Nama : **Ereen Lourenza Natalia Mamahi**

Kelas / no : **SIB3C / 09**

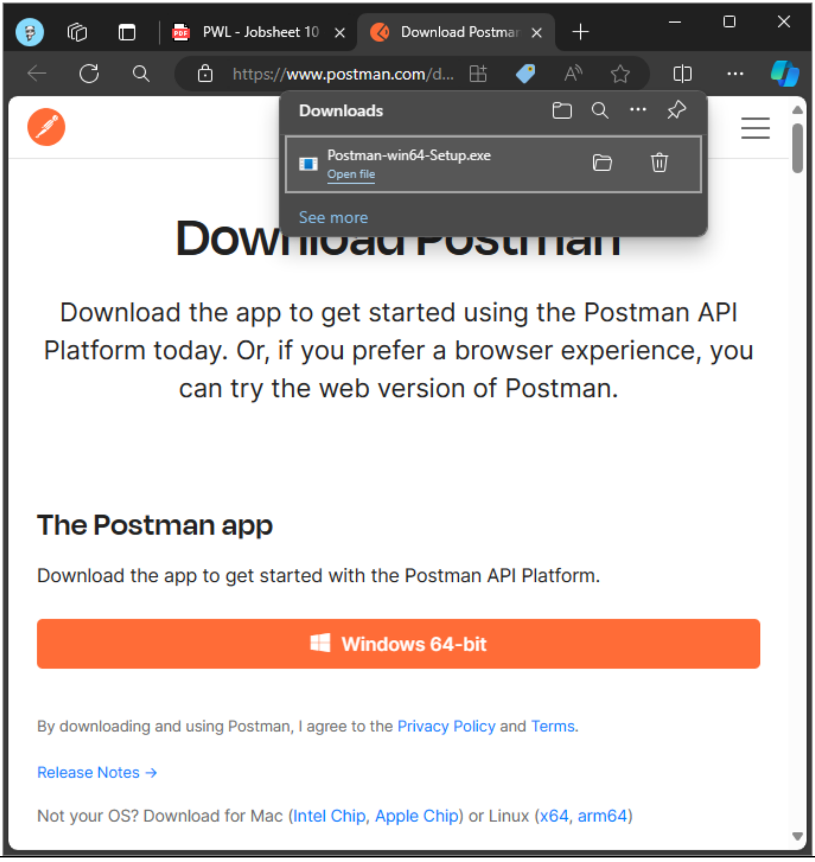
NIM : **2141762044**

Jobsheet 10

Restful API

Praktikum 1 – Membuat RESTful API Registrasi

1 Mendownload postman



Downloads

Postman-win64-Setup.exe

Open file

See more

Download Postman

Download the app to get started using the Postman API Platform today. Or, if you prefer a browser experience, you can try the web version of Postman.

The Postman app

Download the app to get started with the Postman API Platform.

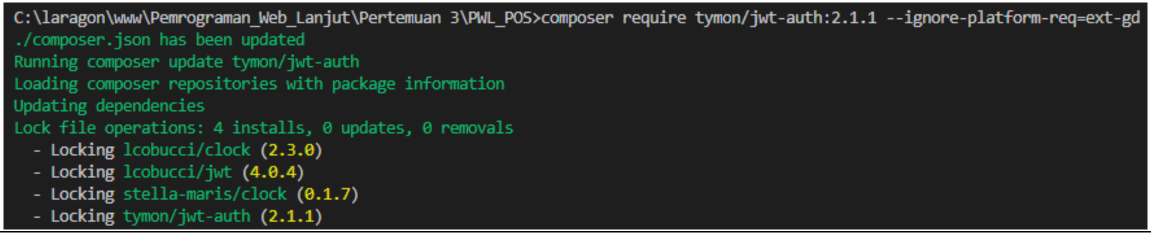
Windows 64-bit

By downloading and using Postman, I agree to the Privacy Policy and Terms.

Release Notes →

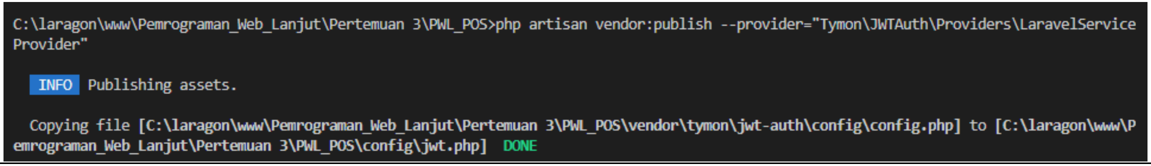
Not your OS? Download for Mac (Intel Chip, Apple Chip) or Linux (x64, arm64)

2 Menginstall JWT



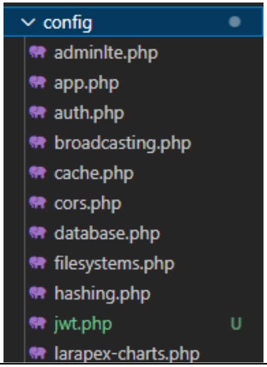
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>composer require tymon/jwt-auth:2.1.1 --ignore-platform-req=ext-gd
./composer.json has been updated
Running composer update tymon/jwt-auth
Loading composer repositories with package information
Updating dependencies
Lock file operations: 4 installs, 0 updates, 0 removals
  - Locking lcobucci/clock (2.3.0)
  - Locking lcobucci/jwt (4.0.4)
  - Locking stella-maris/clock (0.1.7)
  - Locking tymon/jwt-auth (2.1.1)
```

3 Konfigurasi file



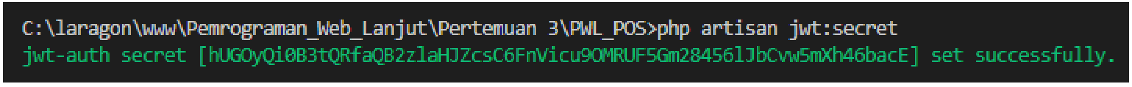
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan vendor:publish --provider="Tymon\JWTAuth\Providers\LaravelService
Provider"

  INFO  Publishing assets.

  Copying file [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\vendor\tymon\jwt-auth\config\config.php] to [C:\laragon\www\P
emrograman_Web_Lanjut\Pertemuan 3\PWL_POS\config\jwt.php]  DONE
```

4 Mendapatkan file jwt.php



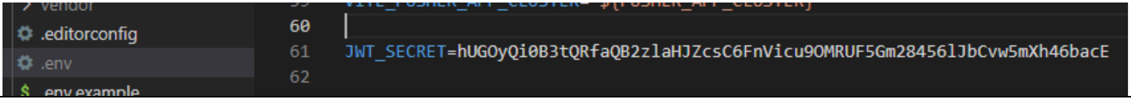
```
config
  adminlte.php
  app.php
  auth.php
  broadcasting.php
  cache.php
  cors.php
  database.php
  filesystems.php
  hashing.php
  jwt.php              U
  larapex-charts.php
```

5 Membuat secret key JWT



```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan jwt:secret
jwt-auth secret [hUGOyQi0B3tQRfaQB2zlaHJZcsC6FnVicu9OMRUF5Gm28456lJbCvw5mXh46bacE] set successfully.
```

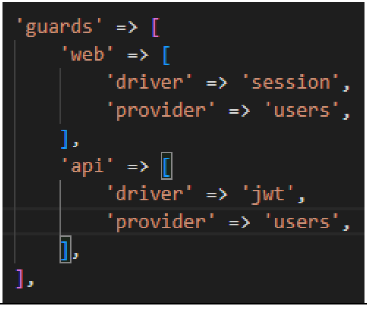
File .env ditambahkan baris berisi nilai key JWT_SECRET



```
.editorconfig
.env
env.example
```

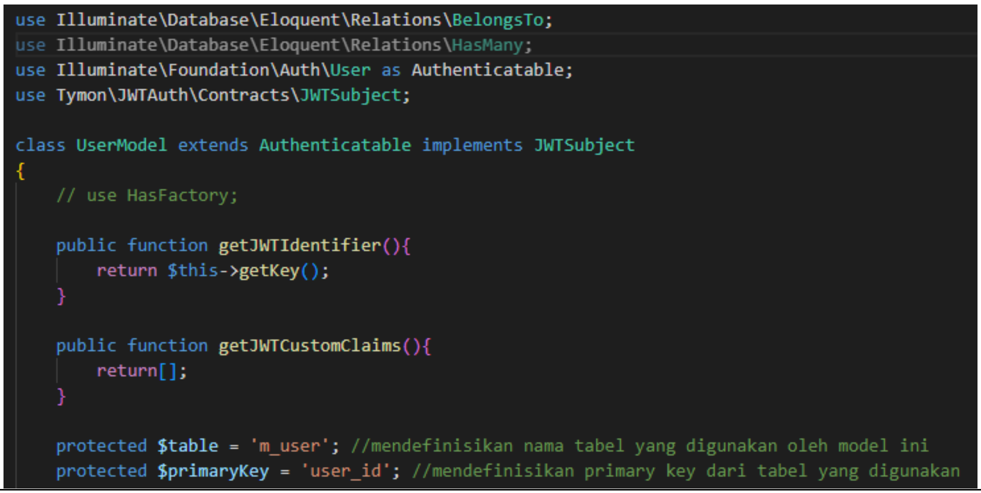
```

JWT_SECRET=hUGOyQi0B3tQRfaQB2zlaHJZcsC6FnVicu9OMRUF5Gm28456lJbCvw5mXh46bacE

```

6 Melakukan konfigurasi guard API di config/auth.php bagian 'guards'



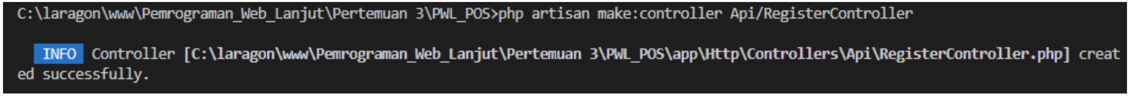
```php
'guards' => [
    'web' => [
        'driver' => 'session',
        'provider' => 'users',
    ],
    'api' => [
        'driver' => 'jwt',
        'provider' => 'users',
    ],
],
```

7 Menambahkan kode di model UserModel

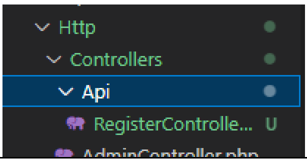


```php
use Illuminate\Database\Eloquent\Relations\BelongsTo;
use Illuminate\Database\Eloquent\Relations\HasMany;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Tymon\JWTAuth\Contracts\JWTSubject;

class UserModel extends Authenticatable implements JWTSubject
{
    // use HasFactory;

    public function getJWTIdentifier(){
        return $this->getKey();
    }

    public function getJWTCustomClaims(){
        return[];
    }

    protected $table = 'm_user'; //mendefinisikan nama tabel yang digunakan oleh model ini
    protected $primaryKey = 'user_id'; //mendefinisikan primary key dari tabel yang digunakan
```

8 Membuat controller register di folder API

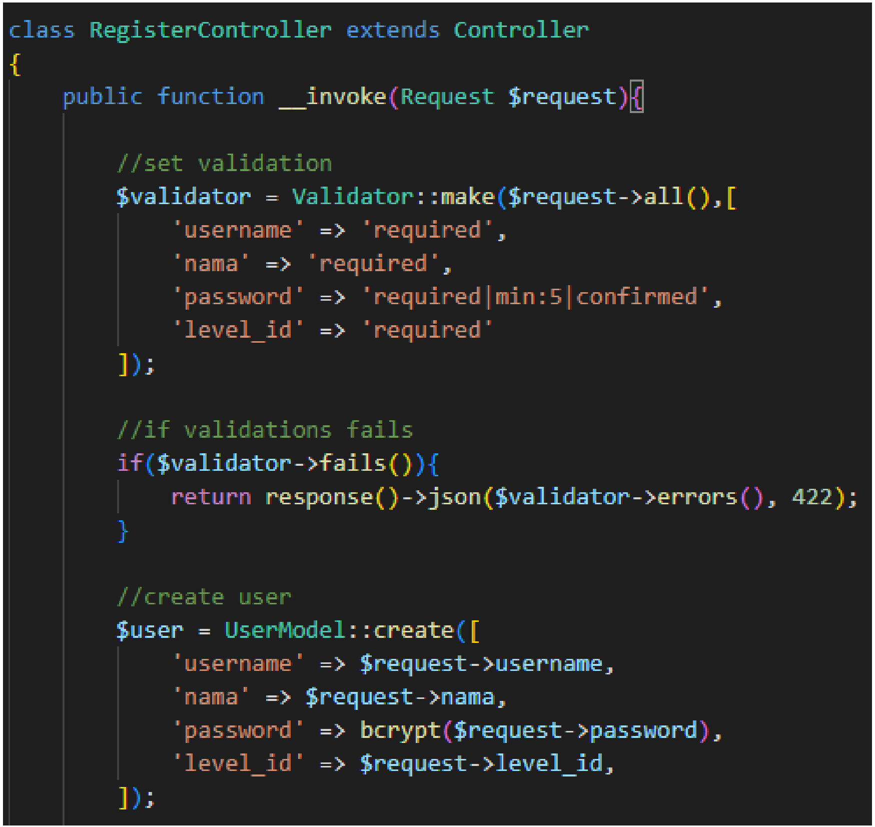


```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/RegisterController

  INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\RegisterController.php] creat
ed successfully.
```

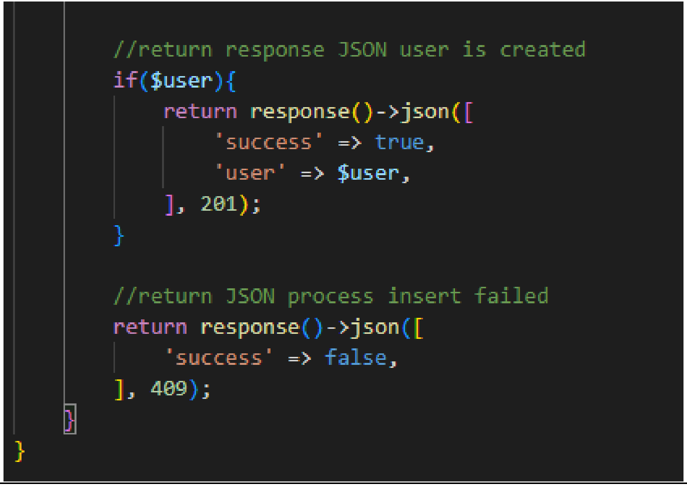


```
Http
  Controllers
    Api
      RegisterControlle...  U
    AdminController.php
```

