

Webpro-1727

This project is a simple full-stack e-commerce application built as part of the **Web Programming '23 Project**. It consists of a backend powered by Node.js and Express, and a frontend built with Vite + React. The application allows users to manage products (add, view, update, and delete).

Developed by:

- **Dahayu Azhka Daeshawnda (10231027)**
 - **Ansellma Tita Pakartiwuri Putri (10231017)**
-

Table of Contents

1. [How to Run the Project](#)
 2. [Project Structure](#)
 3. [API Endpoint List](#)
 4. [Technologies Used](#)
 5. [Screenshots](#)
 6. [Contribution](#)
-

How to Run the Project

Prerequisites

Before running this project, ensure you have the following installed:

- **Node.js**: Version 16 or higher ([Download Node.js](#))
- **npm**: Node Package Manager (comes bundled with Node.js)
- **Git**: For cloning the repository ([Download Git](#))

Usage Steps

1. Clone the Repository

Clone this repository to your local machine:

```
git clone https://github.com/wounderfv1/webpro-1727.git
cd webpro-1727
```

2. Install Dependencies

Install all required dependencies for both the backend and frontend:

Backend

```
cd backend
npm install
```

Frontend

```
cd ../frontend/eccommerce-frontend
npm install
```

3. Run the Backend Application

Start the backend server:

```
node index.js
```

The backend will run on port **3001**. You can access it at:

```
http://localhost:3001
```

4. Run the Frontend Application

Start the frontend development server:

```
npm run dev
```

The frontend will run on port **5173**. You can access it at:

```
http://localhost:5173
```

Project Structure

The project is organized into two main directories: **backend** and **frontend**.

```
webpro-1727/
├── backend/
│   ├── index.js
│   ├── package.json
│   └── routes/
└── frontend/
```

```
├── ecommerce-frontend/
│   ├── src/
│   │   ├── components/
│   │   ├── App.jsx
│   │   └── index.css
│   ├── package.json
│   └── vite.config.js
└── README.md
```

API Endpoint List

Below is a list of available API endpoints in the backend:

1. GET /api/produk

- **Description:** Retrieves a list of all products.
- **Example Response:**

```
[
  { "id": 3, "nama": "Azhka", "harga": 0 },
  { "id": 4, "nama": "Ansel", "harga": 0 }
]
```

2. POST /api/products

- **Description:** Adds a new product.
- **Request Body:**

```
{
  "name": "Azhka",
  "harga": "0"
}
```

- **Example Response:**

```
{
  "id": 3,
  "name": "Azhka",
  "harga": "0"
}
```

3. GET /api/products/:id

- **Description:** Retrieves product details by ID.

- **Example Response:**

```
{
  "id": 1,
  "name": "Alhaitham",
  "harga": "0"
}
```

4. PUT /api/products/:id

- **Description:** Updates product data by ID.
- **Request Body:**

```
{
  "name": "Alhaitham C6R5",
  "harga": "0"
}
```

- **Example Response:**

```
{
  "id": 1,
  "name": "Alhaitham C6R5",
  "harga": "0"
}
```

5. DELETE /api/products/:id

- **Description:** Deletes a product by ID.
- **Example Response:**

```
{
  "message": "Product deleted successfully"
}
```

Technologies Used

This project leverages the following technologies:

Backend

- **Node.js:** JavaScript runtime environment.
- **Express.js:** Minimalist web framework for building APIs.

