

LAPORAN PROJECT

Nama : Achmad nuril huda NIM : 362458302108 Kelas : TRPL 2E

Pendahuluan

Laporan ini dibuat untuk memenuhi tugas Ujian Akhir Semester, di mana mahasiswa berperan sebagai Vendor A (Warung Legacy). Vendor A dimaksudkan sebagai simulasi sistem warung lama yang masih menggunakan cara pengelolaan data sederhana dan belum menerapkan tipe data secara ketat. Sistem ini dibangun menggunakan Node.js dan Express sebagai backend, serta SQLite sebagai database. Data dikirim dan diterima dalam format JSON, sesuai dengan ketentuan yang terdapat pada soal UAS.

Pembahasan

1. Konsep Vendor A (Warung Legacy)

Vendor A menggambarkan sistem warung legacy atau sistem lama, di mana pencatatan data masih sederhana. Pada sistem ini, seluruh data produk disimpan dalam bentuk string, termasuk data harga. Hal ini bertujuan untuk menyesuaikan dengan karakteristik sistem lama yang belum memanfaatkan tipe data numerik secara optimal. Selain itu, informasi stok barang tidak ditampilkan dalam bentuk angka atau boolean, tetapi menggunakan keterangan teks yaitu “ada” dan “habis”, sebagaimana yang biasa digunakan pada pencatatan warung tradisional.

2. Struktur Data Vendor A

Data produk pada Vendor A disusun dalam format JSON. Setiap produk memiliki empat data utama, yaitu kode produk (kd_produk), nama barang (nm_brg), harga (hrg), dan keterangan stok (ket_stok). Nilai harga tetap disimpan sebagai string angka, sedangkan keterangan stok hanya menggunakan nilai “ada” atau “habis”. Struktur data tersebut sudah disesuaikan dengan format JSON yang diminta pada soal UAS.

3. Arsitektur Aplikasi

Aplikasi backend Vendor A terdiri dari beberapa file utama yang saling terhubung. File index.js digunakan sebagai file utama server. Pada file ini dilakukan proses inisialisasi server Express, penggunaan middleware untuk membaca data JSON, pemanggilan database, serta pengaturan route Vendor A dengan prefix /api/vendor-a. Pengolahan data produk tidak langsung ditulis di dalam index.js, melainkan dipisahkan ke dalam file vendorA.routes.js. Pemisahan ini dilakukan agar kode lebih terstruktur dan lebih mudah dipahami. Sementara itu, file db.js berfungsi untuk mengatur koneksi ke database SQLite. File ini akan dijalankan saat server aktif sehingga database sudah siap sebelum menerima request dari client.

4. Implementasi REST API Vendor A

REST API Vendor A menyediakan fitur CRUD (Create, Read, Update, Delete) untuk mengelola data produk. Proses Create digunakan untuk menambahkan data produk baru. Pada proses ini dilakukan pengecekan agar semua data wajib diisi, harga tetap berupa string angka, serta keterangan stok hanya berisi “ada” atau “habis”. Validasi ini diperlukan agar data yang masuk tetap sesuai dengan konsep sistem legacy. Proses Read digunakan untuk menampilkan data produk, baik seluruh data maupun data tertentu berdasarkan kode produk. Jika data yang diminta tidak ditemukan, sistem akan memberikan respons yang sesuai. Proses Update digunakan untuk mengubah data produk yang sudah ada berdasarkan kode produk. Perubahan hanya dapat dilakukan apabila data produk tersebut benar-benar ada di dalam database. Proses Delete digunakan untuk menghapus data produk berdasarkan kode produk tertentu. Jika data berhasil dihapus, sistem akan memberikan pesan konfirmasi, sedangkan jika data tidak ditemukan maka sistem akan memberikan respons error.

5. Penjelasan Kode vendorA.routes.js

File vendorA.routes.js digunakan untuk menangani seluruh request yang berkaitan dengan Vendor A. File ini memanfaatkan `express.Router()` agar setiap endpoint Vendor A terkelola dengan baik dan tidak tercampur dengan bagian lain dari sistem. Endpoint `GET /api/vendor-a` berfungsi untuk mengambil seluruh data produk yang tersimpan di database. Data yang diambil kemudian dikirimkan kembali ke client dalam bentuk JSON. Endpoint `GET /api/vendor-a/:kd_produk` digunakan untuk mengambil satu data produk berdasarkan kode produk. Kode produk diperoleh dari parameter URL. Jika data tidak ditemukan, sistem akan mengembalikan pesan bahwa produk tidak tersedia. Endpoint `POST /api/vendor-a` digunakan untuk menambahkan data produk baru. Pada endpoint ini dilakukan validasi untuk memastikan seluruh data telah diisi dan sesuai dengan format yang ditentukan, terutama pada data harga dan keterangan stok. Endpoint `PUT /api/vendor-a/:kd_produk` digunakan untuk memperbarui data produk yang sudah ada. Sistem akan mencari data berdasarkan kode produk, lalu memperbarui data jika produk tersebut ditemukan. Endpoint `DELETE /api/vendor-a/:kd_produk` digunakan untuk menghapus data produk berdasarkan kode produk. Jika penghapusan berhasil, sistem akan mengembalikan pesan bahwa data telah dihapus. Dengan adanya pemisahan endpoint tersebut, proses pengelolaan data produk menjadi lebih terstruktur dan mudah diuji.

Penutup

Dari hasil pembuatan dan pengujian sistem, dapat disimpulkan bahwa Vendor A berhasil diimplementasikan sebagai REST API menggunakan Express dan SQLite. Sistem ini mampu mensimulasikan warung legacy dengan format data berbasis string dan menyediakan fitur CRUD yang berjalan dengan baik.