Nama : Anisha Trie A

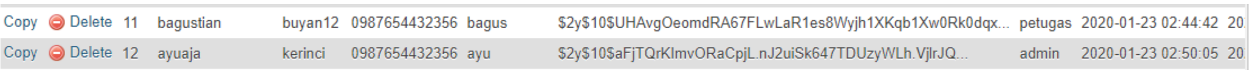
Kelas : XI RPL 6 (Laravel)

No. Absen : 06

Tugas 4

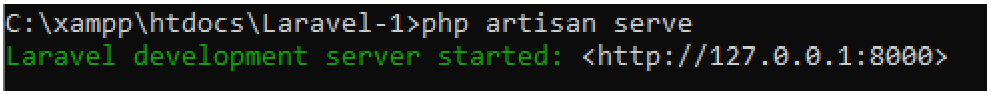
Login & Register

1. Database

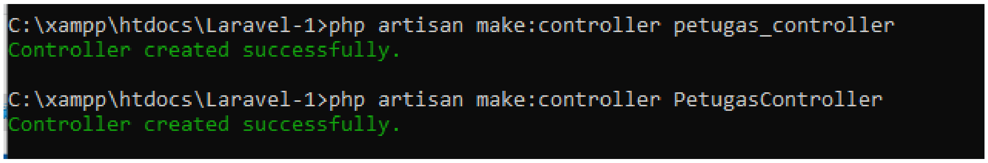


Copy	Delete	11	bagustian	buyan12	0987654432356	bagus	$2y$10$UHAvgOeomdRA67FLwLaR1es8Wyjh1XKqb1Xw0Rk0dqx...	petugas	2020-01-23 02:44:42
Copy	Delete	12	ayuaja	kerinci	0987654432356	ayu	$2y$10$aFjTQrKlmvORaCpjL.nJ2uiSk647TDUzyWLh.VjlrJQ...	admin	2020-01-23 02:50:05

2. Cmd

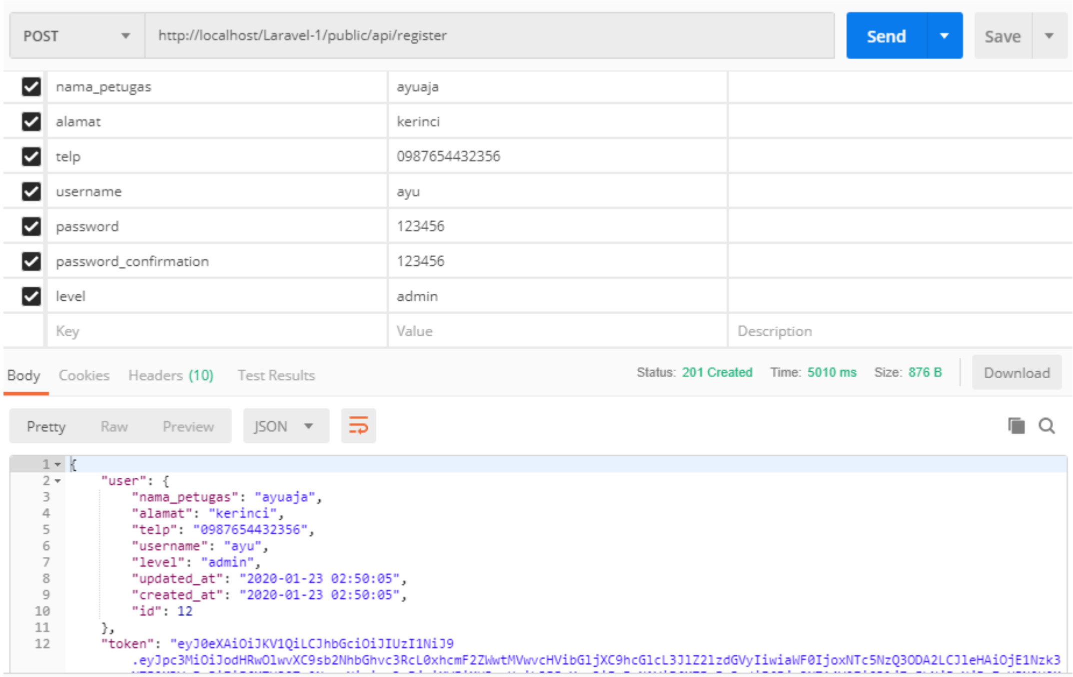


```
C:\xampp\htdocs\Laravel-1>php artisan serve
Laravel development server started: <http://127.0.0.1:8000>
```



```
C:\xampp\htdocs\Laravel-1>php artisan make:controller petugas_controller
Controller created successfully.

C:\xampp\htdocs\Laravel-1>php artisan make:controller PetugasController
Controller created successfully.
```

