

NIM : 2241720095
Nama : Ana Bellatus Mustaqfiro
Kelas : D4 TI - 2H
No. Urut : 04
Mata Kuliah : Pemrograman Web Lanjut 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\PWL_POS> php artisan make:migration add_image_to_m_user_table  
  
[INFO] Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_05_15_065234_add_image_to_m_user_table.php] created successfully.
```

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

```
database > migrations > 2024_05_15_065234_add_image_to_m_user_table.php > ...  
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 };
```

4. Lakukan jalankan update migrasi dengan cara:

`php artisan migrate`

```
PS C:\laragon\www\PWL_POS> php artisan migrate  
  
[INFO] Running migrations.  
  
2024_05_15_065234_add_image_to_m_user_table ..... 234ms DONE
```

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

```
app > Models > UserModel.php > PHP Intelephense > UserModel
30     protected $fillable = [
31         'user_id',
32         'username',
33         'nama',
34         'password',
35         'level_id',
36         'image',
37     ];
38
39     public function level(): BelongsTo
40     {
41         return $this->belongsTo(LevelModel::class, 'level_id', 'level_id');
42     }
43
44     protected function image(): Attribute
45     {
46         return Attribute::make(
47             get: fn ($image) => url('/storage/posts/' . $image),
48         );
49     }
```

6. Lakukan modifikasi pada Controllers/Api/RegisterController

```
app > Http > Controllers > Api > RegisterController.php > PHP Intelephense > RegisterController > __invoke
12     public function __invoke(Request $request)
13     {
14         $validator = Validator::make($request->all(), [
15             'username' => 'required',
16             'nama' => 'required',
17             'password' => 'required|min:5|confirmed',
18             'level_id' => 'required',
19             'image' => 'required'
20         ]);
21
22         if ($validator->fails()) {
23             return response()->json($validator->errors(), 422);
24         }
25
26         $user = UserModel::create([
27             'username' => $request->username,
28             'nama' => $request->nama,
29             'password' => bcrypt($request->password),
30             'level_id' => $request->level_id,
31             'image' => $request->image
32         ]);
33     }
```

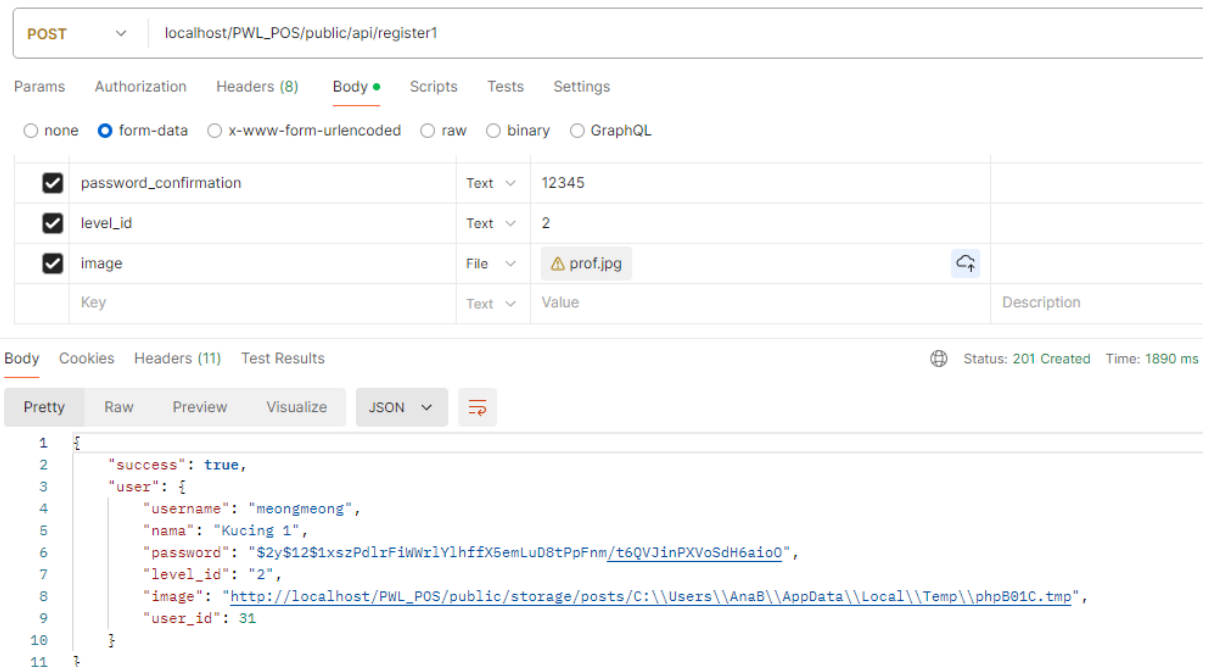
7. Anda dapat menambahkan detail untuk spesifikasi image pada validator

