



Nama : Avicenna Mumtaza
Kelas/Absen : TI-2H/07
NIM : 2241720112

Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 4 (empat) / 6 (enam)
Pertemuan ke- : 11 (sebelas)

JOBSHEET 11

RESTFUL API 2

Praktikum 1 – Implementasi Eloquent Accessor

1. Sebelum memulai pastikan REST API, terlebih dahulu pastikan sudah ter instal aplikasi Postman.
2. Kita akan memodifikasi Table `m_user` dengan menambahkan column : `image`, buka terminal lalu ketikkan

`php artisan make:migration add_image_to_m_user_table`

```
PS C:\laragon\www\PNL_POS> php artisan make:migration add_image_to_m_user_table  
[INFO] Migration [C:\laragon\www\PNL_POS\database\Migrations\2024_04_30_040822_add_image_to_m_user_table.php] created successfully.
```

```
PS C:\laragon\www\PNL_POS> php artisan make:migration add_image_to_m_user_table  
[INFO] Migration [C:\laragon\www\PNL_POS\database\Migrations\2024_05_12_135804_add_image_to_m_user_table.php] created successfully.  
PS C:\laragon\www\PNL_POS> █
```

3. Buka file migrasi tersebut lalu modifikasi seperti ini lalu simpan:



```
<?php use
Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends
Migration
{
    /**
     * Run the migrations.
     */
    public function up():
void
    {
        Schema::table('m_user', function (Blueprint $table) {
            $table->string('image');
        });
    }
    /**
     * Reverse the
migrations.
     */
    public function down():
void
    {
        Schema::table('m_user', function (Blueprint $table) {
            $table->dropColumn('image');
        });
    }
};
```

4. Lakukan jalankan update migrasi dengan cara: php artisan migrate
5. Lalu lakukan modifikasi models pada App/Models/UserModel.php Lakukan modifikasi pada Controllers/Api/RegisterController

```
<?php
namespace App\Http\Controllers\Api;
use App\Models\UserModel; use
App\Http\Controllers\Controller;
use Illuminate\Http\Request;
```



```
use Illuminate\Support\Facades\Validator;
class RegisterController extends
Controller
{
    public function __invoke(Request
$request)
    {
        //set validation
        $validator = Validator::make($request->all(), [
            'username' => 'required',
            'nama' => 'required',
            'password' => 'required|min:5|confirmed',
            'level_id' => 'required',
            'image' => 'required'

        ]);

        //if validations fails        if($validator-
>fails()){            return response()->json($validator->errors(),
422);        }

        //create user
        $user = UserModel::create([
            'username' => $request->username,
            'nama' => $request->nama,
            'password' => bcrypt($request->password),
            'level_id' => $request->level_id,
            'image' => $request->image
        ]);

        //return response JSON user is
created        if($user){
return response()->json([
            'success' => true,
            'user' => $user,
        ], 201);
        }

        //return JSON process insert failed
return response()->json([
            'success' => false,
        ], 409);    }
```



6. Anda dapat menambahkan detail untuk spesifikasi image pada validator

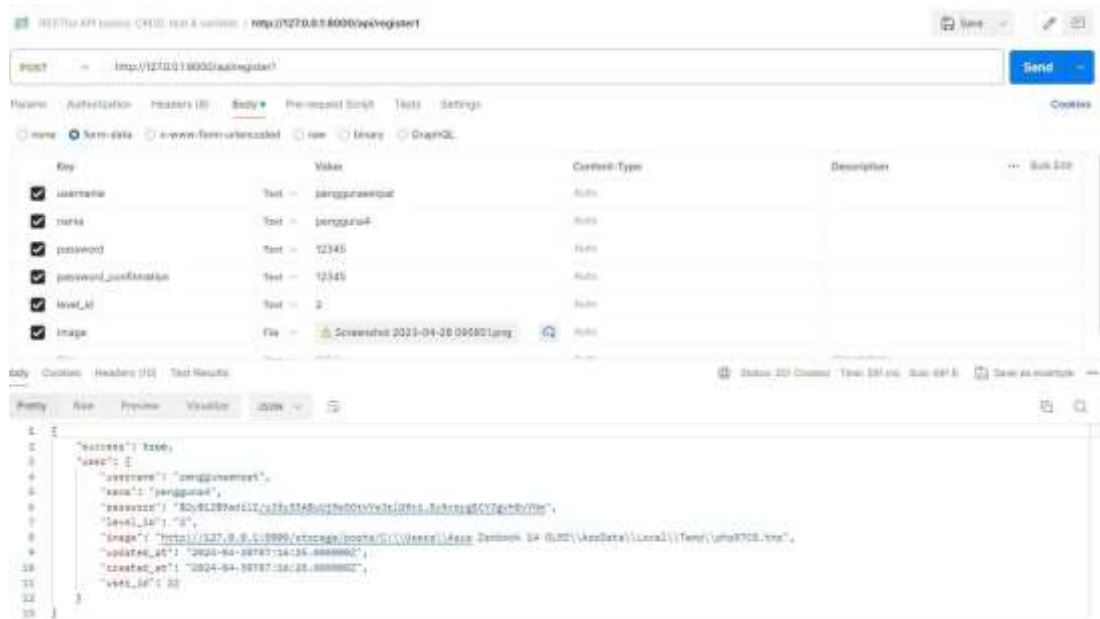
```
'image' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048',
```