3. Register
   - Admin



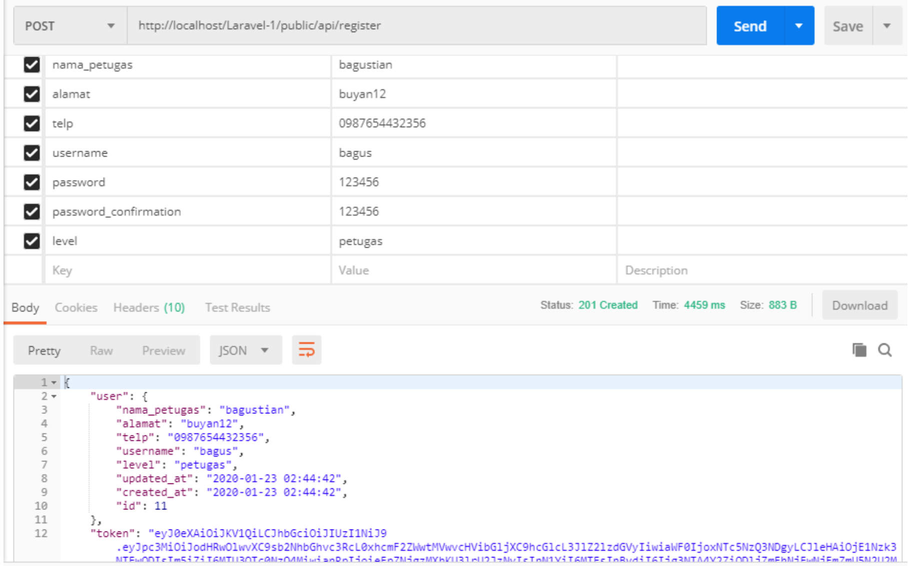
http://localhost/Laravel-1/public/api/register

	Key	Value	Description
☑	nama_petugas	ayuaja	
☑	alamat	kerinci	
☑	telp	0987654432356	
☑	username	ayu	
☑	password	123456	
☑	password_confirmation	123456	
☑	level	admin	

Status: 201 Created Time: 5010 ms Size: 876 B

```
{
    "user": {
        "nama_petugas": "ayuaja",
        "alamat": "kerinci",
        "telp": "0987654432356",
        "username": "ayu",
        "level": "admin",
        "updated_at": "2020-01-23 02:50:05",
        "created_at": "2020-01-23 02:50:05",
        "id": 12
    },
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL0xhcmF2ZWwtMVwvcHVibGljXC9hcGlcL3JlZ2lzdGVyIiwiaWF0IjoxNTc5NzQ3ODA2LCJleHAiOjE1Nzk3
```

- Petugas



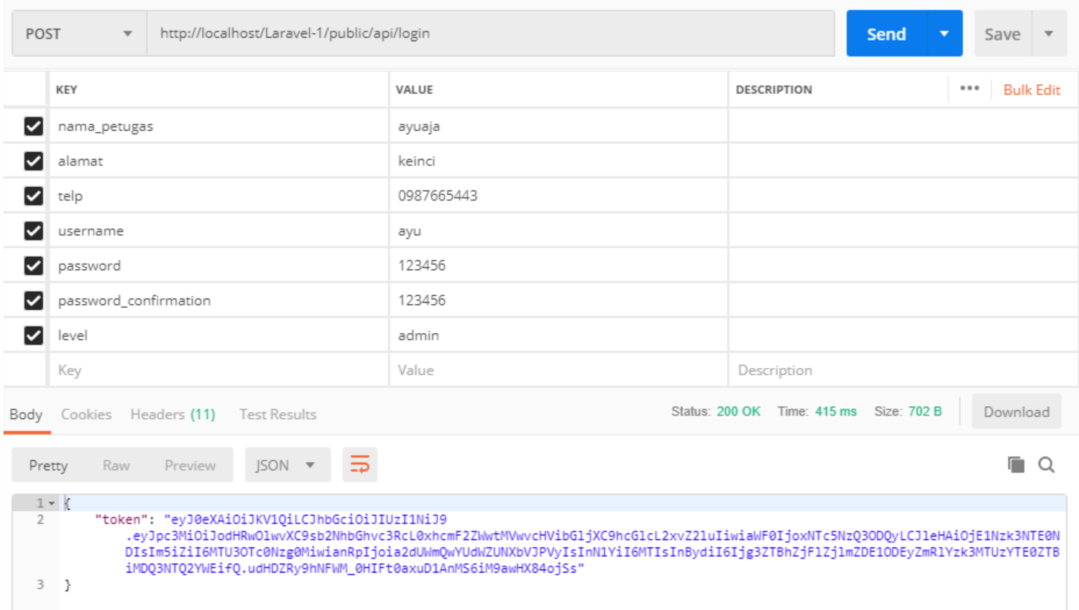
POST http://localhost/Laravel-1/public/api/register

