|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Project React Resto | 90 |

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

1. Bisa mereuse komponen (memakai komponen ulang dari komponen yang telah dibuat sebelumnya (komponen yang sama)). Namun bisa diubah setelahnya sesuai dengan kegunaan (namun tidak membuat dari awal, hanya menambahkan sesuai keperluan, seperti MENU BUAH pada video).

2. Filter, digunakan untuk memilih mana data yang akan digunakan (tidak semuanya bisa tampil nantinya, yang dipilih saja nanti yang akan ditampilkan).

3. Membuat text document pada folder project react resto untuk menuliskan hal-hal yang diperlukan nantinya (seperti penginstallan).

4. Axios, untuk pembuatan http request yang bisa memunculkan code (nomer seperti 400 dan lain-lainnya) bila ada suatu errornya.

5. Bisa menginstall dan menggunakan axios sesuai dengan yang ada di video.

6. Menggunakan link cdn untuk menggunakan axios nya (online, harus dengan internet bila ingin menampilkannya).

7. Membuat codingnya harus di dalam <script><script/>, karena axios termasuk ke dalam javascript.

8. Pada bagian get index (kategoricontroller) harus diletakkan diatas middleware (agar tidak terkena filter dari middlewarenya).

9. Console.log, untuk memunculkan hasil yang dibuatnya bisa tampil di bagian console pada inspect.

10. Get, untuk mendapatkan hasil dari data yang sudah dipilih sebelumnya.

11. Pada bagian Acces-Control-Allow-Headers di CorsMiddleware.php di depan coding yang sudah ada sebelumnya (di dalam petik atas (‘ ’)) perlu ditambahkan api\_token karena di dalam databasenya sudah tercantum token, maka bila tidak diberi token nanti hasilnya akan Unauthorized.

12. Pada app.php dibuka command untuk $app->middleware nya.

13. Menggunakan bootstrap online (dengan link cdn, harus tersambung dengan internet pada saat memakainya).

14. Display-4, untuk membesarkan dari ukuran tulisan yang akan ditampilkan.

15. Margin top (mt), digunakan untuk memberikan jarak pada bagian atas.

16. Membuat button dengan bootstrap (tombol yang bila ditekan setelah diatur nantinya akan mengeluarkan sesuatu dari yang sudah diatur sebelumnya) untuk get, show, post, delete, update.

17. Bisa menampilkan hasil button get pada tampilan luarnya (bukan di dalam console bagian inspect).

18. Bisa membuat get dengan axios (sesuai dengan yang ada di video).

19. Show, untuk menampilkan data yang sudah dipilih sebelumnya (seperti filter, yang nantinya tidak semua data akan keluar).

20. Bisa membuat show dengan axios (sesuai dengan yang ada di video).

21. Bisa menampilkan hasil button show pada tampilan luarnya (bukan di dalam console bagian inspect).

22. Post, untuk mengirim data ke database (apirestoran) yang sudah dibuat sebelumnya di dalam coding.

23. Bisa membuat post dengan axios (sesuai dengan yang ada di video). Data yang dibuat berbentuk object.

24. Bisa menampilkan hasil button post pada tampilan luarnya (bukan di dalam console bagian inspect). Yang nantinya akan memunculkan hasil tulisan ‘Data sudah dimasukkan’ bila data yang dibuat sebelumnya sudah terbuat (muncul) di dalam databasenya (apirestoran).

25. Menambahkan beberapa coding pada api-lumen yang telah dibuat sebelumnya (pada bagian middleware, routes, bootstrap, dan controller).

26. Document.querySelector, untuk pengambilan dom ((Document Object Model), diperlukan oleh JavaScript yang akan mengubah tampilan sebuah website).

27. Delete, untuk menghapus data pada database (apirestoran) yang sudah dipilih id nya dibagian coding (dipilih id nomer berapa yang nantinya akan dihapus, pada coding dituliskan let id = ….; (titik-titik berisikan nomer dari id yang nantinya akan dihapus)).

28. Bisa membuat delete dengan axios (sesuai dengan yang ada di video).

29. Bisa menampilkan hasil button delete pada tampilan luarnya (bukan di dalam console bagian inspect). Yang nantinya akan memunculkan hasil tulisan ‘Data sudah dihapus’ bila data yang dipilih sudah sukses terhapus pada databasenya (apirestoran).

30. If, digunakan untuk dua kondisi. Bila hasil sudah benar, maka akan menjalankan kondisi yanh pertama, dan bila gagal nantinya akan dimunculkan pada kondisi keduanya (else).

31. Update, untuk mengubah data pada database (apirestoran) yang sebelumnya sudah pernah dibuat (seperti penggantian nama pelanggan, alamat, dan nomer telepon).

32. Php artisan route:list, untuk menampilkan hasil dari route yang pernah dibuat sebelumnya (ditunjukkan bila pada saat proses apa menggunakan apa).

33. Bisa membuat update (put) dengan axios (sesuai dengan yang ada di video).

34. Bisa menampilkan hasil button update pada tampilan luarnya (bukan di dalam console bagian inspect). Yang nantinya akan memunculkan hasil tulisan ‘Data sudah di update’ bila data yang dipilih sudah sukses diubah pada databasenya (apirestoran).

