

## CRUD Buku dan Anggota

### 1. 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('tampil_buku', 'BukuController@index')->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('tampil_anggota', 'AnggotaController@index')->middleware('jwt.verify');
```

### 2. Model Buku

```
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class ModelBuku extends Model
8  {
9      //
10     protected $table = "buku";
11     protected $primaryKey = "id";
12 }
```

### 3. Controller Buku

```
3 namespace App\Http\Controllers;
4 use App\ModelBuku;
5 use Illuminate\Http\Request;
6 use Validator;
7 use JWTAuth;
8 use Auth;
9
10 class BukuController extends Controller
11 {
12     /**
13      * Display a listing of the resource.
14      *
15      * @return \Illuminate\Http\Response
16      */
17     public function index()
18     {
19         //
20         if(Auth::user()->level=="admin") {
21             $data=ModelBuku::all();
22             return response()->json($data);
23         }
24         else {
25             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
26         }
27     }
28 }
```

```
46     public function store(Request $request)
47     {
48         //
49         if(Auth::user()->level=="admin") {
50             $validator=Validator::make($request->all(),[
51                 'judul'=>'required',
52                 'penerbit'=>'required',
53                 'pengarang'=>'required',
54                 'foto'=>'required'
55             ]);
56             if($validator->fails()){
57                 return response()->json($validator->errors()->toJson(),400);
58             }
59             else {
60                 $insert=ModelBuku::insert([
61                     'judul'=>$request->judul,
62                     'penerbit'=>$request->penerbit,
63                     'pengarang'=>$request->pengarang,
64                     'foto'=>$request->foto
65                 ]);
66                 if($insert){
67                     $status="sukses";
68                 }
69                 else {
70                     $status="gagal";
71                 }
72                 return response()->json(compact('status'));
73             }
74         }
75         else {
76             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
77         }
78     }
```

```

110     public function update(Request $request, $id)
111     {
112         //
113         if(Auth::user()->level=="admin") {
114             $validator=Validator::make($request->all(),
115             [
116                 'judul'=>'required',
117                 'penerbit'=>'required',
118                 'pengarang'=>'required',
119                 'foto'=>'required'
120             ]
121         );
122         if($validator->fails()) {
123             return response()->json($validator->errors());
124         }
125         $ubah=ModelBuku::where('id', $id)->update([
126             'judul'=>$request->judul,
127             'penerbit'=>$request->penerbit,
128             'pengarang'=>$request->pengarang,
129             'foto'=>$request->foto
130         ]);
131         if($ubah){
132             return response()->json(['status'=>1]);
133         }
134         else {
135             return response()->json(['status'=>0]);
136         }
137     }
138     else {
139         echo "Mohon maaf, data hanya bisa diakses oleh Admin";
140     }
141 }
142 }

```

```

150     public function destroy($id)
151     {
152         //
153         if(Auth::user()->level=="admin") {
154             $hapus=ModelBuku::where('id',$id)->delete();
155             if($hapus){
156                 return response()->json(['status'=>1]);
157             }
158             else {
159                 return response()->json(['status'=>0]);
160             }
161         }
162         else {
163             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
164         }
165     }


```

#### 4. Hasil tambah buku

POST ▼ http://localhost/Laravel4/public/api/tambah\_buku

<input checked="" type="checkbox"/>	judul	Magic Shop
<input checked="" type="checkbox"/>	penerbit	Text <span>▼</span> BTSMEDIA
<input checked="" type="checkbox"/>	pengarang	fara
<input checked="" type="checkbox"/>	foto	-
<input type="checkbox"/>	nama_anggota	Dia
<input type="checkbox"/>	alamat	dimana ya ?
<input type="checkbox"/>	no_hp	1234567890
	Key	Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ▼ 

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


#### 5. Hasil edit buku

PUT ▼ http://localhost/Laravel4/public/api/edit\_buku/7

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ Gra

	KEY	VALUE
<input checked="" type="checkbox"/>	judul	Young Forever
<input checked="" type="checkbox"/>	penerbit	BTS MEDIA
<input checked="" type="checkbox"/>	pengarang	Fara Nisha
<input checked="" type="checkbox"/>	foto	-
	Key	Value


Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ▼ 

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

## 6. Hasil hapus buku

DELETE ▼ http://localhost/Laravel4/public/api/hapus\_buku/7


Pretty Raw Preview Visualize BETA JSON ▼ 

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

## 7. Hasil tampil buku

GET ▼ http://localhost/Laravel4/public/api/tampil\_buku|

body cookies headers (1) response

Pretty Raw Preview Visualize BETA JSON ▼ 

