|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Pembuatan Front end web restoran ci 4 | 90 |

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

1.Mengatur routes untuk default page front yaitu pada homepage/index

2.Membuat view home front utama di dalam template,berisi navbar history,logout, dan indikator pelanggan. kategori dan semua menu pada controller hompegae akan otomatis tampil dari function index karena sudah di atur di routes.

3.Dalam folder front,controller homepage membuat function index yg akan menjadi default homepage front. Yaitu menampilkan kategori menu dengan model kategori dan menu dengan model menu yg dikirim dengan array dan return view juga menggunakan pager,membuat view untuk default tampilan kategori & menu (semua data).

4. View untuk menampilkan data default (semua data menu),dapat menggunakan card dari bootstrap.

5.Untuk menampilkan menu per kategori,maka menambahkan if (!iempty($kategori) foreach ($kategori as $r) untuk looping data yang akan di tampilkan,lalu memanggil function read,untuk menampilkan menu per kategori, Juga dengan idkategori nya sebagai indikator menu dari kategori yang akan di tampilkan.

6. Dalam controller homepage membuat function read dgn parameter $id = null,menggunakan pager lalu if (isset($id)) membungkus model kategori dan menu dengan objek agar dapat digunakan,lalu memanggil data menu dengan $model\_menu->where(‘idkategori’,$id)->findAll(); menggunakna count untuk paging halaman dan if else & $\_GET[‘page’] untuk mengatur pembagian halaman.

7. Memanggil lagi data menu dengan paging berada dalam find all

8. Mengirim data yang akan di tampilkan ke dalam bentuk array yg nantinya akan di tampilkan dlama view menu per kategori.

9. Return view,lalu membuat view untuk menu per kategori dengan card bootstrap.

10. Dalam setiap view menu di card terdapat button beli,yang diarahkan dengan base url ke function index pada folder front controller keranjang belanja dan function index

11. Membuat controller keranjangbelanja pada folder front,lalu membuat default beli yaitu function index parameter nya ($id=null) lalu menggunakan model menu dan kategori yg dibungkus objek lalu $total = 0.lalu if isset $id $this->function isi() dan return redirect front/keranjang belanja.

12. Lalu pada keadaan else,maka menggunakan foreach session get $key => $value untuk mendapatkan email,pelanggan,dan idpelanggan dengan session dari code igniter \_\_ci\_last\_regenerate dan \_\_ci\_previous\_url. Isi foreach nya yaitu untuk mengambil $id = substr($key,1) dan menggunakan model menu dalam bentuk array untuk menemukan id menu $menu[] = $model\_menu->where('idmenu', $id)->findAll(),lalu mengguanakan jumlah untuk mengambil value nya $jumlah[] = $value;

13. Untuk mengirim data array maka,menambahkan if(isset($menu)) = { $menu = []; & $jumlah = []; }

14. Mengirim data dengan array ke view beli,yaitu judul,kategori,menu,jumlah,dan total

15. Membuat view untuk beli yang berisi tabel,menu jumlah menu yg dipesa,total,hapus, + dan – untuk kuantitas,dan total akhir.

16. membuat double foreach untuk mengambil nilai dari array yg dikirim oleh function index untuk beli tadi, foreach pertama untuk mengambil nilai missal,$menu as $key => $r lalu di dalam foreach ditambahkan foreach untuk mengambil value dari menu, $value as $r => $p

17. Lalu untuk menampilkan menu pada kolom tinggal memanggil data menu $p[‘menu’]

18. Untuk kolom harga tinggal $p[‘harga’]

19. Untuk kolom tambah kurang kuantitas,maka menggunakan function tambah dan kurang dari folder controller keranjang belanja.

20. Dalam controller keranjangbelanja membuat function isi() untuk menjalankan proses beli barang yang berisi if session has $id maka akan set session get $id + 1, dan else nya akan session get $id ,1

21. Membuat function tambah untuk kuantitas pada view beli,yaitu menggunakan session set $id lalu session get $id + 1 dan return redirect ke keranjangbelanja

22. Untuk pengurangan,$id di simpan dalam variable terlebih dahulu karena akan ada pengujian,jadi session set $variable lalu session get $variable-1 selanjutnya if session get-> $variable == 0 maka session-remove($variable).Sehingga datanya akan terhapus jadi tidak akan jadi minus -.Llu redirect ke keranjang

23. Untuk delete pesanan,maka pada doreach kolom delete diarahkan ke function delete pada controller keranjangbelanja,yang berisi session remove lalu redirect

24. Untuk harga produk,maka menggunakan $jumlah[$key] untuk mengambil data harga berdasarkan id menu

25. untuk total harga maka menggunakan $total = ($jumlah[$key] + $p[‘harga’])

26. Untuk grand total tinggal menggunakan $total saja

27. Lalu menambahkan button untuk checkout yang akan muncul apabila if ($total > 0)

28. Button checkout akan mengarahkan ke function checkout pada controller keranjang belanja

29. Function checkout pada keranjang belanja (parameter nya mengambil dari $total = null),lalu pada checkout dikoneksikan dengan database menggunakan db config karena akan menggunaakn sql nantinya

