

Rancangan Pembelajaran Mentor - Minggu 7: React.js Lanjutan

Berikut rencana pembelajaran untuk Minggu 7 yang berfokus pada fitur React lanjutan dengan intensitas yang lebih menantang, membangun di atas dasar yang telah dipelajari di minggu sebelumnya.

Senin

Materi (2 jam):

- React Router
- Navigasi antar halaman
- Dynamic routing dan parameter URL

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat aplikasi dengan 3 halaman menggunakan React Router:
 - Home (landing page)
 - About (profil dan pengalaman)
 - Project Detail (dengan dynamic route `/:id`)
 - Implementasi navigasi dan parameter URL
- **Proyek Kecil (4 jam):**
 - Tambahkan navigasi multi-halaman ke portofolio
 - Implementasi dynamic routing untuk detail proyek
 - Commit ke branch baru di Git
- **Review (1 jam):**
 - Uji navigasi di browser
 - Periksa dynamic route
 - Validasi log commit

Tugas Harian:

- Submit kode React Router
- Portofolio dengan navigasi
- Log branch Git

Selasa

Materi (2 jam):

- State management dengan Context API
- Global state dan reducer
- Best practices penggunaan context

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat Context untuk manajemen state global:
 - Theme Context (light/dark mode)
 - Project Context (daftar dan detail proyek)
 - Implementasi reducer untuk state kompleks
- **Proyek Kecil (4 jam):**
 - Terapkan Context API ke portofolio
 - Implementasi fitur tema light/dark
 - Buat flowchart alur state
- **Review (1 jam):**
 - Uji perubahan state
 - Validasi flowchart
 - Commit ke Git

Tugas Harian:

- Submit kode Context API
- Flowchart alur state
- Portofolio dengan fitur tema

Rabu

Materi (2 jam):

- Custom hooks
- Reusable logic
- Best practices pembuatan hooks

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat 2 custom hooks:
 - useFetch (untuk mengambil data API)
 - useLocalStorage (untuk state persistence)
 - Implementasi di komponen berbeda
- **Proyek Kecil (4 jam):**
 - Integrasikan custom hooks ke portofolio
 - Implementasi fetching data proyek
 - Commit ke branch baru
- **Review (1 jam):**
 - Uji custom hooks
 - Periksa kode
 - Validasi log commit

Tugas Harian:

- Submit kode custom hooks
- Portofolio dengan fitur API

- Log branch Git

Kamis

Materi (2 jam):

- Flowchart aplikasi React
- Optimasi performa dengan memo
- useCallback dan useMemo

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat flowchart aplikasi React:
 - Alur navigasi
 - State management
 - Data flow
 - Implementasi memo dan useCallback
- **Proyek Kecil (4 jam):**
 - Optimalkan portofolio dengan memo/useCallback
 - Implementasi lazy loading
 - Push ke GitHub dengan pull request
- **Review (1 jam):**
 - Uji performa aplikasi
 - Validasi flowchart
 - Periksa pull request

Tugas Harian:

- Submit flowchart
- Kode optimasi
- Link pull request

Jumat (Ujian):

- **Ujian Praktik (5 jam):**
 - Buat aplikasi manajemen proyek dengan:
 - React Router untuk navigasi
 - Context API untuk state global
 - Custom hooks untuk data fetching
 - Optimasi performa dengan memo/useCallback
 - Kelola kode dengan Git:
 - Buat branch
 - 5 commit terstruktur
 - Push ke GitHub
 - Buat pull request
 - Sertakan flowchart alur aplikasi
- **Persiapan Presentasi (3 jam):**

- Validasi kode
- Validasi flowchart
- Validasi repositori
- Siapkan slide untuk demo dan penjelasan
- Sabtu: Presentasi hasil ujian

Tugas Mingguan:

- Kembangkan portofolio dengan fitur React lanjutan:
 - Navigasi multi-halaman
 - State global dengan Context API
 - Custom hooks untuk data fetching
 - Optimasi performa
- Kelola dengan Git:
 - Branch
 - Merge
 - Pull request
- Unggah ke GitHub dengan README yang menjelaskan fitur React

Catatan:

- **Koneksi dengan Minggu Sebelumnya:** Portofolio diperbarui dengan fitur React lanjutan (navigasi, state global, custom hooks), menggantikan logika sederhana dari minggu sebelumnya, dikelola dengan Git.
- **Intensitas Lebih Berat:** Latihan mencakup skenario kompleks (dynamic routing, custom hooks, optimasi), proyek kecil mengintegrasikan fitur ke portofolio, dan pull request untuk kolaborasi.
- **Flowchart:** Digunakan untuk memvisualisasikan alur aplikasi React, memperkuat desain aplikasi.
- **Git:** Menggunakan branch, commit terstruktur, dan pull request untuk praktik profesional.