

PRAKTIKUM PEMROGRAMAN WEB LANJUT

TUGAS PERTEMUAN 11

RESTFUL API 2

DISUSUN OLEH :

GILANG PURNOMO

NIM:2341720042



PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

APRIL 2025

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
Jawab : membuat kolom baru untuk foto sudah dibuat pada jobsheet 8
3. Buka file migrasi tersebut lalu modifikasi seperti ini lalu simpan:

```
Minggu 8 > PWL_POS > database > migrations > 2025_04_10_143039_add_foto_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('photo')->nullable()->after('password');
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(['photo']);
26           });
27       }
28
29  };
```

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

```
Minggu 11 > PWL_POS > app > Models > UserModel.php > PHP Intelephense > UserModel > photo
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Model;
6  use Tymon\JWTAuth\Contracts\JWTSubject;
7  use Illuminate\Database\Eloquent\Casts\Attribute;
8  use Illuminate\Database\Eloquent\Relations\BelongsTo;
9  use Illuminate\Foundation\Auth\User as Authenticatable;
10
11  class UserModel extends Authenticatable implements JWTSubject
12  {
13      public function getJWTIdentifier()
14      {
15          return $this->getKey();
16      }
17
18      public function getJWTCustomClaims()
19      {
20          return [];
21      }
22
23      protected $table = 'm_user';
24      protected $primaryKey = 'user_id';
25
26      protected $fillable = ['level_id', 'username', 'nama', 'password', 'photo', 'create_at', 'update_at'];
27
28      protected $hidden = ['password'];
29
30      protected $casts = ['password' => 'hashed'];
31
32      public function level(): BelongsTo
33      {
34          return $this->belongsTo(LevelModel::class, 'level_id', 'level_id');
35      }
36
37      protected function photo(): Attribute
38      {
39          return Attribute::make(
40              get: fn ($photo) => url('/storage/posts/' . $photo),
41          );
42      }
43  }
```

6. Lakukan modifikasi pada Controllers/Api/RegisterController

```
Minggu 11 > PWL_POS > app > Http > Controllers > Api > RegisterController.php > PHP Inte
1  <?php
2
3  namespace App\Http\Controllers\Api;
4
5  use App\Models\UserModel;
6  use Illuminate\Http\Request;
7  use App\Http\Controllers\Controller;
8  use Illuminate\Support\Facades\Validator;
9
10 class RegisterController extends Controller
11 {
12     public function __invoke(Request $request)
13     {
14         $validator = Validator::make($request->all(), [
15             'username' => 'required',
16             'nama' => 'required',
17             'password' => 'required',
18             'level_id' => 'required',
19             'photo' => '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' => $request->password,
30             'level_id' => $request->level_id,
31             'photo' => $request->photo
32         ]);
33
34         if ($user) {
35             return response()->json([
36                 'success' => true,
37                 'user' => $user,
38             ], 201);
39         }
40
41         return response()->json([
42             'success' => false,
43         ], 409);
44     }
45 }
46
```

7. Anda dapat menambahkan detail untuk spesifikasi image pada validator

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

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

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

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

POST localhost/PWL_POS/public/api/register1 Send

Params Auth Headers (8) Body • Scripts Settings Cookies

form-data

<input checked="" type="checkbox"/>	username	Text	penggunaempat
<input checked="" type="checkbox"/>	nama	Text	pengguna4
<input checked="" type="checkbox"/>	password	Text	123456
<input checked="" type="checkbox"/>	password_confir...	Text	123456
<input checked="" type="checkbox"/>	level_id	Text	2
<input checked="" type="checkbox"/>	photo	File	ironman.jpeg

Body 201 Created 10.86 s 675 B

{ } JSON Preview Visualize

```

1 {
2   "success": true,
3   "user": {
4     "username": "penggunaempat",
5     "nama": "pengguna4",
6     "level_id": "2",
7     "photo": "http://localhost/PWL_POS/public/storage/posts/
8       C:\\Users\\ASUS\\AppData\\Local\\Temp\\php528F.tmp",
9     "updated_at": "2025-04-25T14:44:58.000000Z",
10    "created_at": "2025-04-25T14:44:58.000000Z",
11    "user_id": 39
12  }
13 }