☑	nama_petugas	bagustian	
☑	alamat	buyan12	
☑	telp	0987654432356	
☑	username	bagus	
☑	password	123456	
☑	password_confirmation	123456	
☑	level	petugas	
	Key	Value	Description

Status: 201 Created Time: 4459 ms Size: 883 B

```
{
    "user": {
        "nama_petugas": "bagustian",
        "alamat": "buyan12",
        "telp": "0987654432356",
        "username": "bagus",
        "level": "petugas",
        "updated_at": "2020-01-23 02:44:42",
        "created_at": "2020-01-23 02:44:42",
        "id": 11
    },
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL0xhcmF2ZWwtMVwvcHVibGljXC9hcGlcL3J1Z2lzdGVyIiwiaWF0IjoxNTc5NzQ3NDgyLCJleHAiOjE1Nzk3
```

4. Login
   - Admin



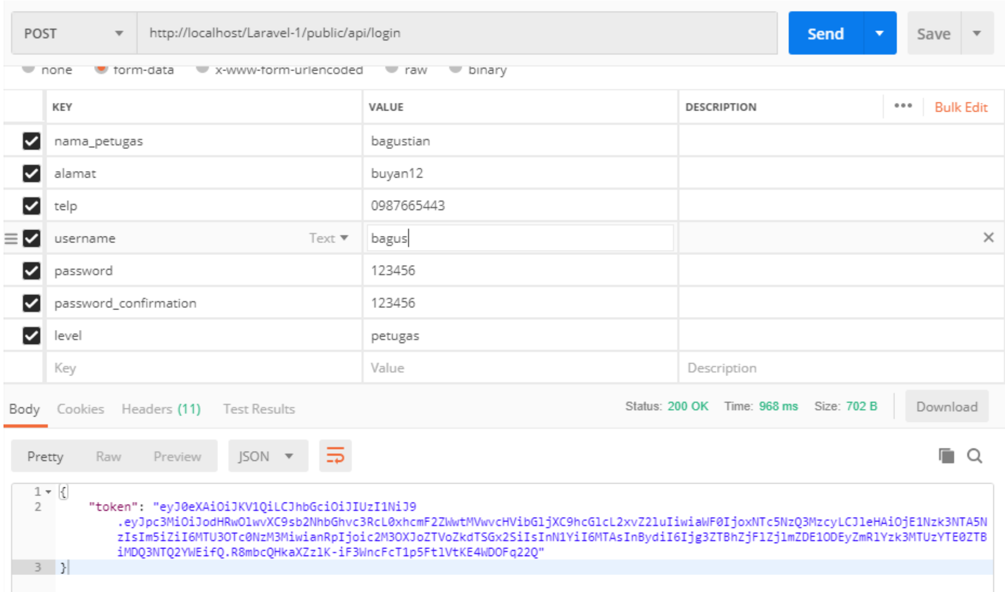
POST http://localhost/Laravel-1/public/api/login

	KEY	VALUE	DESCRIPTION
☑	nama_petugas	ayuaja	
☑	alamat	keinci	
☑	telp	0987665443	
☑	username	ayu	
☑	password	123456	
☑	password_confirmation	123456	
☑	level	admin	
	Key	Value	Description

Status: 200 OK Time: 415 ms Size: 702 B

```
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL0xhcmF2ZWwtMVwvcHVibGljXC9hcGlcL2xvZ2luIiwiaWF0IjoxNTc5NzQ3ODQyLCJleHAiOjE1Nzk3NTE0N
        DIsIm5iZiI6MTU3OTc0Nzg0MiwianRpIjoia2dUWmQwYUdWZUNXbVJPVyIsInN1YiI6MTIsInBydiI6Ijg3ZTBhZjF1ZjlmZDE1ODEyZmR1Yzk3MTUzYTE0ZTB
        iMDQ3NTQ2YWEifQ.udHDZRy9hNFWM_0HIFt0axuD1AnMS6iM9awHX84ojSs"
}
```

