# PEMROGRAMAN API
# REST API MENGGUNAKAN NEXT.JS

**Dosen Pengampu:**

**Saiful Nur Budiman, M.Kom**


**Nama Kelompok:**

| | |
|---|---|
| **Rizqi Harisma Uchrowi** | **23104410001** |
| **Dimas Akbar Maulana** | **23104410056** |
| **Angga Novantara** | **24104410044** |
| **Andrecaesar Setiawan** | **23104410041** |
| **Intan Lusiana** | **23104410106** |
| **Enjing Dwi Retno Wulan** | **23104410049** |

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ISLAM BALITAR**

**2025**

# Pendahuluan

Proyek ini merupakan pengembangan API backend menggunakan framework Next.js versi 16.0.5. Nama proyek adalah "project-api-nextjs" dengan versi 0.1.0. Tujuan proyek ini adalah membangun sebuah sistem API yang aman, dengan fitur autentikasi dan otorisasi, serta integrasi database menggunakan Prisma. Proyek ini dirancang untuk mendukung manajemen data seperti students dan users, dengan penekanan pada keamanan akses berdasarkan role pengguna (misalnya, role ADMIN).

Proyek ini dibuat sebagai bagian dari Ujian Akhir Semester (UAS) kelompok, yang menunjukkan penerapan konsep-konsep seperti server-side rendering, middleware untuk keamanan, dan konfigurasi lingkungan development. Semua file konfigurasi dan dependencies telah disesuaikan untuk mendukung pengembangan yang efisien dan skalabel.

**Deskripsi Proyek**

Proyek ini adalah aplikasi API berbasis Next.js yang fokus pada backend. Fitur utama meliputi:

1. Autentikasi dan Otorisasi: Menggunakan JWT (JSON Web Token) untuk verifikasi token pengguna.
2. Proteksi Route API: Middleware memastikan hanya pengguna terautentikasi yang dapat mengakses route tertentu, dan role ADMIN diperlukan untuk akses ke /api/users.
3. Integrasi Database: Menggunakan Prisma sebagai ORM untuk berinteraksi dengan database (misalnya PostgreSQL via adapter PG).
4. Rate Limiting: Untuk mencegah abuse API.
5. Validasi Data: Menggunakan Zod untuk schema validation.

Proyek ini private (tidak untuk publikasi langsung) dan dioptimalkan untuk environment development, build, dan production.

**Teknologi dan Dependencies**

Berdasarkan file package.json dan package-lock.json, berikut adalah daftar dependencies utama yang digunakan:

**Dependencies Utama (Runtime):**

| Nama Package | Versi | Deskripsi |
|---|---|---|
| @prisma/adapter-pg | ^7.1.0 | Adapter Prisma untuk PostgreSQL. |
| @prisma/client | ^7.0.1 | Client untuk Prisma ORM, digunakan untuk query database. |
| bcryptjs | ^3.0.3 | Library untuk hashing password (keamanan autentikasi). |
| jose | ^6.1.2 | Library untuk JWT verification (menggantikan jsonwebtoken untuk performa lebih baik). |
| next | 16.0.5 | Framework utama untuk building API dan aplikasi web. |
| rate-limiter-flexible | ^9.0.0 | Untuk membatasi request API guna mencegah DDoS atau abuse. |
| react | 19.2.0 | Library UI (meskipun fokus backend, mungkin untuk komponen hybrid). |
| react-dom | 19.2.0 | DOM renderer untuk React. |
| zod | ^4.1.13 | Library untuk schema validation dan parsing data. |

**DevDependencies (Development):**

| Nama Package | Versi | Deskripsi |
|---|---|---|
| @tailwindcss/postcss | ^4 | Plugin PostCSS untuk Tailwind CSS. |
| @types/node | ^20 | Type definitions untuk Node.js. |
| @types/react | ^19 | Type definitions untuk React. |
| @types/react-dom | ^19 | Type definitions untuk React DOM. |
| baseline-browser-mapping | ^2.8.32 | Mapping untuk compatibility browser. |
| dotenv | ^17.2.3 | Untuk loading environment variables dari .env. |
| eslint | ^9 | Tool untuk linting code. |
| eslint-config-next | 16.0.5 | Konfigurasi ESLint khusus Next.js. |
| prisma | ^7.0.1 | CLI Prisma untuk migrasi dan generate schema. |
| tailwindcss | ^4 | Framework CSS utility-first. |

| | | |
|---|---|---|
| typescript | ^5 | Bahasa superset JavaScript untuk type safety. |

Total packages terinstal: Lebih dari 100 (berdasarkan package-lock.json), dengan lockfile version 3 untuk konsistensi instalasi.

