

CRUD – ADMIN

1. Create_Buku_Admin

POST localhost/Laravel-1/public/api/tambah_buku Send Save

<input checked="" type="checkbox"/>	judul	Anomaly
<input checked="" type="checkbox"/>	penerbit	PT Gramedia
<input checked="" type="checkbox"/>	pengarang	Dian Mu
<input checked="" type="checkbox"/>	foto	nggak ada
<input type="checkbox"/>	nama_anggota	Ninis
<input type="checkbox"/>	alamat	Malang
<input type="checkbox"/>	no_telp	053628175438
	Key	Value
		Description

Body Cookies Headers (11) Test Results Status: 200 OK Time: 392ms Size: 362 B Save Response

Pretty Raw Preview Visualize BETA JSON 🔍

```
1 {
2   "status": 1
3 }
```

2. Read_Buku_Admin

GET localhost/Laravel-1/public/api/buku Send Save

Pretty Raw Preview Visualize BETA JSON 🔍

```
48 {
49   "id": 7,
50   "judul": "Tidak Ada Judulnya",
51   "penerbit": "Penerbit",
52   "pengarang": "Apasih",
53   "foto": "aaaa",
54   "created_at": null,
55   "updated_at": null
56 },
57 {
58   "id": 8,
59   "judul": "aaaaa",
60   "penerbit": "bbbbbb",
61   "pengarang": "cccccc",
62   "foto": "dddddd",
63   "created_at": null,
64   "updated_at": null
65 },
66 {
67   "id": 9,
68   "judul": "Anomaly",
69   "penerbit": "PT Gramedia",
70   "pengarang": "Dian Mu",
71   "foto": "nggak ada",
72   "created_at": null,
73   "updated_at": null
74 }
```

3. Update_Buku_Admin

PUT localhost/Laravel-1/public/api/edit_buku/9 Send Save

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL BETA

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	judul	Anomaly			
<input checked="" type="checkbox"/>	penerbit	PT Gramedia			
<input checked="" type="checkbox"/>	pengarang	Dian_Mu			
<input checked="" type="checkbox"/>	foto	apaya			
	Key	Value	Description		

Body Cookies Headers (11) Test Results Status: 200 OK Time: 3.18s Size: 362 B Save Response

