

Nama : Norbertino Eurakha Nandatoti

NIM : 10231071

## Pemrograman Web

---



## Sistem Manajemen E-Commerce Sederhana

---



### Deskripsi Aplikasi




Sistem Manajemen E-Commerce Sederhana adalah aplikasi web yang memungkinkan pengguna untuk mengelola daftar produk secara efisien. Aplikasi ini menyediakan antarmuka yang intuitif untuk menampilkan, menambah, mengedit, dan menghapus produk dengan informasi seperti nama dan harga. Aplikasi ini dirancang dengan mempertimbangkan kemudahan penggunaan dan performa yang optimal.

---





### Teknologi yang Digunakan

#### Frontend

-  **React.js dengan TypeScript**
  - Menggunakan Vite sebagai build tool
  - TypeScript untuk type safety
  - React Hooks untuk state management
-  **Axios**
  - HTTP client untuk komunikasi dengan API
  - Penanganan error yang robust
-  **CSS**
  - Styling modern dengan flexbox
  - Responsive design
  - Animasi dan transisi

#### Backend

-  **Node.js & Express.js**
    - RESTful API architecture
    - Middleware untuk CORS dan error handling
  -  **MySQL**
    - Database relasional
    - Query yang dioptimalkan
- 



### Persyaratan Sistem

- Node.js v14 atau lebih tinggi

- MySQL v8.0 atau lebih tinggi
  - NPM v6 atau lebih tinggi
- 

## Instruksi Instalasi

### 1. Clone repository ini

```
git clone [URL_REPOSITORY]
cd proweb
```

### 2. Setup Database

- Buat database MySQL baru
- Import schema dari file `schema.sql` (jika tersedia)
- Konfigurasi kredensial di `db.js`

### 3. Install dependencies untuk frontend:

```
cd proweb-frontend
npm install
```

### 4. Install dependencies untuk backend:

```
cd proweb-backend
npm install
```

### 5. Konfigurasi Environment

- Buat file `.env` di folder backend
- Isi dengan konfigurasi yang diperlukan:

```
DB_HOST=localhost
DB_USER=your_username
DB_PASSWORD=your_password
DB_NAME=your_database
PORT=3001
```

### 6. Jalankan backend:

```
cd proweb-backend
npm start
```

## 7. Jalankan frontend:

```
cd proweb-frontend
npm run dev
```

## 📁 Struktur Proyek

```
proweb/
├── proweb-frontend/
│   ├── src/
│   │   ├── components/
│   │   │   └── ProdukList.tsx    # Komponen utama untuk manajemen produk
│   │   ├── App.tsx              # Komponen root
│   │   └── App.css               # Styling global
│   ├── public/                  # Asset statis
│   └── package.json             # Dependencies frontend
└── proweb-backend/
    ├── index.js                 # Entry point backend
    ├── db.js                    # Konfigurasi database
    └── package.json             # Dependencies backend
```

## 🔗 API Endpoints

### Produk

#### 1. GET /produk

- Mendapatkan semua produk
- Response: Array of products
- Status: 200 OK

#### 2. POST /produk

- Menambah produk baru
- Body: { nama: string, harga: number }
- Response: Created product
- Status: 201 Created

#### 3. PUT /produk/:id

- Mengupdate produk berdasarkan ID
- Body: { nama: string, harga: number }
- Response: Updated product
- Status: 200 OK

#### 4. DELETE /produk/:id

- Menghapus produk berdasarkan ID
  - Response: Success message
  - Status: 200 OK
- 

## ✦ Fitur-fitur Aplikasi

### 1. ✦ Fitur Tambah Produk



- **Form Input**

- Input field untuk nama produk
- Input field untuk harga produk
- Validasi input real-time
- Format harga otomatis
- Tombol submit yang responsif

- **User Experience**

- Form yang bersih dan mudah digunakan
- Validasi input yang jelas
- Feedback visual saat input
- Loading state saat proses submit
- Success/error notifications

- **Validasi**

- Nama produk tidak boleh kosong
- Harga harus berupa angka positif
- Maksimum karakter untuk nama
- Format harga yang sesuai
- Handling karakter khusus

### 2. 📋 Tampilan Daftar Produk



- **Layout & Desain**

- Grid layout untuk daftar produk
- Card design untuk setiap produk
- Spacing dan padding yang konsisten
- Warna yang kontras untuk keterbacaan

- **Fungsionalitas**

- Auto-refresh saat data berubah

- Sorting berdasarkan nama/harga
- Pagination untuk data yang banyak
- Search/filter produk

