



Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Sistem Informasi Bisnis
Semester : 5

Kelas : SIB 3D
NIM : 2241760026
Nama : Siti Faiqoh
Jobsheet Ke- : 11

Laporan Jobsheet

Praktikum 1 – Implementasi Eloquent Accessor

| Langkah | Jawaban/Deskripsi |
|---------|--|
| 1 | Sebelum memulai pastikan REST API, terlebih dahulu pastikan sudah ter instal aplikasi Postman. |
| 2 | <p>Kita akan memodifikasi Table m_user dengan menambahkan column : image, buka terminal lalu ketikkan</p> <p>php artisan make:migration add_image_to_m_user_table</p> <pre>PS C:\laragon\www\PWL_POS> php artisan make:migration add_image_to_m_user_table</pre> <p>INFO Migration [C:\laragon\www\PWL_POS\database\migrations\2024_11_06_091026_add_image_to_m_user_table.php] created successfully.</p> |
| 3 | <p>Buka file migrasi tersebut lalu modifikasi seperti ini lalu simpan:</p> <pre>database > migrations > 2024_11_06_091026_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 * Jalankan migration. 11 */ 12 public function up(): void 13 { 14 // Menambahkan kolom 'image' pada tabel 'm_user' 15 Schema::table('m_user', function (Blueprint \$table) { 16 \$table->string('image'); // Menambah kolom bertipe string untuk menyimpan nama file gambar 17 }); 18 } 19 20 /** 21 * Membatalkan migration. 22 */ 23 public function down(): void 24 { 25 // Menghapus kolom 'image' dari tabel 'm_user' 26 Schema::table('m_user', function (Blueprint \$table) { 27 \$table->dropColumn('image'); // Menghapus kolom 'image' jika migration dibatalkan 28 }); 29 } 30 };</pre> |



| | |
|---|---|
| 4 | <p>Lakukan jalankan update migrasi dengan cara php artisan migrate</p> <pre>PS C:\laragon\www\PWL_P05> php artisan migrate [INFO] Running migrations. 2024_11_06_091026_add_image_to_m_user_table 439ms DONE</pre> |
| 5 | <p>Lalu lakukan modifikasi models pada App/Models/UserModel.php</p> <pre>app > Models > UserModel.php > UserModel > hasRole 1 <?php 2 3 namespace App\Models; 4 5 use Illuminate\Database\Eloquent\Factories\HasFactory; 6 // use Illuminate\Database\Eloquent\Model; 7 use Illuminate\Database\Eloquent\Relations\BelongsTo; 8 use Illuminate\Foundation\Auth\User as Authenticatable; 9 use Tymon\JWTAuth\Contracts\JWTSubject; 10 use Illuminate\Database\Eloquent\Casts\Attribute; 11 12 class UserModel extends Authenticatable implements JWTSubject 13 { 14 public function getJWTIdentifier() 15 { 16 return \$this->getKey(); 17 } 18 19 public function getJWTCustomClaims() { 20 return []; 21 } 22 23 use HasFactory; 24 protected \$table = 'm_user'; 25 protected \$primaryKey = 'user_id'; 26 protected \$fillable = [27 'username', 28 'nama', 29 'password', 30 'level_id', 31 'image' 32]; 33 34 protected \$casts = ['password' => 'hashed']; 35 36 public function level() : BelongsTo 37 { 38 return \$this->belongsTo(LevelModel::class, 'level_id', 'level_id'); 39 } 40 41 //nama role 42 public function getRoleName(): string{ 43 return \$this->level->level_nama; 44 } 45 46 // apakah user memiliki role tertentu 47 public function hasRole(\$role): bool{ 48 return \$this->level->level_kode == \$role; 49 } 50 51 //Mendapatkan kode role 52 public function getRole() 53 { 54 return \$this->level->level_kode; 55 } 56 57 protected function image(): Attribute 58 { 59 return Attribute::make(60 get: fn (\$image) => url('/storage/posts/'.\$image), 61); 62 } 63 }</pre> |
| 6 | <p>Lakukan modifikasi pada Controllers/Api/RegisterController</p> |



```
app > Http > Controllers > Api > RegisterController.php > RegisterController > __invoke
1  <?php
2
3  namespace App\Http\Controllers\Api;
4
5  use App\Http\Controllers\Controller;
6  use App\Models\UserModel;
7  use Illuminate\Http\Request;
8  use Illuminate\Support\Facades\Validator;
9
10 class RegisterController extends Controller
11 {
12     public function __invoke(Request $request)
13     {
14         // Validasi input: 'username', 'nama', 'password', dan 'level_id' wajib diisi
15         $validator = Validator::make($request->all(), [
16             'username' => 'required',
17             'nama' => 'required',
18             'password' => 'required|min:5|confirmed', // password minimal 5 karakter dan konfirmasi
19             'level_id' => 'required',
20             'image' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048'
21         ]);
22
23         // Jika validasi gagal, kirim respons error 422
24         if ($validator->fails()) {
25             return response()->json($validator->errors(), 422);
26         }
27
28         $image = $request->file('image');
29
30         // Membuat user baru dengan data yang valid
31         $user = UserModel::create([
32             'username' => $request->username,
33             'nama' => $request->nama,
34             'password' => bcrypt($request->password), // Enkripsi password
35             'level_id' => $request->level_id,
36             'image' => $request->image, // url nya sesuai dengan nama penyimpanan di laptop
37             // 'image' => $image->hashName(), // urlnya hash
38         ]);
39
40         // Jika user berhasil dibuat, kirim respons sukses 201
41         if ($user) {
42             return response()->json([
43                 'success' => true,
44                 'user' => $user,
45             ], 201);
46         }
47
48         // Jika gagal menyimpan, kirim respons error 409
49         return response()->json([
50             'success' => false,
51         ], 409);
52     }
53 }
```