```
app > Http > Controllers > Api > RegisterController.php > PHP Intelephense > RegisterController > __invoke
12     public function __invoke(Request $request)
13     {
14         $validator = Validator::make($request->all(), [
15             'username' => 'required',
16             'nama' => 'required',
17             'password' => 'required|min:5|confirmed',
18             'level_id' => 'required',
19             'image' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048'
20         ]);
21     }
```

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

```
routes > api.php > ...  
27 | Route::post('/register1', App\Http\Controllers\Api\RegisterController::class)->name('register1');
```

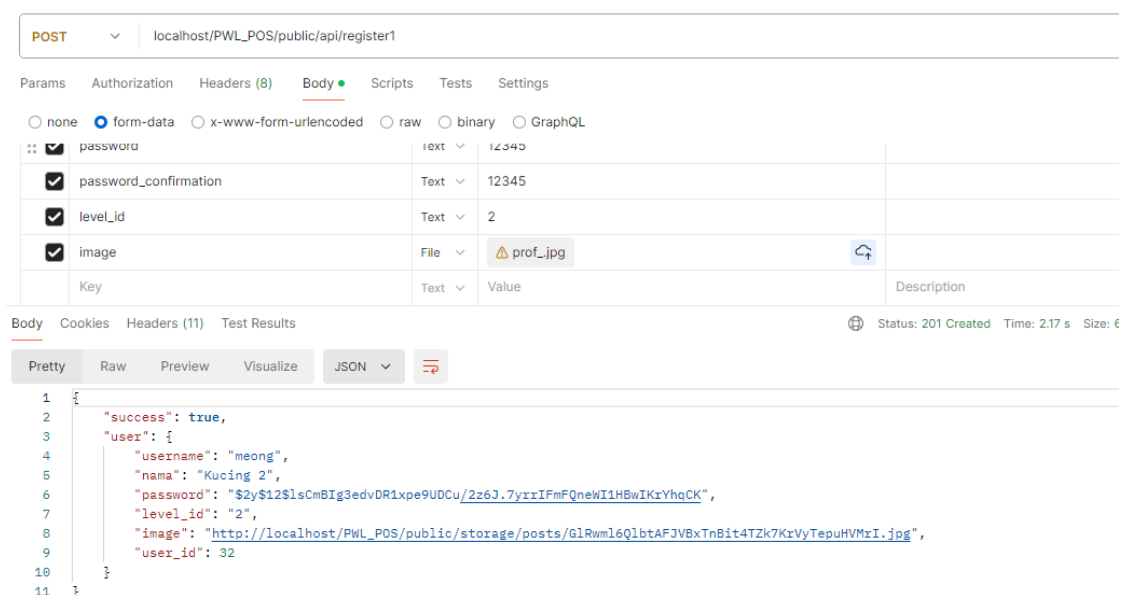
9. Simpan dan akses pada aplikasi Postman, atur pada Body isi manual Key dan Valuenya pada Key image tambahkan value File dan upload gambar http://127.0.0.1:8000/api/register1 dengan method POST dan klik send



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

```
app > Http > Controllers > Api > RegisterController.php > PHP Intelephense > RegisterController > _invoke  
19 | 'image' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048'
```

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



perbedaan dari sebelumnya adalah, pada langkah sebelumnya path dan nama file terlihat jelas, sedangkan pada langkah ini nama file dienkripsi dan disamarkan sehingga lebih aman

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 diinputkan.

