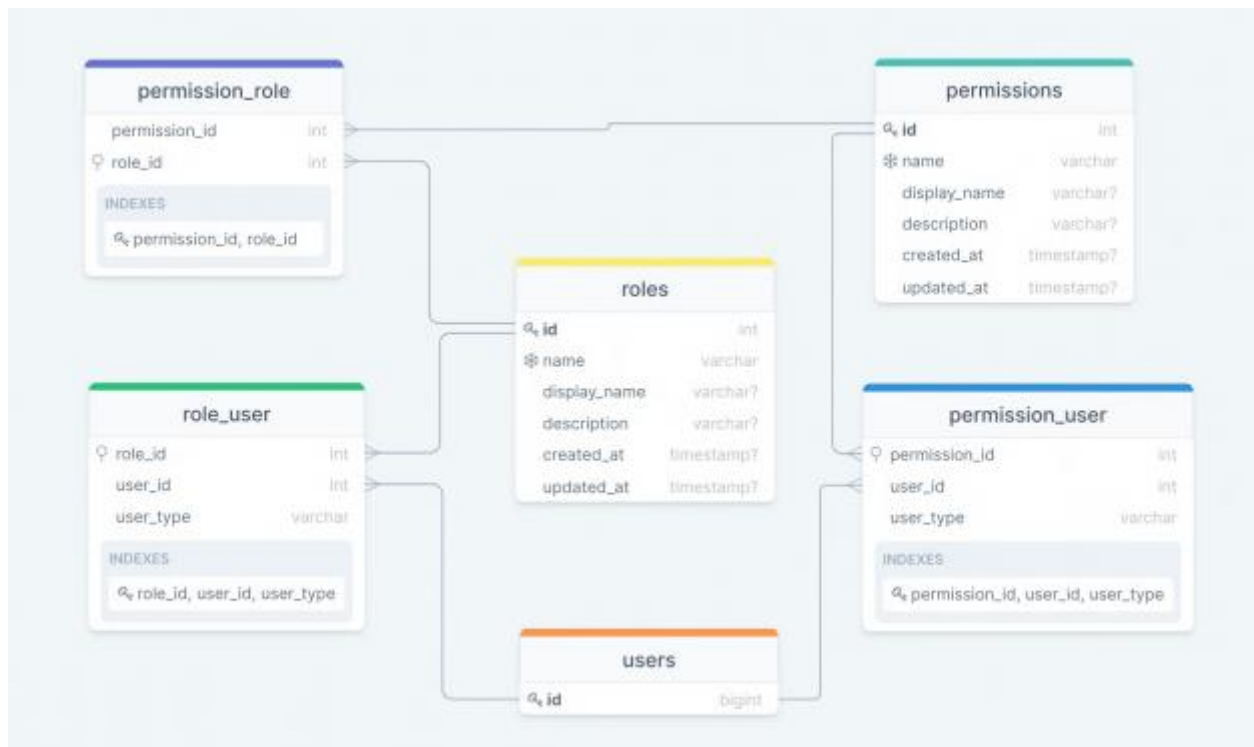


Day 4

Pada pertemuan ini kita akan fokus di User Role yang berfungsi sebagai Gate (gerbang / pembatas) untuk menentukan fitur-fitur yang berhak diakses atau tidak oleh si User.



1. Install package bernama : Laratrust

```
1 composer require santigarcor/laratrust
```

2. Kemudian jalankan perintah :

```
1 php artisan vendor:publish --tag="laratrust"
2 php artisan config:clear
3 php artisan laratrust:setup (kemudian tekan tombol Y)
4 composer dump-autoload
5 php artisan migrate:fresh --seed
```

3. Rubah code Cache berikut di file config\laratrust.php

```
1 'enabled' => env('LARATRUST_ENABLE_CACHE', false),
```

4. Untuk menambah role kepada User, kita bisa membuat perintah dengan merubah & menambahkan code berikut di file UserSeeder.php yang telah kita buat kemarin.

```
1 $admin = Role::create([
2     'name' => 'admin',
3     'display_name' => 'User Administrator'
4 ]);
5
6 $pengguna = Role::create([
7     'name' => 'pengguna',
8     'display_name' => 'User Biasa'
9 ]);
10
11 $user = new User();
12 $user->name = 'Agung Wahyudi';
13 $user->email = 'admin@gmail.com';
14 $user->password = Hash::make('12345678');
15 $user->save();
16
17 $pengunjung = new User();
18 $pengunjung->name = 'Pengunjung';
19 $pengunjung->email = 'pengunjung@gmail.com';
20 $pengunjung->password = Hash::make('12345678');
21 $pengunjung->save();
22
23 $user->attachRole($admin);
24 $pengunjung->attachRole($pengguna);
```

5. Jalankan perintah :

```
1 php artisan migrate:fresh --seed
```

6. Untuk mengecek, apakah user & role yang sudah kita masukan sesuai dengan keinginan, jalankan perintah

```
1 php artisan tinker
2 User::where('email', 'admin@gmail.com')->first()->hasRole('admin');
3 User::where('email', 'admin@gmail.com')->first()->hasRole('pengunjung');
4
5 User::where('email', 'pengunjung@gmail.com')->first()->hasRole('admin');
6 User::where('email', 'pengunjung@gmail.com')->first()->hasRole('pengunjung');
```

7. Sedangkan untuk mengecek melalui browser, kita bisa menambahkan code file resources/views/home.blade.php menjadi berikut

```
1 <div class="card-body">
2     @if (session('status'))
3         <div class="alert alert-success" role="alert">
4             {{ session('status') }}
5         </div>
6     @endif
7     @role('admin')
8     <p>This is visible to users with the admin role. Gets translated to
9         \Laratrust::hasRole('admin')</p>
10    @endrole
11
12    @role('pengguna')
13    <p>This is visible to users with the pengguna role. Gets translated to
14        \Laratrust::hasRole('pengguna')</p>
15    @endrole
16    {{ __('You are logged in!') }}
17 </div>
```

8. Sedangkan untuk mendaftarkan middleware & role berdasarkan route, kita menambahkan code berikut di file web.php

```
1 // Hanya untuk role admin
2 Route::group(['prefix' => 'admin', 'middleware' => ['auth', 'role:admin']], function(){
3     Route::get('/', function(){
4         return 'halaman admin';
5     });
6
7     Route::get('profile', function(){
8         return 'halaman profile admin';
9     });
10 });
11
12
13 // Hanya untuk role pengguna
14 Route::group(['prefix' => 'pengguna', 'middleware' => ['auth', 'role:pengguna']], function(){
15     Route::get('/', function(){
16         return 'halaman pengguna';
17     });
18
19     Route::get('profile', function(){
20         return 'halaman profile pengguna';
21     });
22 });
```

9. Upload project kalian ke github.