   - Petugas


POST
Send
Save
○ none ● form-data ○ x-www-form-urlencoded ○ raw ○ binary
Bulk Edit
Body Cookies Headers (11) Test Results
Download
Pretty Raw Preview JSON

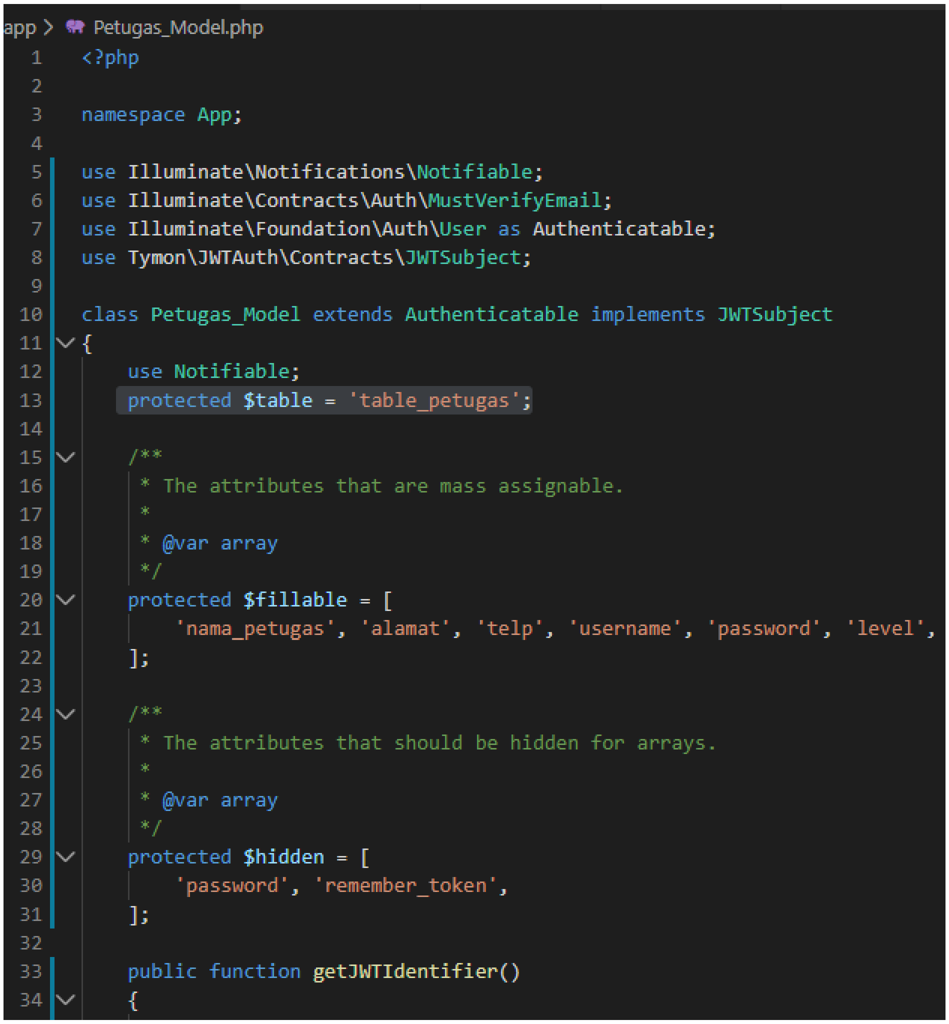
http://localhost/Laravel-1/public/api/login

	KEY	VALUE	DESCRIPTION
☑	nama_petugas	bagustian	
☑	alamat	buyan12	
☑	telp	0987665443	
☑	username Text	bagus	
☑	password	123456	
☑	password_confirmation	123456	
☑	level	petugas	
	Key	Value	Description

