

LAPORAN PRAKTIKUM  
PEMROGRAMAN WEB LANJUT



DISUSUN OLEH:  
THORIQ FATHURROZI  
2H/28  
2241720052

PROGRAM STUDI D-IV TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
2024

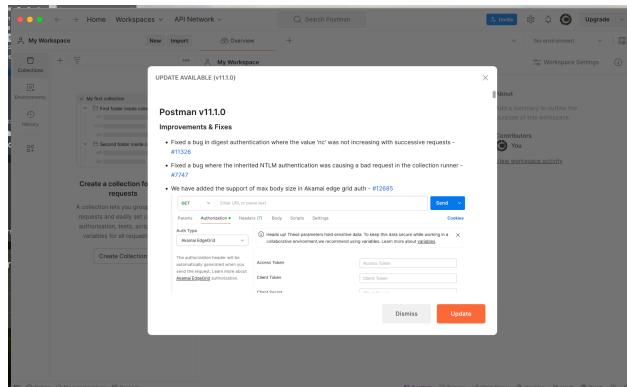


Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

<b>Nama</b>	:	Thoriq Fathurrozi
<b>Nim</b>	:	2241720052
<b>Kelas</b>	:	2H

**Praktikum 1. Membuat RESTful API Register**

1. Sebelum memulai membuat REST API, terlebih dahulu download aplikasi Postman di <https://www.postman.com/downloads>. Aplikasi ini akan digunakan untuk mengerjakan semua tahap praktikum pada Jobsheet ini.

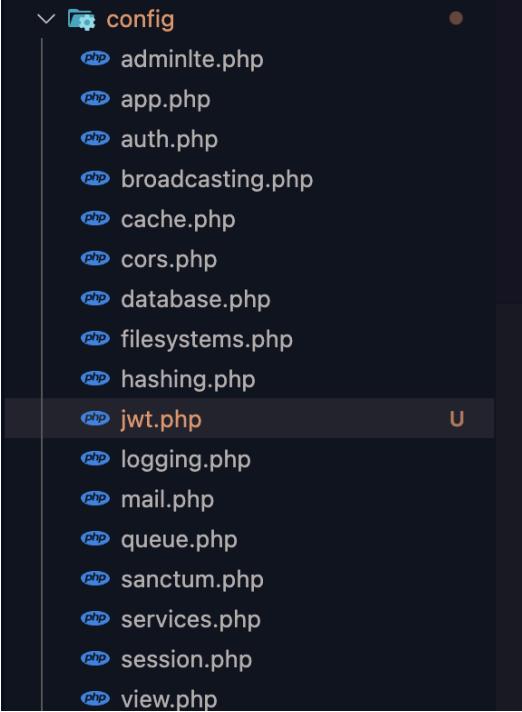


2. Lakukan instalasi JWT dengan mengetikkan perintah berikut:  
composer require tymon/jwt-auth:2.1.1  
Pastikan Anda terkoneksi dengan internet.

```
● frrzy@Thoriqs-MacBook-Pro PWL_POS % composer require tymon/jwt-auth:2.1.1
./composer.json has been updated
Running composer update tymon/jwt-auth
Loading composer repositories with package information
Updating dependencies
Lock file operations: 3 installs, 0 updates, 0 removals
- Locking lcobucci/clock (3.2.0)
- Locking lcobucci/jwt (4.3.0)
- Locking tymon/jwt-auth (2.1.1)
Writing lock file
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

3.	<p>Setelah berhasil menginstall JWT, lanjutkan dengan publish konfigurasi file dengan perintah berikut:</p> <pre>php artisan vendor:publish --provider="Tymon\JWTAuth\Providers\LaravelServiceProvider"</pre> <p>No security vulnerability advisories found. ● frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan vendor:publish --provider="Tymon\JWTAuth\Providers\LaravelServiceProvider" INFO Publishing assets. Copying file [vendor/tymon/jwt-auth/config/config.php] to [config/jwt.php] ..... DONE</p>
4.	<p>Jika perintah di atas berhasil, maka kita akan mendapatkan 1 file baru yaitu config/jwt.php. Pada file ini dapat dilakukan konfigurasi jika memang diperlukan.</p> 
5.	<p>Setelah itu jalankan perintah berikut untuk membuat secret key JWT.</p> <pre>php artisan jwt:secret</pre> <p>Jika berhasil, maka pada file .env akan ditambahkan sebuah baris berisi nilai key JWT_SECRET.</p> <pre>frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan jwt:secret jwt-auth secret [63ryA1IyoXf7o5wn7AdQpG3qAx73ycwt7kou1myXa9zwDuGzP3PvzBswtohlp32e] set successfully.</pre> <p>JWT_SECRET=63ryA1IyoXf7o5wn7AdQpG3qAx73ycwt7kou1myXa9zwDuGzP3PvzBswtohlp32e</p>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

6.	<p>Selanjutnya lakukan konfigurasi guard API. Buka config/auth.php. Ubah bagian 'guards' menjadi seperti berikut.</p> <pre>'guards' =&gt; [     'web' =&gt; [         'driver' =&gt; 'session',         'provider' =&gt; 'users',     ],     'api' =&gt; [         'driver' =&gt; 'jwt', You, 1         'provider' =&gt; 'users',     ], ],</pre>
7.	<p>Kita akan menambahkan kode di model UserModel, ubah kode seperti berikut:</p> <pre>use Illuminate\Database\Eloquent\Relations\BelongsTo; use Tymon\JWTAuth\Contracts\JWTSubject; use Illuminate\Foundation\Auth\User as Authenticatable; You, 30 se  // class UserModel extends Model implements Authenticatable 27 references   0 implementations   You, 31 seconds ago   2 authors (You and others) class UserModel extends Authenticatable implements JWTSubject  {     // use AuthenticatableTrait;     // use HasFactory;     0 references   0 overrides     public function getJWTIdentifier()     {         return \$this-&gt;getKey();     }      0 references   0 overrides     public function getJWTCustomClaims()     {         return [];     }      0 references     protected \$table = 'm_user'; //m_user is the table name     0 references     protected \$primaryKey = 'user_id'; //user_id is the primary key }</pre>
8.	<p>Berikutnya kita akan membuat controller untuk register dengan menjalankan perintah berikut.</p> <pre>php artisan make:controller Api/RegisterController</pre> <p>Jika berhasil maka akan ada tambahan controller pada folder Api dengan nama RegisterController.</p> <pre>jwt-auth secret 1o3TyA11y0X170SWn7AuqPG5qAx7SyCwL7K0U1mYxa9zWDDGZFSFVzbSwt0ntpszeJ set successfully • frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/RegisterController  INFO Controller [app/Http/Controllers/Api/RegisterController.php] created successfully.  • frrzy@Thoriqs-MacBook-Pro PWL_POS %</pre>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

9. Buka file tersebut, dan ubah kode menjadi seperti berikut.