**Struktur dan Konfigurasi Proyek**

Proyek ini memiliki konfigurasi standar Next.js dengan penambahan custom untuk keamanan dan database. Berikut ringkasan file utama:

1. **middleware.js**:
    1. Fungsi middleware untuk memverifikasi token JWT pada route API tertentu (/api/students/* dan /api/users/*).
    2. Menggunakan jose untuk verifikasi token dengan secret dari environment variable JWT_SECRET.
    3. Menambahkan header x-user-role dari payload token.
    4. Memblokir akses jika token invalid atau role bukan ADMIN untuk /api/users.
    5. Error handling: Mengembalikan response JSON dengan status 401 atau 403 jika gagal.
2. **eslint.config.mjs**:
    1. Konfigurasi ESLint menggunakan eslint-config-next untuk core web vitals dan TypeScript.
    2. Override ignores untuk file seperti .next/**, out/**, dll., agar linting fokus pada source code.
3. **.env.example**:
    1. Contoh environment variables: DATABASE_URL untuk koneksi database dan JWT_SECRET untuk JWT.
4. **next.config.ts**:
    1. File konfigurasi Next.js dasar (kosong, siap untuk custom seperti rewrites atau images).
5. **.gitignore**:

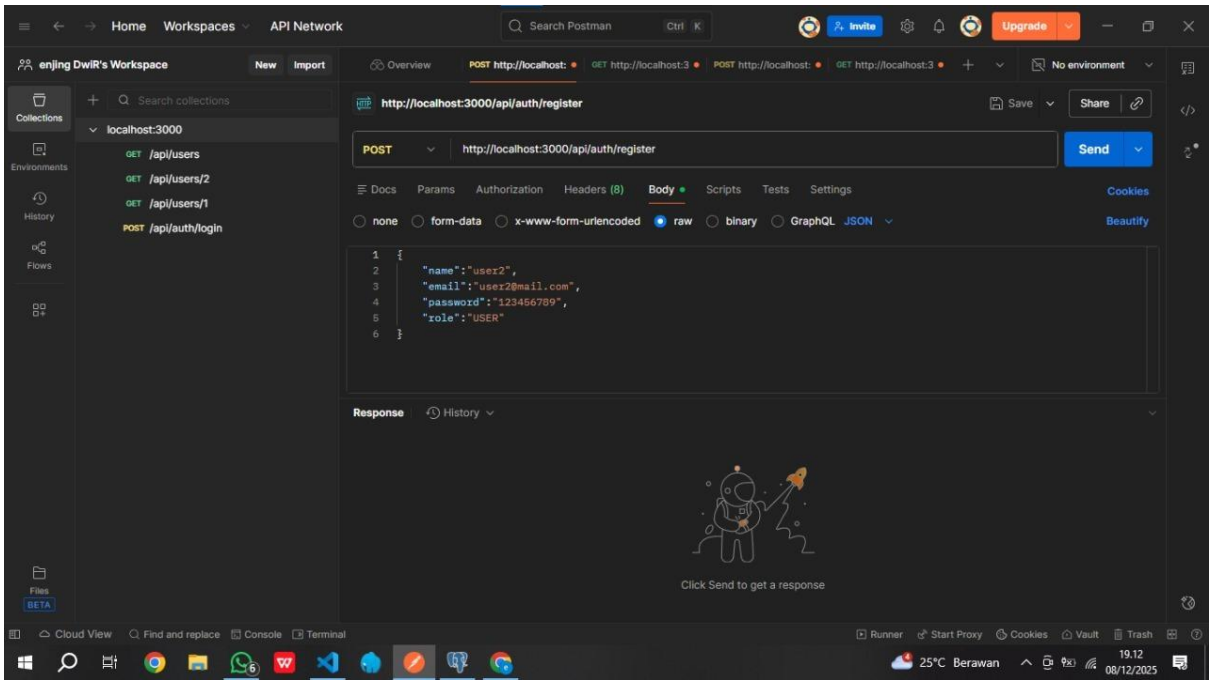
    1. Mengabaikan file seperti node_modules, .env, .next, dll., untuk menjaga repo bersih dari file sensitif atau generated.
6. **README.md**:
    1. Instruksi dasar untuk menjalankan proyek: npm run dev untuk development server di localhost:3000.
    2. Referensi ke dokumentasi Next.js dan deployment di Vercel.
7. **postcss.config.mjs**:
    1. Konfigurasi PostCSS dengan plugin Tailwind CSS untuk processing CSS.
8. **tsconfig.json**:
    1. Konfigurasi TypeScript: Target ES2017, include JSX, paths alias (@/* untuk root), dan exclude node_modules.
    2. Mendukung incremental compilation untuk performa.
9. **prisma.config.ts**:
    1. Konfigurasi Prisma: Schema di prisma/schema.prisma, migrations di prisma/migrations, dan datasource dari DATABASE_URL.