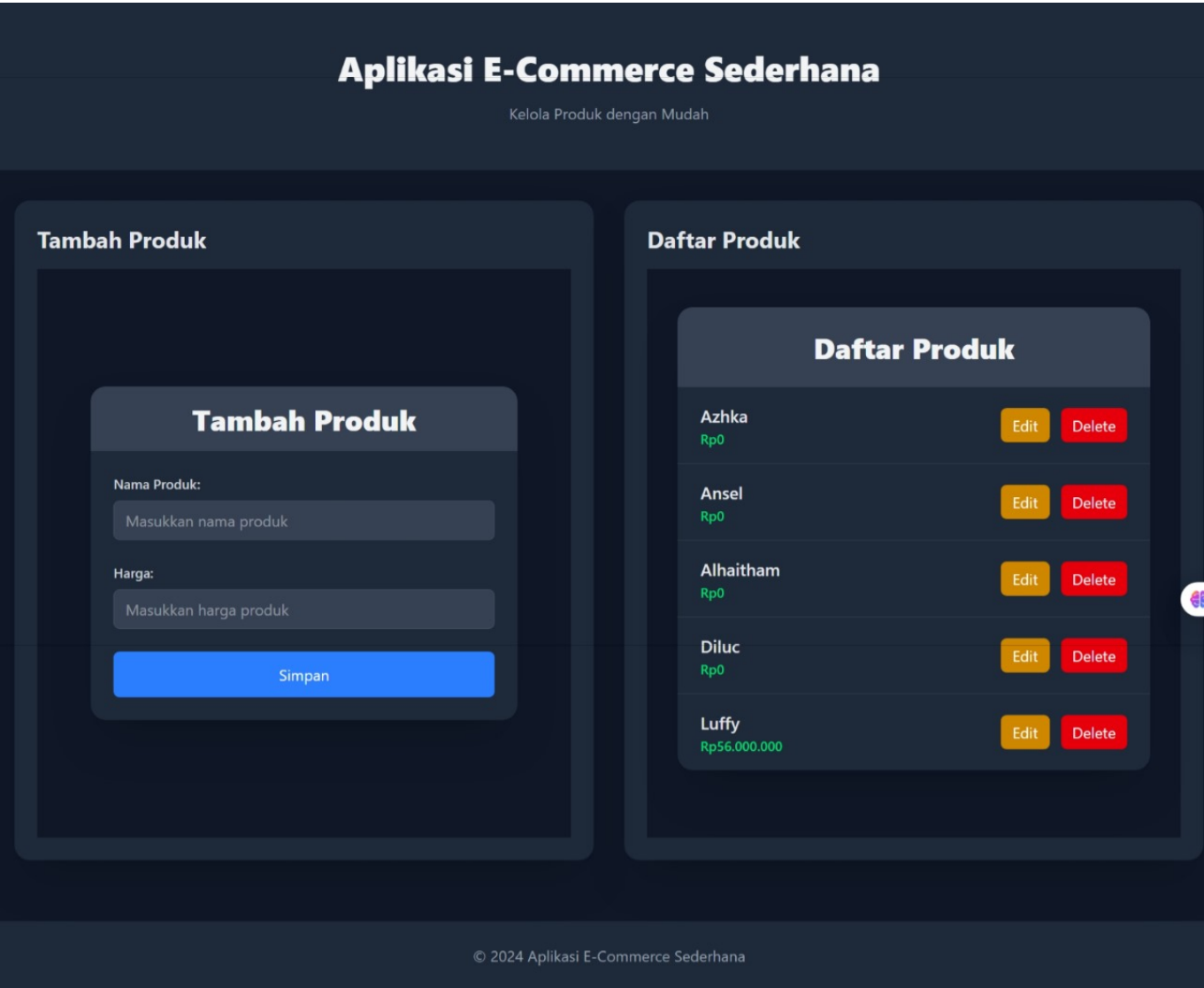
Frontend

- **React:** JavaScript library for building user interfaces.
- **Vite:** Fast build tool and development server for modern web projects.
- **Tailwind CSS:** Utility-first CSS framework for styling.

Screenshots

Here are some screenshots of the application:

1. Home Page



The home page displays a list of products and provides options to add, edit, or delete products.