```
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\UserModel;
use Illuminate\Support\Facades\Validator;

0 references | 0 implementations
class RegisterController extends Controller
{
    0 references | 0 overrides
    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 validation 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
        ]);

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

        //Return response JSON user is not created
        return response()→json([
            'success' => false,
        ], 409);
    }
}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

10. Selanjutnya buka routes/api.php, ubah semua kode menjadi seperti berikut.

```
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\Api\RegisterController;|    You, 1 second ago • Uncommited

/*
|--------------------------------------------------------------------------
| API Routes
|--------------------------------------------------------------------------
|
|
| Here is where you can register API routes for your application. These
| routes are loaded by the RouteServiceProvider and all of them will
| be assigned to the "api" middleware group. Make something great!
|
*/

Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
    return $request->user();
});

Route::post('/register', RegisterController::class)->name('register');
```

11. Jika sudah, kita akan melakukan uji coba REST API melalui aplikasi Postman.  
Buka aplikasi Postman, isi URL localhost/PWL\_POS/public/api/register serta method POST. Klik Send.

The screenshot shows the Postman interface with the following details:

- Method:** POST
- URL:** localhost:8000/api/register
- Params:** Key Value Description
- Body:** Key Value Description
- Headers:** (9)
- Test Results:** 422 Unprocessable Content 45 ms 470 B
- Pretty:** JSON
- Response:** (14 lines of JSON validation errors for required fields: username, nama, password, and level\_id.)

Jika berhasil akan muncul error validasi seperti gambar di atas.  
Lakukan percobaan yang sama dan berikan screenshoot hasil percobaan Anda.



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

12

Sekarang kita coba masukkan data. Klik tab Body dan pilih form-data. Isikan key sesuai dengan kolom data, serta isikan data registrasi menggunakan nilai yang Anda inginkan.

The screenshot shows a Postman interface with a 'POST' request to 'localhost:8000/api/register'. The 'Body' tab is selected, showing 'form-data' selected. The data fields are:

Key	Type	Value	Description	... Bulk Edit
username	Text	penggunasatu		
nama	Text	Pengguna 1		
password	Text	12345		
password_confirmation	Text	12345		
level_id	Text	2		

Below the table, the response is shown in 'Pretty' format:

```
1 {
  "success": true,
  "user": {
    "username": "penggunasatu",
    "nama": "Pengguna 1",
    "password": "$2y$12$EL1hKbb2UugRKX/eYIe8d.CpqxFxCnPxmWf3uZamvXMZBxXiqMS",
    "level_id": "2",
    "updated_at": "2024-05-14T18:42:29.000000Z",
    "created_at": "2024-05-14T18:42:29.000000Z",
    "user_id": 13
  }
}
```

Setelah klik tombol Send, jika berhasil maka akan keluar pesan sukses seperti gambar di atas.

Lakukan percobaan yang sama dan berikan screenshoot hasil percobaan Anda.

13.

Lakukan commit perubahan file pada Github.



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

**Praktikum 2. Membuat RESTful API Login**

1. Kita buat file controller dengan nama LoginController.

```
php artisan make:controller Api/LoginController
```

Jika berhasil maka akan ada tambahan controller pada folder Api dengan nama LoginController.

```
● frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/LoginController
INFO Controller [app/Http/Controllers/Api/LoginController.php] created successfully.
```