**Cara Menjalankan Proyek**

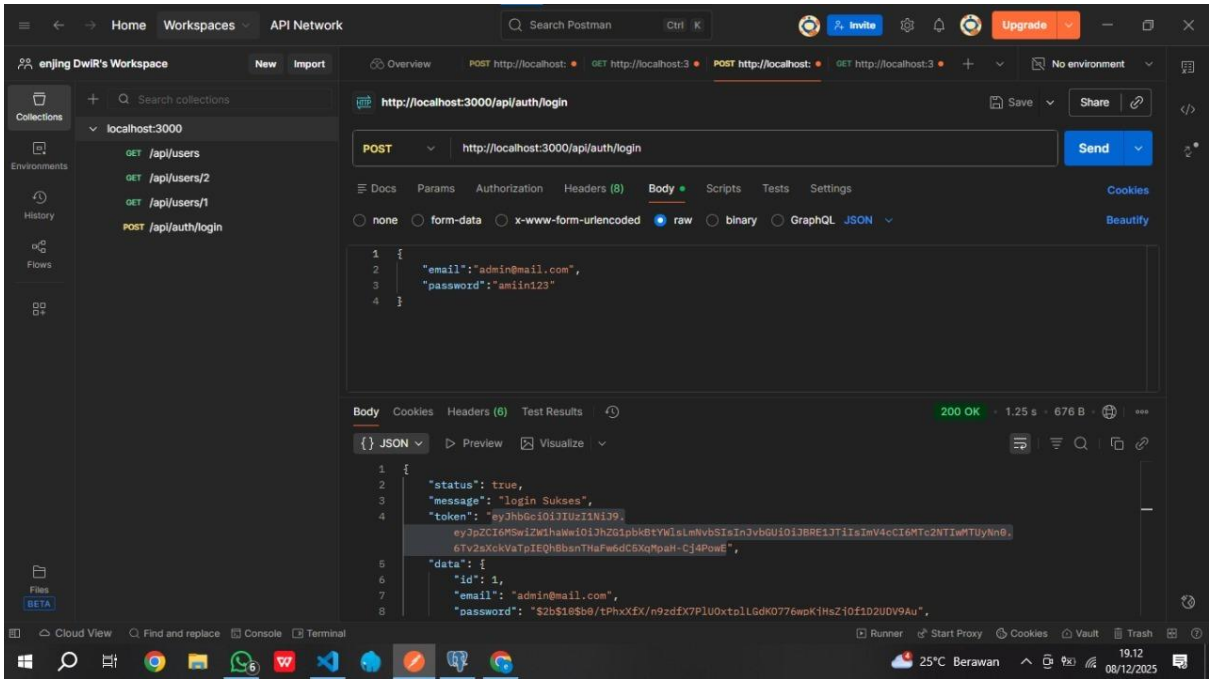
1. Install dependencies: npm install.
2. Setup environment: Copy .env.example ke .env dan  isi variabel.
3. Jalankan Prisma: npx prisma generate dan npx prisma migrate dev.
4. Development: npm run dev (server di http://localhost:3000).