9 Mengubah kode RegisterController

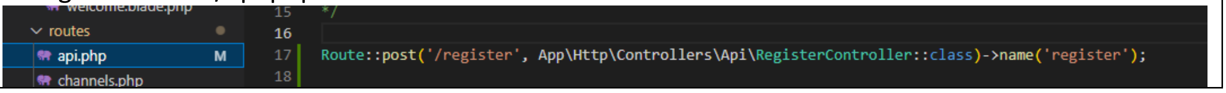


```php
class RegisterController extends Controller
{
    public function __invoke(Request $request){

        //set validation
        $validator = Validator::make($request->all(),[
            'username' => 'required',
            'nama' => 'required',
            'password' => 'required|min:5|confirmed',
            'level_id' => 'required'
        ]);

        //if validations fails
        if($validator->fails()){
            return response()->json($validator->errors(), 422);
        }

        //create user
        $user = UserModel::create([
            'username' => $request->username,
            'nama' => $request->nama,
            'password' => bcrypt($request->password),
            'level_id' => $request->level_id,
        ]);
```



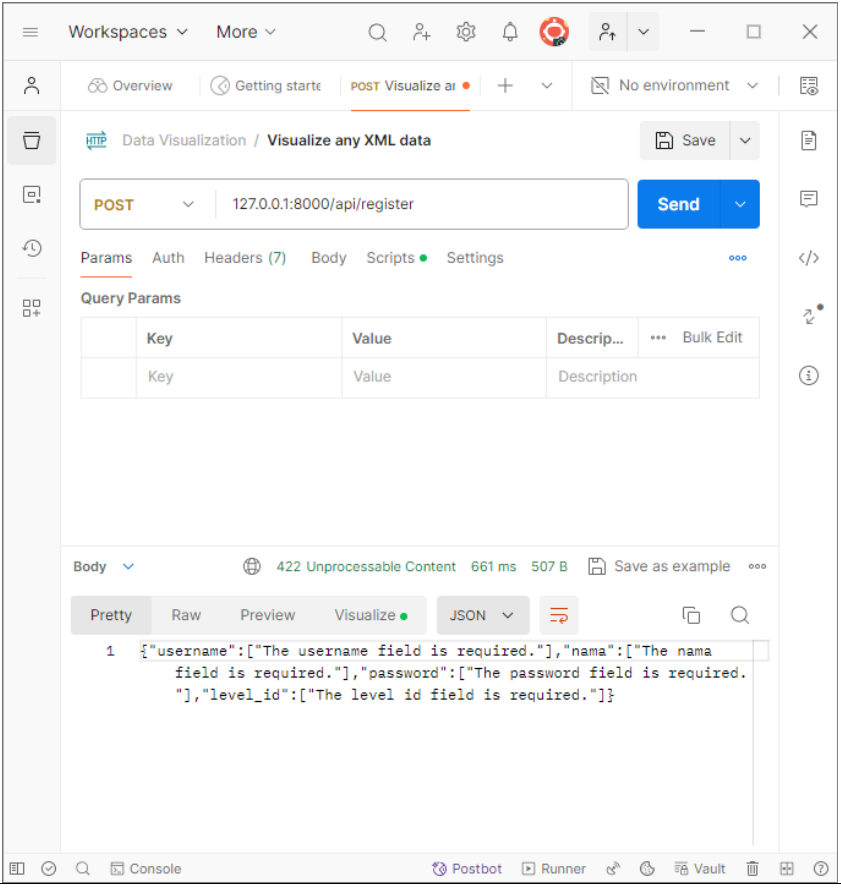
```php
        //return response JSON user is created
        if($user){
            return response()->json([
                'success' => true,
                'user' => $user,
            ], 201);
        }

        //return JSON process insert failed
        return response()->json([
            'success' => false,
        ], 409);
    }
}
```

10 Mengubah routes/api.php



```php
*/

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

11 Mengisi URL method POST pada aplikasi Postman (127.0.0.1:8000/api/register)

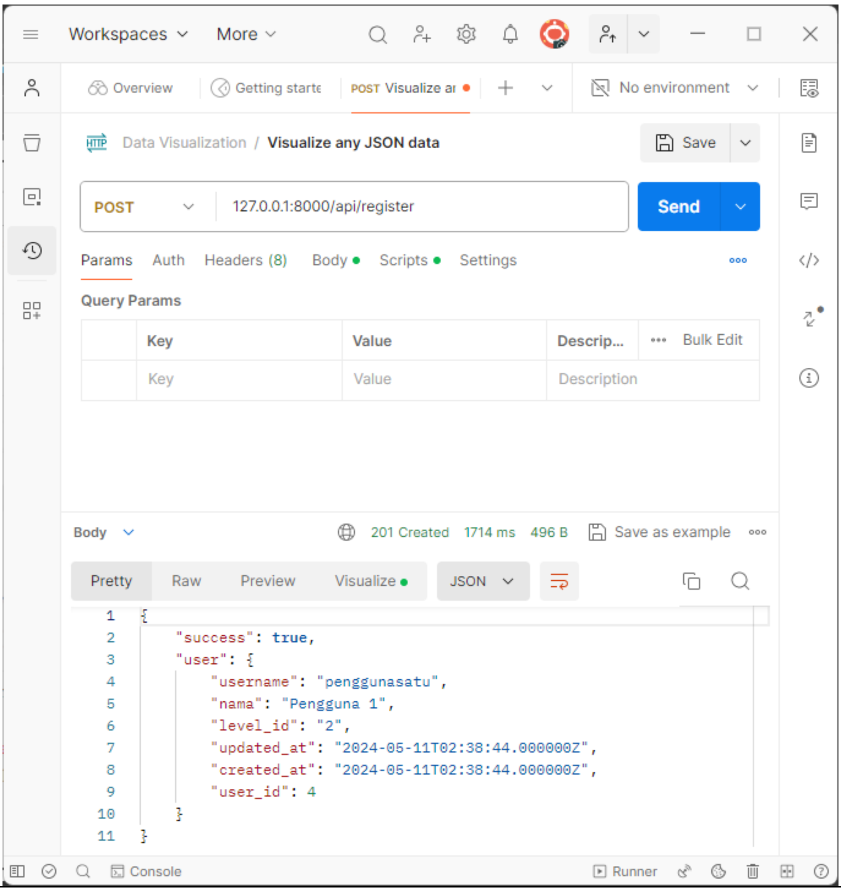


POST 127.0.0.1:8000/api/register

422 Unprocessable Content 661 ms 507 B

```
{"username":["The username field is required."],"nama":["The nama field is required."],"password":["The password field is required."],"level_id":["The level id field is required."]}
```

12 Mencoba memasukkan data



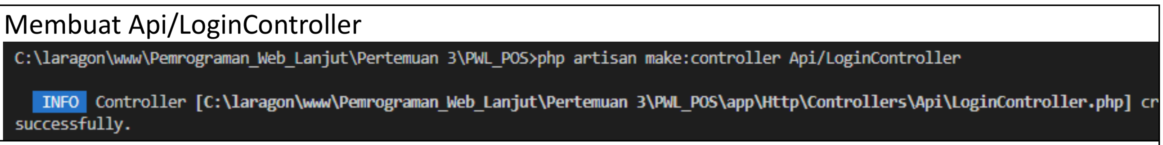
POST 127.0.0.1:8000/api/register

201 Created 1714 ms 496 B

```
{
    "success": true,
    "user": {
        "username": "penggunasatu",
        "nama": "Pengguna 1",
        "level_id": "2",
        "updated_at": "2024-05-11T02:38:44.000000Z",
        "created_at": "2024-05-11T02:38:44.000000Z",
        "user_id": 4
    }
}
```

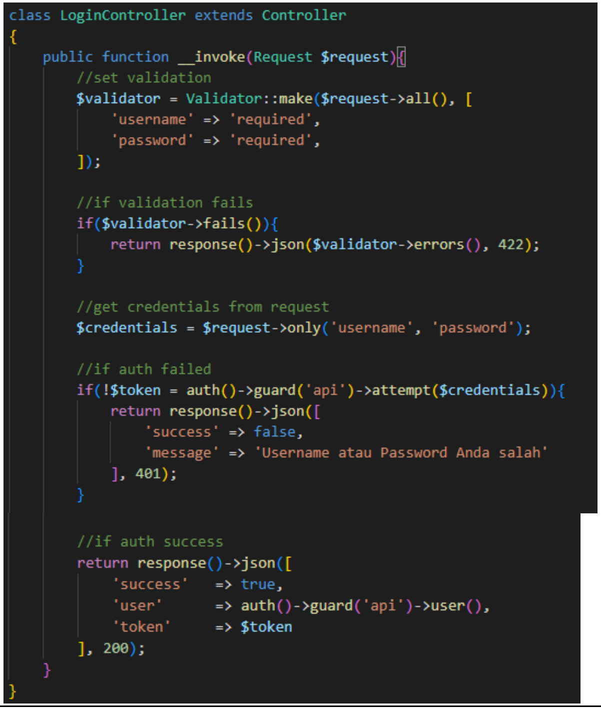
Praktikum 2 – Membuat RESTful API Login

1 Membuat Api/LoginController



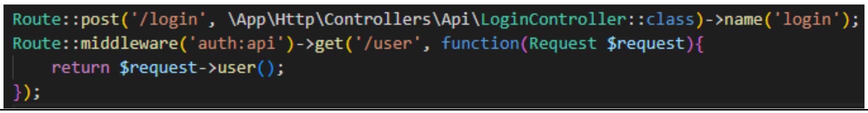
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/LoginController

  INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\LoginController.php] cr
successfully.
```

2 Modifikasi LoginController



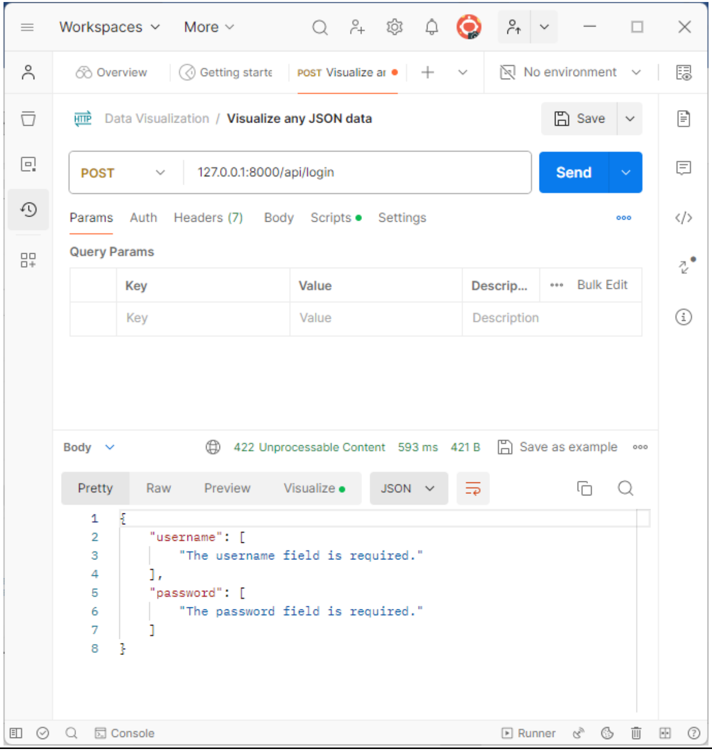
```php
class LoginController extends Controller
{
    public function __invoke(Request $request){
        //set validation
        $validator = Validator::make($request->all(), [
            'username' => 'required',
            'password' => 'required',
        ]);

        //if validation fails
        if($validator->fails()){
            return response()->json($validator->errors(), 422);
        }

        //get credentials from request
        $credentials = $request->only('username', 'password');

        //if auth failed
        if(!$token = auth()->guard('api')->attempt($credentials)){
            return response()->json([
                'success' => false,
                'message' => 'Username atau Password Anda salah'
            ], 401);
        }

        //if auth success
        return response()->json([
            'success'   => true,
            'user'      => auth()->guard('api')->user(),
            'token'     => $token
        ], 200);
    }
}
```

3 Menambahkan route



```php
Route::post('/login', \App\Http\Controllers\Api\LoginController::class)->name('login');
Route::middleware('auth:api')->get('/user', function(Request $request){
    return $request->user();
});
```

4 Mengisi URL method POST pada aplikasi Postman (127.0.0.1:8000/api/login)



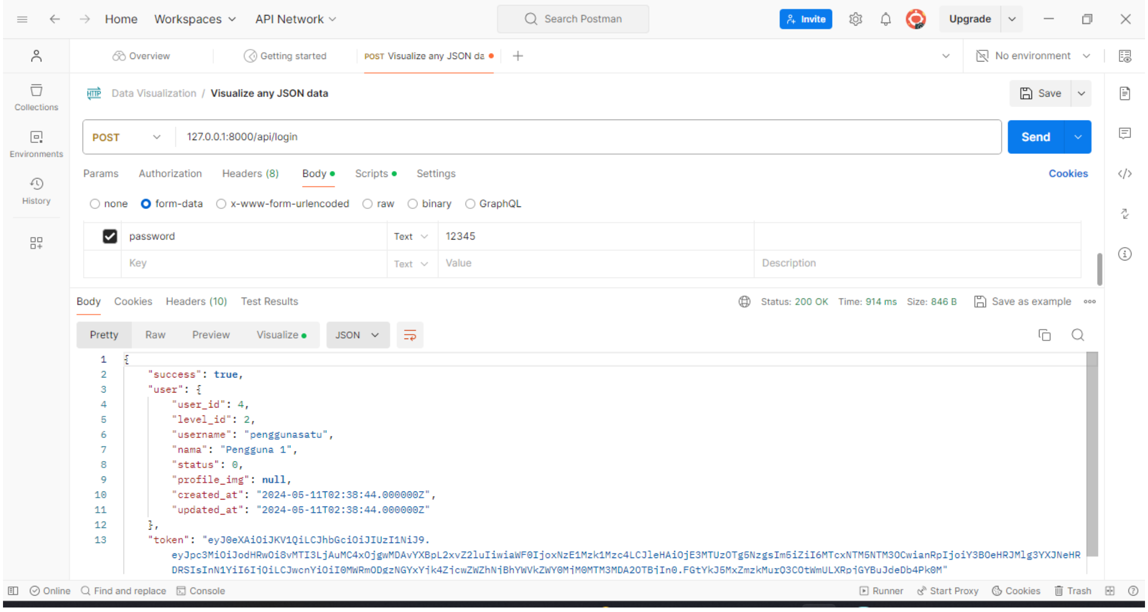
POST 127.0.0.1:8000/api/login

Key	Value	Descrip...
Key	Value	Description

422 Unprocessable Content 593 ms 421 B

```json
{
    "username": [
        "The username field is required."
    ],
    "password": [
        "The password field is required."
    ]
}
```

