29 / XI Laravel

Tugas 5 CRUD Master Laravel dgn Authentication JWT

1. Controller

Anggota

```
* anggotacontroller.php ×
app > Http > Controllers > 💝 anggotacontroller.php
        namespace App\Http\Controllers;
        use Illuminate\Http\Request;
        use Auth;
        use\App\Anggota_model;
        use Validator;
        class anggotacontroller extends Controller
               public function store(Request $req)
                    $validator=Validator::make($req->all(),
                          'alamat'=>'required',
'telp'=>'required'
                    if($validator->fails()){
                         return Response()->json($validator->errors());
                    $simpan=Anggota_model::create([
                          'nama_anggota'=>$req->nama_anggota,
'alamat'=>$req->alamat,
                          'telp'=>$req->telp
                    if($simpan){
                         return Response()->json(['status'=>1, 'message'=>"Data Berhasil Ditambahkan!"]);
                         return Response()->json(['status'=>0]);
            public function tampil anggota()
                $data_anggota=Anggota_model::count();
$data_anggota1=Anggota_model::all();
return Response()->json(['count'=> $data_anggota, 'anggota'=> $data_anggota1, 'status'=>1]);
           public function update($id,Request $req)
                $validator=Validator::make($req->all(),
                      'nama_anggota'=>'required',
                     'alamat'=>'required',
'telp'=>'required'
               ]);
if($validator->fails()){
    return Response()->json($validator->errors());
    return Response()->json($validator->errors());
                $\text{Subah=Anggota_model::where('id',$id)->update([
   'nama_anggota'=>$req->nama_anggota,
   'alamat'=>$req->lamat,
   'telp'=>$req->telp
                    return Response()->json(['status'=>0]);
                $hapus=Anggota_model::where('id',$id)->delete();
```

```
if($hapus){
    return Response()->json(['status'=>1, 'message'=>"Data Berhasil Dihapus!"]);
} else {
    return Response()->json(['status'=>0]);
}

}

}

}

7

}
```

Buku

```
anggotacontroller.php
                        🦬 bukucontroller.php ×
app > Http > Controllers > ♥ bukucontroller.php
      namespace App\Http\Controllers;
      use Illuminate\Http\Request;
      use\App\Buku_model;
      use Illuminate\Support\Facades\Validator;
      class bukucontroller extends Controller
           public function index()
 12
               if(Auth::user()->level=="admin"){
                   $dt_buku=Buku_model::get();
                   return response()->json($dt_buku);
               else {
                   return response()->json(['status'=>'anda bukan admin']);
           public function store(Request $req)
               $validator=Validator::make($req->all(),
                   'judul'=>'required',
                   'pengarang'=>'required',
                   'foto'=>'required'
               if($validator->fails()){
                   return Response()->json($validator->errors());
               $simpan=Buku_model::create([
```

```
judul'=>$req->judul,
        'penerbit'=>$req->penerbit,
        'pengarang'=>$req->pengarang
    if($simpan){
       return Response()->json(['status'=>1, 'message'=>"Data Berhasil Ditambahkan!"]);
        return Response()->json(['status'=>0]);
public function tampil_buku()
    $data_buku=Buku_model::count();
    $data_buku1=Buku_model::all();
    return Response()->json(['count'=> $data_buku, 'buku'=> $data_buku1, 'status'=>1]);
public function update($id,Request $req)
    $validator=Validator::make($req->all(),
        'penerbit'=>'required',
        'pengarang'=>'required',
'foto'=>'required'
    if($validator->fails()){
        return Response()->json($validator->errors());
    $ubah=Buku_model::where('id',$id)->update([
        'judul'=>$req->judul,
        'penerbit'=>$req->penerbit,
         'pengarang'=>$req->pengarang
    if($ubah){
          return Response()->json(['status'=>1, 'message'=>"Data Berhasil Diubah!"]);
           return Response()->json(['status'=>0]);
  public function destroy($id)
       $hapus=Buku_model::where('id',$id)->delete();
      if($hapus){
          return Response()->json(['status'=>1, 'message'=>"Data Berhasil Dihapus!"]);
           return Response()->json(['status'=>0]);
```

2. Model

Anggota

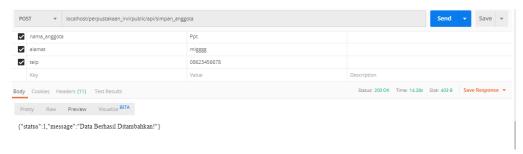
Buku

```
manggotacontroller.php
app > manggotacontroller.php
comparison
app > manggotacontroller.php
app >
```

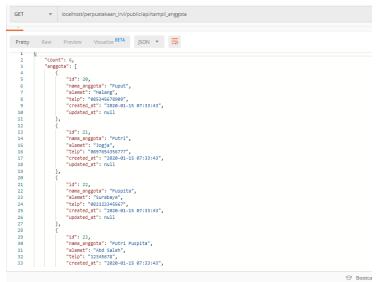
3. Testing di Postman

- Anggota

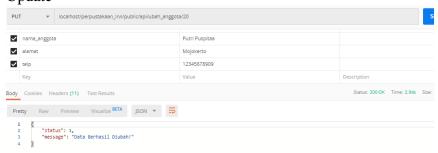
Create



Read



Update

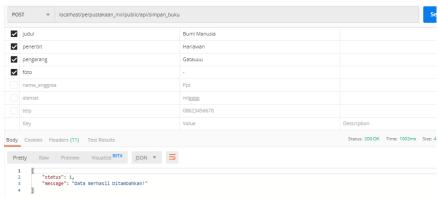


Delete

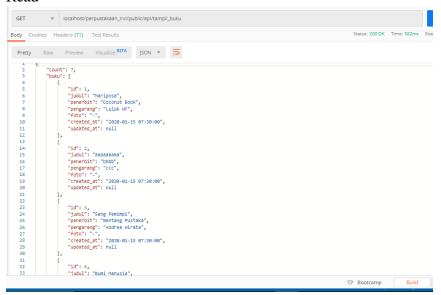


- Buku

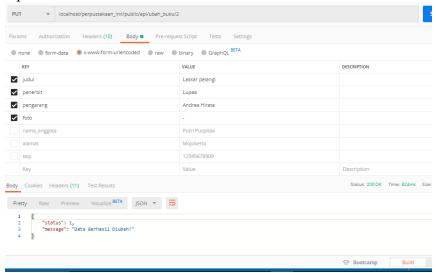
Create



Read



Update



Delete