2. Add Product Form



Tambah Produk

Nama Produk:

Masukkan nama produk

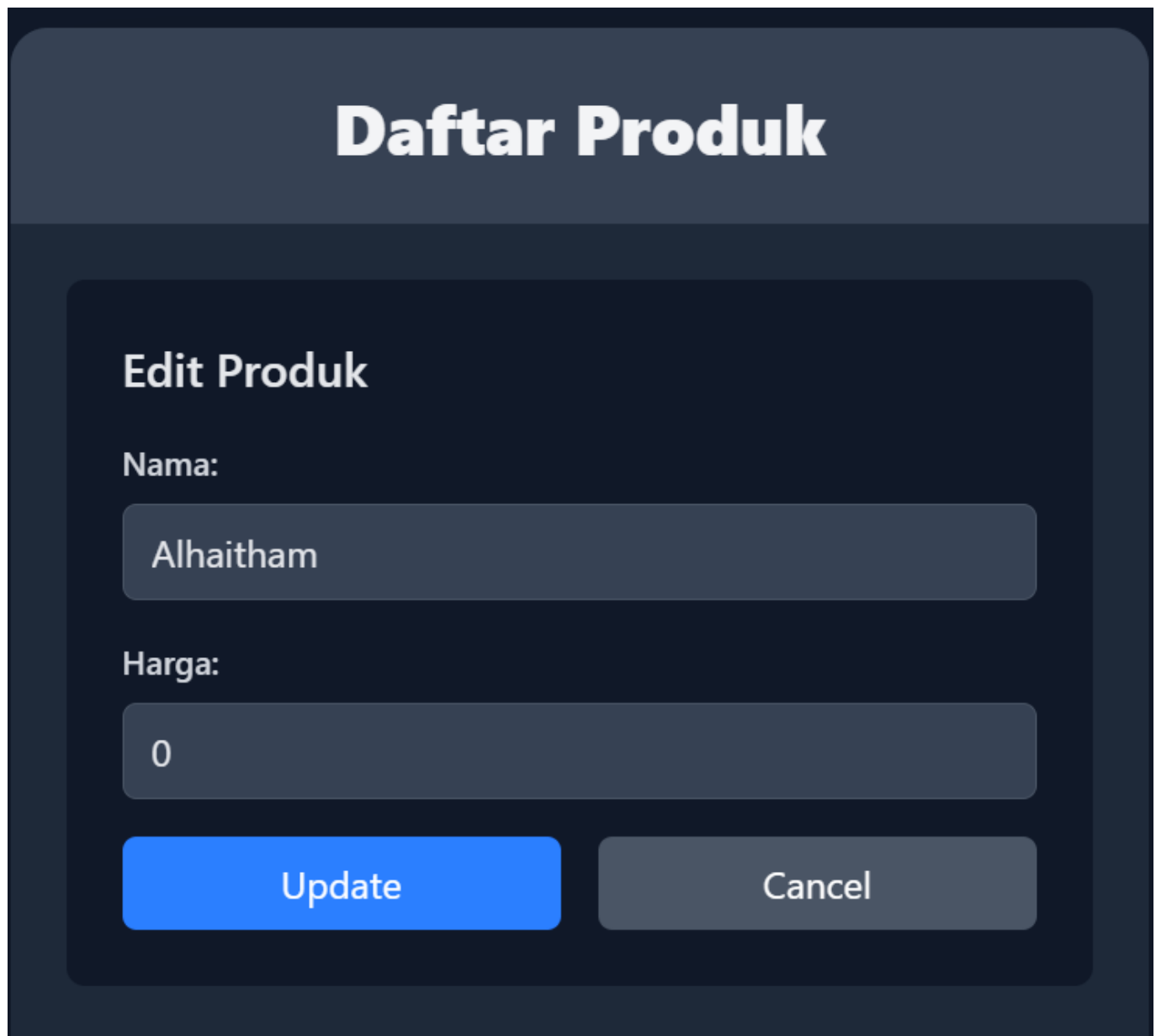
Harga:

Masukkan harga produk

Simpan

A form for adding new products with fields for name and price.

3. **Edit Product Form**



Daftar Produk

Edit Produk

Nama:

Harga:

Update Cancel

A form for updating existing product details.

Contribution

We welcome contributions! If you would like to contribute, please follow these steps:

1. Fork the repository.
2. Create a new branch (`git checkout -b feature/YourFeatureName`).
3. Commit your changes (`git commit -m "Add YourFeatureName"`).
4. Push to the branch (`git push origin feature/YourFeatureName`).
5. Submit a pull request with a clear explanation of the changes.

If you encounter any issues or have suggestions, feel free to open an issue in this repository.