5 Mencoba memasukkan data



POST 127.0.0.1:8000/api/login

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

	Key		Value	Description
☑	password	Text	12345	
	Key	Text	Value	Description

Status: 200 OK Time: 914 ms Size: 846 B

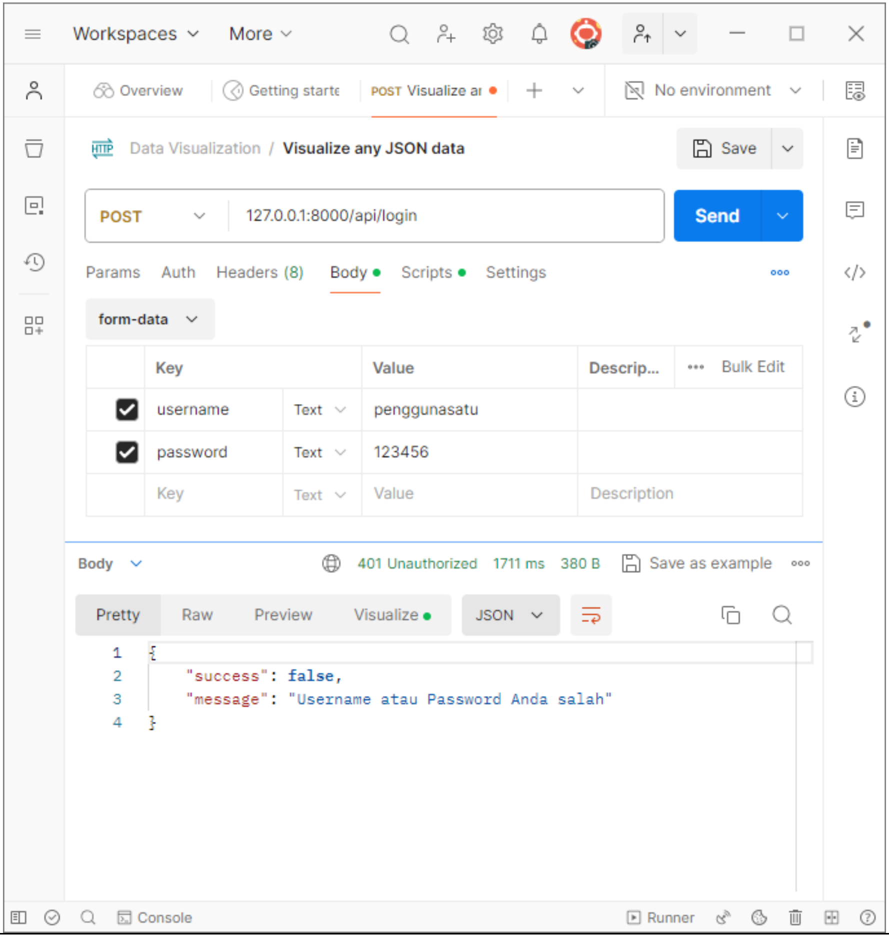
JSON

```
{
    "success": true,
    "user": {
        "user_id": 4,
        "level_id": 2,
        "username": "penggunasatu",
        "nama": "Pengguna 1",
        "status": 0,
        "profile_img": null,
        "created_at": "2024-05-11T02:38:44.000000Z",
        "updated_at": "2024-05-11T02:38:44.000000Z"
    },
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.
        eyJpc3MiOiJodHRwOi8vMTI3LjAuMC4xOjgwMDAvYXBpL2xvZ2luIiwiaWF0IjoxNzE1Mzk1Mzc4LCJleHAiOjE3MTUzOTg5NzgsIm5iZiI6MTcxNTM5NTM3OCwianRpIjoiY3BOeHRJMlg3YXJNeHR
        DRSIsInN1YiI6IjQiLCJwcnYiOiI0MWRmODgzNGYxYjk4ZjcwZWZhNjBhYWVkZWY0MjM0MTM3MDA2OTBjIn0.FGtYkJ5MxZmzkMurQ3COtWmULXRpjGYBuJdeDb4Pk0M"
```

Copy token login yg nanti diperlukan untuk logout

eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOi8vMTI3LjAuMC4xOjgwMDAvYXBpL2xvZ2luIiwiaWF0IjoxNzE1Mzk2MTUyLCJleHAiOjE3MTUzOTk3NTIsIm5iZiI6MTcxNTM5NjE1MiwianRpIjoiZ3NBYXdCbXVnM3E5d2tpZyIsInN1YiI6IjQiLCJwcnYiOiI0MWRmODgzNGYxYjk4ZjcwZWZhNjBhYWVkZWY0MjM0MTM3MDA2OTBjIn0.nNeGx8rDwI9yNlsy7mpGo-4SAh1x85W8RPmm9aVrfjQ

6 Percobaan untuk data yang salah



POST 127.0.0.1:8000/api/login

form-data

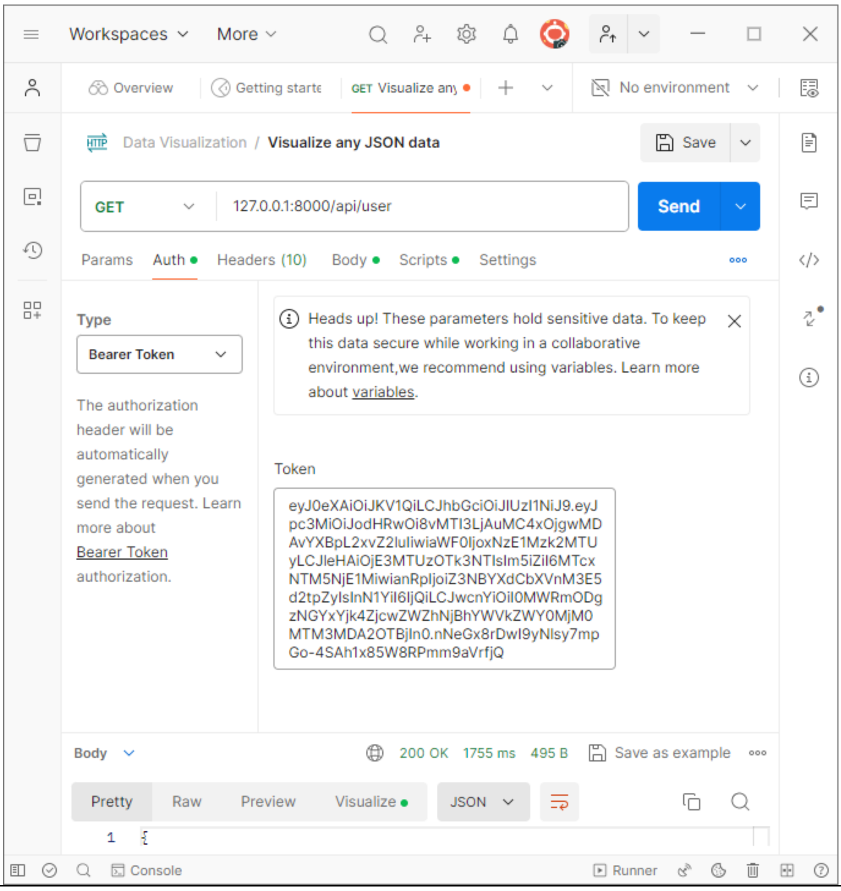
	Key		Value	Descrip...
☑	username	Text	penggunasatu	
☑	password	Text	123456	
	Key	Text	Value	Description

Body 401 Unauthorized 1711 ms 380 B

JSON

```
{
    "success": false,
    "message": "Username atau Password Anda salah"
}
```

7 Memasukkan token di auth – bearer token



GET 127.0.0.1:8000/api/user

Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment,we recommend using variables. Learn more about variables.

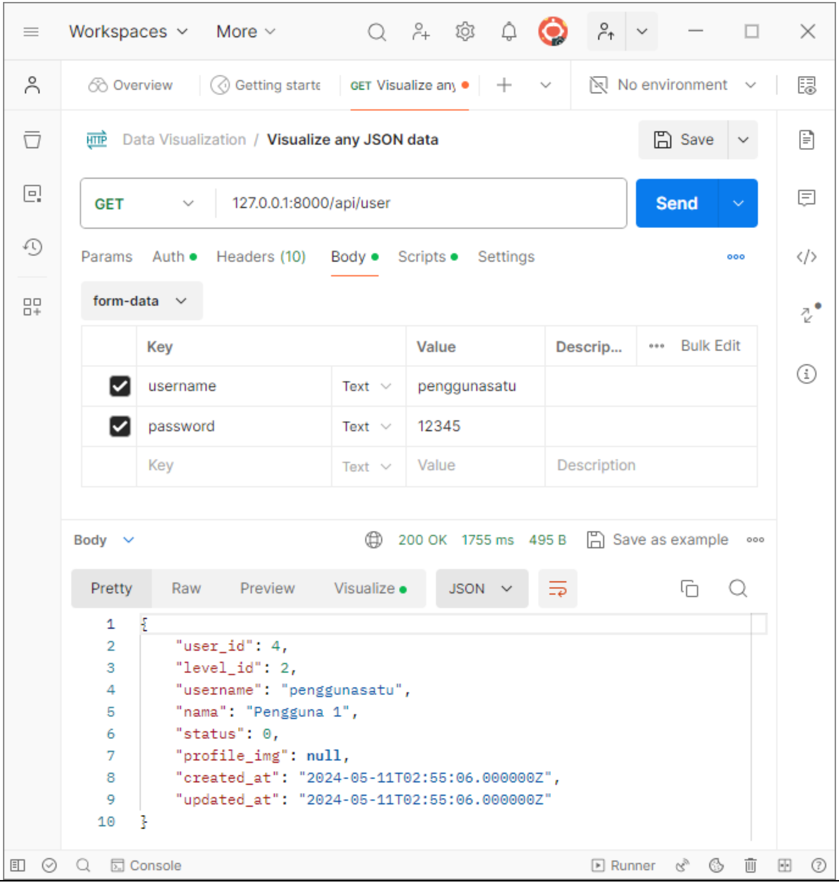
Token

```
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ
pc3MiOiJodHRwOi8vMTI3LjAuMC4xOjgwMD
AvYXBpL2xvZ2luIiwiaWF0IjoxNzE1Mzk2MTU
yLCJleHAiOjE3MTUzOTk3NTIsIm5iZiI6MTcx
NTM5NjE1MiwianRpIjoiZ3NBYXdCbXVnM3E5
d2tpZyIsInN1YiI6IjQiLCJwcnYiOiI0MWRmODg
zNGYxYjk4ZjcwZWZhNjBhYWVkZWY0MjM0
MTM3MDA2OTBjIn0.nNeGx8rDwI9yNlsy7mp
Go-4SAh1x85W8RPmm9aVrfjQ
```

200 OK 1755 ms 495 B

```
{
```

8 Melakukan send request method GET, maka akan ditampilkan data user yang sedang login



GET 127.0.0.1:8000/api/user

form-data

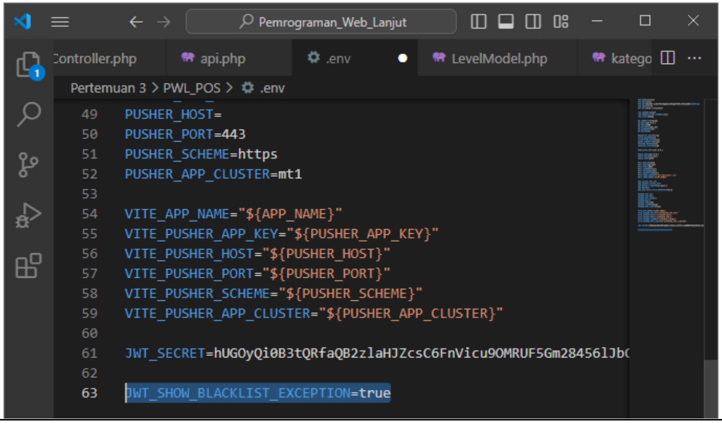
	Key		Value	Descrip...
☑	username	Text	penggunasatu	
☑	password	Text	12345	
	Key	Text	Value	Description

200 OK 1755 ms 495 B

```
{
    "user_id": 4,
    "level_id": 2,
    "username": "penggunasatu",
    "nama": "Pengguna 1",
    "status": 0,
    "profile_img": null,
    "created_at": "2024-05-11T02:55:06.000000Z",
    "updated_at": "2024-05-11T02:55:06.000000Z"
}
```

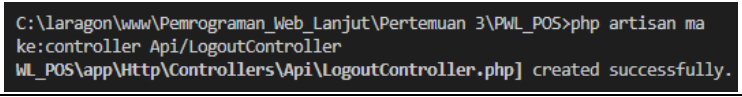
Praktikum 3 – Membuat RESTful API Logout

1 Menambahkan kode pada file .env



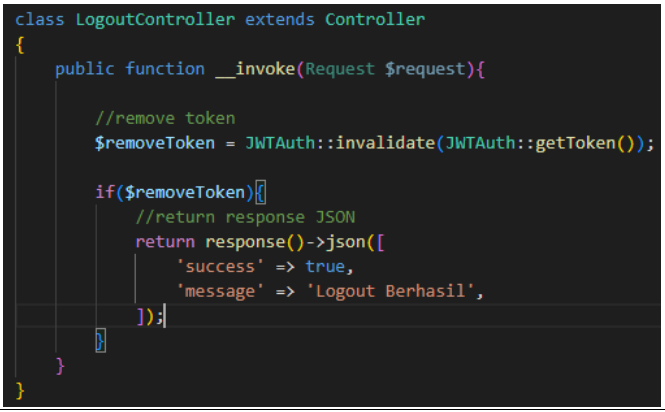
```
PUSHER_HOST=
PUSHER_PORT=443
PUSHER_SCHEME=https
PUSHER_APP_CLUSTER=mt1

VITE_APP_NAME="${APP_NAME}"
VITE_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
VITE_PUSHER_HOST="${PUSHER_HOST}"
VITE_PUSHER_PORT="${PUSHER_PORT}"
VITE_PUSHER_SCHEME="${PUSHER_SCHEME}"
VITE_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"

JWT_SECRET=hUGOyQi0B3tQRfaQB2zlaHJZcsC6FnVicu9OMRUF5Gm28456lJb(

JWT_SHOW_BLACKLIST_EXCEPTION=true
```