```
11 {  
12   "id": 2,  
13   "judul": "Bintang",  
14   "penerbit": "Butterfly Media",  
15   "pengarang": "Fara Nisha",  
16   "foto": "-",  
17   "created_at": "2020-01-15 07:51:44",  
18   "updated_at": null  
19 },  
20 {  
21   "id": 3,  
22   "judul": "Matahari",  
23   "penerbit": "Butterfly Media",  
24   "pengarang": "Fara Nisha",  
25   "foto": "-",  
26   "created_at": "2020-01-15 07:51:44",  
27   "updated_at": null  
28 },  
29 {  
30   "id": 4,  
31   "judul": "Purple",  
32   "penerbit": "Butterfly Media",  
33   "pengarang": "Fara Nisha",  
34   "foto": "-",  
35   "created_at": "2020-01-15 07:51:44",  
36   "updated_at": null  
37 },
```

## 8. Bukan admin Buku

POST http://localhost/Laravel4/public/api/tambah\_buku

<input checked="" type="checkbox"/>	judul	Text ▼	Wings
<input checked="" type="checkbox"/>	penerbit		BTSMEDIA
<input checked="" type="checkbox"/>	pengarang		Fara Nisha Sukma Gustika
<input checked="" type="checkbox"/>	foto		-
<input type="checkbox"/>	nama_anggota		Dia
<input type="checkbox"/>	alamat		dimana ya ?
<input type="checkbox"/>	no_hp		1234567890
	Key		Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA HTML ▼

```
1  Mohon maaf, data hanya bisa diakses oleh Admin
```

## 9. Model Anggota

```
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class ModelAnggota extends Model
8  {
9      //
10     protected $table = "anggota";
11     protected $primaryKey = "id";
12 }
```

## 10. Controller Anggota

```
3 namespace App\Http\Controllers;
4 use App\ModelAnggota;
5 use Illuminate\Http\Request;
6 use Validator;
7 use JWTAuth;
8 use Auth;
9
10 class AnggotaController extends Controller
11 {
12     /**
13      * Display a listing of the resource.
14      *
15      * @return \Illuminate\Http\Response
16      */
17     public function index()
18     {
19         //
20         if(Auth::user()->level=="admin") {
21             $data=ModelAnggota::all();
22             return response()->json($data);
23         }
24         else {
25             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
26         }
27     }
28 }
```



```
45     public function store(Request $request)
46     {
47         //
48         if(Auth::user()->level=="admin") {
49             $validator=Validator::make($request->all(),[
50                 'nama_anggota'=>'required',
51                 'alamat'=>'required',
52                 'no_hp'=>'required'
53             ]);
54             if($validator->fails()){
55                 return response()->json($validator->errors()->toJson(),400);
56             }
57             else {
58                 $insert=ModelAnggota::insert([
59                     'nama_anggota'=>$request->nama_anggota,
60                     'alamat'=>$request->alamat,
61                     'no_hp'=>$request->no_hp
62                 ]);
63                 if($insert){
64                     $status="sukses";
65                 }
66                 else {
67                     $status="gagal";
68                 }
69                 return response()->json(compact('status'));
70             }
71         }
72         else {
73             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
74         }
75     }
```