2. Buka file tersebut, dan ubah kode menjadi seperti berikut.

```
namespace App\Http\Controllers\Api;

use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Validator;

0 references | 0 implementations
class LoginController extends Controller
{
    0 references | 0 overrides
    public function __invoke(Request $request)
    {
        $validator = Validator::make($request->all(), [
            'username' => 'required',
            'password' => 'required'
        ]);

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

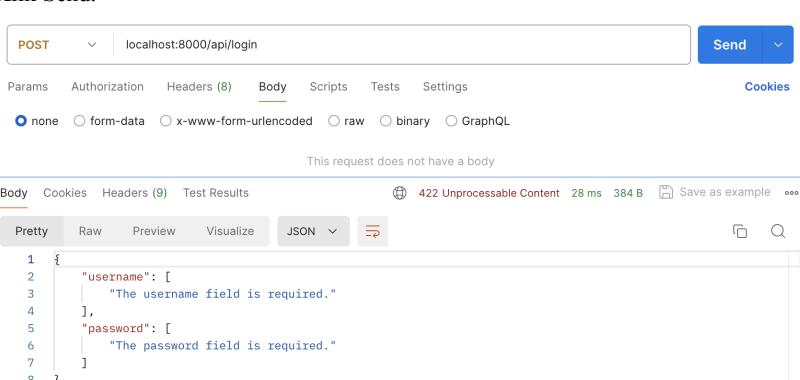
        $credentials = $request->only('username', 'password');

        if (!$token = auth()->guard('api')->attempt($credentials)) {
            return response()->json([
                'success' => false,
                'message' => 'Username atau password salah!'
            ], 200);
        }

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



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

3.	<p>Berikutnya tambahkan route baru pada file api.php yaitu /login dan /user.</p> <pre>22 23 Route::post('/register', RegisterController::class)→name('register'); 24 Route::post('/login', LoginController::class)→name('login'); 25 Route::middleware('auth:api')→get('/user', function (Request \$request) { 26     return \$request→user(); 27 }); 28 You, 4 seconds ago • Uncommitted changes</pre>
4.	<p>Jika sudah, kita akan melakukan uji coba REST API melalui aplikasi Postman. Buka aplikasi Postman, isi URL localhost/PWL_POS/public/api/login serta method POST. Klik Send.</p>  <p>This request does not have a body</p> <p>Body Cookies Headers (9) Test Results 422 Unprocessable Content 28 ms 384 B Save as example ...</p> <p>Pretty Raw Preview Visualize JSON</p> <pre>1 { 2     "username": [ 3         "The username field is required." 4     ], 5     "password": [ 6         "The password field is required." 7     ] 8 }</pre> <p>Jika berhasil akan muncul error validasi seperti gambar di atas. Lakukan percobaan yang sama dan berikan screenshoot hasil percobaan Anda.</p>
5.	<p>Selanjutnya, isikan username dan password sesuai dengan data user yang ada pada database. Klik tab Body dan pilih form-data. Isikan key sesuai dengan kolom data, serta isikan data user. Klik tombol Send, jika berhasil maka akan keluar tampilan seperti berikut. Copy nilai token yang diperoleh pada saat login karena akan diperlukan pada saat logout.</p>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

	<p>POST localhost:8000/api/login</p> <p>Send</p> <p>Params Authorization Headers (9) Body Scripts Tests Settings Cookies</p> <p><input type="radio"/> none <input checked="" type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input type="radio"/> raw <input type="radio"/> binary <input type="radio"/> GraphQL</p> <table border="1"><thead><tr><th>Key</th><th>value</th><th>Description</th><th>... Bulk Edit</th></tr></thead><tbody><tr><td>username</td><td>Text penggunasatu</td><td></td><td></td></tr><tr><td>password</td><td>Text 12345</td><td></td><td></td></tr><tr><td>Key</td><td>Value</td><td>Description</td><td></td></tr></tbody></table> <p>body Cookies Headers (9) Test Results</p> <p>Pretty Raw Preview Visualize JSON <code>≡</code></p> <pre>1 { 2     "success": true, 3     "user": { 4         "user_id": 13, 5         "level_id": 2, 6         "username": "penggunasatu", 7         "nama": "Pengguna 1", 8         "password": "\$2y\$12\$ELlhKbb2UugRKX/eYIe8d.CpqrxMfxCnPxmWF3uZamvXMZBxXiqMS", 9         "created_at": "2024-05-14T18:42:29.000000Z", 10        "updated_at": "2024-05-14T18:42:29.000000Z" 11    }, 12    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9. eyJpc3MiOiJodHRw0i8vbG9jYWxob3N0OjgwMDAvYXBpL2xvZ2luIiwiaWF0IjoxNzE1NzEyODYwLCJleHAiOjE3MTU3MTY0Nj AsImSiZi6MtCxNTcxMjg2MCwianRpIjoidw5oRmJ1WnxQ0XFpOXJRS1sInN1Yii6ijEzIwiCH2Ijo1NDFkZjg4MzRmWI5 OGY3MGVmYTYwYWFlZGVmNDIzNDEzNzAwNjkwyJ9.5L6-gR4AS30el_j0mFjGiNBavz0_a6tmimB1u0X4VG0" 13 }</pre>	Key	value	Description	... Bulk Edit	username	Text penggunasatu			password	Text 12345			Key	Value	Description	
Key	value	Description	... Bulk Edit														
username	Text penggunasatu																
password	Text 12345																
Key	Value	Description															
	<p>Lakukan percobaan yang sama dan berikan screenshot hasil percobaan Anda.</p>																
6.	<p>Lakukan percobaan yang untuk data yang salah dan berikan screenshot hasil percobaan Anda.</p> <p>POST localhost:8000/api/login</p> <p>Send</p> <p>Params Authorization Headers (9) Body Scripts Tests Settings Cookies</p> <p><input type="radio"/> none <input checked="" type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input type="radio"/> raw <input type="radio"/> binary <input type="radio"/> GraphQL</p> <table border="1"><thead><tr><th>Key</th><th>value</th><th>Description</th><th>... Bulk Edit</th></tr></thead><tbody><tr><td>username</td><td>Text penggunasatu</td><td></td><td></td></tr><tr><td>password</td><td>Text 123456</td><td></td><td></td></tr><tr><td>Key</td><td>Value</td><td>Description</td><td></td></tr></tbody></table> <p>Body Cookies Headers (9) Test Results</p> <p>Pretty Raw Preview Visualize JSON <code>≡</code></p> <pre>1 { 2     "success": false, 3     "message": "Username atau password salah!" 4 }</pre>	Key	value	Description	... Bulk Edit	username	Text penggunasatu			password	Text 123456			Key	Value	Description	
Key	value	Description	... Bulk Edit														
username	Text penggunasatu																
password	Text 123456																
Key	Value	Description															



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

7. Coba kembali melakukan login dengan data yang benar. Sekarang mari kita coba menampilkan data user yang sedang login menggunakan URL `localhost/PWL_POS/public/api/user` dan method GET. **Jelaskan hasil dari percobaan tersebut.**

The screenshot shows a Postman request configuration for a GET method to `localhost:8000/api/user`. The Authorization tab is selected, showing a dropdown set to "Bearer Token". Below it, a note says: "The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization." To the right, there's a "Token" field containing a long JWT token: `eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...`. The response section shows a 200 OK status with 37 ms latency and 504 B size. The JSON response body is displayed in pretty print:

```
1 {
2   "user_id": 13,
3   "level_id": 2,
4   "username": "penggunaSatu",
5   "nama": "Pengguna 1",
6   "password": "$2y$12$ELlhKbb2UugRKX/eYIe8d.CpqMfxCnPxmWF3uZamvXMZBxXiqMS",
7   "created_at": "2024-05-14T18:42:29.000000Z",
8   "updated_at": "2024-05-14T18:42:29.000000Z"
9 }
```

kita simpan dulu token yang dikembalikan pada saat login kemudian kita simpan di authorization dengan tipe bearer token kemudian access user. ini bertujuan meng authorization dari token yang telah diberikan dari login. maka saat mengakses api di path tersebut akan mengembalikan credential user

8. Lakukan commit perubahan file pada Github.

**Praktikum 3. Membuat RESTful API Logout**

1. Tambahkan kode berikut pada file `.env`  
`JWT_SHOW_BLACKLIST_EXCEPTION=true`

```
JWT_SECRET=63ryA1IyoXF7o5wn7AdQpG3qAx73ycwt7kou1myXa9zwDuGzP3PvzBswtohlp32e
JWT_SHOW_BLACKLIST_EXCEPTION=true
```

2. Buat Controller baru dengan nama `LogoutController`.  
`php artisan make:controller Api/LogoutController`