## Screenshot Hasil Testing

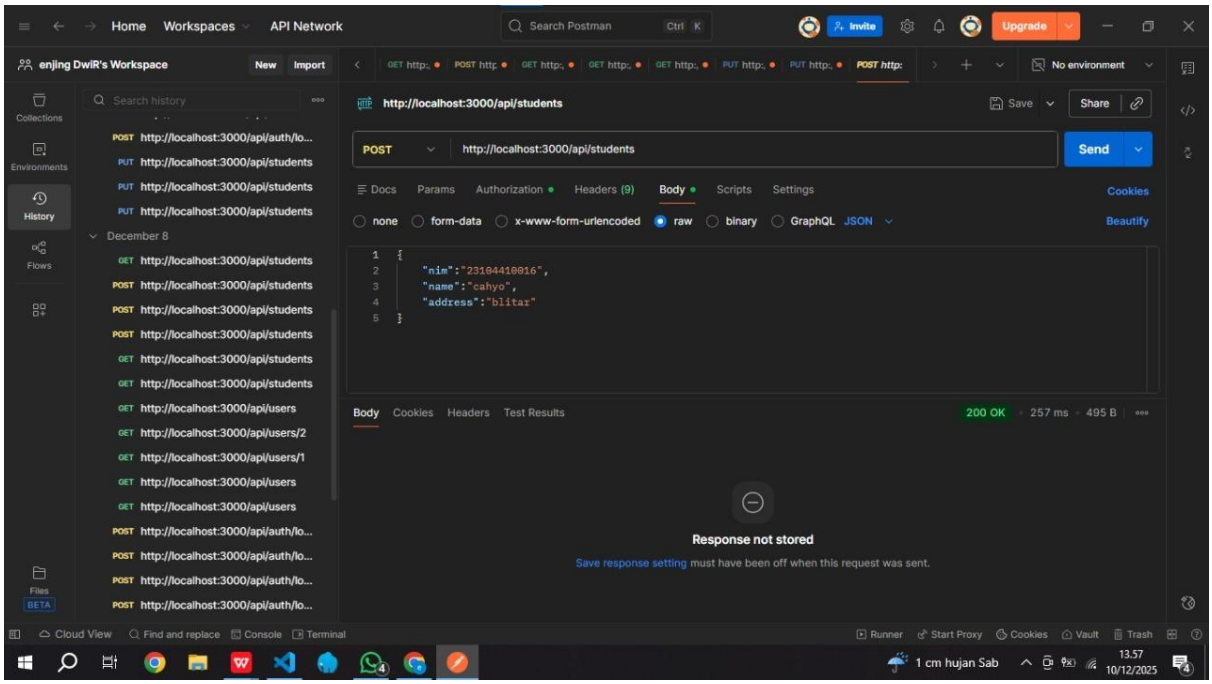
1. Menambahkan / mendaftar sebagai admin dan user