Status: 200 OK Time: 968 ms Size: 702 B

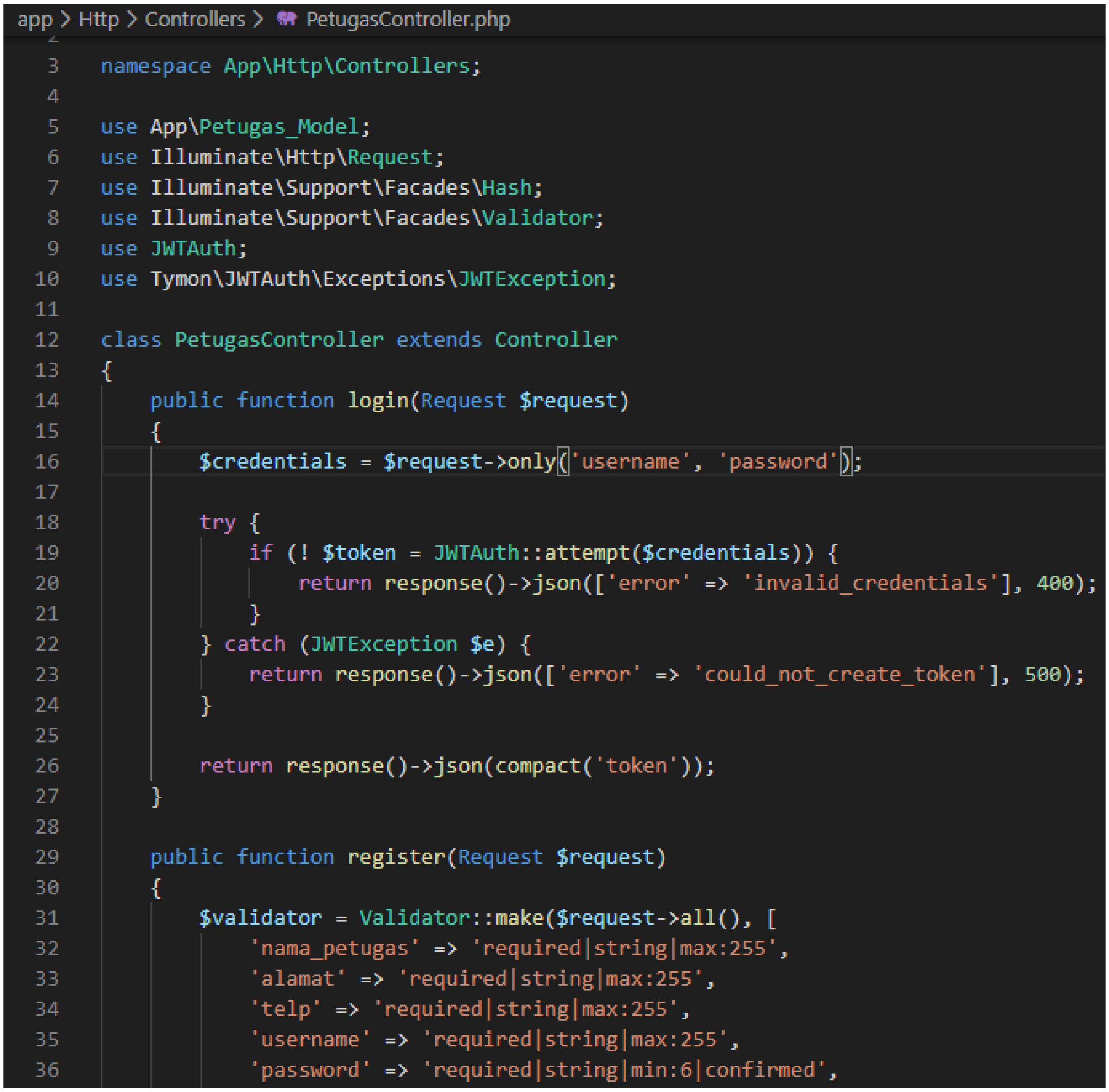
```
{
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJpc3MiOiJodHRwOlwvXC9sb2NhbGhvc3RcL0xhcmF2ZWwtMVwvcHVibGljXC9hcGlcL2xvZ2luIiwiaWF0IjoxNTc5NzQ3MzcyLCJleHAiOjE1Nzk3NTA5N
        zIsIm5iZiI6MTU3OTc0NzM3MiwianRpIjoic2M3OXJoZTVoZkdTSGx2SiIsInN1YiI6MTAsInBydiI6Ijg3ZTBhZjF1ZjlmZDE1ODEyZmR1Yzk3MTUzYTE0ZTB
        iMDQ3NTQ2YWEifQ.R8mbcQHkaXZz1K-iF3WncFcT1p5Ft1VtKE4WDOFq22Q"
}
```

