User Guide: Aplikasi Tanya Database Jersey dengan Next.js + OpenRouter GPT

# 1. Prasyarat dan Instalasi

- Node.js (disarankan versi LTS)  
- NPM (biasanya sudah ada dalam Node.js)  
- MySQL atau XAMPP untuk database  
- Koneksi internet untuk akses OpenRouter API

Langkah instalasi:

1. Clone atau buat project Next.js:

npx create-next-app@latest tanya-jersey

2. Masuk ke folder proyek:

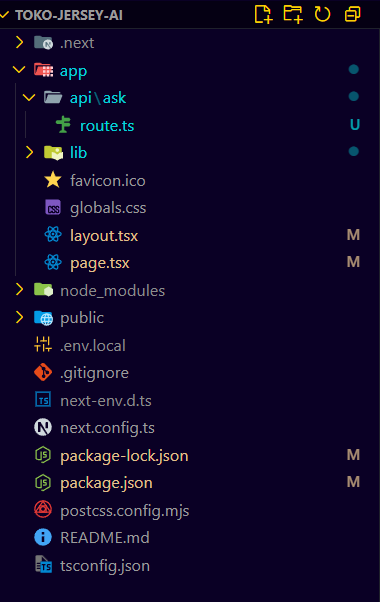
cd tanya-jersey

3. Install dependency yang dibutuhkan:

npm install mysql2 @langchain/openai openai @langchain/core

# 2. Struktur Folder

Struktur utama folder:  
  
app/  
 └─ page.tsx <- Halaman utama (frontend)  
 └─ api/ask/route.ts <- API route untuk memproses pertanyaan  
lib/  
 └─ db.ts <- Konfigurasi koneksi ke MySQL  
 └─ gpt.ts <- Konfigurasi GPT melalui OpenRouter  
.env.local <- Menyimpan API key dan kredensial database



# 3. Penjelasan Kode

## 3.1 file: lib/db.ts

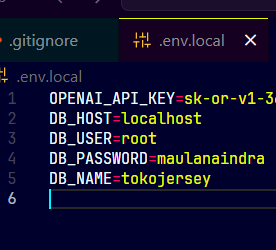
Berisi konfigurasi koneksi ke database MySQL menggunakan mysql2.

import mysql from 'mysql2/promise';  
export const db = await mysql.createConnection({  
 host: process.env.DB\_HOST,  
 user: process.env.DB\_USER,  
 password: process.env.DB\_PASSWORD,  
 database: process.env.DB\_NAME,  
});

## 3.2 file: lib/gpt.ts

Konfigurasi LangChain untuk menggunakan model GPT melalui OpenRouter.

Buat akun dulu di: <https://openrouter.ai/>, setelah buat aku kemudian akses link untuk mendapatkan API Key: <https://openrouter.ai/settings/keys>, setelah dapat copy dan tempel ke file .env.local



Untuk nama, password dan username database, di sesuaikan saja

Kemudian di file code seperti ini:

import { ChatOpenAI } from "@langchain/openai";  
export const gpt = new ChatOpenAI({  
 temperature: 0,  
 openAIApiKey: process.env.OPENROUTER\_API\_KEY,  
 configuration: { baseURL: "https://openrouter.ai/api/v1" },  
 modelName: "openai/gpt-3.5-turbo",  
});

## 3.3 file: app/api/ask/route.ts

Menerima pertanyaan dari frontend, menghasilkan query SQL via GPT, dan eksekusi ke database.

import { NextRequest, NextResponse } from "next/server";  
import { db } from "@/lib/db";  
import { gpt } from "@/lib/gpt";  
import { HumanMessage } from "@langchain/core/messages";  
  
export async function POST(req: NextRequest) {  
 const { question } = await req.json();  
 const prompt = `...`; // termasuk instruksi SQL & nama tabel  
 const messages = [new HumanMessage(prompt)];  
 const response = await gpt.call(messages);  
 const sql = response.content;  
 const [rows] = await db.execute(sql);  
 return NextResponse.json({ data: rows });  
}

## 3.4 file: app/page.tsx

Halaman antarmuka pengguna untuk mengirim pertanyaan dan melihat hasil.

'use client';  
import { useState } from "react";  
export default function Home() {  
 const [question, setQuestion] = useState("");  
 const [result, setResult] = useState<any[]>([]);  
 const [loading, setLoading] = useState(false);  
 const [error, setError] = useState("");  
 const askGPT = async () => { ... };  
 return (<main>...</main>);  
}

# 4. Menjalankan Aplikasi

1. Pastikan database MySQL aktif dan sudah diisi struktur tabel.  
2. Atur file `.env.local` dengan konfigurasi berikut:

DB\_HOST=localhost  
DB\_USER=root  
DB\_PASSWORD=  
DB\_NAME=tokojersey  
OPENROUTER\_API\_KEY=sk-or-xxxx

3. Jalankan aplikasi dengan:

npm run dev

4. Akses di browser: http://localhost:3000

# 5. Catatan Tambahan

- Gunakan pertanyaan natural seperti: "Tampilkan semua produk harga di atas 100"  
- Pastikan produk ditulis sesuai nama jika ingin query cocok