35. Status, untuk mengirimkan data balik bila hal yang dikirim awalnya sudah sukses terjalankan.

36. Bisa membuat import dan export sesuai dengan yang ada di video.

37. Import (function), untuk mengambil data dari function yang sudah diberikan export di depannya (nanti akan dimunculkan pada console bagian inspect).

38. Export (function), untuk membuat tampilan dari data yang nantinya akan ditampilkan lewat import. Di depan functionnya diberikan export agar dapat tampil nantinya.

39. Antara import dan export saling berkaitan satu sama lain. Apa yang ada di dalam export yang nantinya akan ditampilkan oleh import (meskipun kedua hal ini berbeda file, tinggal menuliskan link dari file nya (pada bagian import dituliskan link dari exportnya)). Import bisa memunculkan banyak export di dalam kurung objectnya, seperti {get, show, siswa} pada video. Namun hasilnya tetap masing-masing sesuai dengan exportnya (isi get dan show berbeda, maka tampilannya juga sesuai dan tidak disatukan. Get sendiri, show sendiri (hasil tampilannya)).

40. Bisa memisah-misahkan bagian yang sudah dibuat sebelumnya di dalam index.html dengan membuat file tersendiri pada tiap bagiannya seperti get.js, link.js, dan lain-lainnya (di dalam folder layout pada project-react-resto). Isi di dalam index awalnya dipindah dahulu di dalam file backup.html, yang kemudian baru dipisah-pisahkan sesuai nama filenya.

41. Format document, untuk merapikan spasi di dalam code (langsung rapi tanpa manual). Dengan cara klik kanan, lalu pilih Format Document.

42. Hasil dari dijadikan satu dengan yang dipisah-pisah sesuai tempatnya tetap sama. Hanya saja lebih rapid an mudah mencari error nantinya bila dipisah-pisahkan sesuai kebutuhannya (seperti get disendirikan, show, dan lain-lainnya). Memanggilnya (menyatukannya agar bisa ditampilkan) dengan menggunakan export dan import. Pada index diletakkan bagian importnya (namun disendirikan per bagian, seperti import untuk get sendiri, show sendiri, dan lain-lainnya), sedangkan untuk exportnya diletakkan di depan setiap dari functionnya masing-masing.

43. Pada saat menggunakan export dan import, data di dalamnya tidak boleh berupa angka. Seperti nomer telepon pada update yang harus diberikan petik satu (‘ ’). Bila tidak maka akan error nantinya.

44. Bisa menginstall node.js sesuai dengan yang ada di video (hingga sukses).

45. Semua penginstallan yang dilakukan (pada video 19) kebanyakan menggunakan command prompt.

46. Bisa menginstall yarn dan menamainya dengan react-resto sesuai dengan yang ada di video (hingga sukses).

47. Pada bagian nama tab dan folder dibagian atas library computer bisa dengan mengklik kanan lalu diketikkan cmd untuk mempemudah lokasi dari cmd nya (tidak perlu dituliskan manual dengan cd, namun bisa juga dengan cara manual (menambahkan cd)).

48. Bisa menginstall react router dom sesuai dengan yang ada di video (hingga sukses).

49. Pada saat menginstall package tambahan lainnya dengan yarn tidak boleh menggunakan kata install, namun diganti dengan add. Bila tidak, maka penginstallan yang dilakukan akan gagal.

50. Bisa menginstall axios untuk yarn sesuai dengan yang ada di video (hingga sukses).

51. Bisa menginstall react bootstrap sesuai dengan yang ada di video (hingga sukses).

52. Bisa menginstall react hook form sesuai dengan yang ada di video (hingga sukses).

53. Bisa memulai (start) yarn dengan terminal yang ada di dalam vs code (di terminal diketikkan yarn start untuk memulainya). Dan hasilnya akan muncul pada localhost:3000. Bisa juga dilakukan di dalam cmd (command prompt) bila di dalam vs code tidak dapat memulai, dengan mengetikkan hal yang sama (yarn start).

54. Menghapus beberapa command yang tidak diperlukan (yang ada di dalam App.js dan index.html (di dalam folder public untuk index.html nya) pada react-resto).

55. Bisa memunculkan code yang telah dibuat dengan yarn pada browser penampilnya (localhost:300). Seperti Belajar Raect salah satunya.

56. Menginstall JS JSX Snippets di dalam extension pada visual studio code (vs code). Digunakan untuk mempermudah membuat tag nantinya.

57. Menginstall html to JSX di dalam extension pada visual studi code (vs code). Digunakan untuk mengubah (diconvert) dari yang awalnya berupa tag html, bisa langsung berubah menjadi tag jsx.

58. Membuat beberapa folder yang nantinya akan digunakan selanjutnya.

59. Bisa membuat import dan export di dalam yarn (react-resto).

60. Bisa menjalankan snippet yang baru saja dibuat (langsung otomatis muncul untuk mempermudah pembuatan codingnya), kecuali tag yang ditambahkan di dalam emmet (include language, yang menambahkan javascript di dalam kotakan item dan javascript react di dalam kotakan value).

**Saya Belum Mengerti**

1. Pada saat menambahkan bahasa dibagian include language pada emmet (preferences -> setting) belum bisa ditambahkan dikarenakan terjadi error dan tidak muncul kotakan item dan value pada add itemnya. Memunculkan tulisan Setting has an invalid type, expected “object”. Fix in JSON. Saya akan mencoba menambahkannya nanti agar tag nya dapat digunakan (mempermudah, seperti tag <h1>).

2.

3.