5. Coding
    - Model petugas

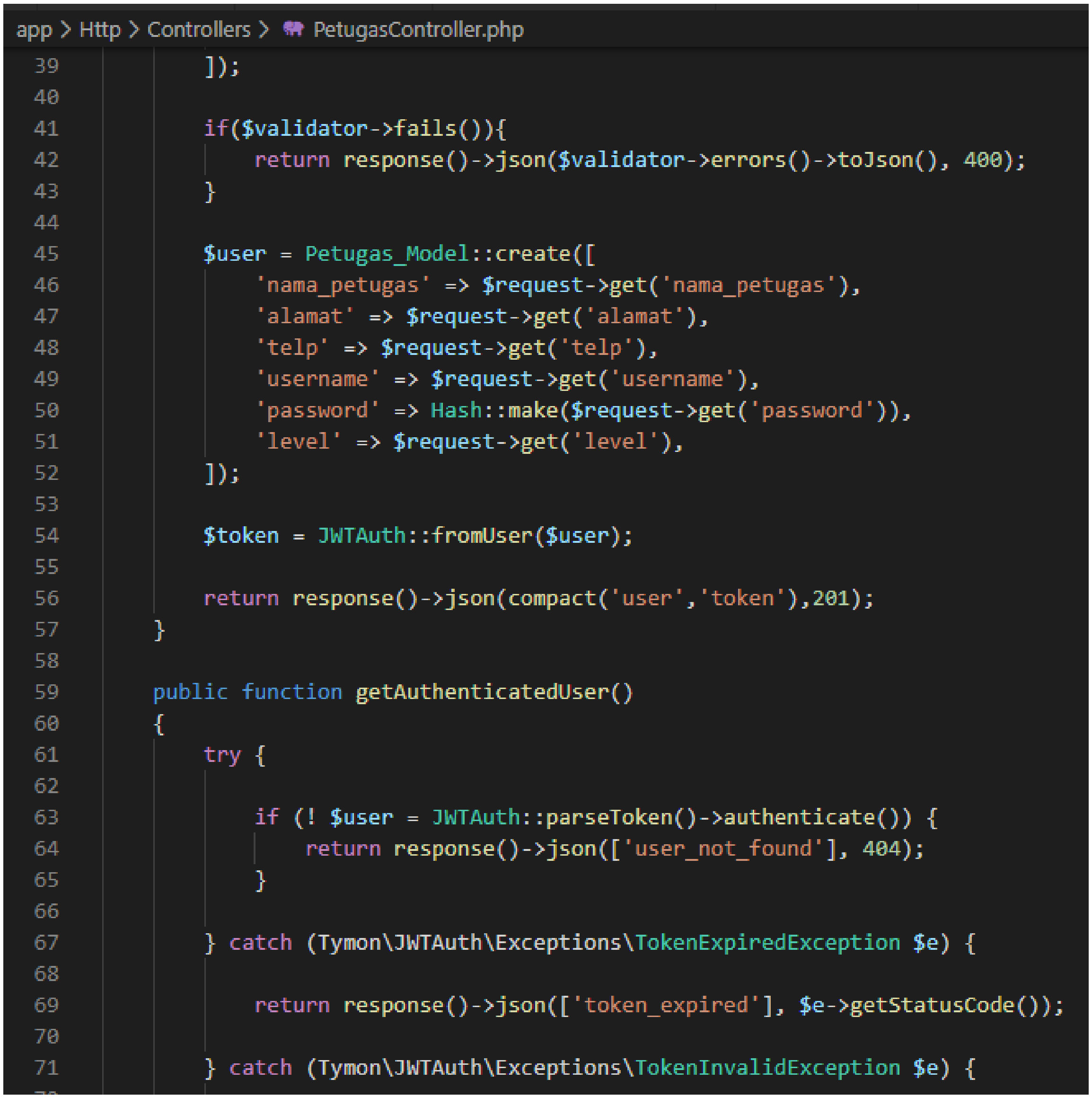


```php
<?php

namespace App;

use Illuminate\Notifications\Notifiable;
use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Tymon\JWTAuth\Contracts\JWTSubject;

class Petugas_Model extends Authenticatable implements JWTSubject
{
    use Notifiable;
    protected $table = 'table_petugas';

    /**
     * The attributes that are mass assignable.
     *
     * @var array
     */
    protected $fillable = [
        'nama_petugas', 'alamat', 'telp', 'username', 'password', 'level',
    ];

    /**
     * The attributes that should be hidden for arrays.
     *
     * @var array
     */
    protected $hidden = [
        'password', 'remember_token',
    ];

    public function getJWTIdentifier()
    {
```