Pretty Raw Preview Visualize BETA JSON ⌵

```
1 {  
2   "status": 1  
3 }
```

4. Delete_Buku_Admin

DELETE localhost/Laravel-1/public/api/hapus_buku/9 Send Save

Pretty Raw Preview Visualize BETA JSON ⌵

```
1 {  
2   "status": 1  
3 }
```

5. Create_Anggota_Admin

POST localhost/Laravel-1/public/api/tambah_anggota Send Save

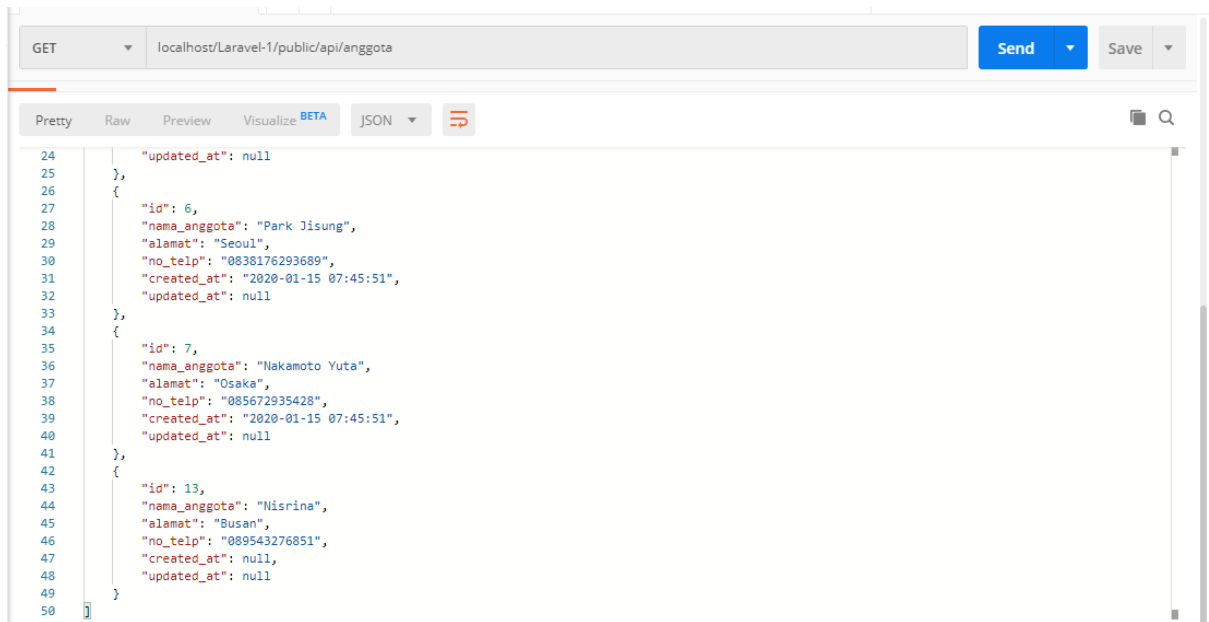
<input checked="" type="checkbox"/>	nama_anggota	Nisrina	
<input checked="" type="checkbox"/>	alamat	Busan	
<input checked="" type="checkbox"/>	no_telp	089543276851	
	Key	Value	Description

Body Cookies Headers (11) Test Results Status: 200 OK Time: 4.37s Size: 362 B Save Response

Pretty Raw Preview Visualize BETA JSON ⌵

```
1 {  
2   "status": 1  
3 }
```

6. Read_Anggota_Admin

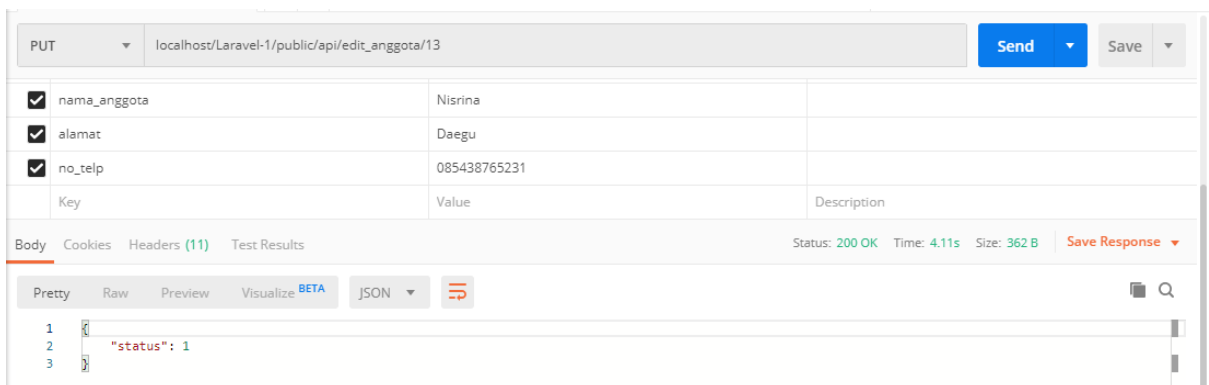


```
GET localhost/Laravel-1/public/api/anggota

Pretty Raw Preview Visualize BETA JSON

24      "updated_at": null
25    },
26    {
27      "id": 6,
28      "nama_anggota": "Park Jisung",
29      "alamat": "Seoul",
30      "no_telp": "0838176293689",
31      "created_at": "2020-01-15 07:45:51",
32      "updated_at": null
33    },
34    {
35      "id": 7,
36      "nama_anggota": "Nakamoto Yuta",
37      "alamat": "Osaka",
38      "no_telp": "085672935428",
39      "created_at": "2020-01-15 07:45:51",
40      "updated_at": null
41    },
42    {
43      "id": 13,
44      "nama_anggota": "Nisrina",
45      "alamat": "Busan",
46      "no_telp": "089543276851",
47      "created_at": null,
48      "updated_at": null
49    }
50  ]
```

