Nama: Muhammad Bagas Ramadhan

Absen: 19

Controller Anggota

```
namespace App\Http\Controllers;
use Auth;
use App\Buku_model;
use App\User;
use App\Anggota_model;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
use Tymon\JWTAuth\Exceptions\JWTException;
class anggota extends Controller
     public function simpan(Request $req)
          $validator=Validator::make($req->all(),
              'nama_anggota'=>'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
              $pesan['message'] = 'berhasil tambah anggota broh';
return Response()->json($pesan);
             $pesan['message'] = 'gagal tambah anggota broh';
return Response()->json($pesan);
     public function update($id, Request $req)
          $validator=Validator::make($req->all(),
              'nama_anggota'=>'required',
              'alamat'=>'required',
              'telp'=>'required'
              return Response()->json($validator->errors());
          $ubah=Anggota_model::where('id',$id)->update([
              'nama_anggota'=>$req->nama_anggota,
```

```
'alamat'=>$req->alamat,
   'telf'=>$req->telf'
]);
if($ubah){
   $data['message'] = 'Anggota Berhasil diubah';
   return Response()->json($data);
} else {
   $data['message'] = 'Anggota gagal diubah';
   return Response()->json($data);
}

public function destroy($id)
{
   $hapus=Anggota_model::where('id',$id)->delete();
   if($hapus){
    $pesan['message'] = 'Anggota berhasil dihapus';
    return Response()->json($pesan);
}else(
   $pesan['message'] = 'Anggota gagal dihapus';
   return Response()->json($pesan);
}
}
```

Controller Buku

```
class buku extends Controller
     public function store(Request $req)
           $validator=Validator::make($req->all(),
               'judul'=>'required',
               'penerbit'=>'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,
'foto'=>$req->foto
          if($simpan){
   $pesan['message'] = 'berhasil tambah buku broh';
   return Response()->json($pesan);
               $pesan['message'] = 'gagal tambah buku broh';
return Response()->json($pesan);
     public function update($id, Request $req)
          $validator=Validator::make($req->all(),
               'pengarang'=>'required',
'penerbit'=>'required',
               'foto'=>'required',
           if($validator->fails()){
                return Response()->json($validator->errors());
          $ubah=Buku_model::where('id',$id)->update([
   'judul'=>$req->judul,
   'pengarang'=>$req->pengarang,
   'penerbit'=>$req->penerbit,
```

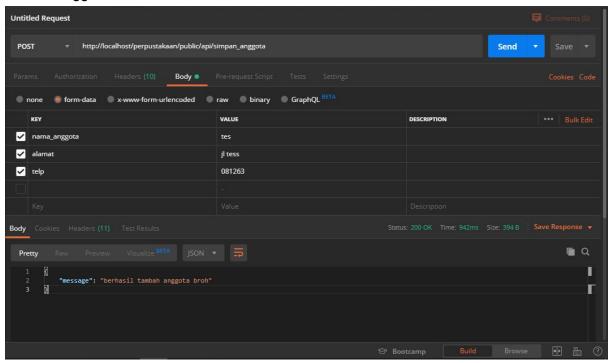
```
if($ubah){
       $data['message'] = 'Buku Berhasil diubah';
       return Response()->json($data);
       $data['message'] = 'Buku gagal diubah';
       return Response()->json($data);
public function destroy($id)
   $hapus=Buku_model::where('id',$id)->delete();
       $pesan['message'] = 'Buku berhasil dihapus';
       return Response()->json($pesan);
       $pesan['message'] = 'Buku gagal dihapus';
       return Response()->json($pesan);
    if(Auth::User()->level=="admin"){
       $dt_buku=Buku_model::get();
       return Response()->json($dt_buku);
       $pesan['message'] = 'kamu buka admin bro maap';
       return Response()->json($pesan);
```

Model Anggota

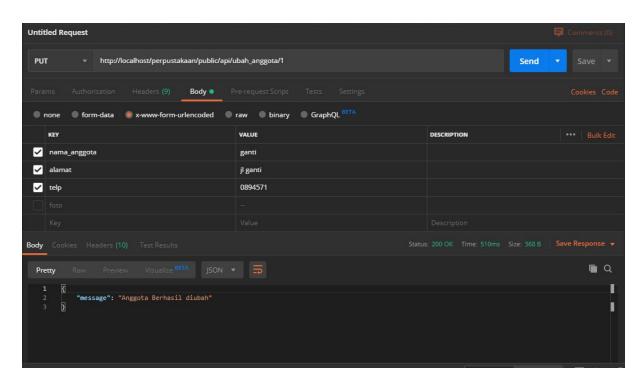
Model Buku

Test postman

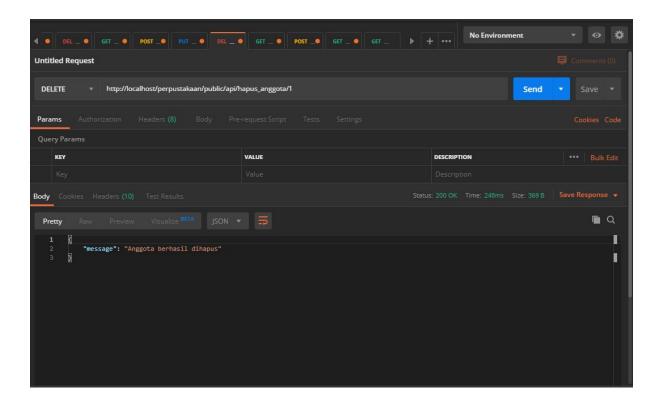
1. tambah anggota



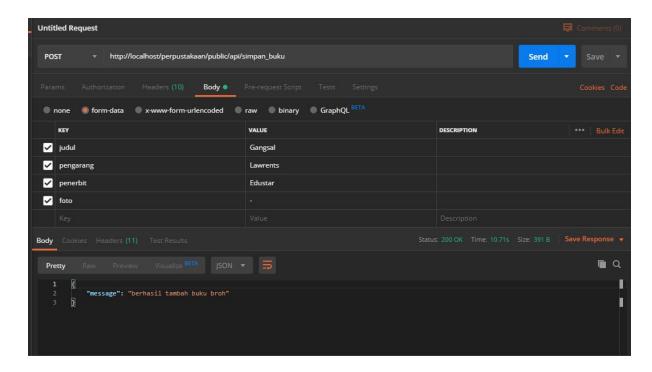
2. edit anggota



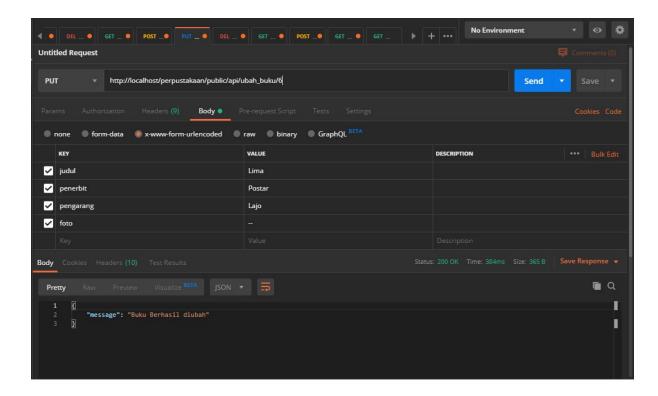
3. hapus anggota



4. tambah buku



5. edit buku



6. hapus buku