```
• frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/LogoutController
INFO Controller [app/Http/Controllers/Api/LogoutController.php] created successfully.
```

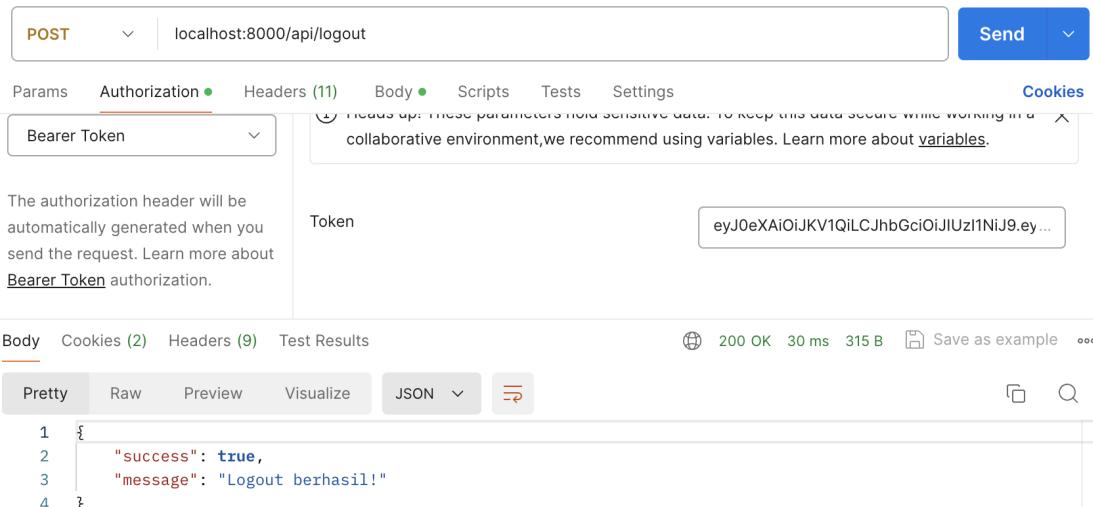


Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

3.	<p>Buka file tersebut dan ubah kode menjadi seperti berikut.</p> <pre>use App\Http\Controllers\Controller; use Illuminate\Http\Request; use Tymon\JWTAuth\Facades\JWTAuth;  0 references   0 implementations class LogoutController extends Controller {     0 references   0 overrides     public function __invoke(Request \$request)     {         \$removeToken = JWTAuth::invalidate(JWTAuth::getToken());          if (\$removeToken) {             return response()-&gt;json([                 'success' =&gt; true,                 'message' =&gt; 'Logout berhasil!'             ], 200);         }     } }</pre>
4.	<p>Lalu kita tambahkan routes pada api.php</p> <pre>  Route::post('/register', RegisterController::class)→name('register');   Route::post('/login', LoginController::class)→name('login');   Route::post('/logout', App\Http\Controllers\Api\LogoutController::class)→name('logout');   Route::middleware('auth:api')→get('/user', function (Request \$request) {       return \$request→user();   });  </pre>
5.	<p>Jika sudah, kita akan melakukan uji coba REST API melalui aplikasi Postman. Buka aplikasi Postman, isi URL localhost/PWL_POS/public/api/logout serta method POST.</p>
6.	<p>Isi token pada tab Authorization, pilih Type yaitu Bearer Token. Isikan token yang didapat saat login. Jika sudah klik Send.</p>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

	
7.	Lakukan commit perubahan file pada Github.
<b>Praktikum 4. Implementasi CRUD dalam RESTful API</b>	
1.	<p>Pertama, buat controller untuk mengolah API pada data level. php artisan make:controller Api/LevelController</p> <pre>frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/LevelController INFO Controller [app/Http/Controllers/Api/LevelController.php] created successfully.</pre>
2.	Setelah berhasil, buka file tersebut dan tuliskan kode seperti berikut yang berisi fungsi CRUDnya.



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
Pengampu: Tim Ajar Pemrograman Web Lanjut  
May 2024

```
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Models\LevelModel;

0 references | 0 implementations
class LevelController extends Controller
{
    0 references | 0 overrides
    public function index()
    {
        return LevelModel::all();
    }

    0 references | 0 overrides
    public function store(Request $request)
    {
        $level = LevelModel::create($request->all());
        return response()->json($level, 201);
    }

    0 references | 0 overrides
    public function show(LevelModel $level)
    {
        return LevelModel::find($level);
    }

    0 references | 0 overrides
    public function update(Request $request, LevelModel $level)
    {
        $level->update($request->all());
        return LevelModel::find($level);
    }

    0 references | 0 overrides
    public function destroy(LevelModel $level)
    {
        $level->delete();
        return response()->json([
            'success' => true,
            'message' => 'Data terhapus'
        ], 200);
    }
}
```

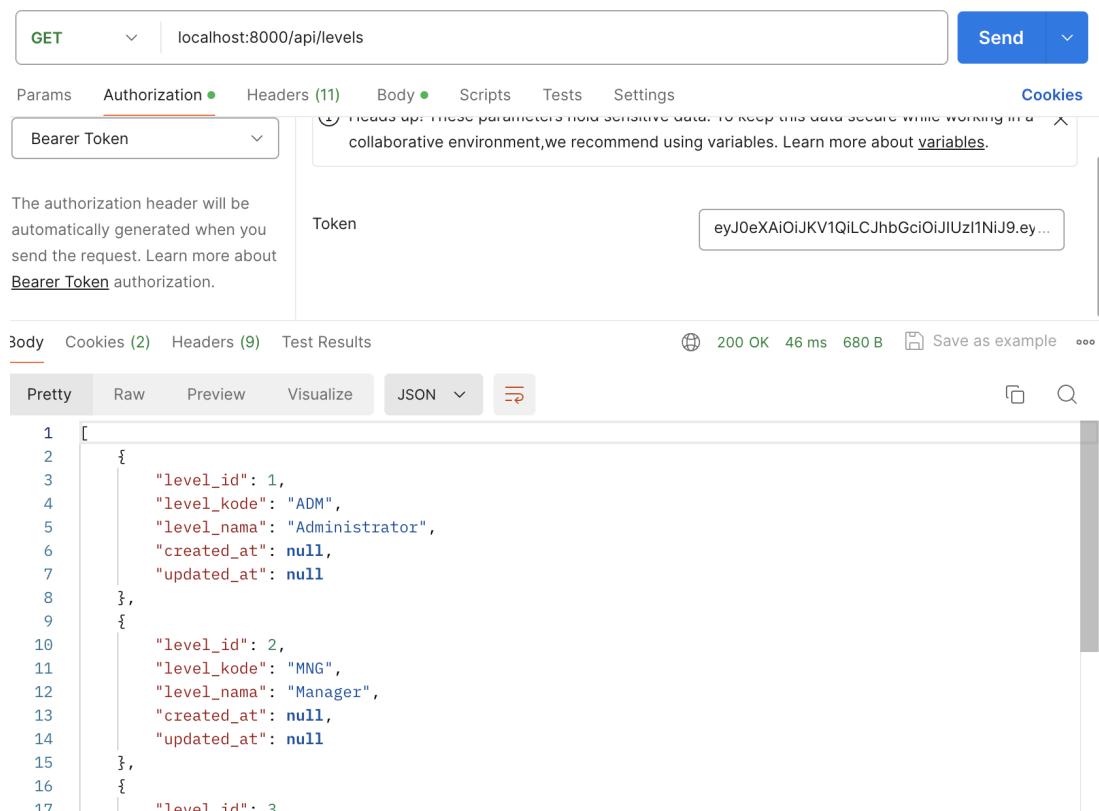


Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

3. Kemudian kita lengkapi routes pada api.php.

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

4. Jika sudah. Lakukan uji coba API mulai dari fungsi untuk menampilkan data. Gunakan URL: localhost/PWL\_POS-main/public/api/levels dan method GET. **Jelaskan dan berikan screenshot hasil percobaan Anda.**



The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Token eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9eyJ...

Body Cookies (2) Headers (9) Test Results

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "level_id": 1,
4     "level_kode": "ADM",
5     "level_nama": "Administrator",
6     "created_at": null,
7     "updated_at": null
8   },
9   {
10    "level_id": 2,
11    "level_kode": "MNG",
12    "level_nama": "Manager",
13    "created_at": null,
14    "updated_at": null
15  },
16  {
17    "level_id": 3,
```

pada saat mengakses api/levels nilai yang dikembalikan adalah levels yang ada di database

5. Kemudian, lakukan percobaan penambahan data dengan URL : localhost/PWL\_POS-main/public/api/levels dan method POST seperti di bawah ini.