7. Update_Anggota_Admin



```
PUT localhost/Laravel-1/public/api/edit_anggota/13

nama_anggota Nisrina
alamat Daegu
no_telp 085438765231
Key Value Description

Body Cookies Headers (11) Test Results Status: 200 OK Time: 4.11s Size: 362 B Save Response

Pretty Raw Preview Visualize BETA JSON

1  {
2    "status": 1
3  }
```

8. Delete_Anggota_Admin



```
DELETE localhost/Laravel-1/public/api/hapus_anggota/13

Pretty Raw Preview Visualize BETA JSON

1  {
2    "status": 1
3  }
```

9. CRUD Bukan Admin



10. Buku Controller

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Buku_Model;
7  use Illuminate\Support\Facades\Validator;
8  use Auth;
9
10 class BukuController extends Controller
11 {
12     public function show(){
13         if(Auth::user()->level=="admin"){
14             $data=Buku_Model::all();
15             return response()->json($data);
16         } else {
17             return response()->json(['Hanya Admin yang Bisa Mengakses']);
18         }
19     }
20
21     public function store(Request $req)
22     {
23         if(Auth::user()->level=="admin"){
24             $validator=Validator::make($req->all(), [
25                 'judul' => 'required',
26                 'penerbit' => 'required',
27                 'pengarang' => 'required',
28                 'foto' => 'required'
29             ]);
30             if($validator->fails()){
31                 return response()->json($validator->errors());
32             }
33
34             $simpan=Buku_Model::insert([
35                 'judul' => $req->judul,
36                 'penerbit' => $req->penerbit,
37                 'pengarang' => $req->pengarang,
38                 'foto' => $req->foto
39             ]);
40             if($simpan){
41                 return response()->json(['status'=>1]);
42             } else {
43                 return response()->json(['status'=>0]);
44             }
45         }
46     }
47 }
```

```

21 public function store(Request $req)
22 {
23     if(Auth::user()->level=="admin"){
24         $validator=Validator::make($req->all(), [
25             'judul' => 'required',
26             'penerbit' => 'required',
27             'pengarang' => 'required',
28             'foto' => 'required'
29         ]);
30         if($validator->fails()){
31             return response()->json($validator->errors());
32         }
33
34         $simpan=Buku_Model::insert([
35             'judul' => $req->judul,
36             'penerbit' => $req->penerbit,
37             'pengarang' => $req->pengarang,
38             'foto' => $req->foto
39         ]);
40         if($simpan){
41             return response()->json(['status'=>1]);
42         } else {
43             return response()->json(['status'=>0]);
44         }
45     } else {
46         return response()->json(['Hanya Admin yang Bisa Mengakses']);
47     }
48 }
49
50 public function update($id, Request $req)
51 {
52     if(Auth::user()->level=="admin"){
53         $validator=Validator::make($req->all(), [
54             'judul' => 'required',
55             'penerbit' => 'required',
56             'pengarang' => 'required',
57             'foto' => 'required'
58         ]);
59         if($validator->fails()){
60             return response()->json($validator->errors());
61         }
62
63         $ubah=Buku_Model::where('id', $id)->update([
64             'judul' => $req->judul,
65             'penerbit' => $req->penerbit,
66             'pengarang' => $req->pengarang,
67             'foto' => $req->foto
68         ]);
69         if($ubah){
70             return response()->json(['status'=>1]);
71         } else {
72             return response()->json(['status'=>0]);
73         }
74     } else {
75         return response()->json(['Hanya Admin yang Bisa Mengakses']);
76     }
77 }

```