7

Anda dapat menambahkan detail untuk spesifikasi image pada validator

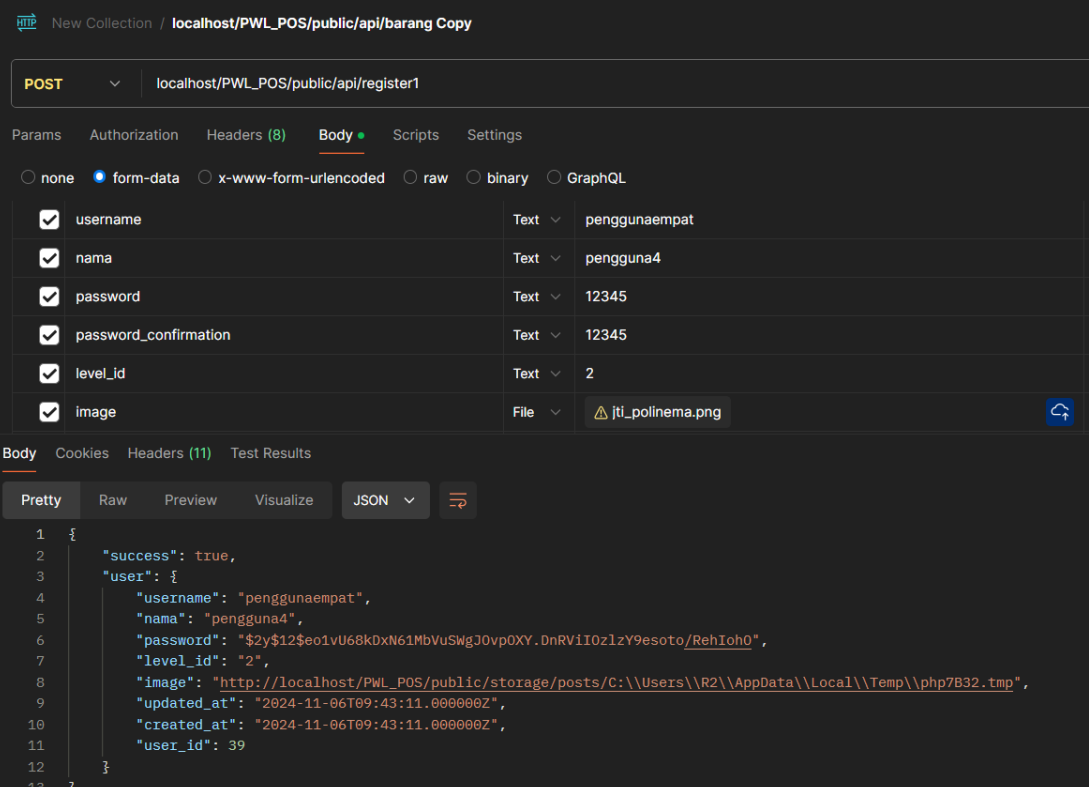
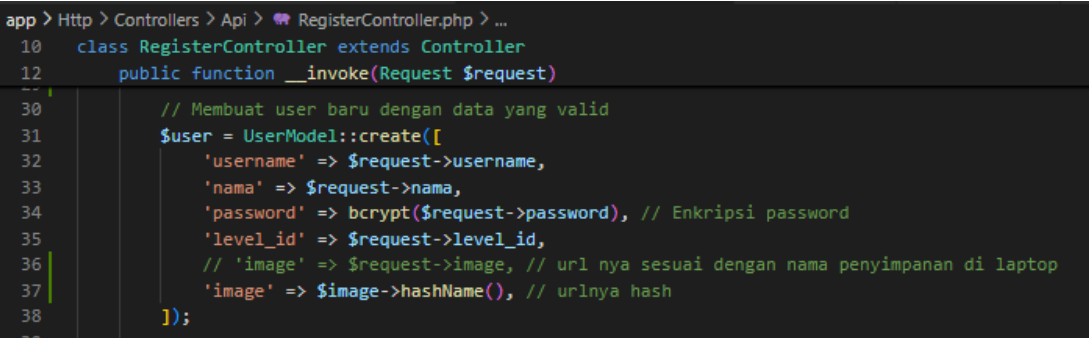
```
app > Http > Controllers > Api > RegisterController.php > RegisterController > __invoke
10 class RegisterController extends Controller
12     public function __invoke(Request $request)
20         'image' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048'
21     ];
22 }
```

