```php
<?php

namespace App\Http\Controllers;

use App\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
use JWTAuth;
use Tymon\JWTAuth\Exceptions\JWTException;
use App\Modeladmin;

class UserController extends Controller
{
    public function login(Request $request)
    {
        $credentials = $request->only('username', 'password');

        try {
            if (! $token = JWTAuth::attempt($credentials)) {
                return response()->json(['error' => 'invalid_credentials'], 400);
            }
        } catch (JWTException $e) {
            return response()->json(['error' => 'could_not_create_token'], 500);
        }

        return response()->json(compact('token'));
    }
//-----------------------------------------------------------------------
    public function register(Request $request)
    {
        $validator = Validator::make($request->all(), [
            'nama_admin' => 'required|string|max:255',
            'role' => 'required|string|max:255',
            'username' => 'required|string|max:255',
            'password' => 'required|string',
        ]);
```

```php
// EVENT \\
Route::get('/event', 'event@tampil')->middleware('jwt.verify');
Route::put('/event/{id}', 'event@update')->middleware('jwt.verify');
Route::delete('/event/{id}', 'event@delete')->middleware('jwt.verify');
Route::post('/event', 'event@store')->middleware('jwt.verify');

// ADMIN \\
Route::get('/admin', 'admin@login');
Route::get('/tampiladmin', 'admin@tampiladmin')->middleware('jwt_ok');
Route::post('/admin', 'admin@register');
Route::put('/admin/{id}', 'admin@update');
Route::delete('/admin/{id}', 'admin@delete');

// TRANSAKSI \\
Route::get('/tampil_transaksi', 'transaksi@tampil_transaksi')->middleware('jwt.verify');
Route::put('/transaksi/{id}', 'transaksi@update')->middleware('jwt.verify');
Route::delete('/transaksi/{id}', 'transaksi@delete')->middleware('jwt.verify');
Route::post('/transaksi', 'transaksi@store')->middleware('jwt.verify');
```

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Validator;
use App\Modelevent;

class event extends Controller
{
        public function store(Request $req)
    {
        $validator = Validator::make($req->all(),
        [
            'nama_event' => 'required',
            'tgl_pelaksana' => 'required',
            'info'=> 'required',
            'harga' => 'required',
            'jumlah_tiket' => 'required',
        ]);

        if($validator->fails()){
            return Response()->json($validator->errors());
        }

        $event = Modelevent::create([
            'nama_event' => $req->nama_event,
            'tgl_pelaksana' => $req->tgl_pelaksana,
            'info'=> $req->info,
            'harga'=> $req->harga,
            'jumlah_tiket' => $req->jumlah_tiket,
        ]);
        if($event){
            return Response()->json(['status'=>1,'message'=>'Data event berhasil ditambahkan!']);
        }
        else{
            return Response()->json(['status'=>0,'message'=>'Data event tidak berhasil ditambahkan!']);
        }
    }
}
```

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Modeladmin extends Model
{
    protected $table="admin";
    protected $primarykey="id";
    protected $fillable = [
        'nama_admin', 'username', 'password', 'role',
     ];
    public $timestamps=false;
}
```

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Modeltransaksi extends Model
{
    protected $table="transaksi";
    protected $primarykey="id";
    protected $fillable = [
        'id_admin', 'id_event',
    ];
    public $timestamps=false;
}
```

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Validator;
use App\Modeltransaksi;
use App\Modelevent;
use App\Modeladmin;
use Auth;
use DB;

class transaksi extends Controller
{
    public function store(Request $req)
    {
        $validator = Validator::make($req->all(),
        [
            'id_admin' => 'required',
            'id_event' => 'required',

        ]);

        if($validator->fails()){
            return Response()->json($validator->errors());
        }

        $event = Modeltransaksi::create([
            'id_admin' => $req->id_admin,
            'id_event' => $req->id_event,

        ]);
        if($event){
            return Response()->json(['status'=>1,'message'=>'Data Event berhasil ditambahkan!']);
        }
        else{
            return Response()->json(['status'=>0,'message'=>'Data Event tidak berhasil ditambahkan!']);
        }
    }
```