2 Membuat Api/LogoutController



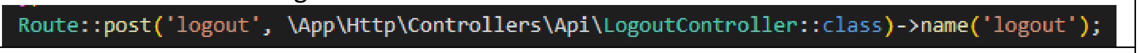
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan ma
ke:controller Api/LogoutController
WL_POS\app\Http\Controllers\Api\LogoutController.php] created successfully.
```

3 Modifikasi kode



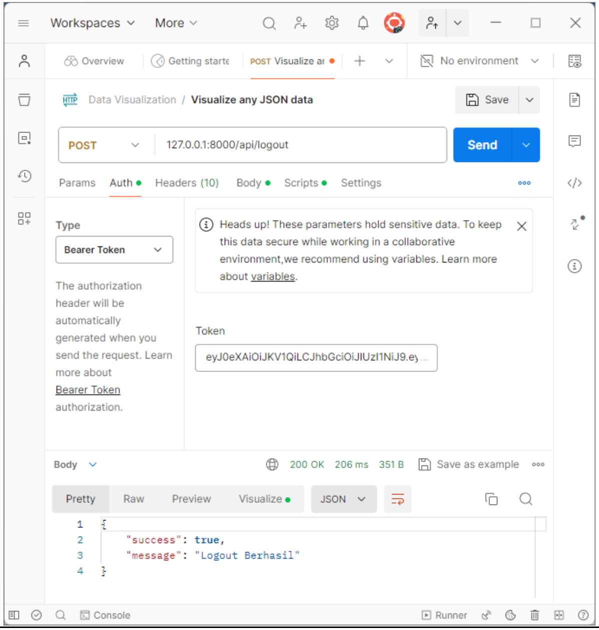
```php
class LogoutController extends Controller
{
    public function __invoke(Request $request){

        //remove token
        $removeToken = JWTAuth::invalidate(JWTAuth::getToken());

        if($removeToken){
            //return response JSON
            return response()->json([
                'success' => true,
                'message' => 'Logout Berhasil',
            ]);
        }
    }
}
```

4 Menambahkan routes logout



```php
Route::post('logout', \App\Http\Controllers\Api\LogoutController::class)->name('logout');
```

5 Mengisi URL method POST pada aplikasi Postman (127.0.0.1:8000/api/logout) dan memasukkan token saat login di Auth – Bearer Token



POST 127.0.0.1:8000/api/logout

Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment,we recommend using variables. Learn more about variables.

Token: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey...

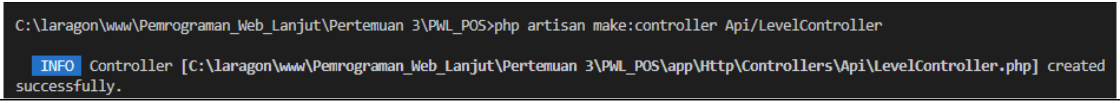
200 OK 206 ms 351 B

```
{
    "success": true,
    "message": "Logout Berhasil"
}
```

Praktikum 4 – Implementasi CRUD dalam RESTful API

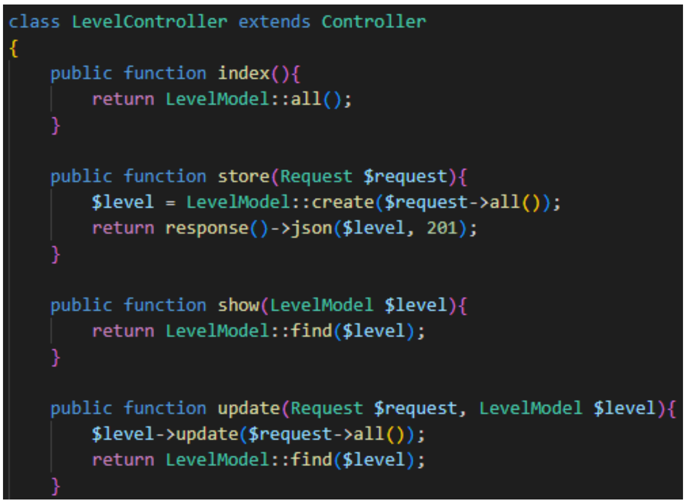
Data Level

1 Membuat Api/LevelController

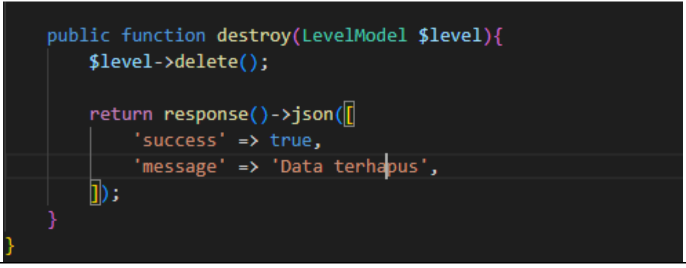


```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/LevelController

   INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\LevelController.php] created successfully.
```

2 Menambahkan kode

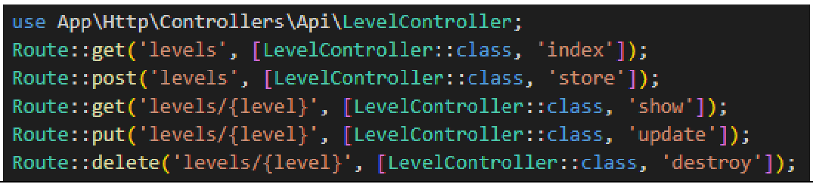


```php
class LevelController extends Controller
{
    public function index(){
        return LevelModel::all();
    }

    public function store(Request $request){
        $level = LevelModel::create($request->all());
        return response()->json($level, 201);
    }

    public function show(LevelModel $level){
        return LevelModel::find($level);
    }

    public function update(Request $request, LevelModel $level){
        $level->update($request->all());
        return LevelModel::find($level);
    }
```



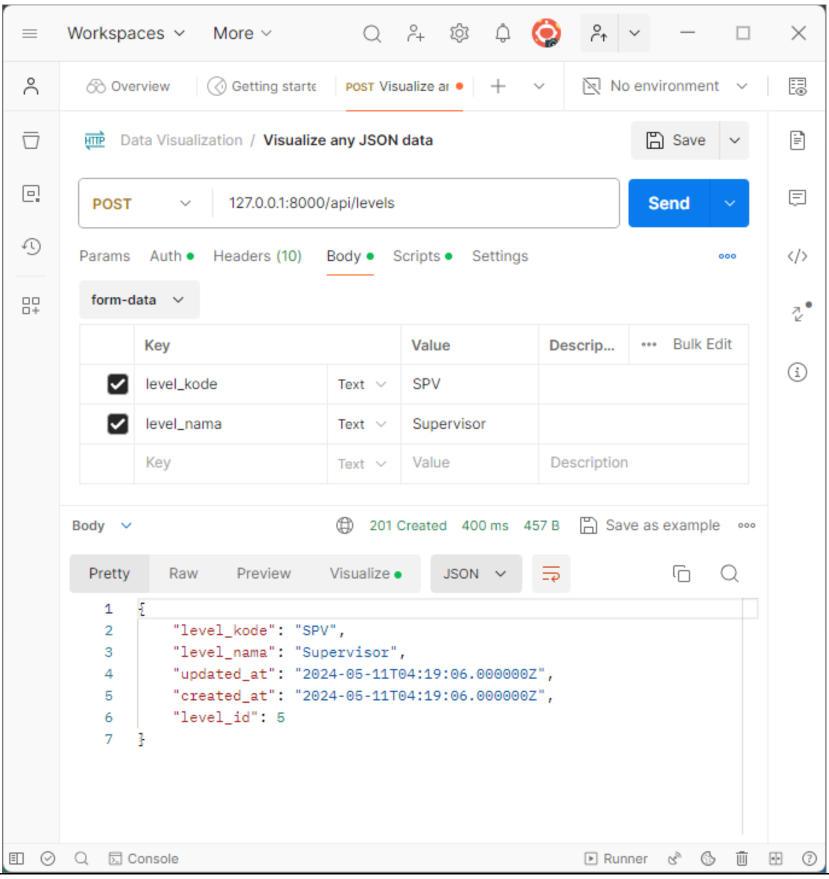
```php
    public function destroy(LevelModel $level){
        $level->delete();

        return response()->json([
            'success' => true,
            'message' => 'Data terhapus',
        ]);
    }
}
```

3 Menambahkan route level



```php
use App\Http\Controllers\Api\LevelController;
Route::get('levels', [LevelController::class, 'index']);
Route::post('levels', [LevelController::class, 'store']);
Route::get('levels/{level}', [LevelController::class, 'show']);
Route::put('levels/{level}', [LevelController::class, 'update']);
Route::delete('levels/{level}', [LevelController::class, 'destroy']);
```

4 Mengisi URL method POST pada aplikasi Postman (127.0.0.1:8000/api/levels) dan memasukkan form-data



POST 127.0.0.1:8000/api/levels

form-data

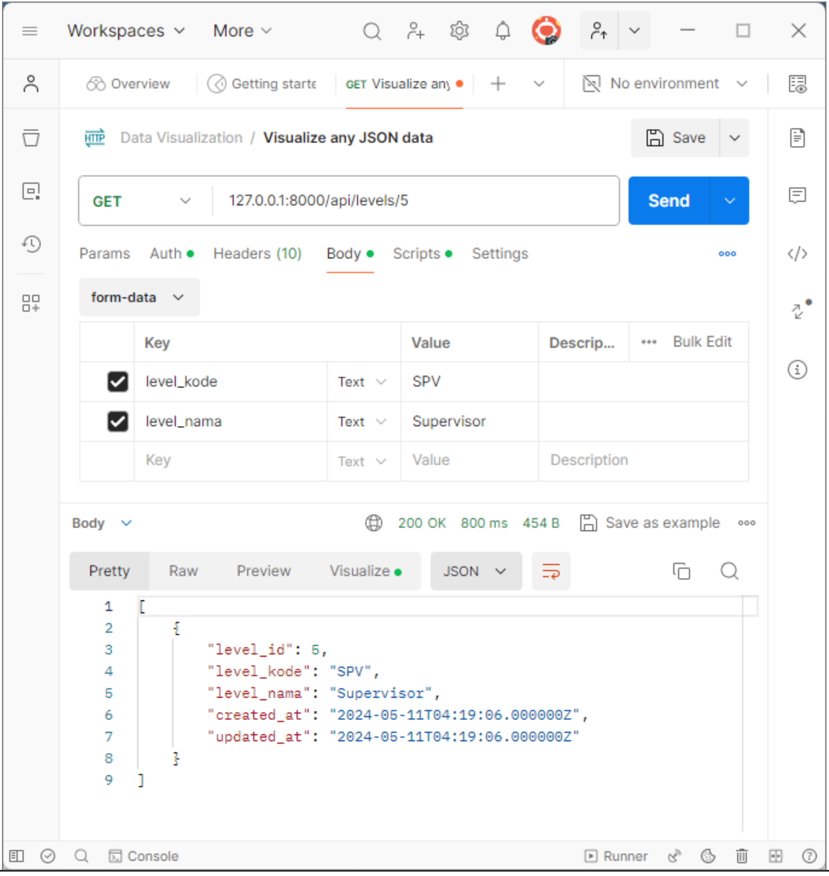
	Key		Value	Descrip...
☑	level_kode	Text	SPV	
☑	level_nama	Text	Supervisor	
	Key	Text	Value	Description

Body 201 Created 400 ms 457 B

JSON

```
{
    "level_kode": "SPV",
    "level_nama": "Supervisor",
    "updated_at": "2024-05-11T04:19:06.000000Z",
    "created_at": "2024-05-11T04:19:06.000000Z",
    "level_id": 5
}
```

5 Menampilkan detail data (127.0.0.1:8000/api/levels/5) sesuai level_id



GET 127.0.0.1:8000/api/levels/5

form-data

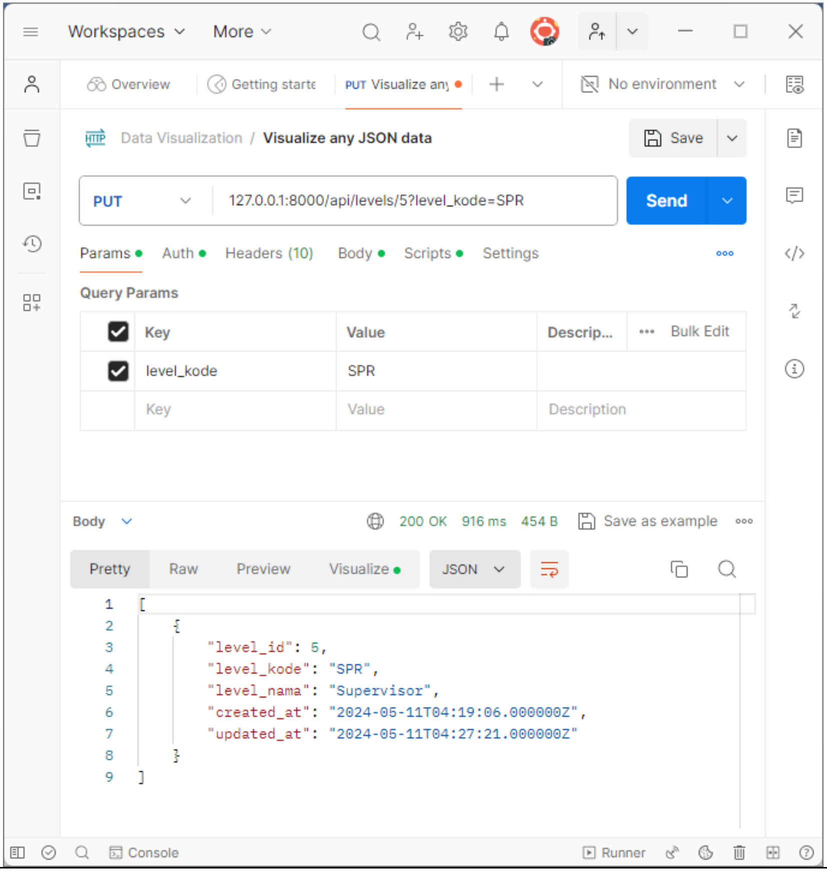
	Key		Value	Descrip...
☑	level_kode	Text	SPV	
☑	level_nama	Text	Supervisor	
	Key	Text	Value	Description

Body 200 OK 800 ms 454 B

JSON

