

XI RPL 6/ 16

POST

localhost:8000/api/register

Send

Save

<input checked="" type="checkbox"/>	nama_petugas	Petugas 1	
<input checked="" type="checkbox"/>	telp	08987654321	
<input checked="" type="checkbox"/>	username	petugas	
<input checked="" type="checkbox"/>	password	mamopapus	
<input checked="" type="checkbox"/>	password_confirmation	mamopapus	
<input checked="" type="checkbox"/>	level	petugas	
	Key	Value	Description

Body

Cookies

Headers (9)

Test Results

Status: 201 Created

Time: 15.15s

Size: 705B

Save Response

Pretty

Raw

Preview

Visualize

.JSON

```

1  {
2    "user": {
3      "nama_petugas": "Petugas 1",
4      "telp": "08987654321",
5      "username": "petugas",
6      "level": "petugas",
7      "updated_at": "2020-04-08 12:43:58",
8      "created_at": "2020-04-08 12:43:58",
9      "id": 3
10   }

```

POST localhost:8000/api/login

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> username	petugas	
<input checked="" type="checkbox"/> password	momopapus	
Key	Value	Description

Status: 200 OK Time: 2.06s Size: 606 B Save Response

Pretty Raw Preview Visualize JSON

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTQwMjZlODQyOTYxIiwiaWF0IjoxNjE0MDA0MDAwfQ=="
}
```

3. CONTROLLER

PetugasController.php

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\User;
6  use App\Petugas_model;
7  use Illuminate\Http\Request;
8  use Illuminate\Support\Facades\Hash;
9  use Illuminate\Support\Facades\Validator;
10 use JWTAuth;
11 use Tymon\JWTAuth\Exceptions\JWTException;
12 use Auth;
13
14 class PetugasController extends Controller
15 {
16     public function login(Request $request)
17     {
18         $credentials = $request->only('username', 'password');
19
20         try {
21             if (! $token = JWTAuth::attempt($credentials)) {
22                 return response()->json(['error' => 'invalid_credentials'], 400);
23             }
24         } catch (JWTException $e) {
25             return response()->json(['error' => 'could_not_create_token'], 500);
26         }
27
28         return response()->json(compact('token'));
29     }
30 }
```

```
31 public function register(Request $request)
32 {
33     $validator = Validator::make($request->all(), [
34         'nama_petugas' => 'required|string|max:255',
35         'telp' => 'required|string|max:255',
36         'username' => 'required|string|max:255|unique:petugas',
37         'password' => 'required|string|min:6|confirmed',
38         'level' => 'required|string|max:255',
39
40     ]);
41
42     if($validator->fails()){
43         return response()->json($validator->errors()->toJson(), 400);
44     }
45
46     $user = User::create([
47         'nama_petugas' => $request->get('nama_petugas'),
48         'telp' => $request->get('telp'),
49         'username' => $request->get('username'),
50         'password' => Hash::make($request->get('password')),
51         'level' => $request->get('level'),
52
53     ]);
54
55     $token = JWTAuth::fromUser($user);
56
57     return response()->json(compact('user', 'token'), 201);
58 }
```

PetugasController.php

```
60 public function getAuthenticatedUser()
61 {
62     try {
63
64         if (! $user = JWTAuth::parseToken()->authenticate()) {
65             return response()->json(['user_not_found'], 404);
66         }
67
68     } catch (Tymon\JWTAuth\Exceptions\TokenExpiredException $e) {
69
70         return response()->json(['token_expired'], $e->getStatusCode());
71
72     } catch (Tymon\JWTAuth\Exceptions\TokenInvalidException $e) {
73
74         return response()->json(['token_invalid'], $e->getStatusCode());
75
76     } catch (Tymon\JWTAuth\Exceptions\JWTException $e) {
77
78         return response()->json(['token_absent'], $e->getStatusCode());
79
80     }
81
82     return response()->json(compact('user'));
83 }
```

```

PetugasController.php

85 public function tampil(){
86     if(Auth::user()->level=="admin"){
87         $datas = Petugas_model::get();
88         $count = $datas->count();
89         $film = array();
90         $status = 1;
91         foreach ($datas as $dt_sw){
92             $film[] = array(
93                 'id' => $dt_sw->id,
94                 'nama_petugas' => $dt_sw->nama_petugas,
95                 'telp' => $dt_sw->telp,
96                 'username' => $dt_sw->username,
97                 'password' => $dt_sw->password,
98                 'level' => $dt_sw->level,
99                 'created_at' => $dt_sw->created_at,
100                 'updated_at' => $dt_sw->updated_at,
101             );
102         }
103         return Response()->json(compact('count','film'));
104     } else {
105         return Response()->json(['status'=> 'Gabisa, kamu bukan admin :(']);
106     }
107 }
108
109 public function update($id,Request $req)
110 {
111     if(Auth::user()->level=="admin"){
112         $validator=Validator::make($req->all(),
113         [
114             'nama_petugas' => 'required',
115             'telp' => 'required',

```

4. MODEL

User.php

```
1 <?php
2
3 namespace App;
4
5 use Illuminate\Notifications\Notifiable;
6 use Illuminate\Contracts\Auth\MustVerifyEmail;
7 use Illuminate\Foundation\Auth\User as Authenticatable;
8 use Tymon\JWTAuth\Contracts\JWTSubject;
9
10 class User extends Authenticatable implements JWTSubject
11 {
12     protected $table = "petugas";
13     // public $timestamps = false;
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array
19      */
20     protected $fillable = [
21         'nama_petugas', 'telp', 'username', 'password', 'level'
22     ];
23
24     /**
25      * The attributes that should be hidden for arrays.
26      *
27      * @var array
28      */
29     protected $hidden = [
30         'password', 'remember_token',
31     ];
32
33     public function getJWTIdentifier()
34     {
35         return $this->getKey();
36     }
37     public function getJWTCustomClaims()
38     {
39         return [];
40     }
41 }
```

Petugas_model.php

```
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class Petugas_model extends Model
8  {
9      protected $table="petugas";
10     protected $primaryKey="id";
11     protected $fillable = ['nama_petugas','telp','username','password','level'];
12 }
```