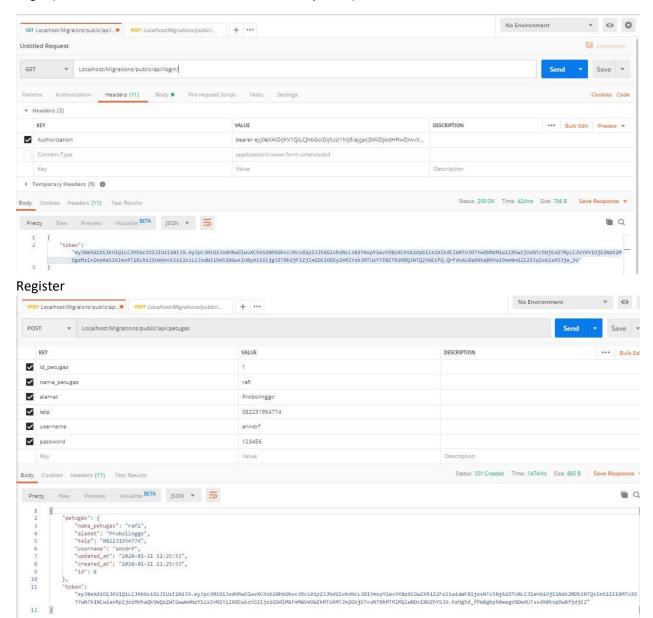
Nama: Ananda Rafi Amanullah

Kelas: XI RPL 7 / 06 (Laravel/04)

Petugas Controller

```
$validator = Validator::make($request->all(),
           'nama_pctugas' -> 'required|string|max:255',
'alamat' -> 'required|string|max:255',
'tclp'-> 'required|string|max:255',
'tclp'-> 'required|string|max:255',
'password' -> 'required|string|min:6|confirmed',
      if($validator->fails()){
    return response()->json($validator->errors()->toJson(), 400);
     $token = JWTAuth::fromUser($petugas);
use App\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
       use Tymon\JWTAuth\Exceptions\JWTException;
                  $credentials = $request->only('username', 'password');
                 try {
  if (! $token = JWFAuth::attompt($credentials)) {
    return response()->json(['error' -> 'invalid_credentials'], 400);
}
                 return response()->json(['error' -> 'invalid_credentials'], 400);
} catch (JMTException $e) {
    return response()->json(['error' -> 'could_not_create_token'], 500);
}
                 return response()->json(compact('token'));
             public function getAuthenticatedUser()
                       if (! $petugas = JMTAuth::parseToken()->authenticate()) {
    return response()->json(['user_not_found'], 484);
                 } catch [Tymon\JWTAuth\Exceptions\TokenExpiredException $e]] {
                       return response()->json(['token_expired'], $e->getStatusCode());
                 } catch (Tymon\JWTAuth\Exceptions\TokenInvalidException Se) {
                       return response()->json(['token_invalid'], $e->getStatusCode());
                  } catch (Tymon\JMTAuth\Exceptions\JWTException $e) {
                       return response()->json(['token_absent'], $e->getStatusCode());
                 return response()->json(compact('petugas'));
```

Login (Post, Cuma waktu di screeenshoot metodnya GET)



USER MODEL