), SQLite para dados internos de controle, pymysql para conexão MySQL remota.  
- Frontend: ReactJS, componentes separados (Login, Cadastro, Dashboard, Sincronismo, Configuração etc).  
- Sincronismo: As tabelas do MySQL do cliente são lidas e copiadas para SQLite local apenas quando o usuário decide.  
- Relacionamento: Modelagem visual inspirada no Power BI, permitindo relacionamentos 1:1, 1:N, N:N via drag and drop.  
- KPIs: Painéis por setor, configuráveis, alimentados a partir das tabelas sincronizadas.  
- Segurança: Cada usuário só acessa sua própria configuração e base.  
- Arquivos de código: Separação clara backend/frontend, convenções modernas, documentação inline.

# 3. Fluxo Principal do Usuário

1. Cadastro/Login  
 - Usuário cria conta (salva no SQLite).  
 - Faz login, cai no Dashboard de indicadores (Financeiro, Comercial, Produção).

2. Configuração da Conexão  
 - Formulário salva dados de conexão MySQL.  
 - Dados persistem para reuso e autenticação.

3. Sincronismo de Tabelas/Views  
 - Usuário visualiza e seleciona tabelas/views do banco MySQL (trazidas via pymysql).  
 - Sincronismo executa cópia de estrutura/dados para SQLite local.  
 - Apenas tabelas/views marcadas ficam disponíveis para criação de indicadores e relacionamento.

4. Relacionamento Visual  
 - Interface drag-and-drop para criar/gerenciar relacionamentos entre tabelas sincronizadas.  
 - Mapeamentos persistidos em tabela `relacionamentos` (com tipo 1:1, 1:N, etc).

5. KPIs Básicos e Avançados  
 - Indicadores prontos por setor, usando SQL gerado automaticamente sobre a base SQLite sincronizada.  
 - Possibilidade de customizar e mapear indicadores específicos do negócio.

# 4. Pontos Críticos/Diferenciais

- Sem dependência de Streamlit: Sistema 100% web, independente de notebooks.  
- Sincronismo seguro: Nunca consulta o MySQL do cliente em tempo real; apenas importa tabelas/views autorizadas.  
- Relacionamento visual intuitivo: Usuário/admin pode ajustar relacionamentos como no Power BI.  
- Personalização: Setores, KPIs, mapeamentos – tudo customizável.

# 5. Comandos de Execução

Backend:

cd backend  
pip install -r requirements.txt  
python -m uvicorn main:app --reload

Frontend:

cd frontend  
npm install  
npm start

# 6. Observações Técnicas

- Arquivo de conexão: database.db (em backend)  
- Usuário SQLite: cada usuário tem campos próprios para salvar suas credenciais de banco remoto  
- Persistência: Todas configurações e logs de relacionamento, sincronismo e indicadores salvos no SQLite  
- Sincronização: Testada usando pymysql, suporte a views e tables do schema  
- Validação: Validação robusta dos dados de conexão e checagem da existência do usuário

# 7. Próximos Passos

- Melhorar interface de relacionamento visual  
- Habilitar criação dinâmica de KPIs customizados por arrasto  
- Logs de sincronismo e validação de estrutura dinâmica  
- Expor documentação OpenAPI sempre em /docs no backend

Se precisar reiniciar o projeto ou abrir um novo chat:

1. Cole esse conteúdo no início da conversa para garantir contexto total.  
2. Ou salve como .docx/.md e suba o arquivo no novo chat.