```
PS C:\laragon\www\PWL_POS> php artisan make:migration add_image_to_m_barang_table
```

```
INFO Migration [C:\laragon\www\PWL_POS\database\migrations\2024_05_15_074624_add_image_to_m_barang_table.php] created successfully.
```

```
database > migrations > 2024_05_15_074624_add_image_to_m_barang_table.php > ...
```

```
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::table('m_barang', 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_barang', function (Blueprint $table) {
25             $table->dropColumn('image');
26         });
27     }
28 };
```

```
PS C:\laragon\www\PWL_POS> php artisan migrate
```

```
INFO Running migrations.
```

```
2024_05_15_074624_add_image_to_m_barang_table ..... 79ms DONE
```

```
app > Models > BarangModel.php > ...
```

```
17 protected $fillable = [
18     'barang_kode',
19     'barang_nama',
20     'harga_beli',
21     'harga_jual',
22     'kategori_id',
23     'image'
24 ];
```

```
app > Models > BarangModel.php > ...
```

```
36 protected function image(): Attribute
37 {
38     return Attribute::make(
39         get: fn ($image) => url('/storage/posts/' . $image),
40     );
41 }
```

POST localhost/PWL_POS/public/api/barangs

Params Authorization Headers (8) Body Scripts Tests Settings

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

	Key	Value	Description
<input checked="" type="checkbox"/>	kategori_id	Text 11	
<input checked="" type="checkbox"/>	barang_kode	Text SPT002	
<input checked="" type="checkbox"/>	barang_nama	Text Ventela	
<input checked="" type="checkbox"/>	harga_beli	Text 300000	
<input checked="" type="checkbox"/>	harga_jual	Text 350000	
<input checked="" type="checkbox"/>	image	File prof_.jpg	

Body Cookies Headers (11) Test Results Status: 201 Created Time: 791 ms Size: 694 B

Body Cookies Headers (11) Test Results Status: 201 Created Time: 791 ms Size

Pretty Raw Preview Visualize JSON

```

1 {
2   "barang_kode": "SPT002",
3   "barang_nama": "Ventela",
4   "harga_beli": "300000",
5   "harga_jual": "350000",
6   "kategori_id": "11",
7   "image": "http://localhost/PWL_POS/public/storage/posts/vfXiY0rAvyofu03KXeMJ0nJg9FK10n2VD10YCBVp.jpg",
8   "updated_at": "2024-05-15T08:07:55.000000Z",
9   "created_at": "2024-05-15T08:07:55.000000Z",
10  "barang_id": 15
11 }

```

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

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

app > Http > Controllers > Api > TransaksiController.php > PHP Intelephense > TransaksiController > show

```

16 public function show(string $id)
17 {
18     $penjualan = TransaksiModel::with('detail.barang')->findOrFail($id);
19
20     $response = [
21         'id penjualan' => $penjualan->penjualan_id,
22         'pembeli' => $penjualan->pembeli,
23         'kode penjualan' => $penjualan->penjualan_kode,
24         'tanggal' => $penjualan->penjualan_tanggal,
25         'barang yang dibeli' => [],
26         'total belanja' => 0,
27     ];
28
29     foreach ($penjualan->detail as $item) {
30         $barang = $item->barang;
31         $subtotal = $item->harga * $item->jumlah;
32         $response['barang yang dibeli'][] = [
33             'nama barang' => $barang->barang_nama,
34             'harga' => $item->harga,
35             'jumlah' => $item->jumlah,
36             'subtotal' => $subtotal,
37             'image' => url('storage/posts/' . $barang->image),
38         ];
39         $response['total belanja'] += $subtotal;
40     }
41
42     return response()->json($response, 200);
43 }
44 }

```

```
routes > api.php > ...
61 Route::get('transaksis/{transaksis}', [TransaksiController::class, 'show']);
```

GET localhost/PWL_POS/public/api/transaksis/2

Params Authorization Headers (6) Body Scripts Tests Settings

Query Params

	Key	Value	Description
--	-----	-------	-------------

Body Cookies Headers (11) Test Results

Status: 200 OK Time: 670 ms Size: 901 B Sa

Pretty Raw Preview Visualize JSON

```
1 {
2   "id penjualan": 2,
3   "pembeli": "Bella",
4   "kode penjualan": "PJI002",
5   "tanggal": "2024-03-06",
6   "barang yang dibeli": [
7     {
8       "nama barang": "Teh Botol",
9       "harga": 7000,
10      "jumlah": 3,
11      "subtotal": 21000,
12      "image": "http://localhost/PWL_POS/public/storage/posts"
13    },
14    {
15      "nama barang": "Kaos Oblong",
16      "harga": 55000,
17      "jumlah": 2,
18      "subtotal": 110000.
19    }
20  ],
21   "total belanja": 201000
22 }
```