```

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

`'photo' => $request->photo->hashName(),`

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

POST localhost/PWL_POS/public/api/register1 Send

Params Auth Headers (8) Body • Scripts Settings Cookies

form-data

<input checked="" type="checkbox"/>	username	Text	penggunasembilan
<input checked="" type="checkbox"/>	nama	Text	pengguna9
<input checked="" type="checkbox"/>	password	Text	123456
<input checked="" type="checkbox"/>	password_confir...	Text	123456
<input checked="" type="checkbox"/>	level_id	Text	2
<input checked="" type="checkbox"/>	photo	File	spiderman.jp...

Body 201 Created 3.60 s 672 B

{ } JSON Preview Visualize

```

1 {
2   "success": true,
3   "user": {
4     "username": "penggunasembilan",
5     "nama": "pengguna9",
6     "level_id": "2",
7     "photo": "http://localhost/PWL_POS/public/storage/posts/
8       XymR4860pr2EncTCBmjNjgbw6bwvzZpAzSns6Nnb.jpg",
9     "updated_at": "2025-04-25T14:47:48.000000Z",
10    "created_at": "2025-04-25T14:47:48.000000Z",
11    "user_id": 40
12  }
13 }

```

Yang berbeda adalah nama filenya menjadi ter hash

Link Commit :

https://github.com/Gilangp/Pemrograman_Web_Lanjut/commit/b6cb9c6f11bb0339d8278201c3493a25cf74f7be

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.

Membuat barang baru

POST localhost/PWL_POS/public/api/barangs Send

Params Auth Headers (8) Body Scripts Settings Cookies

form-data

<input checked="" type="checkbox"/>	kategori_id	Text	5
<input checked="" type="checkbox"/>	barang_kode	Text	BRG-09
<input checked="" type="checkbox"/>	barang_nama	Text	Setrika
<input checked="" type="checkbox"/>	image	File	Setrika.jpeg
<input checked="" type="checkbox"/>	harga_beli	Text	200000
<input checked="" type="checkbox"/>	harga_jual	Text	300000

Body 201 Created 3.51 s 722 B

JSON

```
2  "success": true,
3  "barang": {
4    "kategori_id": "5",
5    "barang_kode": "BRG-09",
6    "barang_nama": "Setrika",
7    "image": "http://localhost/PWL_POS/public/storage/barang/
8    BG2RZLICUUaK048lnXuixfML5kheGE99W7YgZgxp.jpg",
9    "harga_beli": "200000",
10   "harga_jual": "300000",
11   "updated_at": "2025-04-25T16:31:18.000000Z",
12   "created_at": "2025-04-25T16:31:18.000000Z",
    "barang_id": 27
  }
```

data tersimpan juga di database.

<input type="checkbox"/>	Edit	Copy	Delete	27	5	BRG-09	Setrika	BG2RZLICUUaK048lnXuixfML5kheGE99W7YgZgxp.jpg	200000	300000	2025-04-25 16:31:18	2025-04-25 16:31:18
--------------------------	------	------	--------	----	---	--------	---------	--	--------	--------	---------------------	---------------------

Mengambil data dengan method get

GET localhost/PWL_POS/public/api/barangs/27 Send

Params Auth Headers (8) Body • Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body 200 OK • 847 ms • 685 B

JSON Preview Visualize

```
1 {
2   "barang_id": 27,
3   "kategori_id": 5,
4   "barang_kode": "BRG-09",
5   "barang_nama": "Setrika",
6   "image": "http://localhost/PWL_POS/public/storage/barang/
7             BG2RZLICUuaK048lnXuixfML5kheGE99W7YgZgxp.jpg",
8   "harga_beli": 200000,
9   "harga_jual": 300000,
10  "created_at": "2025-04-25T16:31:18.000000Z",
11  "updated_at": "2025-04-25T16:31:18.000000Z"
}
```

Link Commit Tugas :

https://github.com/Gilangp/Pemrograman_Web_Lanjut/commit/dfb5cbcb7e9eea46323943f527673703d550e24e