

Dokumentasi Teknis & Pengguna: SAPA-TAZKIA

Dokumentasi ini mencakup aspek teknis arsitektur sistem (Developer-side) dan panduan operasional (User-side) untuk platform chatbot RAG (Retrieval-Augmented Generation) SAPA-TAZKIA.

1. Dokumentasi Teknis (Technical Documentation)

A. Arsitektur Folder & Modul

Aplikasi ini dibangun menggunakan arsitektur **Service-Controller-Route** untuk memisahkan logika bisnis dari antarmuka API.

- **Backend (/backend):**
 - `src/services/`: Pusat logika bisnis. Contoh: `ragService.js` mengelola alur pengambilan data (Retrieval) dari JSON ke OpenAI.
 - `src/controllers/`: Menangani request HTTP dan memberikan response ke client.
 - `prisma/`: Mengelola skema database (`schema.prisma`) dan seeding data awal (`seed.js`).
 - `data/*.JSON`: *Knowledge base* lokal yang digunakan oleh sistem RAG untuk memberikan jawaban akurat seputar kampus.
 - `src/middleware/`: Keamanan sistem, termasuk `authMiddleware.js` (validasi token) dan `rateLimitMiddleware.js` (pencegahan spam API).
- **Frontend (/frontend):**
 - `src/context/AuthContext.js`: Mengelola state login pengguna di seluruh aplikasi.
 - `src/api/`: Konfigurasi `axiosConfig.js` untuk komunikasi dengan server backend.
 - `src/components/chat/`: Komponen modular UI untuk antarmuka percakapan.

B. Integrasi Basis Data & AI

- **Database:** Menggunakan PostgreSQL (atau SQL lainnya) yang diatur melalui **Prisma ORM**.
- **RAG Engine:** Mengintegrasikan **OpenAI API** dengan data lokal dari folder `backend/data/`.

- **Caching & Rate Limit:** Menggunakan **Redis** melalui `redisService.js` untuk memantau kuota penggunaan user secara real-time guna efisiensi biaya API.

C. Cara Instalasi (Technical Setup)

1. Clone Repository.

Backend:

```
Bash  
cd backend  
npm install  
npx prisma migrate dev # Sinkronisasi Database  
npx prisma db seed    # Mengisi data awal  
npm start
```

Frontend:

```
Bash  
cd frontend  
npm install  
npm run dev
```

2. Dokumentasi Pengguna (User Manual)

A. Akses Aplikasi

1. Buka browser dan akses URL aplikasi yang telah daring (Deployed).
2. Pengguna akan disambut oleh **Landing Page** yang berisi informasi singkat mengenai SAPA-TAZKIA.

B. Sistem Autentikasi

1. Klik tombol **Login**.
2. Gunakan akun yang telah terdaftar atau akun Google (sesuai konfigurasi `AuthModal.jsx`).
3. Sistem akan memvalidasi email melalui `emailDomainValidator.js` untuk memastikan pengguna adalah sivitas akademika yang sah.

C. Melakukan Percakapan (Chat)

1. Setelah login, pengguna diarahkan ke **ChatPage**.

2. Ketik pertanyaan di ChatInput.jsx, contoh: "Apa saja syarat pendaftaran program Magister?".
3. AI akan memproses jawaban berdasarkan data resmi dari file program_magister.JSON.
4. Status kuota chat harian dapat dilihat pada komponen RateLimitStatus.jsx.

D. Penanganan Masalah (Troubleshooting)

- **Gagal Kirim Pesan:** Cek apakah kuota harian Anda habis (dibatasi oleh rateLimitConfig.js).
 - **Data Tidak Muncul:** Pastikan koneksi internet stabil untuk mengambil data dari OpenAI.
-

Kesimpulan Modul

Modul yang dikembangkan telah memenuhi standar industri dengan pemisahan *concern* yang jelas, sistem keamanan berlapis (Auth & Rate Limit), serta pengelolaan data yang terstruktur melalui Prisma.