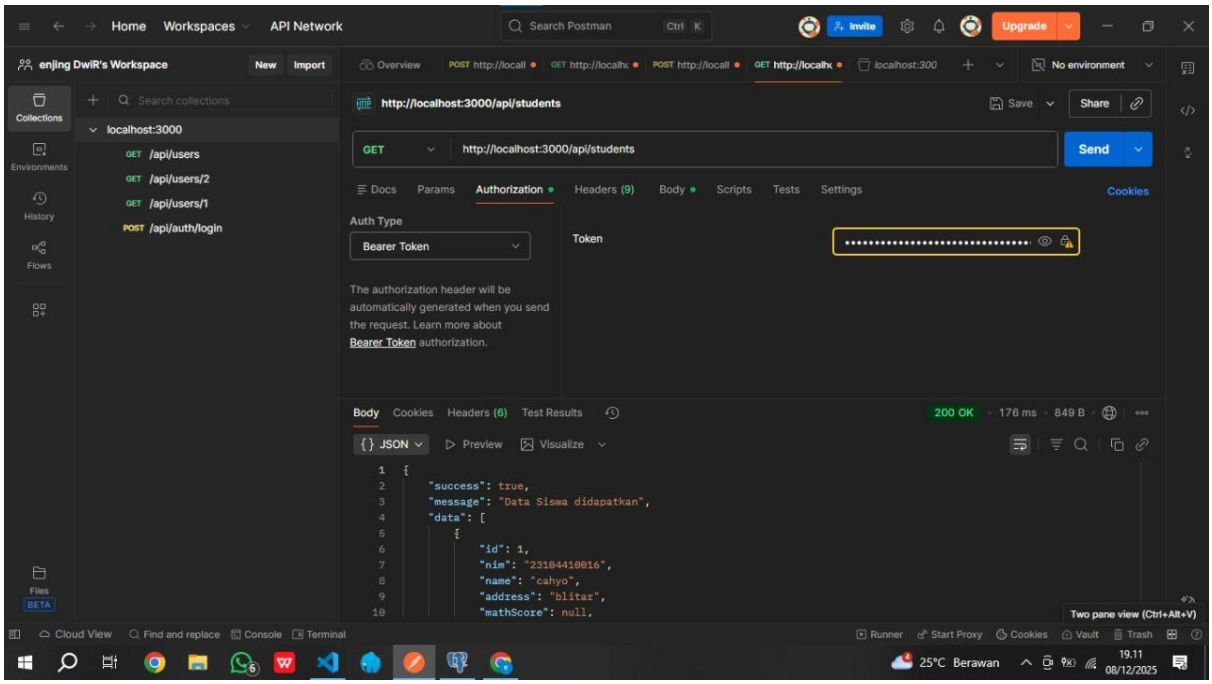

2. Login sebagai admin untuk mendapatkan token



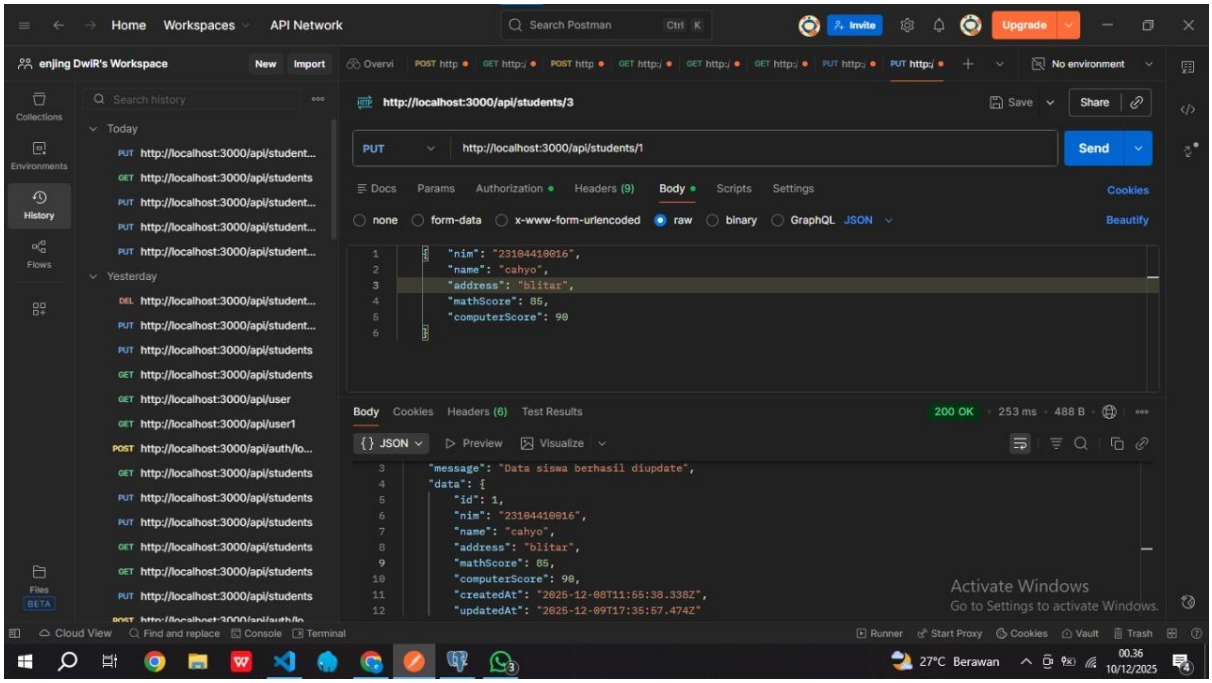
3. Menambahkan data student



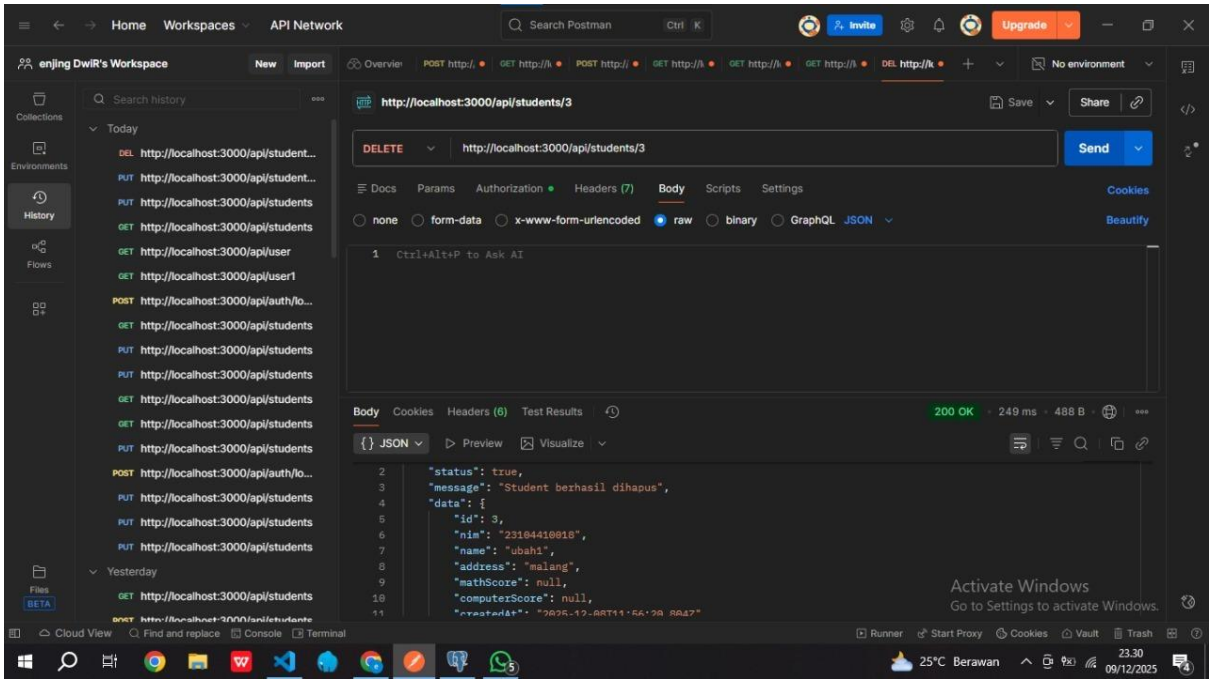

4. Mendapatkan semua data students



5. Memperbarui (Update) data student




6. Delete data student



## Kesimpulan

Proyek "project-api-nextjs" berhasil mengimplementasikan API backend yang aman dan scalable menggunakan Next.js. Ini mencakup autentikasi, otorisasi, dan integrasi database, yang sesuai dengan tujuan UAS kelompok untuk mendemonstrasikan kemampuan pengembangan web modern. Demikian laporan ini kami buat dengan sebenar-benarnya.