```
[
    {
        "level_id": 5,
        "level_kode": "SPV",
        "level_nama": "Supervisor",
        "created_at": "2024-05-11T04:19:06.000000Z",
        "updated_at": "2024-05-11T04:19:06.000000Z"
    }
]
```

6 Melakukan edit data (127.0.0.1:8000/api/levels/{id}) method PUT dengan mengisi data di bagian Params. URL otomatis terisi, jika mengisi query params



PUT 127.0.0.1:8000/api/levels/5?level_kode=SPR

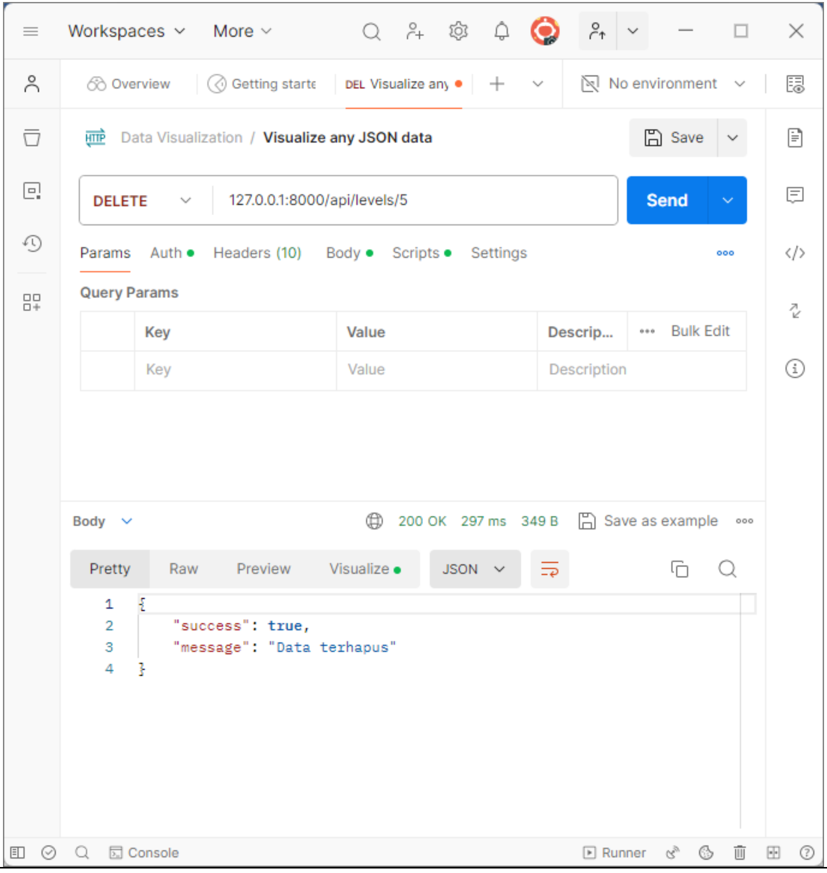
Query Params

	Key	Value	Descrip...
☑	level_kode	SPR	
	Key	Value	Description

200 OK 916 ms 454 B

```
[
    {
        "level_id": 5,
        "level_kode": "SPR",
        "level_nama": "Supervisor",
        "created_at": "2024-05-11T04:19:06.000000Z",
        "updated_at": "2024-05-11T04:27:21.000000Z"
    }
]
```

7 Melakukan hapus data



DELETE 127.0.0.1:8000/api/levels/5

Query Params

Key	Value	Descrip...
Key	Value	Description

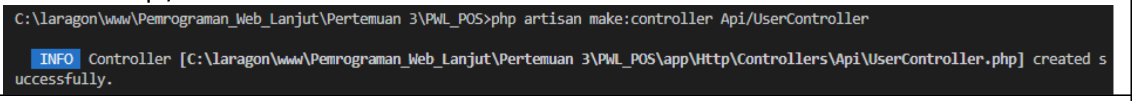
200 OK 297 ms 349 B

```
{
    "success": true,
    "message": "Data terhapus"
}
```

Tugas – Implementasi CRUD API pada tabel m_user, m_kategori, m_barang

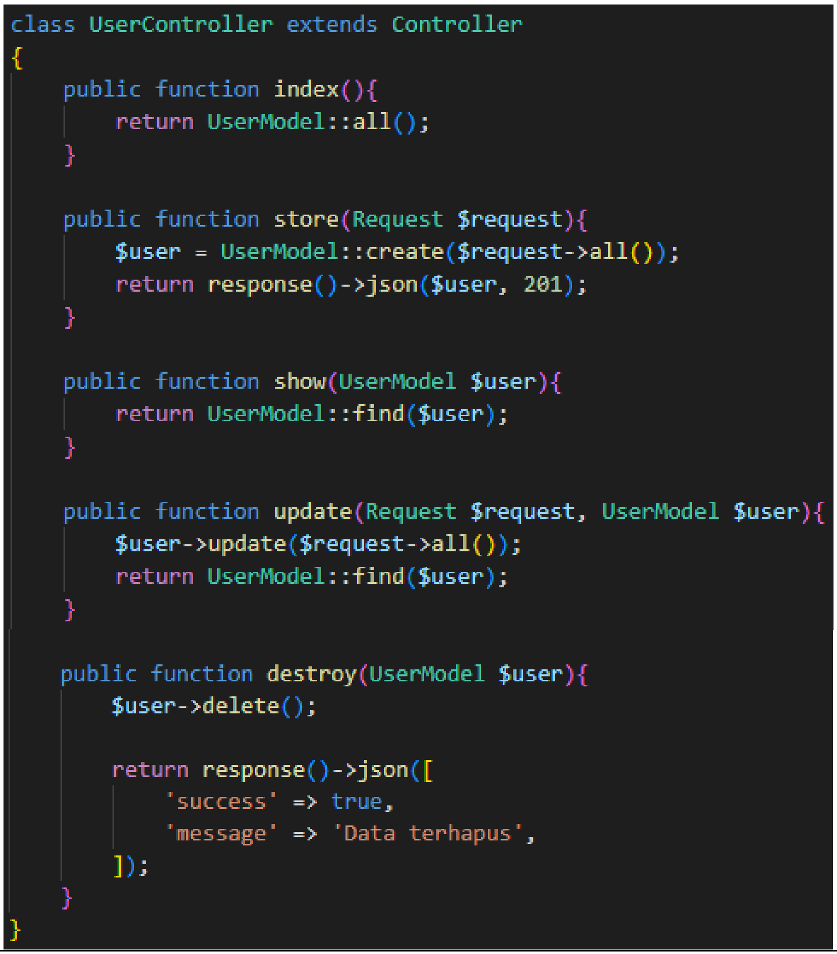
Data User

1 Membuat Api/UserController



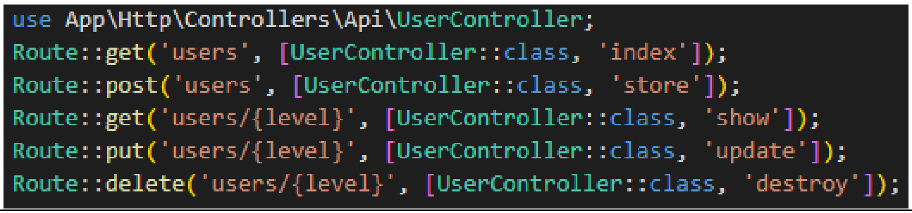
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/UserController

  INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\UserController.php] created s
uccessfully.
```

2 Menambahkan kode



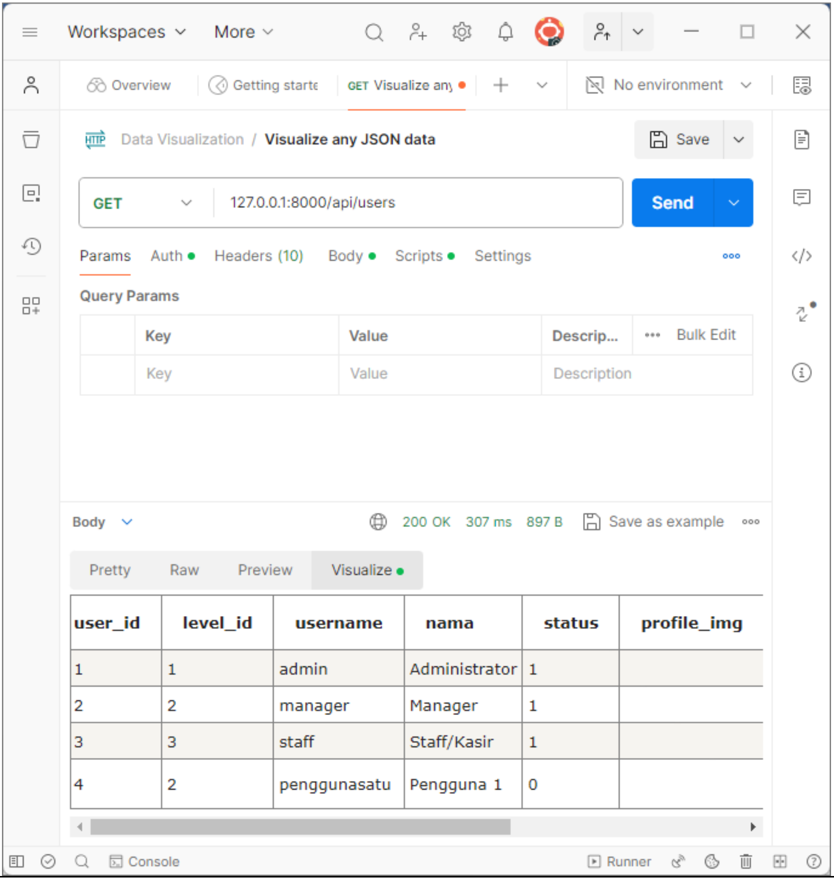
```php
class UserController extends Controller
{
    public function index(){
        return UserModel::all();
    }

    public function store(Request $request){
        $user = UserModel::create($request->all());
        return response()->json($user, 201);
    }

    public function show(UserModel $user){
        return UserModel::find($user);
    }

    public function update(Request $request, UserModel $user){
        $user->update($request->all());
        return UserModel::find($user);
    }

    public function destroy(UserModel $user){
        $user->delete();

        return response()->json([
            'success' => true,
            'message' => 'Data terhapus',
        ]);
    }
}
```

3 Menambahkan route user



```php
use App\Http\Controllers\Api\UserController;
Route::get('users', [UserController::class, 'index']);
Route::post('users', [UserController::class, 'store']);
Route::get('users/{level}', [UserController::class, 'show']);
Route::put('users/{level}', [UserController::class, 'update']);
Route::delete('users/{level}', [UserController::class, 'destroy']);
```

4 Data User yang sudah ada



GET 127.0.0.1:8000/api/users

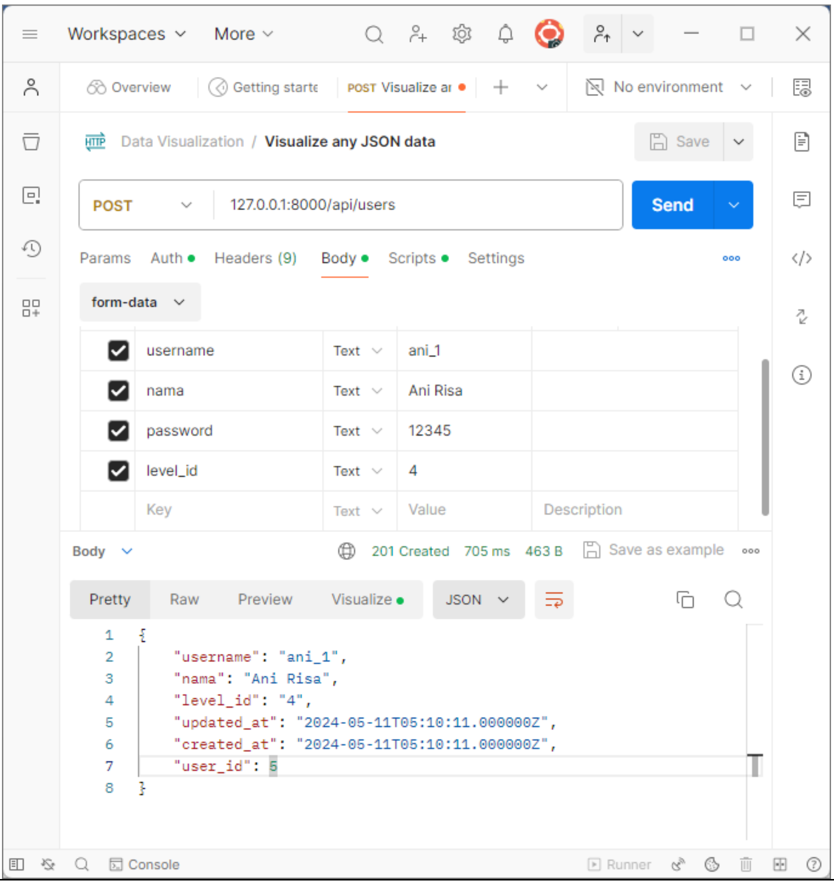
Query Params

Key	Value	Descrip...
Key	Value	Description

200 OK 307 ms 897 B

user_id	level_id	username	nama	status	profile_img
1	1	admin	Administrator	1	
2	2	manager	Manager	1	
3	3	staff	Staff/Kasir	1	
4	2	penggunasatu	Pengguna 1	0	

5 Mengisi URL method POST pada aplikasi Postman (127.0.0.1:8000/api/users) dan memasukkan form-data



POST 127.0.0.1:8000/api/users

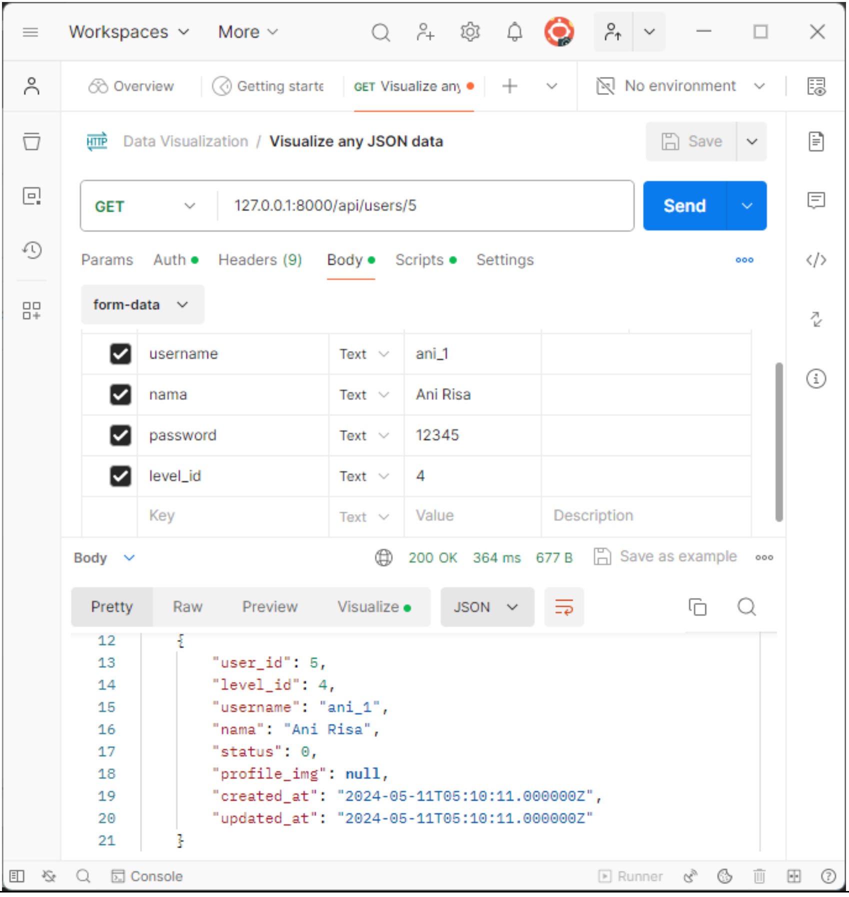
form-data

	Key	Type	Value	Description
☑	username	Text	ani_1	
☑	nama	Text	Ani Risa	
☑	password	Text	12345	
☑	level_id	Text	4	
	Key	Text	Value	Description

