# MODUL PERKULIAHAN

**Mata Kuliah** : Pengembangan Aplikasi Web  
**Materi** : Pengenalan Pengembangan Aplikasi Web dengan React dan Node.js  
**Pertemuan** : 01  
**Fakultas** : Teknik  
**Program Studi** : Teknologi Informasi  
**Kode MK** : AntaraTI501P  
**Disusun Oleh** : Ir.Asroni, S.T., M.Eng

## DAFTAR ISI

1. [PEMBAHASAN](#pembahasan)
2. [PREREQUISITES](#prerequisites)
3. [PRAKTIKUM](#praktikum)
   * [A. Tujuan Praktikum](#a-tujuan-praktikum)
   * [B. Materi](#b-materi)
   * [C. Metode Praktikum](#c-metode-praktikum)
   * [D. Langkah-langkah Praktikum](#d-langkah-langkah-praktikum)
4. [TUGAS](#tugas)

## PEMBAHASAN

1. Pengenalan Pengembangan Aplikasi Web Modern:
   * Konsep full-stack development
   * Arsitektur aplikasi web modern (client-side dan server-side)
   * Tren terkini dalam pengembangan web
2. React.js:
   * Pengenalan React sebagai library JavaScript untuk membangun antarmuka pengguna
   * Konsep komponen dan state dalam React
   * Virtual DOM dan keunggulan React dalam performa
3. Node.js:
   * Pengenalan Node.js sebagai runtime JavaScript di sisi server
   * Event-driven, non-blocking I/O model Node.js
   * Penggunaan Node.js untuk backend development
4. Integrasi React dan Node.js:
   * Arsitektur aplikasi web dengan React di frontend dan Node.js di backend
   * RESTful API dan komunikasi client-server
   * Keuntungan penggunaan JavaScript di seluruh stack
5. Ekosistem dan Tools:
   * NPM (Node Package Manager) untuk manajemen dependensi
   * Create React App untuk memulai proyek React
   * Express.js sebagai framework web untuk Node.js
6. Dasar-dasar JavaScript Modern:
   * ES6+ features yang sering digunakan dalam React dan Node.js
   * Asynchronous programming dengan Promises dan async/await
7. Persiapan Lingkungan Pengembangan:
   * Instalasi Node.js, NPM, dan Create React App
   * Pengaturan editor kode dan developer tools

## PREREQUISITES

Beberapa perangkat lunak yang digunakan untuk menunjang proses pembelajaran:

1. Laptop/komputer
2. Node.js (versi LTS terbaru)
   * Unduh dari https://nodejs.org/en/download/
3. Visual Studio Code atau editor teks lainnya
   * Unduh dari https://code.visualstudio.com/
4. Terminal atau Command Prompt
5. Git (version control)
   * Unduh dari https://git-scm.com/downloads

## PRAKTIKUM

### A. Tujuan Praktikum

Praktikum ini bertujuan untuk memperkenalkan mahasiswa pada pengembangan aplikasi web modern menggunakan React.js untuk frontend dan Node.js untuk backend. Mahasiswa akan memahami konsep dasar kedua teknologi ini dan bagaimana mereka bekerja bersama dalam pengembangan aplikasi web full stack.

### B. Materi

1. Pengenalan React.js dan Node.js
2. Instalasi dan konfigurasi lingkungan pengembangan
3. Membuat proyek React menggunakan Create React App
4. Membuat server Node.js sederhana
5. Integrasi frontend React dengan backend Node.js

### C. Metode Praktikum

1. Penjelasan materi oleh instruktur
2. Demonstrasi langsung penggunaan React.js dan Node.js
3. Praktik mandiri oleh mahasiswa
4. Diskusi dan tanya jawab

### D. Langkah-langkah Praktikum

1. Instalasi Node.js
   * Kunjungi situs resmi Node.js: https://nodejs.org/
   * Unduh versi LTS (Long Term Support) terbaru
   * Ikuti langkah-langkah instalasi sesuai sistem operasi Anda
   * Verifikasi instalasi dengan menjalankan perintah node -v dan npm -v di terminal
2. Membuat Proyek React
   * Buka terminal dan jalankan perintah:
   * npx create-react-app my-react-app  
     cd my-react-app  
     npm start
   * Buka browser dan akses http://localhost:3000 untuk melihat aplikasi React default
3. Membuat Komponen React Sederhana
   * Buka file src/App.js dan ubah kontennya menjadi:
   * import React from 'react';  
       
     function App() {  
      return (  
      <div>  
      <h1>Selamat Datang di Aplikasi React</h1>  
      <p>Ini adalah komponen React sederhana.</p>  
      </div>  
      );  
     }  
       
     export default App;
   * Simpan file dan lihat perubahannya di browser
4. Membuat Server Node.js Sederhana
   * Buat folder baru untuk proyek Node.js:
   * mkdir my-node-server  
     cd my-node-server
   * Inisialisasi proyek Node.js:
   * npm init -y
   * Buat file server.js dan isi dengan kode berikut:
   * const express = require('express');  
     const app = express();  
     const port = 5000;  
       
     app.get('/', (req, res) => {  
      res.json({ message: 'Hello from Node.js Server!' });  
     });  
       
     app.listen(port, () => {  
      console.log(`Server running on http://localhost:${port}`);  
     });
   * Jalankan npm install express
   * Jalankan server dengan perintah node server.js
5. Integrasi React dengan Node.js
   * Di file src/App.js di proyek React, tambahkan fungsi untuk mengambil data dari server:
   * import React, { useEffect, useState } from 'react';  
       
     function App() {  
      const [message, setMessage] = useState('');  
       
      useEffect(() => {  
      fetch('http://localhost:5000')  
      .then(response => response.json())  
      .then(data => setMessage(data.message))  
      .catch(error => console.error('Error:', error));  
      }, []);  
       
      return (  
      <div>  
      <h1>Integrasi React dan Node.js</h1>  
      <p>Pesan dari server: {message}</p>  
      </div>  
      );  
     }  
       
     export default App;
   * Tambahkan proxy di package.json proyek React:
   * "proxy": "http://localhost:5000"
   * Jalankan server Node.js dan aplikasi React secara bersamaan

## TUGAS

1. Buat sebuah aplikasi React sederhana yang menampilkan “Hello, [nama]!” di halaman web.
   * Gunakan Create React App untuk membuat proyek baru
   * Modifikasi komponen App.js untuk menerima input nama dari pengguna
   * Tampilkan pesan selamat datang dengan nama yang diinputkan
2. Buat server Node.js sederhana yang mengembalikan pesan “Hello from Server!” ketika diakses.
   * Gunakan Express.js untuk membuat server
   * Buat satu endpoint GET yang mengembalikan pesan dalam format JSON
3. Push kedua proyek (React dan Node.js) ke GitHub dalam repository terpisah dengan format:
   * React project: PAWPraktikum1\_React\_[NIM]
   * Node.js project: PAWPraktikum1\_Node\_[NIM]
4. Kirim link kedua repository GitHub ke dalam MyKlass praktikum PAW.