```

79     public function destroy($id)
80     {
81         if(Auth::user()->level=="admin"){
82             $hapus=Buku_Model::where('id', $id)->delete();
83             if($hapus){
84                 return response()->json(['status'=>1]);
85             } else {
86                 return response()->json(['status'=>0]);
87             }
88         } else {
89             return response()->json(['Hanya Admin yang Bisa Mengakses']);
90         }
91     }
92 }

```

11. Anggota Controller

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Anggota_Model;
7  use Illuminate\Support\Facades\Validator;
8  use Auth;
9
10 class AnggotaController extends Controller
11 {
12     public function show(){
13         if(Auth::user()->level=="admin"){
14             $data=Anggota_Model::all();
15             return response()->json($data);
16         } else {
17             return response()->json(['Hanya Admin yang Bisa Mengakses']);
18         }
19     }
20
21     public function store(Request $req)
22     {
23         if(Auth::user()->level=="admin"){
24             $validator=Validator::make($req->all(), [
25                 'nama_anggota' => 'required',
26                 'alamat' => 'required',
27                 'no_telp' => 'required'
28             ]);
29             if($validator->fails()){
30                 return response()->json($validator->errors());
31             }
32
33             $simpan=Anggota_Model::insert([
34                 'nama_anggota' => $req->nama_anggota,
35                 'alamat' => $req->alamat,
36                 'no_telp' => $req->no_telp
37             ]);
38             if($simpan){
39                 return response()->json(['status'=>1]);
40             } else {
41                 return response()->json(['status'=>0]);
42             }
43         } else {
44             return response()->json(['Hanya Admin yang Bisa Mengakses']);
45         }
46     }
47 }

```

```

46 }
47
48 public function update($id, Request $req)
49 {
50     if(Auth::user()->level=="admin"){
51         $validator=Validator::make($req->all(), [
52             'nama_anggota' => 'required',
53             'alamat' => 'required',
54             'no_telp' => 'required'
55         ]);
56         if($validator->fails()){
57             return response()->json($validator->errors());
58         }
59
60         $ubah=Anggota_Model::where('id', $id)->update([
61             'nama_anggota' => $req->nama_anggota,
62             'alamat' => $req->alamat,
63             'no_telp' => $req->no_telp
64         ]);
65         if($ubah){
66             return response()->json(['status'=>1]);
67         } else {
68             return response()->json(['status'=>0]);
69         }
70     } else {
71         return response()->json(['Hanya Admin yang Bisa Mengakses']);
72     }
73 }
74 }
75
76 public function destroy($id)
77 {
78     if(Auth::user()->level=="admin"){
79         $hapus=Anggota_Model::where('id', $id)->delete();
80         if($hapus){
81             return response()->json(['status'=>1]);
82         } else {
83             return response()->json(['status'=>0]);
84         }
85     } else {
86         return response()->json(['Hanya Admin yang Bisa Mengakses']);
87     }
88 }
89 }

```

12.API

```

26 Route::post('tambah_buku', 'BukuController@store')->middleware('jwt.verify');
27 Route::put('edit_buku/{id}', 'BukuController@update')->middleware('jwt.verify');
28 Route::delete('hapus_buku/{id}', 'BukuController@destroy')->middleware('jwt.verify');
29 Route::get('buku', 'BukuController@show')->middleware('jwt.verify');
30
31 Route::post('tambah_anggota', 'AnggotaController@store')->middleware('jwt.verify');
32 Route::put('edit_anggota/{id}', 'AnggotaController@update')->middleware('jwt.verify');
33 Route::delete('hapus_anggota/{id}', 'AnggotaController@destroy')->middleware('jwt.verify');
34 Route::get('anggota', 'AnggotaController@show')->middleware('jwt.verify');

```

Nama : Nisrina Izdiyar Ardana Putri
Kelas : XI RPL 6 (Laravel)
Absen : 28