201 Created 705 ms 463 B

```
{
    "username": "ani_1",
    "nama": "Ani Risa",
    "level_id": "4",
    "updated_at": "2024-05-11T05:10:11.000000Z",
    "created_at": "2024-05-11T05:10:11.000000Z",
    "user_id": 5
}
```

6 Menampilkan detail data (127.0.0.1:8000/api/users/6) sesuai level_id



GET 127.0.0.1:8000/api/users/5

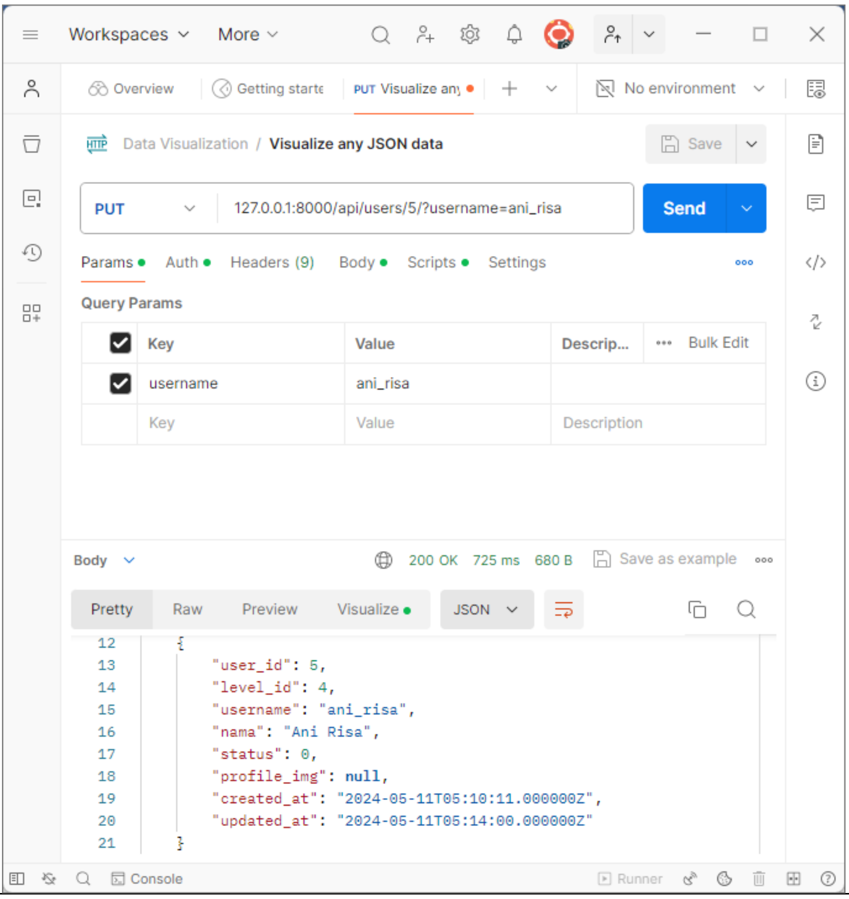
form-data

	Key	Type	Value	Description
☑	username	Text	ani_1	
☑	nama	Text	Ani Risa	
☑	password	Text	12345	
☑	level_id	Text	4	
	Key	Text	Value	Description

Body 200 OK 364 ms 677 B

```
{
    "user_id": 5,
    "level_id": 4,
    "username": "ani_1",
    "nama": "Ani Risa",
    "status": 0,
    "profile_img": null,
    "created_at": "2024-05-11T05:10:11.000000Z",
    "updated_at": "2024-05-11T05:10:11.000000Z"
}
```

7 Melakukan edit data



PUT 127.0.0.1:8000/api/users/5/?username=ani_risa

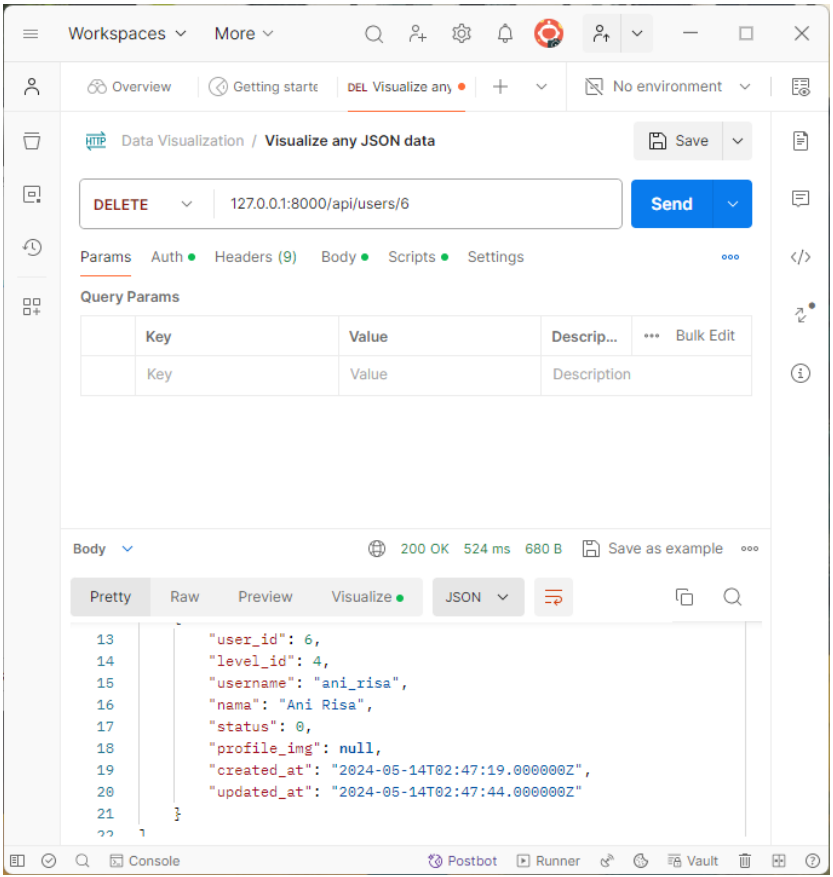
Query Params

	Key	Value	Descrip...
☑	username	ani_risa	
	Key	Value	Description

Body 200 OK 725 ms 680 B

```
{
    "user_id": 5,
    "level_id": 4,
    "username": "ani_risa",
    "nama": "Ani Risa",
    "status": 0,
    "profile_img": null,
    "created_at": "2024-05-11T05:10:11.000000Z",
    "updated_at": "2024-05-11T05:14:00.000000Z"
}
```

8 Delete user



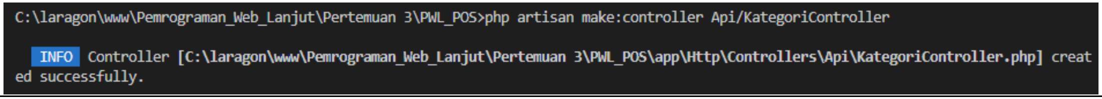
DELETE 127.0.0.1:8000/api/users/6

Body 200 OK 524 ms 680 B

```
        "user_id": 6,
        "level_id": 4,
        "username": "ani_risa",
        "nama": "Ani Risa",
        "status": 0,
        "profile_img": null,
        "created_at": "2024-05-14T02:47:19.000000Z",
        "updated_at": "2024-05-14T02:47:44.000000Z"
    }
]
```

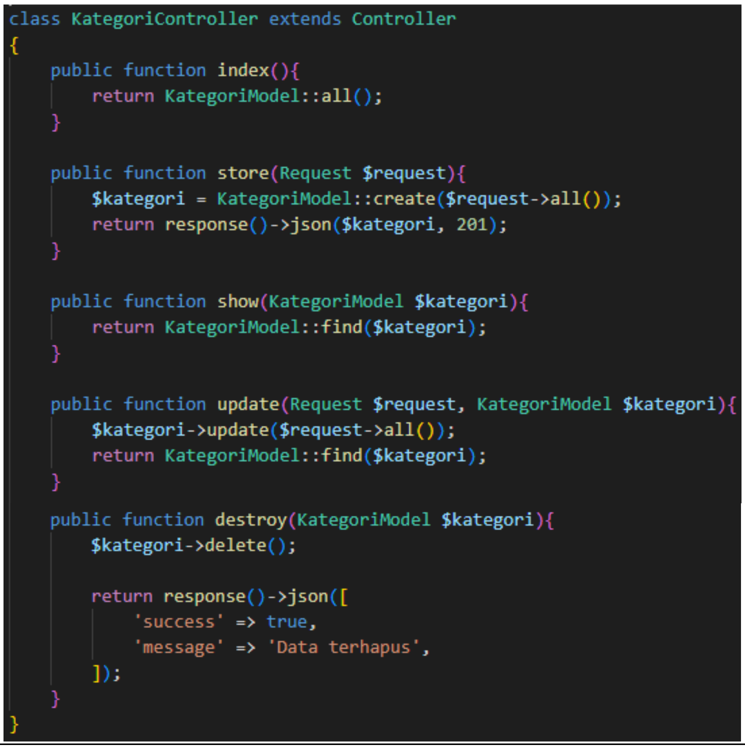
Data Kategori

1 Membuat Api/KategoriController



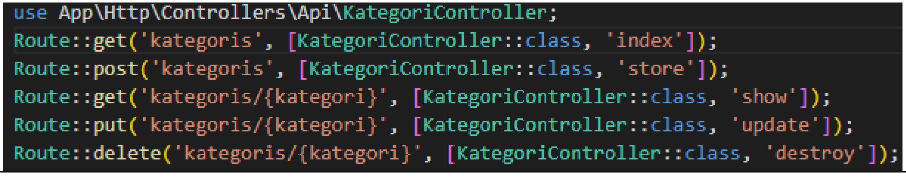
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/KategoriController

  INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\KategoriController.php] created successfully.
```

2 Menambahkan kode



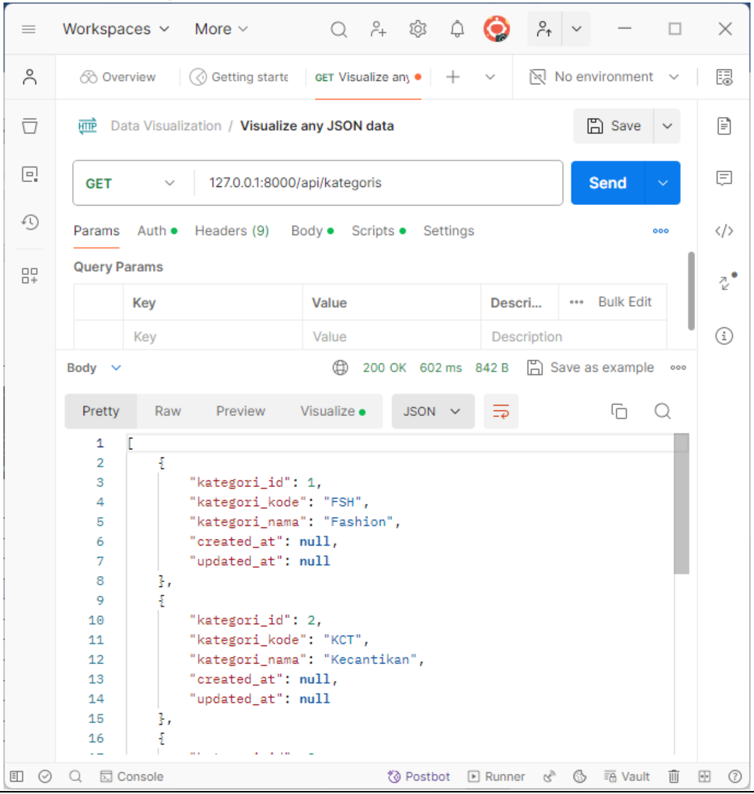
```
class KategoriController extends Controller
{
    public function index(){
        return KategoriModel::all();
    }

    public function store(Request $request){
        $kategori = KategoriModel::create($request->all());
        return response()->json($kategori, 201);
    }

    public function show(KategoriModel $kategori){
        return KategoriModel::find($kategori);
    }

    public function update(Request $request, KategoriModel $kategori){
        $kategori->update($request->all());
        return KategoriModel::find($kategori);
    }
    public function destroy(KategoriModel $kategori){
        $kategori->delete();

        return response()->json([
            'success' => true,
            'message' => 'Data terhapus',
        ]);
    }
}
```

3 Menambahkan route kategori



