

## Register

POST laundry/api/login

POST rental\_mobil/api/register

+

...

No Environment

POST rental\_mobil/api/register

Send

Save

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	nama_petugas	kamu			
<input checked="" type="checkbox"/>	alamat_petugas	Malang			
<input checked="" type="checkbox"/>	username	Text abc			X
<input checked="" type="checkbox"/>	password	abc123			
<input checked="" type="checkbox"/>	password_confirmation	abc123			
<input checked="" type="checkbox"/>	level	petugas			
	Key	Value	Description		

Body Cookies Headers (10) Test Results

Status: 201 Created Time: 12.60s Size: 840 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "user": {
3     "nama_petugas": "kamu",
4     "alamat_petugas": "Malang",
5     "username": "abc",
6     "password": "$2y$10$RQuZ9QZYK4DaBdMgE.MjNuLHLBSN9knuAv4AXVzWhUmTt3crkg0Q.",
7     "level": "petugas",
8     "id": 2
9   },
10  "token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOlwvXC9yZW50YWxkbW9iaWxkL2FwaVwvcmVnaXN0ZXIiLCJpYXQiOiJle1ODY1Mjc2NzAsImV4cCI6MTU4NjUzMTI3MSwibmJmIjoxNTg2NTI3NjcxLCJqdGkiOiJxOFExcFNSG1STko5WEt5Iiwic3ViIjoyLCJwcnYiOiI4N2UwYWYxZWY5ZmQxNTgxMmZkZW5NzE1M2ExNGUwYjA0NzU0NmFhIn0.CErK40-R6IHy48dYtYFoeNEHdMY0RmFYITd4wX-Cpys"
```

## Login

POST laundry/api/login

POST rental\_mobil/api/register

+

...

No Environment

Untitled Request

Comments 0

POST rental\_mobil/api/login

Send

Save

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Cookies Code

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

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	username	abc			
<input checked="" type="checkbox"/>	password	abc123			
	Key	Value	Description		

Body Cookies Headers (10) Test Results

Status: 200 OK Time: 806ms Size: 658 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOlwvXC9yZW50YWxkbW9iaWxkL2FwaVwvcmVnaXN0ZXIiLCJpYXQiOiJle1ODY1Mjc2NzUsImV4cCI6MTU4NjUzMTI3MSwibmJmIjoxNTg2NTI3NjcxLCJqdGkiOiI2akcyUTZFcENjN0EwMG5FIiwic3ViIjoyLCJwcnYiOiI4N2UwYWYxZWY5ZmQxNTgxMmZkZW5NzE1M2ExNGUwYjA0NzU0NmFhIn0.-bfeQx9RV-aUHRicmpEXCKdhXw2ec04yT8DFDaugY1E"
3 }
```

## PetugasController

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\User;
6  use Auth;
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
13 class PetugasController extends Controller
14 {
15     public function login(Request $request)
16     {
17         $credentials = $request->only('username', 'password');
18
19         try {
20             if (!$token = JWTAuth::attempt($credentials)) {
21                 return response()->json(['error' => 'invalid_credentials'], 400);
22             }
23         } catch (JWTException $e) {
24             return response()->json(['error' => 'could_not_create_token'], 500);
25         }
26
27         return response()->json(compact('token'));
28     }
29
30     public function register(Request $request)
31     {
32         $validator = Validator::make($request->all(), [
33             'nama_petugas' => 'required|string|max:255',
34             'alamat_petugas' => 'required|string|max:255',
35             'username' => 'required|string|max:255',
36             'password' => 'required|string|min:3|confirmed',
37             'level' => 'required|string|max:255',
38         ]);
39
40         if($validator->fails()){
41             return response()->json($validator->errors()->toJson(), 400);
42         }
43
44         $user = User::create([
45             'nama_petugas' => $request->get('nama_petugas'),
46             'alamat_petugas' => $request->get('alamat_petugas'),
47             'username' => $request->get('username'),
48             'password' => Hash::make($request->get('password')),
49             'level' => $request->get('level'),
50
51         ]);
52
53         $token = JWTAuth::fromUser($user);
54
55         return response()->json(compact('user', 'token'));
56     }
57
58     public function getAuthenticatedUser()
59     {
60         try {
61             if (!$user = JWTAuth::parseToken()->authenticate()) {
62                 return response()->json(['user_not_found'], 404);
63             }
64         } catch (Tymon\JWTAuth\Exceptions\TokenExpiredException $e) {
65             return response()->json(['token_expired'], $e->getStatusCode());
66         } catch (Tymon\JWTAuth\Exceptions\TokenInvalidException $e) {
67             return response()->json(['token_invalid'], $e->getStatusCode());
68         } catch (Tymon\JWTAuth\Exceptions\JWTException $e) {
69             return response()->json(['token_absent'], $e->getStatusCode());
70         }
71
72         return response()->json(compact('user'));
73     }
74 }
```

## Model

```
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     use Notifiable;
13
14     protected $table = "petugas" ;
15     protected $primaryKey = "id" ;
16
17     /**
18      * The attributes that are mass assignable.
19      *
20      * @var array
21      */
22     protected $fillable = [
23         'nama_petugas', 'alamat_petugas', 'username', 'password', 'level',
24     ];
25     public function getJWTIdentifier()
26     {
27         return $this->getKey();
28     }
29     public function getJWTCustomClaims()
30     {
31         return [];
32     }
33
34     public $timestamps = false ;
35
36 }
37
```

## Api.php

```
1  <?php
2
3  use Illuminate\Http\Request;
4
5  /*
6  |-----
7  | API Routes
8  |-----
9  |
10 | Here is where you can register API routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | is assigned the "api" middleware group. Enjoy building your API!
13 |
14 */
15
16 //Route::middleware('auth:api')->get('/user', function (Request $request) {
17 //    return $request->user();
18 //});
19
20 Route::post('register', 'PetugasController@register');
21 Route::post('login', 'PetugasController@login');
22
```