```

107     public function update(Request $request, $id)
108     {
109         //
110         if(Auth::user()->level=="admin") {
111             $validator=Validator::make($request->all(),
112             [
113                 'nama_anggota'=>'required',
114                 'alamat'=>'required',
115                 'no_hp'=>'required'
116             ]
117         );
118         if($validator->fails()) {
119             return response()->json($validator->errors());
120         }
121         $ubah=ModelAnggota::where('id', $id)->update([
122             'nama_anggota'=>$request->nama_anggota,
123             'alamat'=>$request->alamat,
124             'no_hp'=>$request->no_hp
125         ]);
126         if($ubah){
127             return response()->json(['status'=>1]);
128         }
129         else {
130             return response()->json(['status'=>0]);
131         }
132     }
133     else {
134         echo "Mohon maaf, data hanya bisa diakses oleh Admin";
135     }
136 }
137 }

```

```

145     public function destroy($id)
146     {
147         //
148         if(Auth::user()->level=="admin") {
149             $hapus=ModelAnggota::where('id',$id)->delete();
150             if($hapus){
151                 return response()->json(['status'=>1]);
152             }
153             else {
154                 return response()->json(['status'=>0]);
155             }
156         }
157         else {
158             echo "Mohon maaf, data hanya bisa diakses oleh Admin";
159         }
160     }

```

## 11. Hasil tambah Anggota

POST ▼ http://localhost/Laravel4/public/api/tambah\_anggota

<input checked="" type="checkbox"/>	nama_anggota	Siapa aja boleh
<input checked="" type="checkbox"/>	alamat	Text <span>▼</span> gatau
<input checked="" type="checkbox"/>	no_hp	1234567890
	Key	Value

Body ▼ Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ▼ ≡

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

## 12. Hasil edit Anggota

PUT ▼ http://localhost/Laravel4/public/api/edit\_anggota/45

Params ● Authorization Headers (10) **Body ●** Pre-request Script

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ G

	KEY	VALUE
<input checked="" type="checkbox"/>	nama_anggota	Gak kenal
<input checked="" type="checkbox"/>	alamat	dimana ?
<input checked="" type="checkbox"/>	no_hp	12345678
	Key	Value

Body ▼ Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ▼ ≡

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

### 13. Hasil hapus Anggota

DELETE ▼ http://localhost/Laravel4/public/api/hapus\_anggota/45

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ↺

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

### 14. Hasil tampil Anggota

GET ▼ http://localhost/Laravel4/public/api/tampil\_anggota

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA JSON ↺

```
2 {  
3   "id": 38,  
4   "nama_anggota": "Kim Taehyung",  
5   "alamat": "Daegu",  
6   "no_hp": "0987654321",  
7   "created_at": "2020-01-15 08:04:48",  
8   "updated_at": null  
9 },  
10 {  
11   "id": 39,  
12   "nama_anggota": "Fara Nisha",  
13   "alamat": "Daegu",  
14   "no_hp": "0987612345",  
15   "created_at": "2020-01-15 08:04:48",  
16   "updated_at": null  
17 },  
18 {  
19   "id": 40,  
20   "nama_anggota": "Jeon Jungkook",  
21   "alamat": "Seoul",  
22   "no_hp": "0981234567",  
23   "created_at": "2020-01-15 08:04:48",  
24   "updated_at": null  
25 },  
26 {  
27   "id": 41,  
28   "nama_anggota": "Park Jimin",  
29   "alamat": "Seoul",
```

## 15. Bukan admin Anggota

POST

http://localhost/Laravel4/public/api/tambah\_anggota

<input checked="" type="checkbox"/>	nama_anggota		Dia
<input checked="" type="checkbox"/>	alamat	Text	dimana ya ?
<input checked="" type="checkbox"/>	no_hp		1234567890
	Key		Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize BETA

HTML

```
1  Mohon maaf, data hanya bisa diakses oleh Admin
```

**Nama** : Fara Nisha Sukma Gustika

**Kelas** : Laravel