```
use App\Http\Controllers\Api\KategoriController;
Route::get('kategoris', [KategoriController::class, 'index']);
Route::post('kategoris', [KategoriController::class, 'store']);
Route::get('kategoris/{kategori}', [KategoriController::class, 'show']);
Route::put('kategoris/{kategori}', [KategoriController::class, 'update']);
Route::delete('kategoris/{kategori}', [KategoriController::class, 'destroy']);
```

4 Get data kategori



Data Visualization / Visualize any JSON data

GET 127.0.0.1:8000/api/kategoris Send

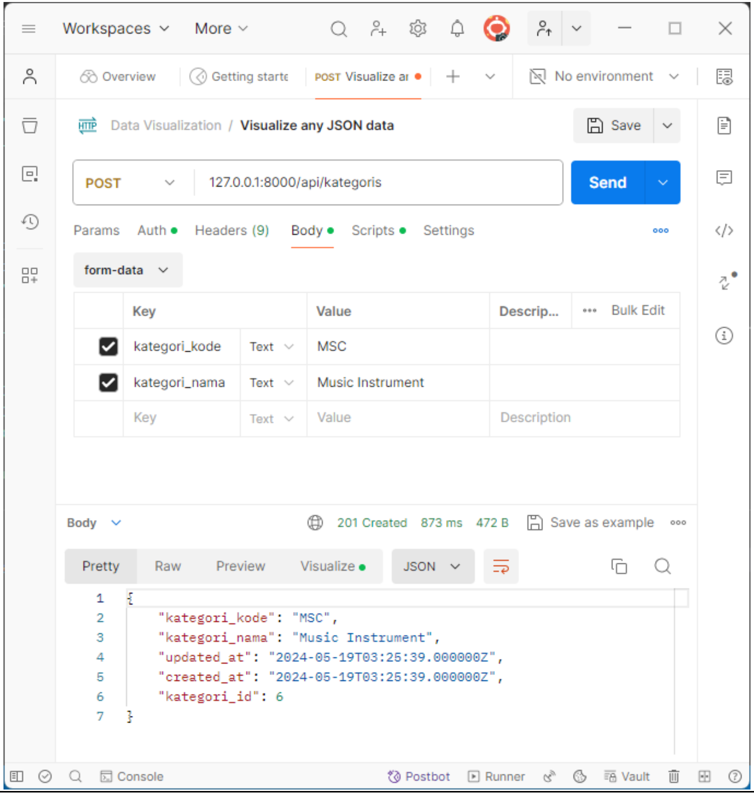
Query Params

Key	Value	Descri...
Key	Value	Description

Body 200 OK 602 ms 842 B

```
[
    {
        "kategori_id": 1,
        "kategori_kode": "FSH",
        "kategori_nama": "Fashion",
        "created_at": null,
        "updated_at": null
    },
    {
        "kategori_id": 2,
        "kategori_kode": "KCT",
        "kategori_nama": "Kecantikan",
        "created_at": null,
        "updated_at": null
    },
    {
```

5 Create kategori



Data Visualization / Visualize any JSON data

POST 127.0.0.1:8000/api/kategoris Send

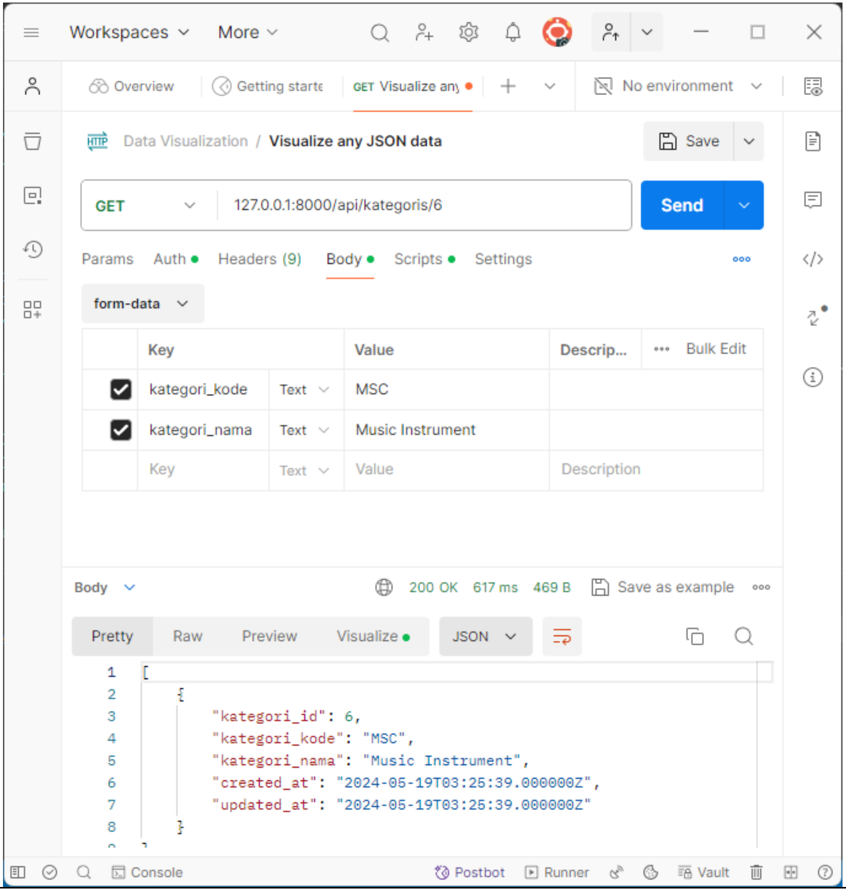
form-data

	Key		Value	Descrip...
☑	kategori_kode	Text	MSC	
☑	kategori_nama	Text	Music Instrument	
	Key	Text	Value	Description

Body 201 Created 873 ms 472 B

```
{
    "kategori_kode": "MSC",
    "kategori_nama": "Music Instrument",
    "updated_at": "2024-05-19T03:25:39.000000Z",
    "created_at": "2024-05-19T03:25:39.000000Z",
    "kategori_id": 6
}
```

6 Show kategori



GET 127.0.0.1:8000/api/kategoris/6

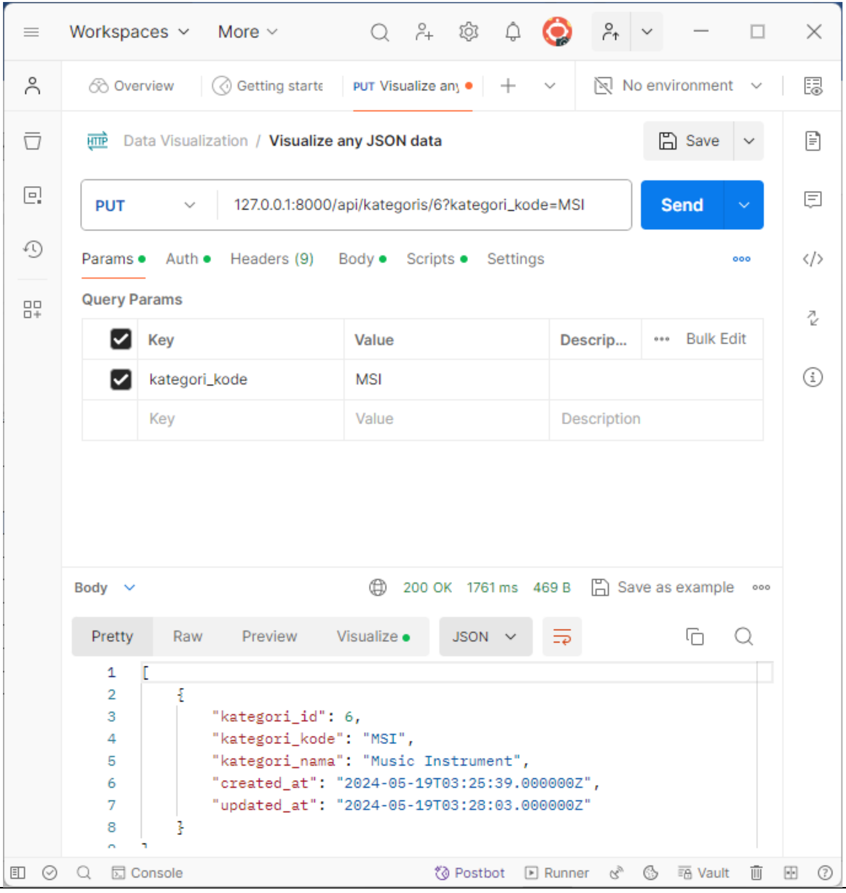
form-data

	Key		Value	Descrip...
☑	kategori_kode	Text	MSC	
☑	kategori_nama	Text	Music Instrument	
	Key	Text	Value	Description

Body 200 OK 617 ms 469 B

```
[
    {
        "kategori_id": 6,
        "kategori_kode": "MSC",
        "kategori_nama": "Music Instrument",
        "created_at": "2024-05-19T03:25:39.000000Z",
        "updated_at": "2024-05-19T03:25:39.000000Z"
    }
```

7 Edit kategori



PUT 127.0.0.1:8000/api/kategoris/6?kategori_kode=MSI

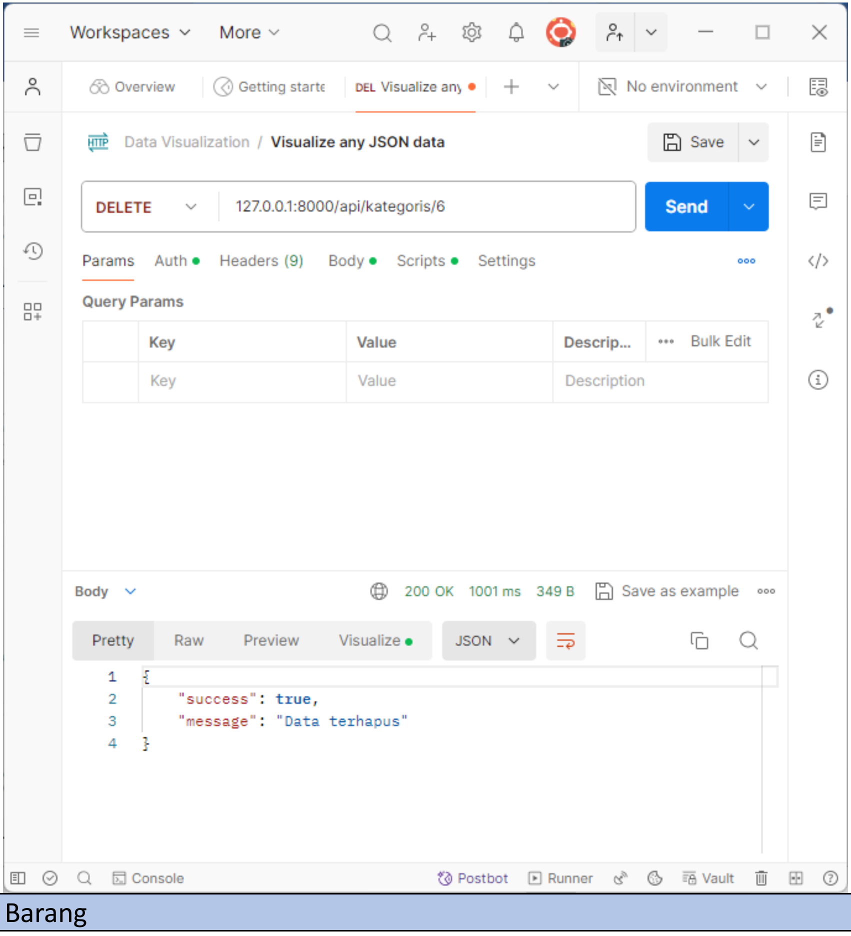
Query Params

	Key	Value	Descrip...
☑	kategori_kode	MSI	
	Key	Value	Description

Body 200 OK 1761 ms 469 B

```
[
    {
        "kategori_id": 6,
        "kategori_kode": "MSI",
        "kategori_nama": "Music Instrument",
        "created_at": "2024-05-19T03:25:39.000000Z",
        "updated_at": "2024-05-19T03:28:03.000000Z"
    }
```

8 Delete kategori



DELETE 127.0.0.1:8000/api/kategoris/6

Key	Value	Descrip...
Key	Value	Description

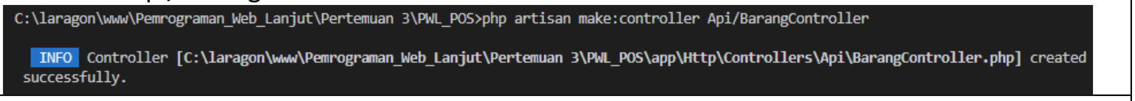
Body 200 OK 1001 ms 349 B

JSON

```
{
    "success": true,
    "message": "Data terhapus"
}
```

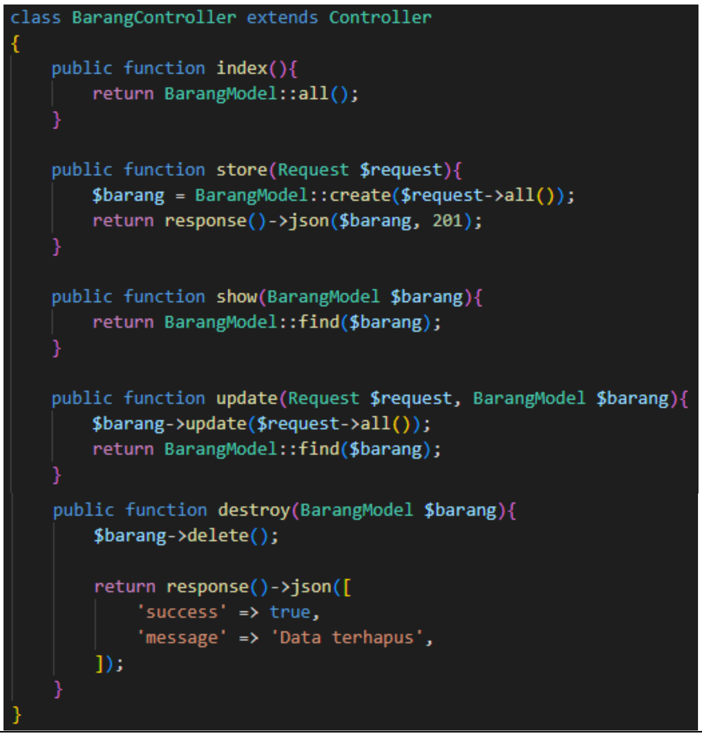
Data Barang

1 Membuat Api/BarangController



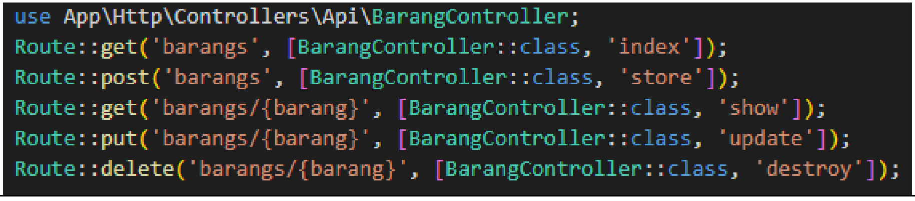
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:controller Api/BarangController

  INFO  Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\Api\BarangController.php] created
 successfully.
```

