|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Pembelajaran API LUMEN dasar | 94 |

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

1. **Membuat function update dengan code**

*Pelanggan*::where('idpelanggan',$id)->update($request->all());

        return response()->json("data sudah diupdate");

1. **“Pelanggan” disini berarti nama dari controller, dan 1 line code diatas menjalankan function update untuk mengganti isi dari tabel di SQL Database**
2. **Untuk line bawahnya, adalah response saat code tersebut berhasil dijalankan**
3. **Membuat validasi, yang jika user memasukkan data yang tidak sesuai dengan kriteria, maka program akan mengeluarkan response berupa error, code validasi berupa**

*$this*->validate($request,[

'pelanggan' => 'required',

'alamat' => 'required',

'telp' => 'required|numeric'

]);

1. **‘required’ = wajib diisi**

**‘number’ = wajib menggunakan angka**

1. **Membuat function create yang digunakan untuk mengupload image, dengan menggunakan code**

$gambar = $request->file('gambar')->getClientOriginalName();

        $request->file('gambar')->move('upload',$gambar);

        return response()->json($gambar);

1. **Untuk baris pertama, adalah perintah untuk meminta nama file beserta ektensinya yang berupa (.jpg), dan file akan dikirimkan menuju folder ‘upload’ pada line ke 2**
2. **Untuk line ke 3, berupa perintah untuk mengirimkan response sukses pada postman, jika gambar telah berhasil dikirim**
3. **Menggunakan function upload untuk mengisi seluruh isi dalam tabel menu didalam database, menggunakan object JSON**
4. **Menggunakan response HTML, contoh jika keluar response berupa angka ‘200’, maka browser berjalan dengan baik, dan jika mengeluarkan response berupa angka ‘400’, maka browser error dan tidak dapat menerima request**
5. **Membuat middleware, agar saat user tidak memiliki permission / bukan admin, user akan langsung diarahkan ke login page melalui routes, middleware disini berfungsi untuk memfilter siapa saja yang bisa mengakses page tertentu**
6. **Penggunaan API Token sama seperti middleware, jika user tidak memiliki token, maka user dianggap tidak memiliki permission untuk mengirim request ke database**
7. **Membuat registrasi user melalui postman, dengan membuat function create yang akan diisi dengan object sesuai dengan isi dari database**
8. **Membuat function login user, yang dimana user akan disuruh untuk memasukan email dan password, jika email & password terdaftar di database, maka program akan otomatis mengubah api\_token tersebut menjadi string secara acak menggunakan function**

*Str*::random(40);

1. **Dan jika email/password yang dimasukkan salah, program akan mengeluarkan response berupa ‘LOGIN GAGAL’, dan akan otomatis mengosongkan isi dari data**
2. **Penggunaan api\_token, agar user bisa mendapatkan data dari database, user harus memiliki sebuah api\_token untuk mengakses**
3. **Cara untuk mendapatkan api\_token, user harus registrasi terlebih dahulu**