|  |  |
| --- | --- |
| Materi | Nilai |
| React-js 21-28 #9 | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Membuat fille file di dalam folder front yang isinya yaitu Nav,Front,Side,Main,dan Footer.
2. Lalu memuat file back.js di dalam folder back sebagai backend.
3. Lalu membuat route yang dipasang di app.js menggynakan codingan import { BrowserRouter as Router, Route } from "react-router-dom";. Digunakan untuk menggunakan fungsi router pada file app.js.
4. Lalu pada app.js kita mngawali dengan <Router> lalu pada masing masing component ditambahkan route lalu path nya seperti berikut.

<Route path="/" component={Front} exact/>

<Route path="/home" component={Front} />

<Route path="/admin" component={Back} />

Jadi artinya yaitu jika di dalam url user tidak menambahkan apa pun setelah localhost maka akan keluar fornt. Jika anda menambahkan /home atau /admin pada url. maka aakan mempilkan back atau front dengan url /home sesuai pada komponen yang di deklarasikan.

1. Exact digunakan saat / pada url agar tidak selalu terpanggil. Jika / tanpa diberi exact maka /admin harusnya memanggil /admin file back saja tetapi akan memanggil front juga, karena / juga dipanggil. Maka agar hal itu tidak terjadi, pada route path / diberi exact setelah component agar / terpanggil jika hanya / saja yang dimasukan.
2. Tetapi jangan lupa untuk menambahkan import dari front dan back.
3. Lalu untuk tampilan front itu mengambil dari file file Nav ,Side,Main,dan Footer.
4. Membuat fungsi link dan routefr pada file side import { Link, useRouteMatch } from "react-router-dom"; lalu memberi fungsi link pada masing masing halaman di file side yang ada pada folder back.
5. Lalu menambahkan perintah const { url } = useRouteMatch(); di dalam const Side di luar dari return untuk menambahkan fungsi route agar jika ingin membuka to={`${url}/kategori`} otomatis sebelumnya akan ada /admin dari path awalnya.
6. Lalu menambahkan const { path } = useRouteMatch(); pada Main.js agar data dari url tadi dapat dibaca jika mengklik link yang menampilkan main sesuai data urlnya.
7. Agar dapat dibaca oleh main maka harus menambahkan path={`${path}/:isi`} jadi isi digunakan untuk variabel dari data url tersebut lalu memanggil content dalam main.js yang isinya memanggil variabel isi.
8. Memahammi bagaimana cara menampilkan isi dari database tabel kategori dari project api lumen mengunakan axios dan ditampilkan ke project react resto pada admin/kategori.
9. Menggunakan use state untuk pengeset an variabel dan pemanggilan variabel fungsinya agar jika halaman itu berubah maka akan otomatis ter update tanpa refresh. FetchData digunakan untuk pengambilan data dari request yang muncul.
10. Lelu membuat isi dari tabel kategori dan keterangan caranya yaitu memanggil request data dari isi, lalu di map dan ada index yg sebagai kunci dan val untuk hasil data nya.
11. Memahami bagaimana membuat fungsi react hook form yaitu me register dulu form yang kita perlukan menggunakan ref={register} dan required nya true.
12. Lelu untuk menghandle error nya kita menggunakan function error nya yaitu mendeklarasikan error nya kolom apa dan pesannya apa.
13. Lelu untuk mengambil data itu menggunakan di dalam form itu terdapat onSubmit={handleSubmit} yang dibawa oleh useForm. Kemudian dia akan menjalankan function simpan yang isinya data dari formnya.
14. Guna function reset() yaitu agar apa yang diketik dan setelah di submit maka akan hilang ap yang di ketikkan tadi.
15. Memahampimi bagaimana mengcreate data menggunakan post. Yaitu pada saat memasukan kategori dan keterangan baru pada form, lalu jika sudah di simpan maka akan menjalankan link.post("/kategori", data).then((res) => setPesan(res.data.pesan));
16. Artinya yaitu apa yang diketikan di kategori dan keterangan tadi akan masuk dan menjalankan funsi post ke halaman /kategori. Lalu jika berhasil maka akan menampilkan setpesan dari data.pesan dari controllr kategori.
17. Use state digunakan larena kita tidak bisa mengirimkan variabel di dalam return. Maka kita menggunakan useState.
18. Mengerti bagaimana pembuatan delete dan fungsinya pembuatannya hampir sama seperti create tetapi itu mencari kategori yang sudah memiliki id yang dapat di hapus, lalu menambahkan fungsi if confirm yaitu jika klik ok maka akan menghapus jika cancel ya tidak menghapus.
19. Membuat fungsi show data agar jika di klik ubah maka data dengan kategori tersebut akan ditampilkan ke form untuk melakukan update data menggunakan use state setpilihan dan pilihan
20. Membuat update data yang digunakan untuk mengubah isi dari kategori dan keterangan

**Saya Belum Mengerti**