**Gambaran Umum**

Aplikasi Todo ini merupakan proyek full-stack yang sederhana namun fungsional, yang digunakan untuk mengelola daftar tugas. Backend dibangun menggunakan Node.js, Express, dan MongoDB, sedangkan frontend dikembangkan menggunakan Flutter. Aplikasi ini memungkinkan pengguna untuk membuat, melihat, memperbarui, dan menghapus tugas (todo).

**Fitur**

* **Operasi CRUD (Create, Read, Update, Delete):**
  + Pengguna dapat menambah tugas baru, melihat daftar tugas, memperbarui rincian tugas, dan menghapus tugas.
* **Integrasi dengan MongoDB:**
  + Data tugas disimpan di database MongoDB untuk memastikan penyimpanan yang persisten.
* **API RESTful:**
  + Backend menggunakan API RESTful yang dibangun dengan Express untuk menangani semua operasi CRUD.
* **Frontend Flutter:**
  + Antarmuka pengguna dikembangkan menggunakan Flutter, menyediakan tampilan yang interaktif dan responsif untuk pengelolaan tugas.
* **Update Data Real-Time:**
  + Aplikasi dirancang untuk mengambil dan memperbarui data terbaru secara real-time.
* **Cross-Platform:**
  + Frontend dapat dijalankan di perangkat Android dan iOS.

**Teknologi yang Digunakan**

* **Backend:**
  + Node.js
  + Express
  + MongoDB
  + Mongoose
  + CORS untuk mengatasi permintaan lintas asal (cross-origin)
* **Frontend:**
  + Flutter
  + Dart
  + dotenv untuk pengelolaan variabel lingkungan
* **Alat Lain:**
  + Postman (untuk pengujian API)
  + Git & GitHub (untuk kontrol versi)

**Struktur Proyek**

* **Backend (Node.js + Express):**
  + config/db.js: Mengonfigurasi dan menghubungkan ke MongoDB.
  + controllers/todoController.js: Menangani logika bisnis untuk operasi CRUD pada tugas.
  + models/Todo.js: Mendefinisikan skema untuk objek Todo yang disimpan di MongoDB.
  + routes/todos.js: Mendefinisikan rute API untuk operasi pada tugas.
  + index.js: Titik masuk utama backend, menginisialisasi server, menghubungkan ke database, dan menyiapkan rute.
* **Frontend (Flutter):**
  + lib/models/todo.dart: Mendefinisikan model Todo untuk aplikasi Flutter, dengan metode untuk mengonversi ke dan dari format JSON.
  + lib/main.dart: Titik masuk aplikasi Flutter, mengatur tema aplikasi dan layar utama.
  + lib/screens/home\_screen.dart: Antarmuka utama untuk mengelola tugas, menampilkan daftar todo, serta menyediakan opsi untuk menambah, memperbarui, atau menghapus tugas.

**Cara Menjalankan**

1. **Backend:**
   * Clone repository ini dan buka folder todo\_api.
   * Instal dependensi yang diperlukan: npm install.
   * Buat file .env di folder todo\_api dan tambahkan URI MongoDB Anda:

MONGO\_URI=<string\_koneksi\_mongodb\_anda>

* + Jalankan server: npm run dev (atau node index.js).
  + Backend akan berjalan di http://localhost:5001.

1. **Frontend:**
   * Buka folder todo\_app.
   * Pastikan Flutter sudah terinstal di komputer Anda.
   * Jalankan flutter pub get untuk menginstal dependensi.
   * Jalankan aplikasi dengan flutter run.

**Cara Menjalankan**

**Backend:**

1. Clone repositori dan navigasikan ke direktori todo\_api.
2. Install dependensi yang diperlukan: npm install.
3. Buat file .env di direktori todo\_api dan tambahkan URI MongoDB Anda: MONGO\_URI=<your\_mongodb\_connection\_string>.
4. Mulai server: npm run dev atau node index.js.
5. Backend akan berjalan di <http://localhost:5001>.

**Frontend:**

1. Navigasikan ke direktori todo\_app.
2. Pastikan Flutter sudah terpasang di komputer Anda.
3. Install dependensi: flutter pub get.
4. Jalankan aplikasi menggunakan: flutter run.