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

POST    localhost:8000/api/levels    Send

Params    Authorization ●    Headers (11)    Body ●    Scripts    Tests    Settings    Cookies

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

<input checked="" type="checkbox"/> level_kode	Text	SPV
<input checked="" type="checkbox"/> level_nama	Text	Supervisor
Key	Text	Value
Description		

Body    Cookies (2)    Headers (9)    Test Results    201 Created    31 ms    420 B    Save as example    ...

Pretty    Raw    Preview    Visualize    JSON    ⚡

```
1 {  
2   "level_kode": "SPV",  
3   "level_nama": "Supervisor",  
4   "updated_at": "2024-05-14T19:32:48.000000Z",  
5   "created_at": "2024-05-14T19:32:48.000000Z",  
6   "level_id": 7  
7 }
```

**Jelaskan dan berikan screenshot hasil percobaan Anda.**  
pada saat berhasil maka api/levels akan mengembalikan nilai yang telah kita inputkan dalam bentuk json ini juga menampilkan level\_id yang keberapa

6. Berikutnya lakukan percobaan menampilkan detail data. **Jelaskan dan berikan screenshot hasil percobaan Anda.**

GET    localhost:8000/api/levels/7    Send

Params    Authorization ●    Headers (9)    Body ●    Scripts    Tests    Settings    Cookies

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

<input type="checkbox"/> level_kode	Text	SPV
<input type="checkbox"/> level_nama	Text	Supervisor
Key	Text	Value
Description		

Body    Cookies (2)    Headers (9)    Test Results    200 OK    32 ms    417 B    Save as example    ...

Pretty    Raw    Preview    Visualize    JSON    ⚡

```
1 [  
2   {  
3     "level_id": 7,  
4     "level_kode": "SPV",  
5     "level_nama": "Supervisor",  
6     "created_at": "2024-05-14T19:32:48.000000Z",  
7     "updated_at": "2024-05-14T19:32:48.000000Z"  
8   }  
9 ]
```

Ketika mengakses api/levels/7 atau diakhiri dengan level\_id maka akan menampilkan detail dari level id tersebut



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

7. Jika sudah, kita coba untuk melakukan edit data menggunakan localhost/PWL\_POS-main/public/api/levels/{id} dan method PUT. Isikan data yang ingin diubah pada tab Param.

PUT localhost:8000/api/levels/7?level\_kode=SPR

Params • Authorization • Headers (11) Body • Scripts Tests Settings Cookies

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

<input checked="" type="checkbox"/> level_kode	Text	SPR
Key	Text	Value
Description		

Body Cookies (2) Headers (9) Test Results

Pretty Raw Preview Visualize JSON ↻

1 [  
2 {  
3 "level\_id": 7,  
4 "level\_kode": "SPR",  
5 "level\_nama": "Supervisor",  
6 "created\_at": "2024-05-14T19:32:48.000000Z",  
7 "updated\_at": "2024-05-14T19:37:00.000000Z"  
8 }  
9 ]

200 OK 41 ms 417 B Save as example ...

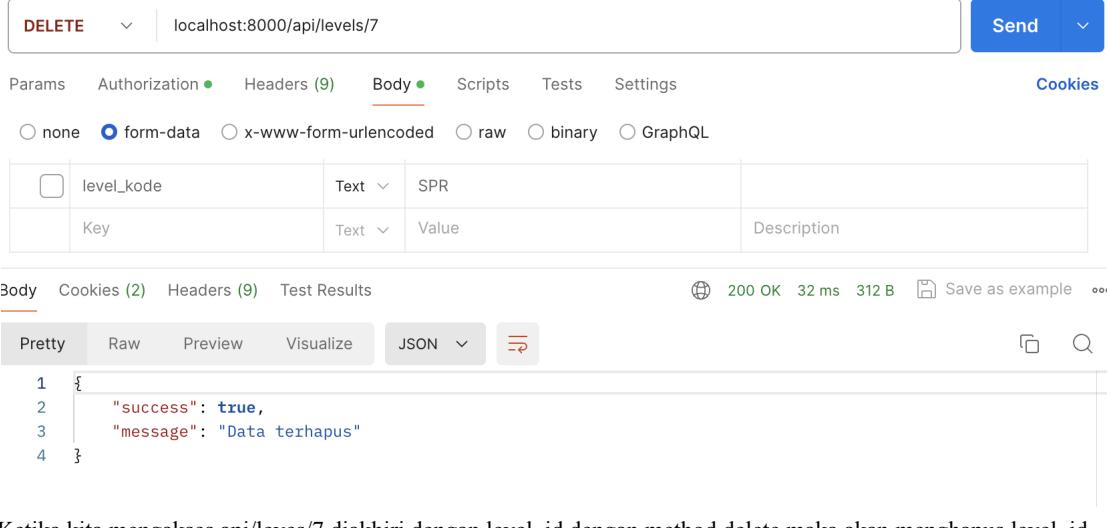
Jelaskan dan berikan screenshot hasil percobaan Anda.

Ketika kita mengakses api/level/7 diakhiri dengan level\_id dan dengan menggunakan kode PUT maka akan mengedit level id tersebut ketika berhasil diedit maka akan mengembalikan nilai level dari level\_id yang telah dirubah

8. Terakhir lakukan percobaan hapus data. **Jelaskan dan berikan screenshot hasil percobaan Anda.**



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

	
9.	Lakukan commit perubahan file pada Github.
<b>Tugas</b>	
1.	Implementasikan CRUD API pada tabel lainnya yaitu tabel m_user, m_kategori, dan m_barang
	membuat controller <pre>frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/BarController INFO Controller [app/Http/Controllers/Api/BarController.php] created successfully.  frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/UserController INFO Controller [app/Http/Controllers/Api/UserController.php] created successfully.  frrzy@Thoriqs-MacBook-Pro PWL_POS % php artisan make:controller Api/KategoriController INFO Controller [app/Http/Controllers/Api/KategoriController.php] created successfully.  frrzy@Thoriqs-MacBook-Pro PWL_POS %</pre>



