



Hari 5 (Jum'at) - Tugas & Presentasi



Tujuan Pembelajaran (2 Jam)

Setelah menyelesaikan sesi ini, santri diharapkan mampu:

1. Menyelesaikan tugas mingguan dengan mengaplikasikan konsep React yang telah dipelajari (Komponen, Props, State, React Router, Styling dengan Tailwind CSS).
2. Mempresentasikan hasil kerja secara efektif, menjelaskan arsitektur, implementasi, dan tantangan yang dihadapi.
3. Memberikan dan menerima feedback konstruktif dari rekan dan mentor.
4. Mendokumentasikan kode dan proses pengembangan dengan baik.



Tugas Mingguan (Waktu Pengerjaan: Sisa Minggu + Akhir Pekan)

Tema Tugas: Membuat Aplikasi Single Page Application (SPA) sederhana menggunakan React.js dengan fitur navigasi dan styling.

Persyaratan Fungsional:

1. **Aplikasi SPA:** Proyek harus berupa aplikasi SPA yang dimuat hanya sekali dan navigasi antar halaman terjadi tanpa refresh penuh.
2. **Minimal 3 Halaman:** Aplikasi harus memiliki minimal 3 halaman berbeda yang dapat diakses melalui navigasi. Contoh: Halaman Beranda (Home), Halaman Tentang (About), Halaman Kontak (Contact), atau halaman lain yang relevan dengan tema pilihan Anda.
3. **Implementasi Navigasi:** Gunakan React Router (atau library routing serupa) untuk mengelola navigasi antar halaman.
 - Harus ada elemen navigasi (misalnya, navbar atau menu) yang memungkinkan pengguna berpindah antar ketiga halaman tersebut.
 - Pastikan URL di browser berubah sesuai dengan halaman yang sedang ditampilkan.
4. **Penggunaan Komponen:** Pecah UI aplikasi menjadi komponen-komponen React yang reusable dan modular.
 - Identifikasi bagian-bagian UI yang bisa dijadikan komponen terpisah (misalnya: Header, Footer, Navbar, Card, Form, dll.).
 - Gunakan Props untuk melewatkan data antar komponen.
 - Gunakan State (dengan `useState` hook) untuk mengelola data yang berubah di dalam komponen (jika ada fitur interaktif sederhana).
5. **Styling dengan Tailwind CSS:** Terapkan styling pada seluruh aplikasi menggunakan Tailwind CSS.
 - Pastikan tampilan aplikasi rapi dan konsisten.
 - Terapkan styling responsif agar aplikasi terlihat baik di berbagai ukuran layar (desktop, tablet, mobile) menggunakan breakpoint Tailwind CSS.
6. **Struktur Proyek Rapi:** Atur file dan folder dalam proyek secara logis dan modular.
 - Pisahkan komponen ke dalam folder `components`.
 - Pisahkan halaman ke dalam folder `pages` atau `views`.
 - Atur file CSS dan aset lainnya dengan rapi.
7. **Dokumentasi:** Buat dokumentasi sederhana untuk proyek Anda.

- Jelaskan cara menjalankan proyek (instalasi dependensi, menjalankan development server).
- Jelaskan struktur folder utama proyek Anda.
- Jelaskan komponen-komponen utama yang Anda buat dan fungsinya.
- Jelaskan bagaimana navigasi diimplementasikan.
- Jelaskan pendekatan styling yang Anda gunakan dengan Tailwind CSS.
- Dokumentasi bisa berupa file **README.md** di root proyek.

Fitur Opsional (Nilai Tambah):

- Menggunakan parameter URL untuk menampilkan detail item (misalnya **/products/:id**).
- Implementasi nested routes.
- Menggunakan form sederhana dengan state management.
- Menambahkan animasi atau transisi antar halaman.
- Menggunakan Context API atau state management library sederhana (misalnya Zustand) untuk data global (jika sudah siap).

Output Tugas:

- Kode sumber proyek yang di-push ke repositori GitHub pribadi Anda.
- File **README.md** yang berisi dokumentasi proyek.

Sesi Presentasi (0.5 Jam per Kelompok/Individu, disesuaikan jumlah santri)

Format Presentasi:

- Setiap santri atau kelompok (jika tugas dikerjakan berkelompok) akan mempresentasikan hasil kerjanya.
- Durasi presentasi disesuaikan dengan jumlah peserta, misalnya 10-15 menit per presentasi.

Materi Presentasi:

1. **Demo Aplikasi:** Tunjukkan aplikasi SPA yang telah Anda buat secara langsung. Navigasikan antar halaman, tunjukkan fitur-fitur yang diimplementasikan, dan demonstrasikan responsivitasnya.
2. **Penjelasan Arsitektur Proyek:**
 - Jelaskan struktur folder proyek Anda dan mengapa Anda memilih struktur tersebut.
 - Jelaskan bagaimana Anda memecah UI menjadi komponen-komponen.
 - Jelaskan bagaimana data mengalir antar komponen (Props, State).
 - Jelaskan bagaimana routing diimplementasikan menggunakan React Router.
 - Jelaskan bagaimana styling diatur dengan Tailwind CSS.
3. **Tantangan dan Solusi:** Ceritakan tantangan teknis yang Anda hadapi selama pengerjaan tugas dan bagaimana Anda menyelesaikannya.
 - Contoh: Kesulitan dalam state management, isu routing, penyesuaian styling responsif, dll.
 - Ini menunjukkan proses berpikir dan kemampuan problem-solving Anda.
4. **Pembelajaran:** Apa saja pembelajaran utama yang Anda dapatkan dari pengerjaan tugas ini?

Sesi Review dan Feedback:

- Setelah setiap presentasi, akan ada sesi tanya jawab singkat.

- Santri lain dan mentor diharapkan memberikan feedback konstruktif (apa yang sudah baik, area yang bisa ditingkatkan, saran teknis).
- Fokus pada pembelajaran bersama dan berbagi pengetahuan.



Tips untuk Tugas & Presentasi

- **Mulai Lebih Awal:** Jangan tunda pengerjaan tugas sampai menit terakhir. Pecah tugas menjadi bagian-bagian kecil dan kerjakan secara bertahap.
- **Gunakan Git Secara Efektif:** Lakukan commit secara rutin dengan pesan yang jelas. Ini membantu melacak perubahan dan memudahkan debugging jika terjadi masalah.
- **Jangan Takut Bertanya:** Jika Anda menemui kesulitan, jangan ragu bertanya kepada mentor atau rekan santri lainnya.
- **Latihan Presentasi:** Latih presentasi Anda sebelum hari H. Pastikan Anda nyaman dengan alur demo dan penjelasan teknisnya.
- **Fokus pada Konsep Dasar:** Pastikan konsep dasar (Komponen, Props, State, Routing, Styling) sudah terimplementasi dengan baik sebelum mengejar fitur opsional.
- **Siapkan Pertanyaan:** Siapkan pertanyaan untuk presentasi rekan Anda. Berpartisipasi aktif dalam sesi feedback akan memperkaya pembelajaran Anda.

Hari terakhir minggu ini didedikasikan untuk mengkonsolidasikan semua pengetahuan yang telah diperoleh melalui pengerjaan tugas praktis dan berbagi pembelajaran melalui presentasi. Ini adalah kesempatan bagus untuk melihat bagaimana konsep-konsep React diterapkan dalam skenario nyata dan belajar dari pendekatan yang berbeda.