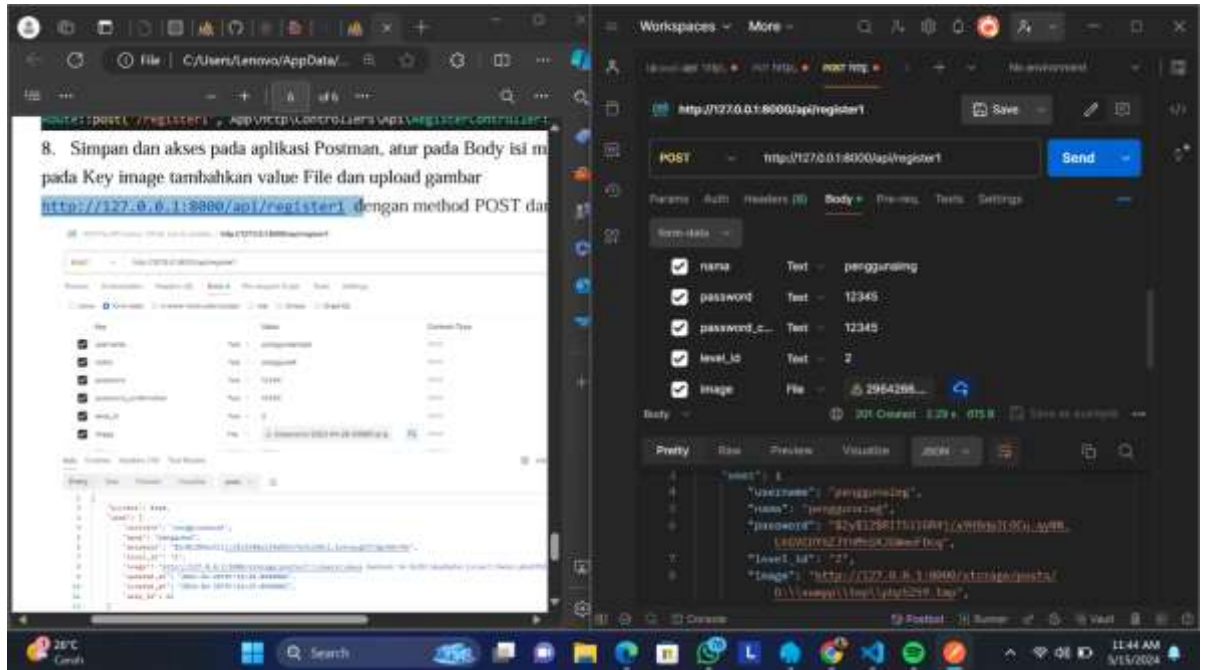
7. Ubah atau tambahkan register1 pada routes/api.php

```
Route::post('/register1', App\Http\Controllers\Api\RegisterController::class)->name('register1');
```

8. Simpan dan akses pada aplikasi Postman, atur pada Body isi manual Key dan Valuenya pada Key image tambahkan value File dan upload gambar