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

UserController

```
0 references | 0 implementations
class UserController extends Controller
{
    0 references | 0 overrides
    public function index()
    {
        return UserModel::all();
    }

    0 references | 0 overrides
    public function store(Request $request)
    {
        $fields = $request→all();
        if (isset($fields['password'])) {
            $fields['password'] = bcrypt($fields['password']);
        }
        $user = UserModel::create(
            $fields
        );
        return response()→json($user, 201);
    }

    0 references | 0 overrides
    public function show(UserModel $user)
    {
        return UserModel::find($user);
    }

    0 references | 0 overrides
    public function update(Request $request, UserModel $user)
    {
        $fields = $request→all();
        if (isset($fields['password'])) {
            $fields['password'] = bcrypt($fields['password']);
        }
        $user→update($fields);
        return UserModel::find($user);
    }

    0 references | 0 overrides
    public function destroy(UserModel $user)
    {
        $user→delete();
        return response()→json([
            'success' => true,
            'message' => 'Data terhapus'
        ], 200);
    }
}
```



KategoriController

```
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'
        ], 200);
    }
}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
Pengampu: Tim Ajar Pemrograman Web Lanjut  
May 2024

```
BarangController
0 references | 0 implementations
class BarangController extends Controller
{
    0 references | 0 overrides
    public function index()
    {
        return BarangModel::all();
    }

    0 references | 0 overrides
    public function store(Request $request)
    {
        $barang = BarangModel::create($request->all());
        return response()->json($barang, 201);
    }

    0 references | 0 overrides
    public function show(BarangModel $barang)
    {
        return BarangModel::find($barang);
    }

    0 references | 0 overrides
    public function update(Request $request, BarangModel $barang)
    {
        $barang->update($request->all());
        return BarangModel::find($barang);
    }

    0 references | 0 overrides
    public function destroy(BarangModel $barang)
    {
        $barang->delete();
        return response()->json([
            'success' => true,
            'message' => 'Data terhapus'
        ], 200);
    }
}
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
Pengampu: Tim Ajar Pemrograman Web Lanjut  
May 2024

	Route API
	<pre>Route::group([     ['prefix' =&gt; 'users', 'middleware' =&gt; 'auth:api'],     function () {         Route::get('/', [UserController::class, 'index']);         Route::post('/', [UserController::class, 'store']);         Route::get('/{user}', [UserController::class, 'show']);         Route::put('/{user}', [UserController::class, 'update']);         Route::delete('/{user}', [UserController::class, 'destroy']);     } );  Route::group([     ['prefix' =&gt; 'barang', 'middleware' =&gt; 'auth:api'],     function () {         Route::get('/', [BarangController::class, 'index']);         Route::post('/', [BarangController::class, 'store']);         Route::get('/{barang}', [BarangController::class, 'show']);         Route::put('/{barang}', [BarangController::class, 'update']);         Route::delete('/{barang}', [BarangController::class, 'destroy']);     } );  Route::group([     ['prefix' =&gt; 'kategori', 'middleware' =&gt; 'auth:api'],     function () {         Route::get('/', [KategoriController::class, 'index']);         Route::post('/', [KategoriController::class, 'store']);         Route::get('/{kategori}', [KategoriController::class, 'show']);         Route::put('/{kategori}', [KategoriController::class, 'update']);         Route::delete('/{kategori}', [KategoriController::class, 'destroy']);     } );</pre>
	Test
	Users Show All User



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET    localhost:8000/api/users

Send

Params    Authorization ●    Headers (11)    Body ●    Scripts    Tests    Settings    Cookies

Auth Type    Bearer Token

The authorization header will be automatically generated when you

Token    eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.e...

Body    Cookies (2)    Headers (9)    Test Results    200 OK    31 ms    2.36 KB    Save as example

Pretty    Raw    Preview    Visualize    JSON

```
1 [ { "user_id": 1, "level_id": 1, "username": "admin", "nama": "Administrator", "password": "$2y$12$SrK136Danq7GtMULsVKzL0Zp1WhfK35XvTizVbA1Uq80CzHaUY5gK", "created_at": null, "updated_at": null }, { "user_id": 2, "level_id": 2, "username": "manager", "nama": "Manager", "password": "$2y$12$Jj09YLNmc3iG5zaBAB9TOTN0c.3J9xdL.apKhMjKz8hkmNtz79.0", "created_at": null, "updated_at": null } ]
```

Create User



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

POST  Send

Params Authorization  Headers (11) Body  Scripts Tests Settings Cookies

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

<input checked="" type="checkbox"/> level_id	Text	1
<input checked="" type="checkbox"/> username	Text	thoriq28
<input checked="" type="checkbox"/> nama	Text	thoriq
<input checked="" type="checkbox"/> password	Text	12345
Key	Text	Value
		Description

Body Cookies (2) Headers (9) Test Results 201 Created 291 ms 502 B Save as example ...

Pretty Raw Preview Visualize JSON

```
1 {
2     "password": "$2y$12$TsmowHmeoUlK8EphUm1a5.tpwzt5KNVBMNdCGWLXJ3diBsDjYYgqq",
3     "level_id": "1",
4     "username": "thoriq28",
5     "nama": "thoriq",
6     "updated_at": "2024-05-14T21:16:51.00000Z",
7     "created_at": "2024-05-14T21:16:51.00000Z",
8     "user_id": 14
9 }
```

Show Detail User



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET  Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

<input type="checkbox"/> level_nama	Text	Supervisor
<input type="checkbox"/> level_kode	Text	SPR
<input type="checkbox"/> level_id	Text	14
<input type="checkbox"/> username	Text	thoriq28
<input type="checkbox"/> nama	Text	thoriq
<input type="checkbox"/> password	Text	12345

Body Cookies (2) Headers (9) Test Results 200 OK 33 ms 446 B Save as example ...

Pretty Raw Preview Visualize JSON JSON

```
1 [ {  
2   "user_id": 2,  
3   "level_id": 2,  
4   "username": "manager",  
5   "nama": "Manager",  
6   "password": "$2y$12$Jj09YLNmcm3iG5zaBAB9TOTNOc.3J9xdL.apKhMjKz8hkmNtz79.0",  
7   "created_at": null,  
8   "updated_at": null  
9 }  
10 ]  
11 ]
```

Update User



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

PUT localhost:8000/api/users/14?nama=thoriqUpdate Send

Params • Authorization • Headers (11) Body • Scripts Tests Settings Cookies

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

<input type="text"/> level_nama	Text	Supervisor
<input type="text"/> level_kode	Text	SPR
<input type="text"/> level_id	Text	14
<input type="text"/> username	Text	thoriq28
<input checked="" type="checkbox"/> nama	Text	thoriqUpdate
<input type="text"/> password	Text	12345

Body Cookies (2) Headers (9) Test Results 200 OK 42 ms 682 B Save as example

Pretty Raw Preview Visualize JSON

```
8     "password": "$2y$12$TsmowHmeoUlK8EphUm1a5.tpwzt5KNVBMNdCGWLXJ3diBsDjYYgqq",
9     "created_at": null,
10    "updated_at": null
11  },
12  {
13    "user_id": 14,
14    "level_id": 1,
15    "username": "thoriq28",
16    "nama": "thoriqUpdate",
17    "password": "$2y$12$TsmowHmeoUlK8EphUm1a5.tpwzt5KNVBMNdCGWLXJ3diBsDjYYgqq",
18    "created_at": "2024-05-14T21:16:51.00000Z",
19    "updated_at": "2024-05-14T21:20:41.00000Z"
20  ]
```

