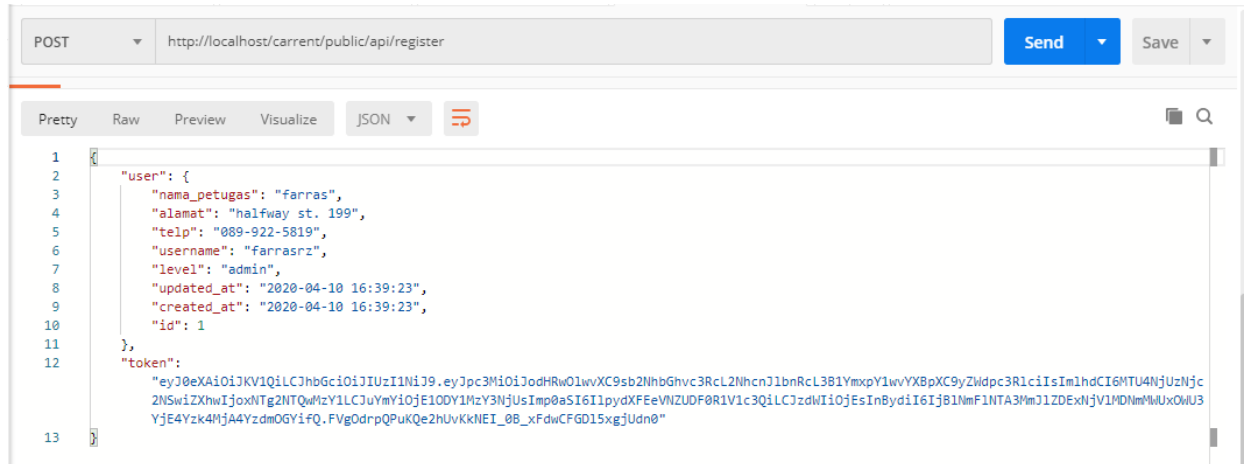


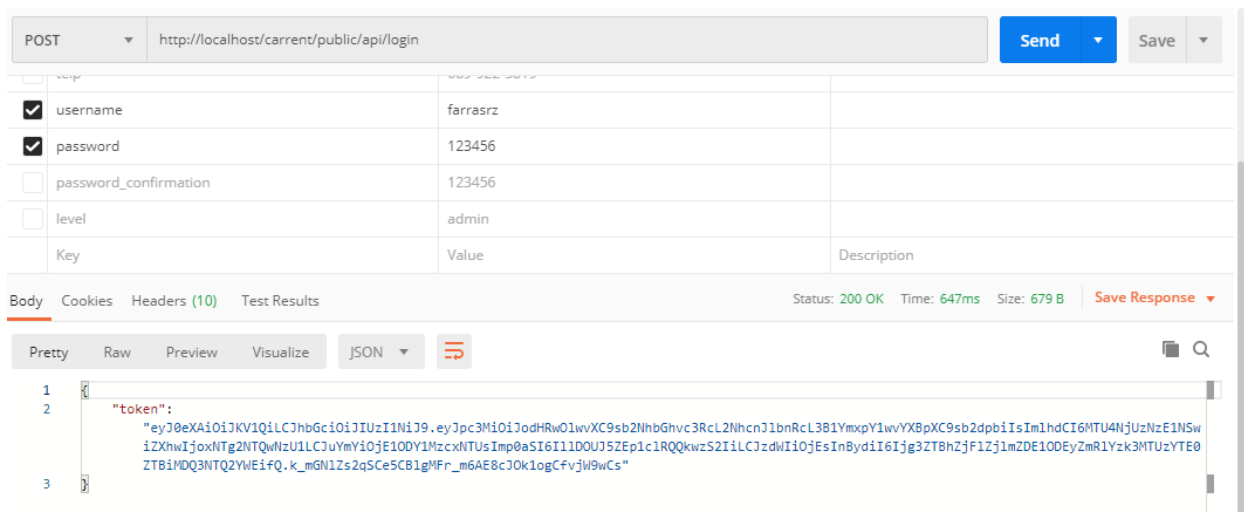
# Luthfarras Tresanizega XIR6/17

## API Auth rental mobil JWT

-register



-login



-api.php

```
Route::post('register', 'PetugasController@register');
Route::post('login', 'PetugasController@login');
```

-controller petugas

```
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_petugas' => 'required|string|max:255',
        'alamat' => 'required',
        'telp' => 'required',
        'username' => 'required|string|max:255|',
        'password' => 'required|string|min:6|confirmed',
        'level' => 'required',
    ]);

    if($validator->fails()){
        return response()->json($validator->errors()->toJson(), 400);
    }

    $user = Petugas::create([
        'nama_petugas' => $request->get('nama_petugas'),
        'alamat' => $request->get('alamat'),
        'telp' => $request->get('telp'),
        'username' => $request->get('username'),
        'password' => Hash::make($request->get('password')),
        'level' => $request->get('level'),
    ]);

    $token = JWTAuth::fromUser($user);

    return response()->json(compact('user', 'token'), 201);
}
```

-model petugas

```
use Notifiable;
protected $table = 'admin';

/**
 * The attributes that are mass assignable.
 *
 * @var array
 */
protected $fillable = [
    'nama_petugas', 'alamat', 'telp', 'username', 'password', 'level'
```