

MODUL PERTEMUAN 7
PRAKTIKUM FULLSTACK DEVELOPMENT

“Menghubungkan Backend Express Database MongoDB dengan Frontend React ”



SEKOLAH TINGGI ILMU KOMPUTER PGRI BANYUWANGI

STIKOM BANYUWANGI

2024

A. Pendahuluan

Untuk modul kali ini kita kan membahas backend dengan menggunakan express.js dan database noSQL MongoDB sebagai tempat penyimpananya. Modul ini bertujuan untuk memberikan pemahaman dasar tentang cara membuat backend menggunakan Express dan menghubungkannya dengan database MongoDB. Peserta diharapkan dapat melakukan operasi CRUD (Create, Read, Update, Delete) pada dokumen `mahasiswa` dalam koleksi `siakad` yang digunakan dalam contoh dibawah dan mengembangkan lebih lanjut.

B. Persiapan/Prasyarat

1. **Node.js dan npm** sudah terinstal di komputer Anda.
2. **MongoDB server** sudah terinstal dan berjalan di komputer Anda.
3. **MongoDB Compass** untuk mempermudah proses pembuatan database.
4. **Text editor** (misalnya VSCode) atau IDE yang mendukung JavaScript.
5. Komputer dengan sistem operasi Windows, MacOS, atau Linux.
6. Koneksi internet untuk mengunduh dependensi.

C. Langkah-langkah Praktikum

1. Bagian 1: Persiapan Lingkungan

a) Instalasi Node.js dan npm

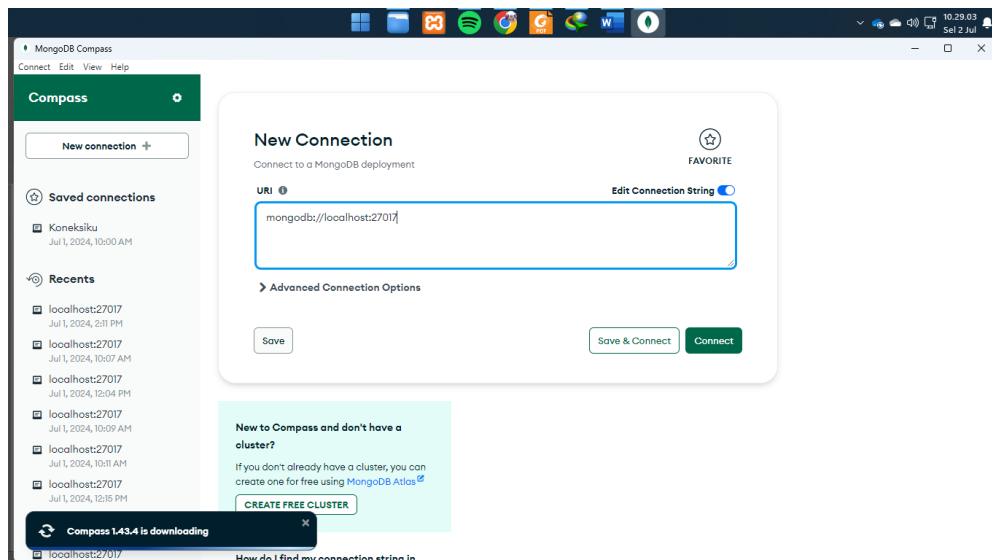
Pastikan Node.js dan npm telah terinstal. Anda dapat memeriksanya dengan perintah berikut:

Perintah pada CMD

```
node -v
npm -v
```

b) Instalasi MongoDB dan MongoDBCompass

Untuk mengecek apakah MongoDB sudah terinstall dapat melakukan koneksi MongoDBCompass ke server MongoDB dengan Alamat : `mongodb://localhost:27017` kemudian klik connect