Delete User



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

	<div style="border: 1px solid #ccc; padding: 10px;"><p>DELETE <span style="margin-left: 10px;">localhost:8000/api/users/14</span> <span style="float: right;">Send</span></p><p>Params Authorization Headers (11) Body Scripts Tests Settings Cookies</p><p><input type="radio"/> none <input checked="" type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input type="radio"/> raw <input type="radio"/> binary <input type="radio"/> GraphQL</p><table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">level_nama</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Supervisor</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">level_kode</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">SPR</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">level_id</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">14</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">username</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">thoriq28</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">nama</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">thoriqUpdate</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/></td><td style="padding: 5px;">password</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">12345</td></tr></table><p>Body Cookies (2) Headers (9) Test Results</p><p>Pretty Raw Preview Visualize JSON <span style="float: right;">Save as example</span></p><pre>1 { 2   "success": true, 3   "message": "Data terhapus" 4 }</pre><p style="text-align: right;">200 OK 37 ms 312 B</p></div>	<input type="checkbox"/>	level_nama	Text	Supervisor	<input type="checkbox"/>	level_kode	Text	SPR	<input type="checkbox"/>	level_id	Text	14	<input type="checkbox"/>	username	Text	thoriq28	<input type="checkbox"/>	nama	Text	thoriqUpdate	<input type="checkbox"/>	password	Text	12345
<input type="checkbox"/>	level_nama	Text	Supervisor																						
<input type="checkbox"/>	level_kode	Text	SPR																						
<input type="checkbox"/>	level_id	Text	14																						
<input type="checkbox"/>	username	Text	thoriq28																						
<input type="checkbox"/>	nama	Text	thoriqUpdate																						
<input type="checkbox"/>	password	Text	12345																						
	Barang																								
	Show all Barang																								



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET localhost:8000/api/barang Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

Body (2) Headers (9) Test Results 200 OK 39 ms 2.24 KB Save as example

Pretty Raw Preview Visualize JSON

<input type="radio"/> none	<input checked="" type="radio"/> form-data	<input type="radio"/> x-www-form-urlencoded	<input type="radio"/> raw	<input type="radio"/> binary	<input type="radio"/> GraphQL
<input type="text"/> level_nama	Text	Supervisor			
<input type="text"/> level_kode	Text	SPR			
<input type="text"/> level_id	Text	14			
<input type="text"/> username	Text	thoriq28			
<input type="text"/> nama	Text	thoriqUpdate			
<input type="text"/> password	Text	12345			

```
1 [ {  
2   "barang_id": 1,  
3   "kategori_id": 2,  
4   "barang_kode": "THF001",  
5   "barang_nama": "Pocari Seared",  
6   "harga_beli": 10000,  
7   "harga_jual": 12000,  
8   "created_at": null,  
9   "updated_at": null  
10 },  
11 {  
12   "barang_id": 2,  
13   "kategori_id": 2,  
14 }
```

Create Barang



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

POST localhost:8000/api/barang Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

<input checked="" type="checkbox"/> kategori_id	Text	2
<input checked="" type="checkbox"/> barang_kode	Text	THF028
<input checked="" type="checkbox"/> barang_nama	Text	Mie Alakadarnya
<input checked="" type="checkbox"/> harga_beli	Text	4000
<input checked="" type="checkbox"/> harga_jual	Text	6000

Body Cookies (2) Headers (9) Test Results 201 Created 39 ms 490 B Save as example ...

Pretty Raw Preview Visualize JSON

```
1 {
2     "kategori_id": "2",
3     "barang_kode": "THF028",
4     "barang_nama": "Mie Alakadarnya",
5     "harga_beli": "4000",
6     "harga_jual": "6000",
7     "updated_at": "2024-05-14T21:26:13.000000Z",
8     "created_at": "2024-05-14T21:26:13.000000Z",
9     "barang_id": 14
10 }
```

Show Detail Barang



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET  Send

Params Authorization Headers (9) Body Scripts Tests Settings Cookies

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

This request does not have a body

Body Cookies (2) Headers (9) Test Results 200 OK 30 ms 428 B Save as example

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "barang_id": 4,
4     "kategori_id": 4,
5     "barang_kode": "THF004",
6     "barang_nama": "Kaos Oblong",
7     "harga_beli": 40000,
8     "harga_jual": 50000,
9     "created_at": null,
10    "updated_at": null
11  }
12 ]
```

Update Barang



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
Pengampu: Tim Ajar Pemrograman Web Lanjut  
May 2024

PUT | localhost:8000/api/barang/14?barang\_nama=Mieeku | Send

Params • Authorization • Headers (11) • Body • Scripts • Tests • Settings • Cookies

Body (1) Authorization (1) Headers (9) Test Results

Body (Pretty, Raw, Preview, Visualize, JSON, ⚙)

200 OK 35 ms 627 B Save as example ⚙

<input type="checkbox"/>	kategori_id	Text	2
<input type="checkbox"/>	barang_kode	Text	THF028
<input checked="" type="checkbox"/>	barang_nama	Text	Mieeku
<input type="checkbox"/>	harga_beli	Text	4000
<input type="checkbox"/>	harga_jual	Text	6000