30. Lalu if isset($total) yang berisi pengambilan id order dengan menggunakan function idOrder, ($idorder = $this->idOrder())

31. Membuat function idOrder setelah function checkout yang akan digunakan untuk mengambil idorder dari database. Dalam function berisi dbconfig untuk koneksi ke database,lalu sql SELECT idorder FROM tblorder ORDER BY idorder.Kemudian men-select nilai dari sql di letakkan dalam variable dengan menggunakan query $result = $db->query($sql); kemudian mengambil nilainnya dengan disimpan pada variable $detail = $result->getResult(‘array’) // nilainya di simpan dalam bentuk array. Kemudian menghitung data dengan $count = count($detail) dan menambahkan statement untuk id yaitu if ($count == 0) $id =1 else $id = $count + 1 yg artinya apabila id nya tidak ada akan bernilai 1 apabila ada maka akan ditambahkan 1 nilainnya. Llau return $id

32. Lanjut dalam function checkout,yaitu mengambil id pelanggan dengan session,$idpelanggan = session()->get(‘idpelanggan’);

33. Untuk men-set tanggal pembelian menggunakan $tgl = date(‘Y-m-d’);

34. Untuk mengambil nilai idorder menggunakan sql, $sql = “SELECT \* FROM tblorder WHERE idorder = $idorder “ lalu menggunakan query $result = $db->query($sql); kemudian diambil nilainya dengan getResult yang berupa array dan dihitung menggunakan count.

35. menjalankan statement untuk insert order ke dalam database dengan menggunanakan if ($count == 0) maka akan menjalankan function $this->insertOrder ($idorder,$idpelanggan,$tgl,$total); dan function $this ->insertOrderDetail($idorder); lalu else nya hanya insertOrderDetail saja.

36. Setelah proses if sudah maka akan menjalankan function $this->kosongkansession(); untuk mengosongkan keranjang belanja yg telah di checkout dan return redirect ke kerajang belanja

37. Membuat function insert order,parameternya berisikan $idorder,$idpelanggan,$tgl,$total karena akan difokuskan ke proses insert data dalam database.Lalu isi dengan dbconfig dan sql untuk insert data $sql = “INSERT INTO tblorder VALUES ( $idorder,$idpelanggan,$tgl,$total,0,0,0)” sesuai pada tabel dan menggunakan query untuk select data nya $db->query($sql)

38. Membuat function insertOrderDetail (parameter $idorder = 1),function ini untuk meng-insertkan menu ke dalam tblorder yg di panggil dalam model orderdetail2.Jadi bungkus model menu dan model orderdetail2 lalu buat foreach session()->get() as $key => $value

39. If dalam foreach Untuk mendapatkan data pelanggan email dan idpelanggan,juga gunakan session dari ci last generate dan ci previous url, isi dari if adalah untuk substr $key yg diletakkan dalam variable $id.

40. Lalu memanggil objek model menu $variable = $model\_menu->where('idmenu', $id)->findAll(); untuk memanggil data menu.

41. menambahkan foreach ($variable as $key) { panggil $idmenu = $key[‘idmenu’]; dan harga menu $key[‘harga’]; lalu simpan dalam array untuk dikirimkan datanya, yaitu idorder,idmenu,jumlah,hargajual kemudian insertkan dengan $model orderdetail 2 ->insert($data/nama array)

42. membuat function kosongkanSession() untuk mengosongkan keranjang apabila proses checkout telah berhasil,yaitu foreach (session()->get() as $key => $value) if seperti pada insertOrderDetail lalu isi if nya session()->remove($key);

43. Pada function checkout isi else dari if(isset($total)) yaitu $id = $this->idOrder() – 1; untuk memanggil Idorder kemudian memanggil model kategori,lalu mengirim data lewat array yaitu thank you message,kategori dan id kemudian return view untuk menampilkannya.

44. Membuat view untuk message bahwa proses checkout telah berhasil,datanya berasal dari else function checkout.

45. Mengatur navbar agar menampilkan button login & juga daftar apabila pelanggan belum login,yaitu pada view front home pada navbar menambahkan if (!empty(session()->get(‘email’) && !empty(session()->get(‘pelanggan’) maka akan menampilkan pelanggan yg sedang login,button history dan button login

46. Lalu else nya berisi button login dan daftar,yang apabila belum login bisa mendaftar pelanggan baru lalu login agar bisa meng-akses full web nya

47. Pada button login diarahkan ke controller Login,di dalam controller tambahkan untuk menampilkan kategori dengan model kategori yg dikirim lewat array.Lalu membuat if ($this->request->getMethod() == ‘post’) untuk mengambil nilai inputan login dari form yaitu email dan password dgn menggunakan request getPost ‘email’ & ‘password’ lalu memanggil model pelangan untuk mengambil data email dan aktif $pelanggan = $model->where(['email' => $email, 'aktif' => 1])->first();

48. menambahkan pengujian apakah data yg di inputkan untuk login sudah tepat,dengan menggunakan validation message sebagai notice nya yg telah di simpan pada info,Jadi if empty($pelanggan) $data[‘info’] = “Email Salah” kemudian pada else isi dengan if (password\_verify($password dan pelanggan[‘password’])) maka akan menjalankan function set session dan redirect ke home front utama.