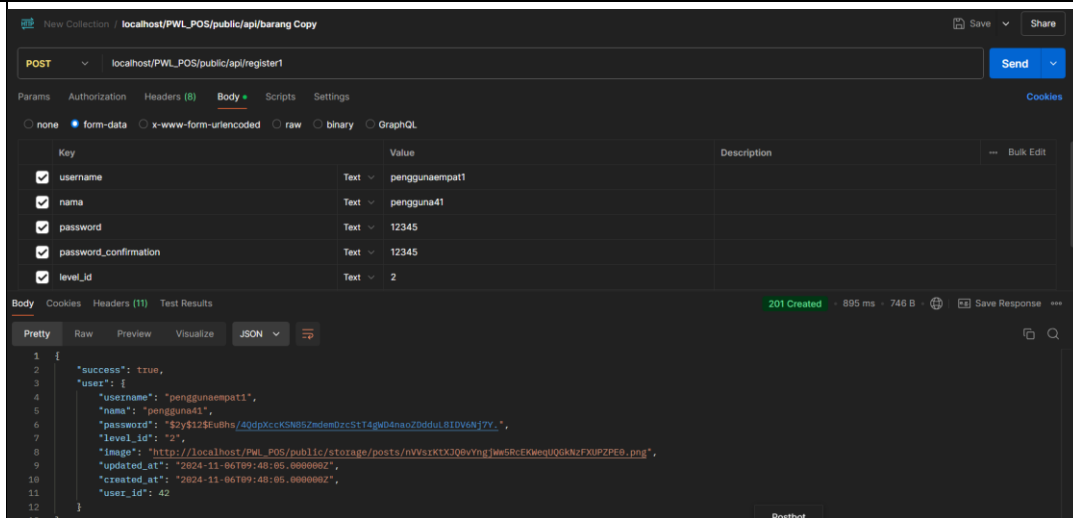
8

Ubah atau tambahkan register1 pada routes/api.php

```
routes > api.php > ...
23 // Register route
24 Route::post('/register', RegisterController::class)->name('register');
25 Route::post('/register1', RegisterController::class)->name('register1');
26
```



| | |
|----|--|
| 9 | <p>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</p>  |
| 10 | <p>Pada Controllers/Api/RegisterController bagian create user ganti dengan</p>  |
| 11 | <p>Uji coba dan screenshot hasilnya apa perbedaan dari yang sebelumnya</p> |