2 Menambahkan kode



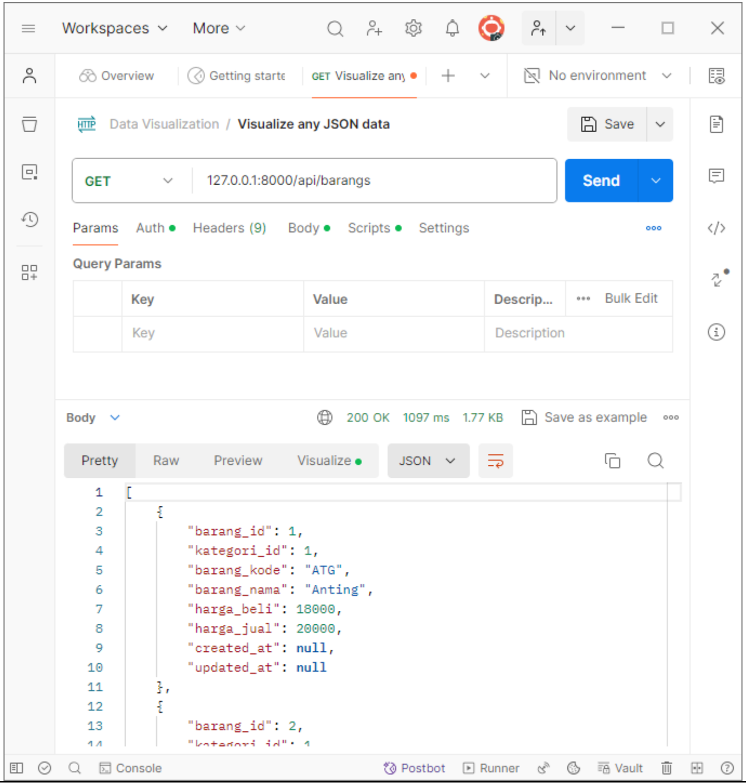
```
class BarangController extends Controller
{
    public function index(){
        return BarangModel::all();
    }

    public function store(Request $request){
        $barang = BarangModel::create($request->all());
        return response()->json($barang, 201);
    }

    public function show(BarangModel $barang){
        return BarangModel::find($barang);
    }

    public function update(Request $request, BarangModel $barang){
        $barang->update($request->all());
        return BarangModel::find($barang);
    }
    public function destroy(BarangModel $barang){
        $barang->delete();

        return response()->json([
            'success' => true,
            'message' => 'Data terhapus',
        ]);
    }
}
```

3 Menambahkan route barang



```
use App\Http\Controllers\Api\BarangController;
Route::get('barangs', [BarangController::class, 'index']);
Route::post('barangs', [BarangController::class, 'store']);
Route::get('barangs/{barang}', [BarangController::class, 'show']);
Route::put('barangs/{barang}', [BarangController::class, 'update']);
Route::delete('barangs/{barang}', [BarangController::class, 'destroy']);
```

4 Get data barang



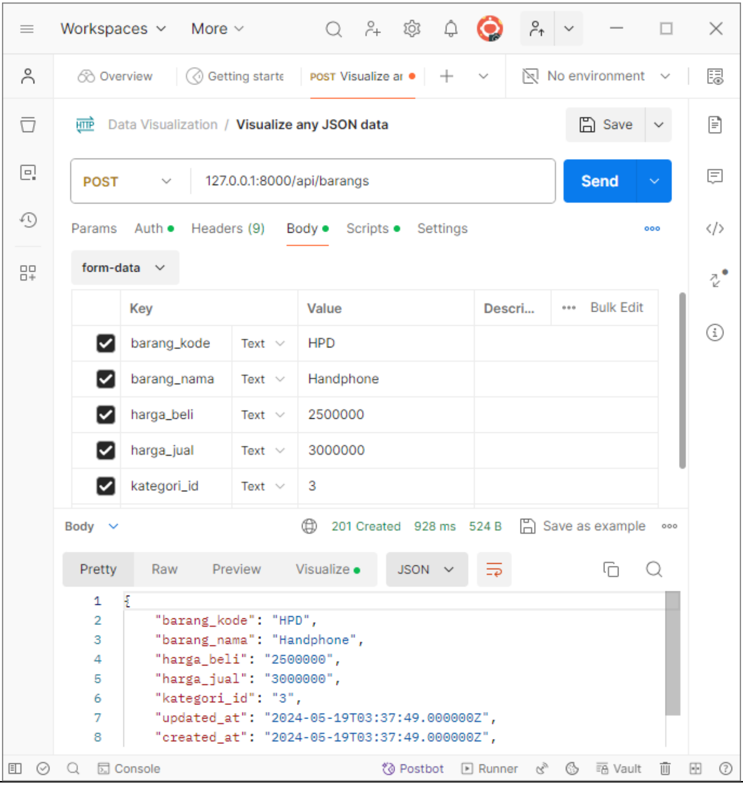
GET 127.0.0.1:8000/api/barangs

Key	Value	Descrip...
Key	Value	Description

Body 200 OK 1097 ms 1.77 KB

```
[
    {
        "barang_id": 1,
        "kategori_id": 1,
        "barang_kode": "ATG",
        "barang_nama": "Anting",
        "harga_beli": 18000,
        "harga_jual": 20000,
        "created_at": null,
        "updated_at": null
    },
    {
        "barang_id": 2,
        "kategori_id": 1
```

5 Create barang



POST 127.0.0.1:8000/api/barangs

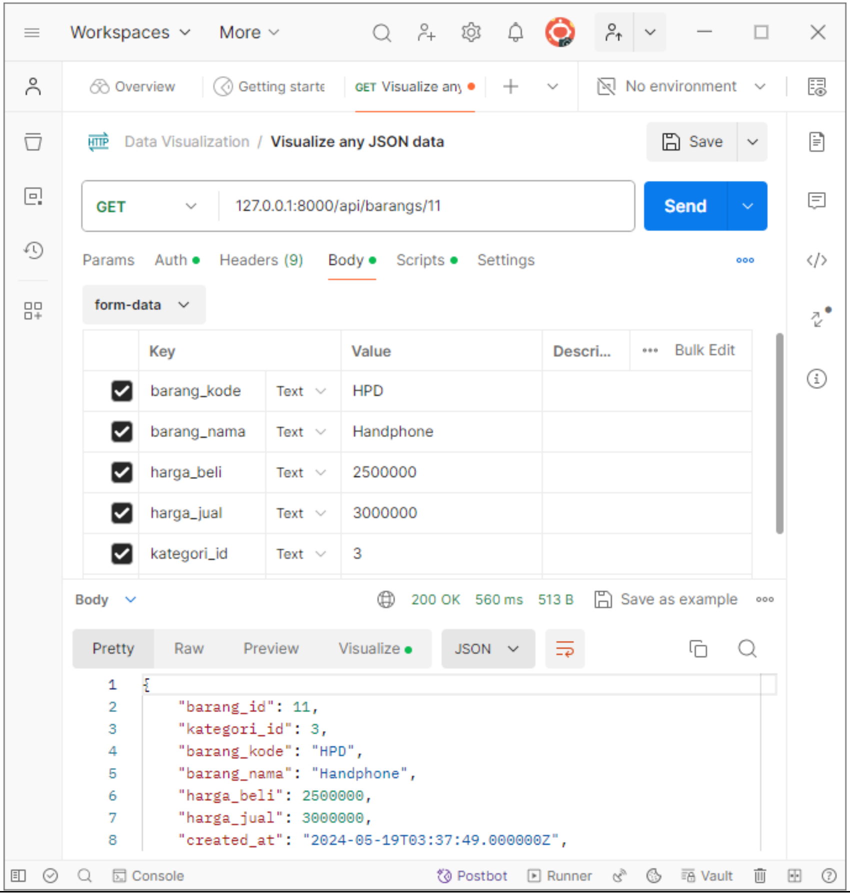
form-data

	Key		Value	Descri...
☑	barang_kode	Text	HPD	
☑	barang_nama	Text	Handphone	
☑	harga_beli	Text	2500000	
☑	harga_jual	Text	3000000	
☑	kategori_id	Text	3	

Body 201 Created 928 ms 524 B

```
{
    "barang_kode": "HPD",
    "barang_nama": "Handphone",
    "harga_beli": "2500000",
    "harga_jual": "3000000",
    "kategori_id": "3",
    "updated_at": "2024-05-19T03:37:49.000000Z",
    "created_at": "2024-05-19T03:37:49.000000Z",
```

6 Show barang



GET 127.0.0.1:8000/api/barangs/11

form-data

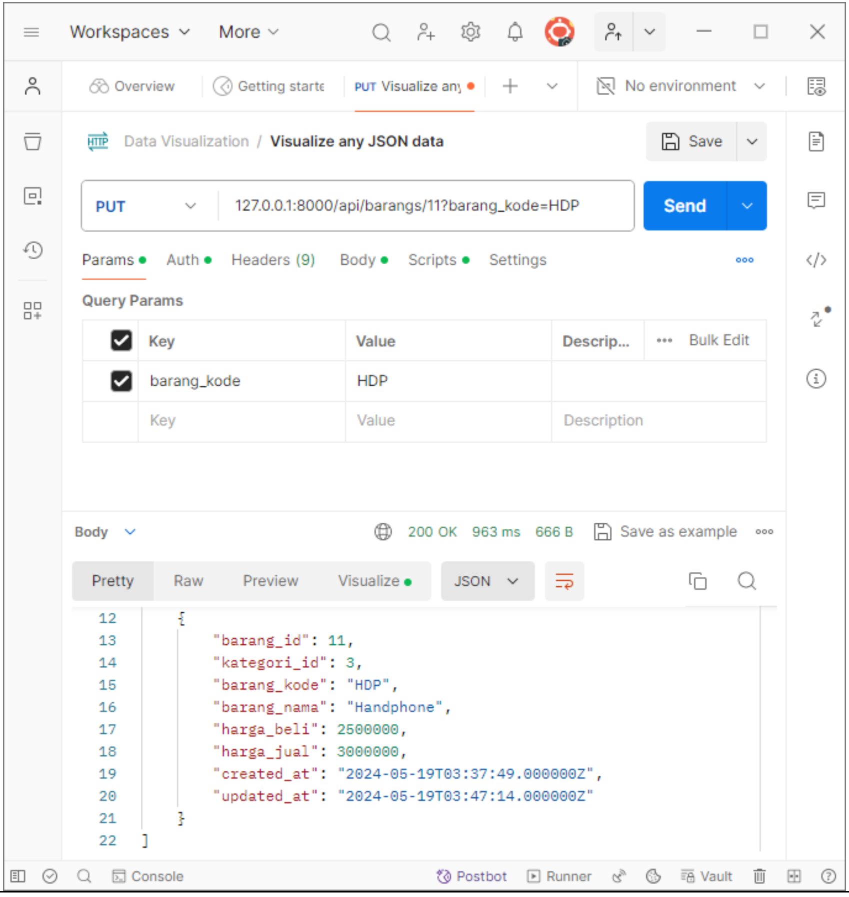
	Key		Value	Descri...
☑	barang_kode	Text	HPD	
☑	barang_nama	Text	Handphone	
☑	harga_beli	Text	2500000	
☑	harga_jual	Text	3000000	
☑	kategori_id	Text	3	

Body 200 OK 560 ms 513 B

JSON

```
{
    "barang_id": 11,
    "kategori_id": 3,
    "barang_kode": "HPD",
    "barang_nama": "Handphone",
    "harga_beli": 2500000,
    "harga_jual": 3000000,
    "created_at": "2024-05-19T03:37:49.000000Z",
```

7 Edit barang



PUT 127.0.0.1:8000/api/barangs/11?barang_kode=HDP

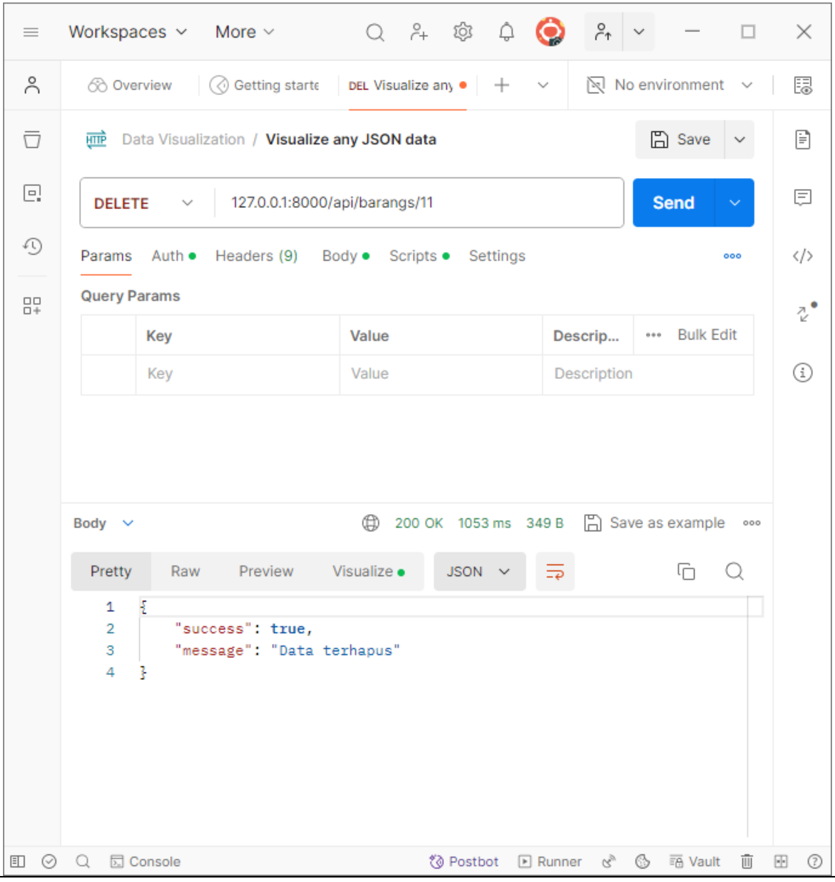
Query Params

☑	Key	Value	Descrip...
☑	barang_kode	HDP	
	Key	Value	Description

Body 200 OK 963 ms 666 B

JSON

```
    {
        "barang_id": 11,
        "kategori_id": 3,
        "barang_kode": "HDP",
        "barang_nama": "Handphone",
        "harga_beli": 2500000,
        "harga_jual": 3000000,
        "created_at": "2024-05-19T03:37:49.000000Z",
        "updated_at": "2024-05-19T03:47:14.000000Z"
    }
]
```

8	Delete barang 

DELETE 127.0.0.1:8000/api/barangs/11

200 OK 1053 ms 349 B

```
{
    "success": true,
    "message": "Data terhapus"
}
```