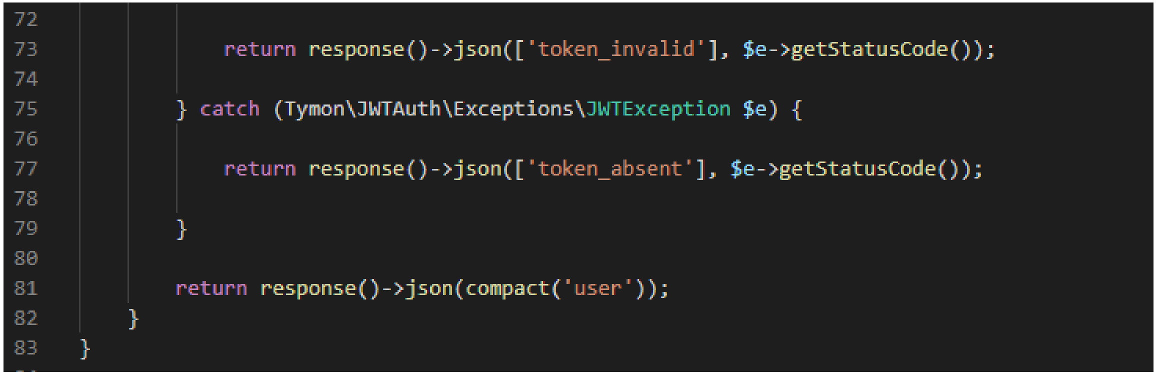
- Controller Petugas



```php
namespace App\Http\Controllers;

use App\Petugas_Model;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
use JWTAuth;
use Tymon\JWTAuth\Exceptions\JWTException;

class PetugasController extends Controller
{
    public function login(Request $request)
    {
        $credentials = $request->only('username', 'password');

        try {
            if (! $token = JWTAuth::attempt($credentials)) {
                return response()->json(['error' => 'invalid_credentials'], 400);
            }
        } catch (JWTException $e) {
            return response()->json(['error' => 'could_not_create_token'], 500);
        }

        return response()->json(compact('token'));
    }

    public function register(Request $request)
    {
        $validator = Validator::make($request->all(), [
            'nama_petugas' => 'required|string|max:255',
            'alamat' => 'required|string|max:255',
            'telp' => 'required|string|max:255',
            'username' => 'required|string|max:255',
            'password' => 'required|string|min:6|confirmed',
```

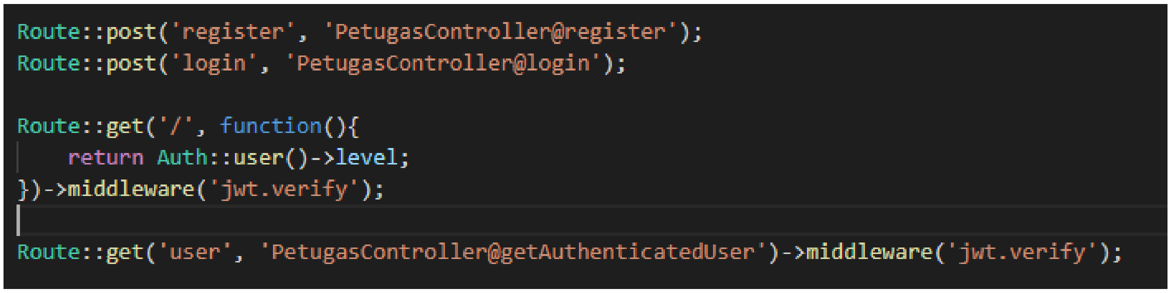


```php
        ]);

        if($validator->fails()){
            return response()->json($validator->errors()->toJson(), 400);
        }

        $user = Petugas_Model::create([
            'nama_petugas' => $request->get('nama_petugas'),
            'alamat' => $request->get('alamat'),
            'telp' => $request->get('telp'),
            'username' => $request->get('username'),
            'password' => Hash::make($request->get('password')),
            'level' => $request->get('level'),
        ]);

        $token = JWTAuth::fromUser($user);

        return response()->json(compact('user','token'),201);
    }

    public function getAuthenticatedUser()
    {
        try {

            if (! $user = JWTAuth::parseToken()->authenticate()) {
                return response()->json(['user_not_found'], 404);
            }

        } catch (Tymon\JWTAuth\Exceptions\TokenExpiredException $e) {

            return response()->json(['token_expired'], $e->getStatusCode());

        } catch (Tymon\JWTAuth\Exceptions\TokenInvalidException $e) {
```