Terdapat perbedaan alamat image

| | | | | | | | | |
|----|---|----------------|------------|------|--|---------------------|---------------------|---|
| 39 | 2 | penggunaempat | pengguna4 | NULL | \$2y\$12\$e01vU68kDxN61MbVuSWgJOvpOXYDnRvIOzIzY9eso... | 2024-11-06 09:43:11 | 2024-11-06 09:43:11 | C:\Users\IR2\AppData\Local\Temp\php7B32.tmp |
| 42 | 2 | penggunaempat1 | pengguna41 | NULL | \$2y\$12\$EuBhs/4QdpKccKSN85ZedemDzcStT4gWd4naoZddul8IDV6N17V. | 2024-11-06 09:48:05 | 2024-11-06 09:48:05 | nVsrKtXJQ0vYngJw5RcEKWeqUQGKnZFXUPZPE0.png |

TUGAS

| Langkah | Jawaban/Deskripsi |
|---------|--|
| 1 | 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. |
| | Membuat file migration untuk menambahkan kolom gambar/image pada tabel t_penjualan <pre>PS C:\laragon\www\PWL_POS> php artisan make:migration add_image_to_t_penjualan_table.php</pre> <p>INFO Migration [C:\laragon\www\PWL_POS\database\migrations\2024_11_06_095549_add_image_to_t_penjualan_table.php.php] created successfully.</p> |



database > migrations > 2024_11_06_095549_add_image_to_t_penjualan_table.php.php > ...

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

Memodifikasi file pada PenjualanModel.php

app > Models > PenjualanModel.php > penjualanmodel

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\Relations\BelongsTo;
8  use Illuminate\Database\Eloquent\Relations\HasMany;
9  use Illuminate\Database\Eloquent\Casts\Attribute;
10
11 class penjualanmodel extends Model
12 {
13     use HasFactory;
14     protected $table = 't_penjualan'; // Mendefinisikan nama tabel yang digunakan oleh model ini
15     protected $primaryKey = 'penjualan_id'; // Mendefinisikan primary key dari tabel yang digunakan
16
17     protected $fillable = ['penjualan_id', 'user_id', 'pembeli', 'penjualan_kode', 'penjualan_tanggal', 'image'];
18
19     // public function detail():HasMany
20     // {
21     //     return $this->hasMany(DetailModel::class, 'detail_id','detail_id');
22     // }
23     public function user():BelongsTo
24     {
25         return $this->belongsTo(UserModel::class, 'user_id','user_id');
26     }
27     public function image(): Attribute
28     {
29         return Attribute::make(
30             get: fn($image) => url('/storage/posts/' . $image),
31         );
32     }
33 }
34
```

Membuat file baru penjualanController.php pada api

PS C:\laragon\www\PWL_POS> php artisan make:controller Api\PenjualanController

INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\Api\PenjualanController.php] created successfully.



hasil

POST <http://127.0.0.1:8000/api/penjualan> **Send**

Params Authorization Headers (8) **Body** Scripts Settings

☐ none ☒ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

| Key | Value |
|-------------------|---|
| penjualan_kode | TRX0001 |
| penjualan_tanggal | 2024-11-4 16:32:40 |
| image | Logo-Polinema-Politeknik-Negeri-Malang-PNG.p... |
| penjualan_id | 20 |

Body Cookies Headers (10) Test Results **201 Created** · 924 ms · 681 B · Save Response

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "success": true,
3   "message": "penjualan berhasil ditambahkan",
4   "data": {
5     "user_id": "1",
6     "pembeli": "angel",
7     "penjualan_kode": "TRX0001",
8     "penjualan_tanggal": "2024-11-4 16:32:40",
9     "image": "http://127.0.0.1:8000/storage/posts/B621qE3aFXLdc8DL3ungv6wzW0y6V00zPz22INw.png",
10    "updated_at": "2024-11-06T09:48:59.000000Z",
11    "created_at": "2024-11-06T09:48:59.000000Z"
12  }
13 }
```

Postbot
Ctrl Alt P