<http://127.0.0.1:8000/api/register1> dengan method POST dan klik send

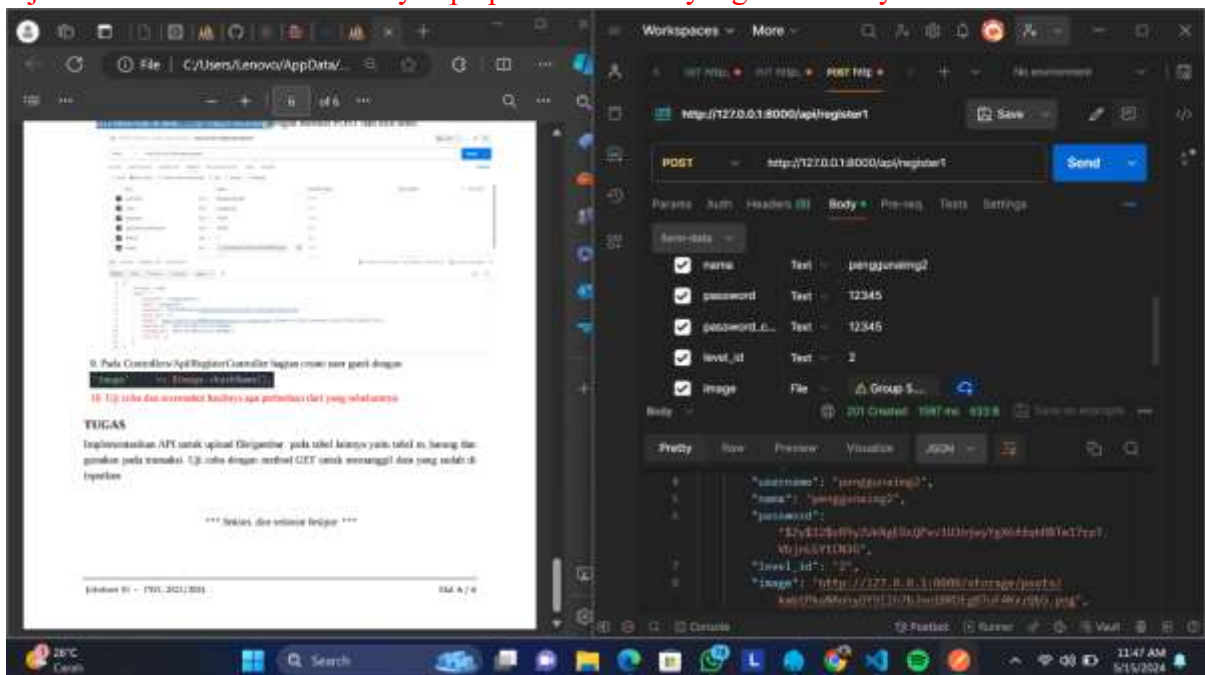




9. Pada Controllers/Api/RegisterController bagian create user ganti dengan

```
'image' => $image->hashName(),
```

10. Uji coba dan screenshot hasilnya apa perbedaan dari yang sebelumnya



Berdasarkan hasil percobaan diatas, sebelum diberi hashName() pada image, detail lokasi atau directory file yang diupload akan ditampilkan, sedangkan setelah diberi hashName() pada image, detail lokasi atau directory file yang diupload akan disembunyikan dan diganti dengan kode acak



dari hash. Kesimpulannya, fungsi `hashName` digunakan untuk merahasiakan lokasi atau directory file, sehingga keamanan file yang diupload akan terjamin.

TUGAS

Implementasikan API untuk upload file/gambar pada tabel lainnya yaitu tabel `m_barang` dan gunakan pada transaksi. Uji coba dengan method GET untuk memanggil data yang sudah di inputkan.

Uji Coba menginputkan data dan menampilkan data terinput :

The screenshot displays a web application on the left and a REST client on the right. The web application shows a form for adding a new item, with fields for `barang_kode`, `barang_nama`, and `harga_beli`. The REST client shows a POST request to `http://127.0.0.1:8000/api/barangs2` with the following body:

```
{
  "barang_kode": "BRG4999",
  "barang_nama": "Pesawat Kertas",
  "harga_beli": "1000",
  "harga_jual": "2000",
  "kategori_id": "3",
  "image": "http://127.0.0.1:8000/storage/posta/1a4jnPbWepwVetfQvB2a1E9uMCH11x32rg.png"
}
```




The screenshot shows the Postman application interface. At the top, there are tabs for 'Workspaces' and 'More'. Below this is a navigation bar with icons for search, user, settings, and notifications. The main interface is divided into several sections:

- Request Bar:** Shows the method 'GET' and the URL 'http://127.0.0.1:8000/api/barangs/15'. There are buttons for 'Save', 'Send', and a dropdown menu.
- Params Section:** Includes tabs for 'Params', 'Auth', 'Headers (6)', 'Body', 'Pre-req.', 'Tests', and 'Settings'. The 'Params' tab is active, showing 'Query Params' with a table:

<input type="checkbox"/>	Key	Value	Descr...	...	Bulk Edit
<input type="checkbox"/>	kategori_id				
<input type="checkbox"/>	barang_kode				

- Body Section:** Shows the response status '200 OK', time '191 ms', and size '576 B'. There are buttons for 'Save as example' and a dropdown menu. Below this are tabs for 'Pretty', 'Raw', 'Preview', and 'Visualize'. The 'Pretty' tab is active, displaying the JSON response:

```
1 {
2   "barang_id": 15,
3   "kategori_id": 2,
4   "barang_kode": "BRG4999",
5   "barang_nama": "Pesawat Kertas",
6   "harga_beli": 1000,
7   "harga_jual": 2000,
8   "created_at": "2024-05-15T05:31:14.000000Z",
9   "updated_at": "2024-05-15T05:31:14.000000Z",
10  "image": "http://127.0.0.1:8000/storage/posts/1akjhPDwBwqWPuekKqvK8raN24tE9bMCFDIX52rg.png"
11 }
```

At the bottom of the interface, there are icons for 'Console', 'Postbot', 'Runner', 'Vault', and other tools.



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

**** Sekian, dan selamat belajar ****