```
10     "updated_at": null
11   },
12 {
13   "barang_id": 14,
14   "kategori_id": 2,
15   "barang_kode": "THF028",
16   "barang_nama": "Mieeku",
17   "harga_beli": 4000,
18   "harga_jual": 6000,
19   "created_at": "2024-05-14T21:26:13.000000Z",
20   "updated_at": "2024-05-14T21:28:12.000000Z"
21 }
22 ]
```

Delete Barang



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

	<div style="border: 1px solid #ccc; padding: 10px;"><p>DELETE <span style="float: right;">localhost:8000/api/barang/14</span> <span style="background-color: #007bff; color: white; padding: 2px 10px; border-radius: 5px; float: right;">Send</span></p><p>Params Authorization <span style="color: green;">●</span> Headers (11) <span style="color: green;">●</span> <b>Body</b> <span style="color: green;">●</span> Scripts Tests Settings Cookies</p><p><input type="radio"/> none <input checked="" type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input type="radio"/> raw <input type="radio"/> binary <input type="radio"/> GraphQL</p><table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="padding: 5px;"><input type="checkbox"/> kategori_id</td><td style="padding: 5px;">Text <span style="float: right;">▼</span></td><td style="padding: 5px;">2</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/> barang_kode</td><td style="padding: 5px;">Text <span style="float: right;">▼</span></td><td style="padding: 5px;">THF028</td></tr><tr><td style="padding: 5px;"><input checked="" type="checkbox"/> barang_nama</td><td style="padding: 5px;">Text <span style="float: right;">▼</span></td><td style="padding: 5px;">Mieeku</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/> harga_beli</td><td style="padding: 5px;">Text <span style="float: right;">▼</span></td><td style="padding: 5px;">4000</td></tr><tr><td style="padding: 5px;"><input type="checkbox"/> harga_jual</td><td style="padding: 5px;">Text <span style="float: right;">▼</span></td><td style="padding: 5px;">6000</td></tr></table><p>Body Cookies (2) Headers (9) Test Results</p><p>Pretty Raw Preview Visualize JSON <span style="color: red;">✖</span></p><pre>1 { 2   "success": true, 3   "message": "Data terhapus" 4 }</pre><p style="text-align: right;">200 OK 35 ms 312 B <span style="color: green;">Save as example</span> <span style="color: blue;">...</span></p></div>	<input type="checkbox"/> kategori_id	Text <span style="float: right;">▼</span>	2	<input type="checkbox"/> barang_kode	Text <span style="float: right;">▼</span>	THF028	<input checked="" type="checkbox"/> barang_nama	Text <span style="float: right;">▼</span>	Mieeku	<input type="checkbox"/> harga_beli	Text <span style="float: right;">▼</span>	4000	<input type="checkbox"/> harga_jual	Text <span style="float: right;">▼</span>	6000
<input type="checkbox"/> kategori_id	Text <span style="float: right;">▼</span>	2														
<input type="checkbox"/> barang_kode	Text <span style="float: right;">▼</span>	THF028														
<input checked="" type="checkbox"/> barang_nama	Text <span style="float: right;">▼</span>	Mieeku														
<input type="checkbox"/> harga_beli	Text <span style="float: right;">▼</span>	4000														
<input type="checkbox"/> harga_jual	Text <span style="float: right;">▼</span>	6000														
	Kategori															
	Show All Kategori															



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET  Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

<input type="checkbox"/>	kategori_id	Text	2
<input type="checkbox"/>	barang_kode	Text	THF028
<input type="checkbox"/>	barang_nama	Text	Mieeku
<input type="checkbox"/>	harga_beli	Text	4000
<input type="checkbox"/>	harga_jual	Text	6000

Body Cookies (2) Headers (9) Test Results 200 OK 42 ms 1.12 KB Save as example

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "kategori_id": 1,
4     "kategori_kode": "MKN",
5     "kategori_nama": "Makanan",
6     "created_at": null,
7     "updated_at": null
8   },
9   {
10    "kategori_id": 2,
11    "kategori_kode": "MMN",
12    "kategori_nama": "Minuman",
13    "created_at": null,
14    "updated_at": null
15  }
]
```

Create Kategori



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

POST localhost:8000/api/kategori Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

<input type="checkbox"/> narga_deil	Text	4000
<input type="checkbox"/> harga_jual	Text	6000
<input checked="" type="checkbox"/> kategori_kode	Text	ELC
<input checked="" type="checkbox"/> kategori_nama	Text	Elektronik
<input type="checkbox"/>	Text	
Key	Value	Description

Body Cookies (2) Headers (9) Test Results 201 Created 34 ms 430 B Save as example ...

Pretty Raw Preview Visualize JSON

```
1 {  
2   "kategori_kode": "ELC",  
3   "kategori_nama": "Elektronik",  
4   "updated_at": "2024-05-14T21:31:33.000000Z",  
5   "created_at": "2024-05-14T21:31:33.000000Z",  
6   "kategori_id": 12  
7 }
```

Show Detail Kategori



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

GET  Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

<input type="checkbox"/> narga_deil	Text	4000
<input type="checkbox"/> harga_jual	Text	6000
<input checked="" type="checkbox"/> kategori_kode	Text	ELC
<input checked="" type="checkbox"/> kategori_nama	Text	Elektronik
<input type="checkbox"/>	Text	
Key	Text	Value

Body Cookies (2) Headers (9) Test Results 200 OK 42 ms 427 B Save as example

Pretty Raw Preview Visualize JSON

```
1 [  
2 {  
3   "kategori_id": 12,  
4   "kategori_kode": "ELC",  
5   "kategori_nama": "Elektronik",  
6   "created_at": "2024-05-14T21:31:33.000000Z",  
7   "updated_at": "2024-05-14T21:31:33.000000Z"  
8 }  
9 ]
```

Update Kategori



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
Pengampu: Tim Ajar Pemrograman Web Lanjut  
May 2024

The screenshot shows the Postman application interface. At the top, there is a header bar with a 'PUT' method dropdown, a URL field containing 'localhost:8000/api/kategori/12?kategori\_nama=Elektronik Rumah', and a 'Send' button. Below the header, a navigation bar includes 'Params', 'Authorization', 'Headers (11)', 'Body', 'Scripts', 'Tests', and 'Settings'. The 'Body' tab is selected, showing a 'form-data' configuration. There are five fields listed: 'harga\_peli' (text, value: 4UUU), 'harga\_jual' (text, value: 6000), 'kategori\_kode' (text, value: ELC), 'kategori\_nama' (text, checked, value: Elektronik Rumah), and an empty text field. A table below the fields maps 'Key' to 'Value' and includes a 'Description' column. Under the 'Body' tab, there are also sections for 'Cookies (2)', 'Headers (9)', and 'Test Results'. On the right side, the response status is shown as '200 OK' with a duration of '39 ms' and a size of '433 B'. Buttons for 'Save as example' and file operations are also present. At the bottom, a code editor displays the JSON response:

```
1 [  
2   {  
3     "kategori_id": 12,  
4     "kategori.kode": "ELC",  
5     "kategori.nama": "Elektronik Rumah",  
6     "created_at": "2024-05-14T21:31:33.000000Z",  
7     "updated_at": "2024-05-14T21:32:43.000000Z"  
8   }  
9 ]
```



Jurusan Teknologi Informasi Politeknik Negeri Malang  
**Pemrograman Web Lanjut**  
**Mata Kuliah Pemrograman Web Lanjut**  
**Pengampu: Tim Ajar Pemrograman Web Lanjut**  
May 2024

DELETE | localhost:8000/api/kategori/12 | Send

Params Authorization Headers (11) Body Scripts Tests Settings Cookies

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

narga_pelli	Text	4000
harga_jual	Text	6000
kategori_kode	Text	ELC
kategori_nama	Text	Elektronik Rumah
Key	Text	

Body Cookies (2) Headers (9) Test Results 200 OK 36 ms 312 B Save as example

Pretty Raw Preview Visualize JSON

```
1 {  
2   "success": true,  
3   "message": "Data terhapus"  
4 }
```