### 3. ✎ Fitur Edit Produk



- **Form Edit**

- Inline editing untuk UX yang lebih baik
- Validasi input real-time
- Format harga otomatis
- Preview perubahan sebelum save

- **User Experience**

- Loading state saat proses edit
- Success/error notifications
- Keyboard shortcuts (Esc untuk cancel)
- Auto-focus pada input pertama

### 4. 🗑️ Fitur Hapus Produk



- **Konfirmasi**

- Modal dialog konfirmasi
- Pesan yang jelas dan informatif
- Opsi untuk membatalkan
- Highlight produk yang akan dihapus

- **Feedback**

- Animasi fade-out saat dihapus
- Success notification
- Undo option (jika diperlukan)
- Error handling yang jelas

### 5. 📱 Responsive Design



- **Desktop (>1024px)**

- Grid layout 3 kolom
- Hover effects
- Full feature set

- **Tablet (768px - 1024px)**

- Grid layout 2 kolom
- Optimized spacing
- Touch-friendly buttons

- **Mobile (<768px)**

- Single column layout
- Stacked elements
- Touch-optimized interface
- Simplified navigation

---

## ☒ Quality Assurance Testing

### CRUD Operations Testing

#### 1. Create (Membuat Produk)

- ☒ Validasi input nama produk
- ☒ Validasi input harga (harus positif)
- ☒ Handling karakter khusus
- ☒ Feedback sukses/error
- ☒ Auto-refresh list setelah create

#### 2. Read (Membaca Produk)

- ☒ Loading state saat fetch data
- ☒ Error handling saat fetch gagal
- ☒ Empty state handling
- ☒ Format harga yang benar
- ☒ Pagination berfungsi

#### 3. Update (Mengupdate Produk)

- ☒ Validasi input saat edit
- ☒ Konfirmasi sebelum update
- ☒ Cancel edit berfungsi
- ☒ Real-time update di UI
- ☒ Error handling saat update gagal

#### 4. Delete (Menghapus Produk)

- ☒ Konfirmasi sebelum delete
- ☒ Animasi saat delete
- ☒ Error handling saat delete gagal
- ☒ UI update setelah delete
- ☒ Undo delete (jika diperlukan)

### UI/UX Testing

- ☒ Responsive di semua ukuran layar
- ☒ Loading states yang jelas
- ☒ Error messages yang informatif
- ☒ Success notifications
- ☒ Smooth transitions dan animations

## Performance Testing

- ☒ Fast initial load
- ☒ Smooth scrolling
- ☒ Efficient data fetching
- ☒ Optimized re-renders
- ☒ Memory leak prevention

---

## Kesimpulan

Sistem Manajemen Produk ini menyediakan solusi yang efisien dan user-friendly untuk mengelola data produk. Dengan kombinasi teknologi modern dan best practices dalam pengembangan web, aplikasi ini menawarkan:

- ⚡ Performa yang optimal
- 👤 User experience yang intuitif
- 🔒 Keamanan data yang terjamin
- 🛠️ Kemudahan maintenance
- 📈 Skalabilitas untuk pengembangan lebih lanjut

---

## Troubleshooting

### 1. 📄 Error Koneksi Database

Jika muncul error "Cannot connect to database":

- Pastikan MySQL server sudah berjalan
- Periksa konfigurasi database di `db.js`
- Pastikan kredensial database benar
- Cek firewall settings
- Verifikasi port MySQL (default: 3306)

### 2. 🌐 Error CORS

Jika muncul error CORS:

- Pastikan backend server berjalan di port 3001
- Periksa konfigurasi CORS di backend
- Pastikan URL API di frontend benar
- Cek header 'Access-Control-Allow-Origin'
- Verifikasi method HTTP yang diizinkan

### 3. Error TypeScript

Jika muncul error TypeScript:

- Jalankan `npm install` untuk memastikan semua dependencies terinstall
- Periksa tipe data di interface `Produk`
- Pastikan semua props memiliki tipe yang sesuai
- Update TypeScript ke versi terbaru
- Clear TypeScript cache

### 4. Error Loading Data

Jika data tidak muncul:

- Periksa koneksi internet
- Pastikan backend server berjalan
- Periksa console browser untuk error detail
- Verifikasi format response API
- Cek network tab di DevTools

### 5. Error Build

Jika terjadi error saat build:

- Clear node\_modules dan package-lock.json
- Jalankan `npm cache clean --force`
- Reinstall dependencies
- Update Node.js ke versi LTS
- Periksa versi dependencies yang kompatibel