Jika sudah terkoneksi maka sudah siap untuk digunakan

2. Bagian 2: Pembuatan Proyek react

- Membuat direktori proyek
- Buat proyek dengan cmd dan ketikkan perintah **"npm create vite@latest"** ikuti perintah pastikan menggunakan react dan Bahasa pemrograman javascript
- Masuk kedalam root proyek dan ketikkan perintah berikut **"npm install"** untuk memanggil semua dependencies yang digunakan
- Tambahkan axios untuk mengakses json dari backend dengan menggunakan perintah **"npm install axios"**
- Buatlah struktur project seperti berikut :

```
vite-react-mahasiswa/
|
├─ src/
|   ├─ components/
|   |   └─ MahasiswaList.js
|   |   └─ MahasiswaDetail.js
|   └─ App.js
|       └─ main.jsx
|           └─ index.css
|
├─ public/
|   └─ index.html
|
├─ vite.config.js
├─ package.json
└─ .gitignore
```

3. Bagian 3: Implementasi frontend

- Buka file src/components/MahasiswaList.js :

```
src/components/MahasiswaList.js:
import React, { useEffect, useState } from 'react';
import axios from 'axios';

const MahasiswaList = () => {
  const [mahasiswa, setMahasiswa] = useState([]);

  useEffect(() => {
    const fetchData = async () => {
      try {
        const response = await
        axios.get('http://localhost:3000/mahasiswa');
        setMahasiswa(response.data);
      } catch (error) {
        console.error('Error fetching data:', error);
      }
    };

    fetchData();
  });
}
```

```

    }, []);

    return (
      <div>
        <h1>Daftar Mahasiswa</h1>
        <ul>
          {mahasiswa.map((mhs) => (
            <li key={mhs._id}>
              {mhs.nama} - {mhs.jurusan} ({mhs.angkatan})
            </li>
          ))}
        </ul>
      </div>
    );
  };
};

export default MahasiswaList;

```

- b) Buka file `src/components/MahasiswaDetail.js` (opsional jika Anda ingin menampilkan detail mahasiswa):

```

src/components/MahasiswaDetail.js
import React, { useEffect, useState } from 'react';
import axios from 'axios';
import { useParams } from 'react-router-dom';

const MahasiswaDetail = () => {
  const { id } = useParams();
  const [mahasiswa, setMahasiswa] = useState(null);

  useEffect(() => {
    const fetchData = async () => {
      try {
        const response = await
        axios.get(`http://localhost:3000/mahasiswa/${id}`);
        setMahasiswa(response.data);
      } catch (error) {
        console.error('Error fetching data:', error);
      }
    };

    fetchData();
  }, [id]);

  if (!mahasiswa) return <div>Loading...</div>;

  return (
    <div>
      <h1>{mahasiswa.nama}</h1>
      <p>Jurusan: {mahasiswa.jurusan}</p>
      <p>Angkatan: {mahasiswa.angkatan}</p>
      <p>Email: {mahasiswa.email}</p>
    </div>
  );
};

export default MahasiswaDetail;

```

c) Lakukan Edit file src/App.js:

```
src/App.js:
import React from 'react';
import MahasiswaList from '../components/MahasiswaList';
// import MahasiswaDetail from '../components/MahasiswaDetail'; //
Jika menggunakan

const App = () => {
  return (
    <div className="App">
      <MahasiswaList />
      { /* Tambahkan route detail jika perlu */ }
    </div>
  );
};

export default App;
```

d) Jalankan server

Perintah pada CMD

```
node run dev
```

Nb:

- 🚀 Untuk pengguna yang sudah menginstall nodemon dapat menjalankan dengan perintah : ***"nodemon run dev"***
- 🚀 Jangan lupa untuk cek database mongodb agar aplikasi dapat mengakses data yang ada dalam database.
- 🚀 Pastikan backend juga sudah dijalankan

4. Bagian 4: Pengujian Frontend

Bukalah browser dan ketikkan Alamat berikut : <http://localhost:3000>

5. Praktikum

- a. Buatkan route untuk halaman kedua untuk menampilkan halaman data matakuliah.
- b. Lakukan percobaan dengan browser untuk mengakses halaman tersebut
- c. Buatlah laporan praktikumnya