

DOKUMENTASI TUGAS UAS BACKEND



Patrick Rivaldy/3124101301

**Program Studi D3 Manajemen Informatika
STIKOM PGRI Banyuwangi
Tahun 2025**

Dokumentasi Proyek Backend E-commerce

Dokumen ini menyediakan panduan lengkap untuk setup, konfigurasi, pengujian, dan pemahaman proyek backend e-commerce yang dibangun menggunakan Express.js dan MySQL.

1. Pendahuluan

Proyek backend ini adalah API RESTful sederhana yang dirancang untuk mendukung aplikasi e-commerce. Fitur utamanya meliputi:

- **Autentikasi Pengguna:** Pendaftaran dan login dengan JSON Web Tokens (JWT).
- **Otorisasi (Role-Based Access Control):** Membedakan akses antara pengguna biasa (user) dan administrator (admin). Hanya admin yang dapat melakukan operasi manajemen produk.
- **CRUD Produk:** Operasi dasar (Create, Read, Update, Delete) untuk mengelola data produk di database.
- **Database:** MySQL.

2. Teknologi yang Digunakan

- **Node.js:** Lingkungan runtime JavaScript.
- **Express.js:** Framework web untuk Node.js.
- **MySQL:** Sistem manajemen database relasional.
- **mysql2:** Driver MySQL untuk Node.js dengan dukungan promise.
- **dotenv:** Untuk mengelola variabel lingkungan.
- **bcryptjs:** Library untuk hashing password secara aman.
- **jsonwebtoken:** Untuk implementasi JSON Web Tokens (JWT).
- **cors:** Middleware untuk mengizinkan permintaan lintas-origin.
- **multer:** Middleware untuk menangani unggahan file multipart/form-data.
- **nodemon (opsional):** Alat untuk otomatis me-restart server saat ada perubahan kode.

3. Struktur Proyek

Proyek ini mengikuti struktur modular untuk memisahkan tanggung jawab dan meningkatkan keterbacaan kode.

backend/

```
|— config/
|   |— db.js      # Konfigurasi koneksi database MySQL
|— controllers/
|   |— authController.js # Logika bisnis untuk pendaftaran dan login pengguna
|   |— productController.js # Logika bisnis untuk operasi CRUD produk
|— middleware/
|   |— authMiddleware.js # Middleware untuk verifikasi token JWT dan otorisasi peran
|— models/
|   |— User.js      # Fungsi untuk berinteraksi dengan tabel 'users' di database
|   |— Product.js   # Fungsi untuk berinteraksi dengan tabel 'products' di database
|— public/
|   |— uploads/     # Folder untuk menyimpan gambar produk yang diunggah
```

```

├── routes/
│   ├── authRoutes.js  # Definisi endpoint API untuk autentikasi
│   └── productRoutes.js # Definisi endpoint API untuk produk
├── .env.example        # Contoh file variabel lingkungan
├── .gitignore          # File yang diabaikan oleh Git
├── app.js              # File utama aplikasi Express (titik masuk server)
├── package.json        # Metadata proyek dan daftar dependensi
└── README.md          # File dokumentasi proyek (ini)

```

4. Persyaratan Sistem

Sebelum memulai, pastikan Anda memiliki hal-hal berikut terinstal di sistem Anda:

- **Node.js:** Versi 14 atau lebih baru.
- **npm (Node Package Manager):** Biasanya terinstal bersama Node.js.
- **MySQL Server:** Versi 5.7 atau 8.x. Pastikan server berjalan.
- **Klien MySQL:** Alat seperti MySQL Workbench, phpMyAdmin, DBeaver, atau MySQL CLI untuk menjalankan skrip SQL.
- **Alat Pengujian API:** Seperti Postman, Insomnia, atau ekstensi VS Code (Thunder Client, REST Client).

5. Langkah-langkah Instalasi & Setup

Ikuti langkah-langkah ini untuk menyiapkan proyek backend Anda:

1. **Kloning Repositori (Jika dari GitHub) atau Buat Proyek Baru:** Jika Anda mengunduh proyek ini dari GitHub, kloning repositori:
2. `git clone <URL_REPOSITORI_ANDA> backend`
3. `cd backend`

Jika Anda membuat proyek dari awal, pastikan Anda berada di folder backend.

4. **Instal Dependensi:** Navigasi ke direktori proyek (backend) di terminal Anda dan instal semua dependensi:
5. `npm install`

6. Konfigurasi Variabel Lingkungan:

- Buat file baru bernama `.env` di root folder backend.
- Salin semua konten dari `.env.example` ke file `.env`.
- **Penting:** Ubah nilai-nilai berikut dengan kredensial MySQL Anda yang sebenarnya dan kunci rahasia yang kuat:
- `PORT=5000`
- `DB_HOST=localhost`
- `DB_USER=root`
- `DB_PASSWORD=your_mysql_password_here` # Ganti dengan password MySQL Anda
- `DB_NAME=e_commerce_db`
- `JWT_SECRET=your_very_strong_and_random_jwt_secret_key` # Ganti dengan string acak yang kuat

Anda bisa membuat JWT_SECRET dengan menjalankan openssl rand -base64 32 di terminal Linux/macOS.

6. Konfigurasi Database MySQL

Jika Anda belum melakukannya, ikuti langkah-langkah ini untuk membuat database dan tabel yang diperlukan. **Jika Anda sudah membuat database e_commerce_db dan tabel users serta products sebelumnya, Anda bisa melewati bagian ini.**

1. **Pastikan MySQL Server Berjalan:** Verifikasi bahwa server MySQL Anda aktif.
2. **Akses Klien MySQL:** Buka klien MySQL pilihan Anda (MySQL Workbench, phpMyAdmin, dll.) dan terhubung ke server MySQL Anda.
3. **Jalankan Skrip SQL:** Salin dan jalankan seluruh skrip SQL berikut di klien MySQL Anda.

-- SQL Script untuk Membuat Database dan Tabel

```
CREATE DATABASE IF NOT EXISTS e_commerce_db;  
USE e_commerce_db;
```

```
CREATE TABLE IF NOT EXISTS users (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  name VARCHAR(255) NOT NULL,  
  email VARCHAR(255) NOT NULL UNIQUE,  
  password VARCHAR(255) NOT NULL,  
  role ENUM('user', 'admin') DEFAULT 'user',  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
CREATE TABLE IF NOT EXISTS products (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  name VARCHAR(255) NOT NULL,  
  description TEXT,  
  price DECIMAL(10, 2) NOT NULL,  
  gender ENUM('Pria', 'Wanita', 'Unisex', 'Anak') NOT NULL,  
  image_url VARCHAR(255),  
  created_by INT,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  FOREIGN KEY (created_by) REFERENCES users(id) ON DELETE SET NULL  
);
```