

Laporan Praktikum Pemrograman Web Lanjut

Jobsheet 11: Restful API



Oleh :

Danica Nasywa Putriniair (2341760122)

Kelas SIB 2B / 05

PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS JURUSAN

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

Jl. Soekarno Hatta No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang, Jawa Timur

65141

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

```
PS E:\Aplik\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 10> php artisan make:migration add_image_to_m_user_table

INFO Migration [E:\Aplik\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 10\database\migrations\2025_04_27_110110_add_image_to_m_user_table.php] created successfully.
```

php artisan make:migration add_image_to_m_user_table

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

```
database > migrations > 2025_04_27_110110_add_image_to_m_user_table.php > ...
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12       public function up(): void
13       {
14           Schema::table('m_user', function (Blueprint $table) {
15               $table->string('image');
16           });
17       }
18
19       /**
20        * Reverse the migrations.
21        */
22       public function down(): void
23       {
24           Schema::table('m_user', function (Blueprint $table) {
25               $table->dropColumn('image');
26           });
27       }
28 };
29
```

4. Lakukan jalankan update migrasi dengan cara: php artisan migrate

```
PS E:\Aplik\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 10> php artisan migrate

INFO Running migrations.

2025_04_27_110110_add_image_to_m_user_table ..... 532ms DONE

PS E:\Aplik\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 10>
```

5. Lalu lakukan modifikasi models pada App/Models/UserModel.php

```

public function getJWTIdentifier(){
    return $this->getKey();
}

public function getJWTCustomClaims(){
    return [];
}

protected $table = 'm_user'; //mendefinisikan nama tabel yang digunakan model
protected $primaryKey = 'user_id'; //mendefinisikan primary key dari tabel

protected $fillable = [
    'level_id',
    'username',
    'nama',
    'password',
    'created_at',
    'updated_at',
    'foto_profile',
    'image'
]; //mendefinisikan kolom yang akan diisi oleh user

protected $hidden = ['password']; // jangan di tampilkan saat select

protected $casts = ['password' => 'hashed']; // casting password agar otomatis dihashing

// relasi ke tabel level
public function level(): BelongsTo
{

```

6. Lakukan modifikasi pada Controllers/Api/RegisterController

```

Http > Controllers > Api > RegisterController.php > RegisterController > __inv
namespace App\Http\Controllers\Api;

use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\UserModel;
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);
    }
}

```

7. Anda dapat menambahkan detail untuk spesifikasi image pada validator

```

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

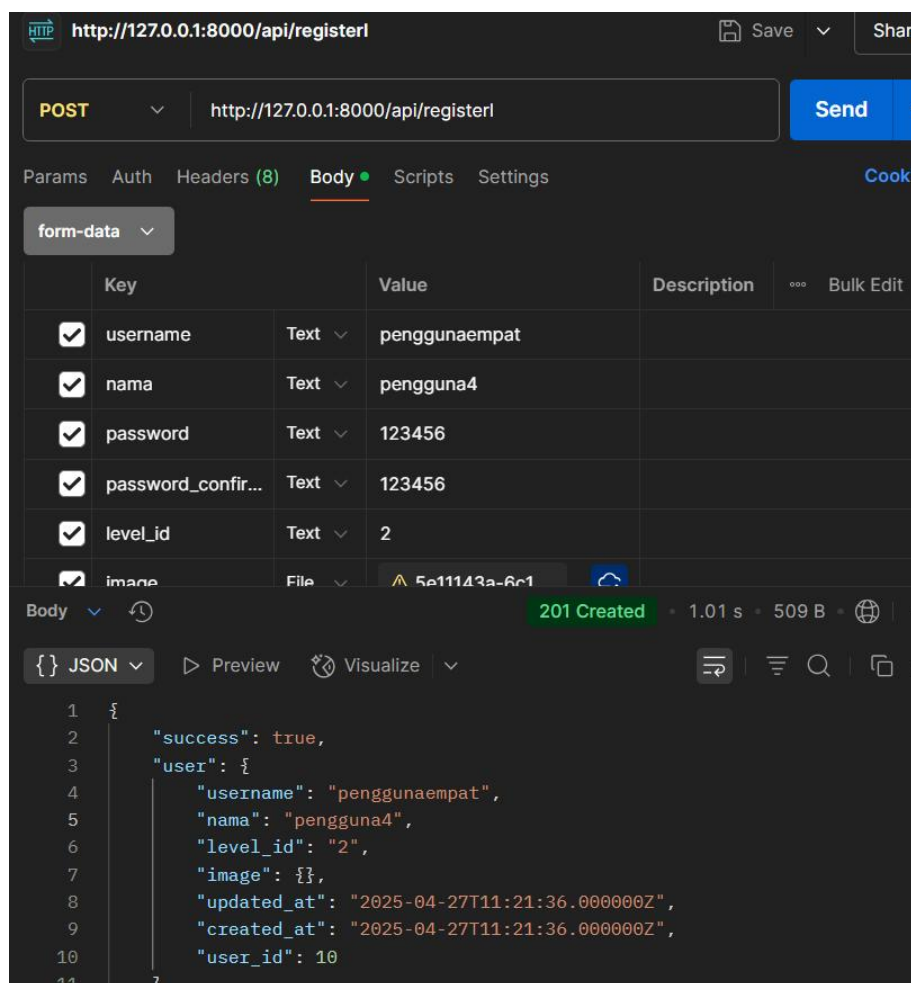
```

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

```
Route::post('/register1', RegisterController::class)->name('register');
```

9. 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



10. Pada Controllers/Api/RegisterController bagian create user ganti dengan Saya ubah seperti ini

```

// Handle image upload
if ($request->hasFile('image')) {
    $image = $request->file('image');
    $imageName = $image->hashName(); // Generate a unique hash for the filename
    $image->storeAs('public/posts', $imageName); // Store the image in storage/app/pu
} else {
    $imageName = null; // Jika tidak ada file yang diunggah (walaupun validasi mengha
}

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

```

Karena Error "Undefined variable '\$image'" terjadi karena kamu mencoba menggunakan variabel \$image untuk mendapatkan hashName()-nya, tetapi variabel \$image belum pernah dideklarasikan atau diberi nilai di dalam scope method __invoke sebelum baris tersebut.

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

HTTP <http://127.0.0.1:8000/api/registerl> Save

POST <http://127.0.0.1:8000/api/registerl> Send

Params Auth Headers (8) Body Scripts Settings

form-data

Key	Text	Value	Description
<input checked="" type="checkbox"/> username	Text	penggunalima	
<input checked="" type="checkbox"/> nama	Text	pengguna5	
<input checked="" type="checkbox"/> password	Text	123456	
<input checked="" type="checkbox"/> password_confirm...	Text	123456	
<input checked="" type="checkbox"/> level_id	Text	2	
<input checked="" type="checkbox"/> image	File	taesan *.jpg	

Body 201 Created • 2.01 s • 552 B •

{ } JSON Preview Visualize

```

1 {
2   "success": true,
3   "user": {
4     "username": "penggunalima",
5     "nama": "pengguna5",
6     "level_id": "2",
7     "image": "SDSCXGdNNAYsLVgiRu1Hf1yI3qUALmEgE5qNNTcQ.jpg",
8     "updated_at": "2025-04-27T11:25:34.000000Z",
9     "created_at": "2025-04-27T11:25:34.000000Z",
10    "user_id": 11

```

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.