49. Untuk else nya maka akan mengeluarkan warning message “PASSWORD SALAH” karena ini berada dalam pengujian password kemudian return view untuk form login

50. Masih dalam controller login,membuat function setSession($pelanggan) untuk menge-set pelanggan yang login,dikirim dengan menggunakan array yaitu pelanggan => $pelanggan[‘pelanggan’],email => ..,idpelanggan => ..dan login => true

51. membuat function untuk logout pelanggan yaitu dengan menggunakan session->detroy,jadi button logout diarahkan ke function ini dan controller login

52. Untuk proses daftar pelanggan baru,button daftar diarahkan ke fuction create karena untuk menampilkan kategori,kemudian return view daftar

53. Pada view untuk daftar form action mengarahkan ke function daftar pada homepage karena akan digunakan untuk proses daftar pelanggan baru

54.Pada controller homepage,membuat function daftar,diisi dengan request getPost karena untuk insert data baru dari form.data nya di kirim dalam bentuk array yaitu pelanggan,alamat,telp,email,password,dan aktif = 1 (cth : 'pelanggan' => $request->getPost('pelanggan') )

55. Untuk enkripsi password menggunakan password hash($request->getPost(‘password’), PASSWORD DEFAULT)

56. Memanggil model pelanggan karena data akan dimasukkan ke tabel pelanggan

57. menambahkan pengujian untuk data yang akan di insert kan ke dalam database yaitu dengan if ($request->getPost(‘password’)) di dalamnya if($model pelanggan->insert($array) === false) maka akan mengeluarkan validation message dari model pelanggan menggunakan session setFlashdata ,untuk else nya akan diarahakan ke page login apabila berhasil daftar

58. Pada model pelanggan harus menambahkan validation rules dan message nya terlebih dahulu

59. Mengapa selalu ditampilkan kategori,karena itu sebagai menu yg tampil tiap saat pada tiap viw karena ini aplikasi restoran

60. Membuat controller untuk menampilkan history pembelian,pada controller homepage menggunakn dbconfig karena akan mengambil data dari database,mengambil data dari view vorder yaitu email nya dengan session get dan getWhere

61.Menambahkan pager dengan mengambil nilai dari data yang sudah di ambil dari vorder tadi,menggunakan getResult(‘array’) lalu menggunakan count untuk menghitung halaman dan if isset $\_GET[‘page’]

62. kemudian menggunakan getWhere lagi untuk menyatukan dengan pager lalu di ambil lagi datanya dengan getResult

63.Mengirim data pada view history menggunakan array,yaitu kategori (harus selalu ditambahkan),data vorder,pager,mulai halaman,banyak halaman,dan total lalu return view history

64.Pada view history di bentuk tabel untuk menampilkan tanggal order dan total belanjaan dan button detail untuk mengarahkan ke controller function detail

65. Membuat function detail pada controller homepage dengan menggunakan model orderdetail2 unutk mengambil data dari tblorderdetail $jumlah = $model\_orderdetail2->where('idorder', $id)->findAll();

66. Menggunakan count untuk menghitung jumlah data kemudian melakukan penerapan untuk paging halaman,tambahkan config pager.

67.Kemudian gabungkan data orderdetail2 dan paging halaman $detail = $model\_orderdetail->where('idorder', $id)->findAll($banyak, $mulai); untuk di tampilkan lewat array,yaitu detail,pager,banyak,count,mulai dan (kategori as always) llau return view detail yang berupa tabel menampilkan nama menu,banyak menu,dan harga nya

68. Untuk mengatur hak akses apabila bukan seorang pelanggan/belum terdaftar & login maka buat file untuk filter front pada folder front kemudian isi dengan if (!session get ‘login’->yg terdapat pada function setSession pada controller login dan juga di tambahkan pada foreach function insertOrder() dan insertOrderDetail() maka result nya adalah redirect ke laman login,jadi artinya apabila pelanggan belum login dan akan membeli sesuatu maka Ketika button beli di klik maka akan mengarah ke page login

69. Pada filters di config menambahkan filetr front setelah filter untuk backend

70.Buat routes group baru untuk add pada filter front yaitu keranjangbelanja index,delete,tambah,kurang,checkout (:any) dan $1 llau untuk checkout lagi tanpa (:any) dan $1.Kemudian untuk homepage history dan detail,untuk detail pakai (:any) dan $1,history tidak.Mengikuti routes sebelumnya