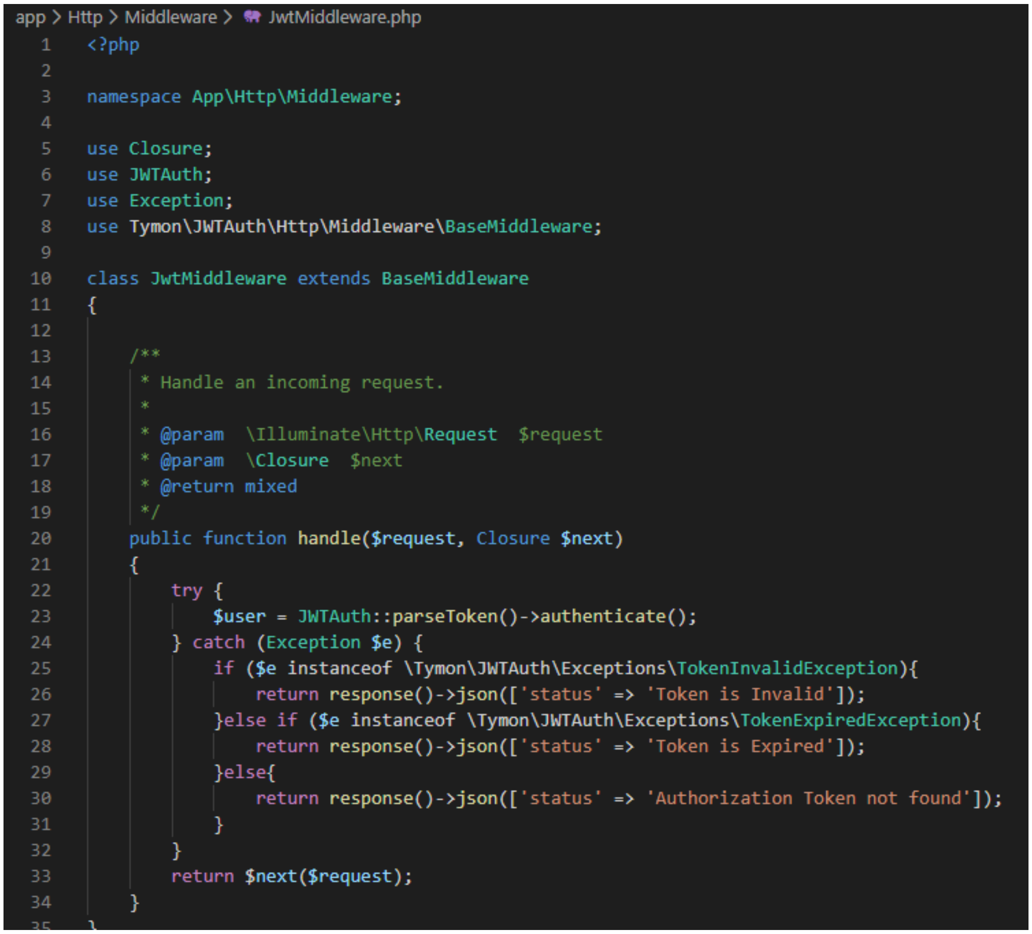
```php

            return response()->json(['token_invalid'], $e->getStatusCode());

        } catch (Tymon\JWTAuth\Exceptions\JWTException $e) {

            return response()->json(['token_absent'], $e->getStatusCode());

        }

        return response()->json(compact('user'));
    }
}
```

- Route



```php
Route::post('register', 'PetugasController@register');
Route::post('login', 'PetugasController@login');

Route::get('/', function(){
    return Auth::user()->level;
})->middleware('jwt.verify');

Route::get('user', 'PetugasController@getAuthenticatedUser')->middleware('jwt.verify');
```

- Jwt Midleware



```php
<?php

namespace App\Http\Middleware;

use Closure;
use JWTAuth;
use Exception;
use Tymon\JWTAuth\Http\Middleware\BaseMiddleware;

class JwtMiddleware extends BaseMiddleware
{

    /**
     * Handle an incoming request.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  \Closure  $next
     * @return mixed
     */
    public function handle($request, Closure $next)
    {
        try {
            $user = JWTAuth::parseToken()->authenticate();
        } catch (Exception $e) {
            if ($e instanceof \Tymon\JWTAuth\Exceptions\TokenInvalidException){
                return response()->json(['status' => 'Token is Invalid']);
            }else if ($e instanceof \Tymon\JWTAuth\Exceptions\TokenExpiredException){
                return response()->json(['status' => 'Token is Expired']);
            }else{
                return response()->json(['status' => 'Authorization Token not found']);
            }
        }
        return $next($request);
    }
}
```