|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| React JS Project Resto Bag.3 (Vid 39-46) | 90 |

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

1. Dapat membuat migration dan controller untuk tabel order menggunakan php artisan pada api-lumen

2. Dapat menyiapkan kolom2 untuk database order menggunakan php artisan pada api-lumen

3. Memahami & dapat merger/join tabel pelanggans dan idpelanggan untuk data order

4. Pada controller order dapat menambahkan opsi untuk update bayar kembali dan status pembayaran di function update

5. Dapat menampilkan data order pada tabel order yaitu

* Menambahkan const array [isi] untuk useGet(‘/order’) untuk dapat memanggil data dari database order
* Mmebuat row untuk judul dan tabel
* Buat tabel lalu isi kolom thead dengan no,pelanggan,tglorder,total,bayar,kembali dan status. Untuk no buat variable terlebih dahulu isi dengan no
* Untuk kolom tbody isi dengan {isi.map((val, index) => (<tr key={index}> untuk menampilkan data menggunakan key dan index misal : <td>{val.pelanggan}</td> lalu untuk nomer tinggal menggunakan {no++} jadi di tiap data akan ter looping nomor yang urut
* Pada kolom status tambahkan button dengan pengujian agar kolom akan menampilkan button belum bayar apabila pesanan bernilai 0/tidak lunas dan akan menampilkan tulisan lunas apabila bernilai 1/lunas : <td>{val.status === 0 ? <button className="btn btn-success">Belum Bayar</button> : <p>Lunas</p>}</td>

6. Memahami cara mengambil data tanggal order untuk di tampilkan menggunakan search bar :

* Pada function show di controller order isi params dengan var a dan b untuk menandai start dan finish tanggal untuk search order
* Panggil model order untuk mengambi data dari kolom tglorder lalu tanda (mulai)>= untuk varA dan (sampai)<= untuk varB lalu menggunakan function get() dan return response agar data bisa di tampilkan di page

7. Memahami cara menampilkan searc bar untuk sortir tanggal pencarian order :

* Pada file order.js di folder back impor useState dan useForm seperti pada kategori
* Buat const useState untuk awal,seAwal dan akhir,setAkhir dengan isi tanggal dimuali pada awal dan var today untuk akhir
* Membuat variable today yang berisi new Date().toISOString().slice(0,10) makshudnya adalah menggunakan function Date untuk menetapkan tanggal yang dibagi 10 bagian
* Membuat const useForm untuk menggunakan function register & handleSubmit
* Membuat form untuk set tanggal awal (label htmlfor tawal dan typeform date nama tawal) begitu juga dengan form untuk set tanggal akhir tinggal di sesuaikan
* Buat button submit
* Pada tag form tambahkan onSubmit yang akan mengarahkan ke function cari untuk menjalankan pencarian data
* Memebuat function cari dengan params (data) lalu isi dengan setAwal(tawal) dan setAkhir(takhir);
* Pada const [isi] useGet arahkan ke `/Order/${awal}/${akhir} seperti yang sudah di set pada const useState agar proses sortir terlaksana sesuai dengan tanggal yang di isikan pada form

8. Memahami penggunaan modal yaitu untuk menampilkan sesuatu dalam mode pop-up sehingga tidak perlu ber alih halaman web untuk menampilkan sebuah opsi

9. Dapat menambahkan modal untuk menampilkan opsi pembayaran order :

* Meng-install modal menggunakan yarn
* Import modal pada file
* Set root modal seperti pada indexs.js Modal.setAppElement('#root');
* Buat modal lalu isi
* Buat const untuk menjalankan modal dengan useState [mopen, setMopen] = useState(false) agar menutup modalnya
* Pada tag untuk menjalankan modal gunakan isOpen={mopen} untuk membuka modal karena mopen akan bernilai true
* Kemudian tambahkan button untuk close dengan onClick mengarahkan untuk menggunakan setMopen karena akan bernilai false yang artinya modal akan menutup

10. Dapat menambahkan form di dalam modal untuk proses pembayaran order :

* Di dalam modal isi dengan row untuk kerapihan layout lalu isi dengan form dan komponen nya label,input,button dsb.
* Atur css dan bootstrap untuk layout agar lebih rapi dan ter-arah

11. Dapat menambahkan data order ke dalam form pembayaran :

* Buat function filterData dengan params id untuk memfilter data mana saja yang akan di tampilkan di dalam form
* Isi function filterData dengan membuat const lalu isi dengan funnction filter menggunakan useGet is untuk mengambil datanya lalu di uji apakah idorder dari database sama dengan id dari data order yang di ambil & akan di tampilkan pada form const data = isi.filter((val) => (val.idorder === id))
* Buat const useState di luar function untuk total dan setTotal bernilai 0 karna jumlah awal lalu const untuk pelanggan & setPelanggan bernilai kosong ‘’
* Di dalam function filterData tambahkan setPelanggan untuk memanggil data (data[0].pelanggan) juga pada total lalu tambahkan setMopen(true) agar modal terbuka
* Pada judul form order tambahkan {pelanggan} untuk menampilkan nama pelanggan yang akan membayar order
* Tambahkan setValue pada const untuk useForm
* Buat function isiForm untuk menampilkan total order dengan menggunakan setValue(‘total’,total);
* Pada awal modal tambahkan onAfterOpen={isiForm} untuk menampilkan total dari order
* Pasang function filterData pada button status untuk mengarahkan ke modal form pembayaran dan mengambil data order yg belum di bayar apabila order belum di bayar onClick={() =>filterData(val.idorder)}>Belum Bayar</button>

12. Dapat meng-update data pembayaran order :

* Menamabhkan onSubmit{handledubmit()} pada form untuk memanggil function simpan untuk melakukan update data
* Membuat function simpan dengan params (data) lalu buat var let hasil untuk isi dari kolom dari order yaitu hasil:data.bayar, kembali:data.bayar – data.total, lalu status nya 1
* Buat const res = await link.put("/Order/" + idorder, hasil); untuk mengirim perubahan/update data setelah di bayar ke database menggunakan api axios (sebelumnya impor axios terlebih dahulu)
* Beri setOpen(false) agar modal pembayran tertutup otomatis setelah transaksi pembayaran selesai di lakukan
* Beri required:true untuk form total register dan required:true ,min:total pada form register bayar karena selain untuk wajib diisi juga agar nilai pembayaran tidak sampai dibawah jumlah yg dibayar yaitu total. Apabila kurang maka akan menheluarkan warning error
* Tambahkan errors pada useForm lalu buat pesan error nya {errors.bayar && "Pembayaran Kurang!"}
* Tambahkan useState untuk idorder & setIdorder karena kedua komponen itu dibutuhkan untuk proses update data
* Pafa function filterData tambahlan setIdorder(data[0].idorder); untuk filter agar idorder dapat ter select